Automated observation of multi-agent based simulations: a statistical analysis approach

Philippe Caillou, Javier Gil-Quijano, Xiao Zhou

Automated observation of multi-agent based simulations

A statistical analysis approach

P. Caillou [*], J. Gil-Quijano [**], X. Zhou [*]

[*] *Laboratoire de Recherche en Informatique*
Université Paris Sud – INRIA, Orsay 91405, France
firstname.name@lri.fr

[**] *CEA, LIST, Laboratoire Information Modèles et Apprentissage, F-91191*
Gif-sur-Yvette, France
firstname.name@cea.fr

Résumé. *Multi-agent based simulations (MABS) have been successfully used to model complex systems in different areas. Nevertheless a pitfall of MABS is that their complexity increases with the number of agents and the number of different types of behavior considered in the model. For average and large systems, it is impossible to validate the trajectories of single agents in a simulation. The classical validation approaches, where only global indicators are evaluated, are too simplistic to give enough confidence in the simulation. It is then necessary to introduce intermediate levels of validation. In this paper we propose the use of data clustering and automated characterization of clusters in order to build, describe and follow the evolution of groups of agents in simulations. These tools provides the modeler with an intermediate point of view on the evolution of the model. Those tools are flexible enough to allow the modeler to define the groups level of abstraction (i.e. the distance between the groups level and the agents level) and the underlying hypotheses of groups formation. We give an online application on a simple NetLogo library model (Bank Reserves) and an offline log application on a more complex Economic Market Simulation.*

Mots-Clés : *Complex systems simulation, multi-agents systems, automated observation, automated characterization, clustering, value-test*

1. Introduction

Multi-agent systems (MAS) are specially well suited to represent complex phenomena from the description of local agent behaviours. The simulation of complex systems using MAS is a cyclic process : the modeler introduces his/her knowledge into the model, runs simulations, discovers bugs, pitfalls or unwanted effects, corrects the model and eventually his/her knowledge, and the cycle restarts. The cycle is over when it is not possible to further improve the model because of technical or knowledge limitations.

Once the agents behaviors are defined, the cyclic modeling process usually focuses on fitting global simulation parameters and/or the initial states of agents in order to reproduce global behaviors observed in the modeled empirical phenomena. The global behaviors are usually reproduced and tested with global variables (for example, in the case of socio-spatial models, variables as populations density and growing rates, or evacuation time and number of death for panic simulations). Those global variables are evaluated both on simulation and on empirical data. The calibration of the simulation is achieved by finding the right set of values (or values intervals) for the agents and global simulation parameters which lead to minimize the difference between trajectories of global variables in simulation and in empirical data. This optimization-like approach is implemented by several existing frameworks (for example in GAMA [TDV10], see section 2 for an extended presentation of these tools).

Nevertheless, this traditional approach may be too simplistic in order to characterize the dynamics of complex systems. Indeed, in a complex system, different phenomena may simultaneously occur at different levels (at the agents and at the global levels, but also at intermediate levels) and influence each other [GQLH10]. For instance, groups of agents (flocks of birds, social groups, coalitions, etc.) following similar trajectories of states may appear, evolve and disappear. To describe and evaluate the evolution of that type of groups, the observation of global variables is not enough. Moreover, because of the emergent properties of complex systems, those groups may be unexpected, and their presence may even be unnoticed because no global variable or any other adapted observation mechanism is provided in the simulator. The significance and even the existence of groups may then be hidden by the usually huge amount of information generated in a MAS simulations.

In this paper, we introduce the use of statistical-based tools to assist the modeler in the discovering, following the evolution and describing groups of agents. After an overview of the state of the art (section 2, in section 3, we present two complementary tools, data clustering used to discover and build groups and value test used to automatically describe those groups. In section 4, we present an observation model that uses those tools in order to produce automated analysis of the evolution of groups in MAS simulations illustrated on a NetLogo simple model. We present a more complex offline application in section 5 and conclude in section 6.

2. State of the art

Multi-agent based simulations have been used with a large number of economic, geographic or social applications. There are several available development frameworks for simulation, some of them user-friendly with specific coding language, such as NetLogo [LM], and with the possibility to interface with Java code parts (like GAMA [TDV10]). Others use only generic language (usually java or C#), such as MODULECO [Pha] or Repast [NCV06, RLJ06]. However, none of these platforms integrate any module for automatic group analysis. Based on these platforms, some analyzer tools, such as LEIA [GKMP08], SimExplorer [LR] and [Cai10] were developed to generate and analyze automatically simulations.

LEIA [GKMP08] is a parameter space browser for the IODA simulation framework[KMP08]. It allows the user to instantly make a visual comparison of numerous simulations by seeing all their results in parallel. It provides the user with a set of transformation and generation tools for model, and a set of tests to browse the simulations space. Scoring rules are applied to help the user in identifying interesting configurations (such as cyclic or regular behaviors).

SimExplorer/OpenMole [LR] is a software, which aims at providing a generic environment for programming and executing experimental designs on complex models. The goals are multiple : (1) to externalize the development of the model exploration, in order to make available some generic methods and tools which can be applied in most of the cases for any model to explore ; (2) to favor the reuse of available components, and therefore lower the investment for good quality model exploration applications ; (3) to facilitate a quality insurance approach for model exploration.

[Cai10] is a tool to automatically generate and run new simulations until the results obtained are statistically valid using a chi-square test. It can generate new simulations and perform statistical tests on the results, with an accuracy that increases gradually as the results are produced. This tool can be applied to any RePast-based simulation. It deduces variables and parameters used and asks the user to choose the configuration of interest. New simulations are generated, computed and analyzed until all the independence tests between parameters/variables are valid. Finally, the test results and their margins of error are presented to the users.

The aim of these tools is to study several simulations (the parameter space), to compare their result and analyze them. However, none of them aim at studying one complex simulation to describe it. To explore one simulation, the only existing tools are the integrated tools (such as the NetLogo graphs and logs), which are limited to global or user-defined clusters, and classic data mining on logs. We aim to combine the advantage of online and agent-oriented analysis of NetLogo with the flexibility and descriptive potential of Data Mining tools. Some work using data mining tool to identify groups and describe them had been realised with specific applications, for exemple in the SimBogota simulation ([GQPD07][EHT07]). In these simulations, social groups in Bogota city regions where identified by data mining, and the group results were perceived by the agent. The goal was however more a multi-scale simulation than a description of simulation dynamic.

3. Analysis tools

In this section, we present the two main tools that we use to automatically analyze MAS simulations. To discover the groups of agents we propose to use data clustering, and then value-test evaluations to describe them. These tools are associated in our analysis model (see section 4) in order to automatically describe the evolution of groups and help the modeler to understand what happens in complex simulations.

3.1. *Finding the groups : clustering*

The goal of clustering algorithms is to "find the structure" of a dataset. Most of the time, data represent objects or individuals that are described by a given number of variables or characteristics [LPM06]. The dataset's structure is represented as a partition or a hierarchy of partitions. Every single object is assigned to a given group (cluster) in the partition, or to a several groups (clusters) when considering a hierarchy of partitions. An object is assigned to a given group g if it is more similar (the sense of similar varies with the algorithm) to the objects in g than to objects in other groups. The main hypothesis when clustering a dataset is that the structure exists and the goal is to make it evident. Similarity between objects usually depends on the distance between them (actually between the vector of variables representing them). One of the most used distance (for quantitative variables) is the Euclidean distance.

As the state of an agent is a vector that includes the instantaneous values of the set of variables that describes the agent's behavior, we can consider the set of states of all the agents in a simulation as a dataset and then to cluster it. In that way, groups of agents whose states similarity is maximal can get formed. To conduct the clustering, the "right" algorithm and distance measure have to be chosen. Indeed, different clustering algorithms can lead to different results as their underlying hypothesis and functionning diverge. In our work, it is the responsibility of the modeler to choose those parameters. By including in our model the Weka machine learning library[1], we provide the modeler with a wide list of clustering algorithms and distances functions. In our experiments

1. http ://weka.wikispaces.com/

we used the X-Means algorithm [PM00], described in the following paragraphs.

One of the most known algorithm of clustering is the *K-Means* algorithm [Llo82] whose objective is to find the k prototypes (average characteristic vectors) that represents the best the data. In that algorithm k initial prototypes are defined (usually by random) and at each iteration every object is assigned to its nearest prototype. Every prototype is then updated to the average vector of the characteristics of the objects that were assigned to it. The process is repeated until there is no significant changes of prototypes between two succesive iterations or until a maximum number of iterations is reached. The main pitfall of K-Means is that the number k of clusters must be known. As the idea of clustering is to find the right and unknown structure describing the dataset, there is no reason to know k beforehand.

An improvement of K-Means is proposed with the X-Means algorithm [PM00]. In that algorithm the "right" number k of clusters is determined by successive K-Means executions. The first execution starts with a k_{min} number of clusters (the minimum number of clusters, a parameter of the model), and at each iteration one of the clusters found in the previous iterations is divided into two new clusters. The cluster to be divided is the one whose internal similarity (the similarity between the objects inside the cluster) is the lowest. The process is repeated until k_{max} number of clusters is reached. Then $k_{max} - k_{min} + 1$ partitions are produced. The chosen partition is the one that maximizes the internal similarity of clusters and maximizes the distance between prototypes.

3.2. *Interpreting the groups : Value tests*

Value test (VT) [Mor84, LPM06] is an indicator that allows the automated interpretation of clusters. It determines the more significant factors (for continuous variables) and modalities (for categorical variables) in a given cluster in comparison with the global dataset. The VT compares the deviation between the variables/modalities in clusters and the variables/modalities in the overall dataset. The main hypothesis in the VT calculation is that the variables follow Gaussian distributions. In that condition, for a level of risk of 5% we can consider

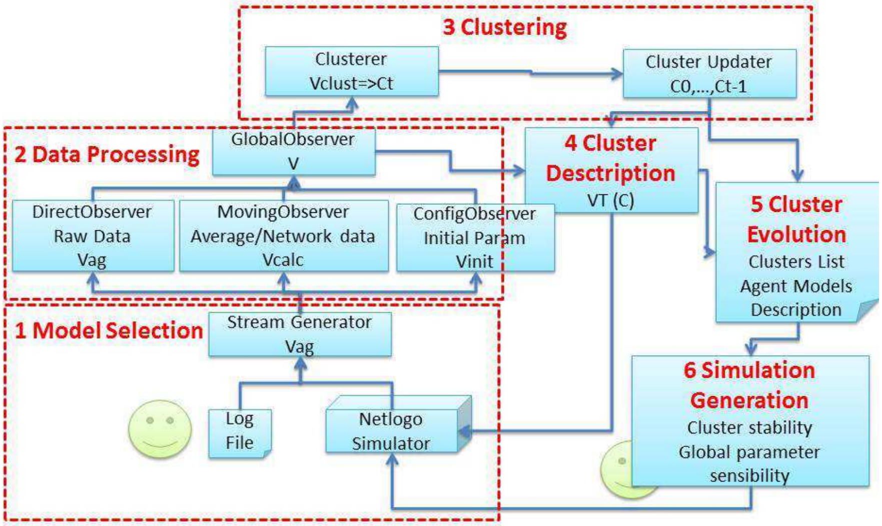


Figure 1 – Overview of the analysis model

that a variable/modality is significant if its VT is greater than 2. The automated description of a group is given by its set of significant variables/modalities. For continous variables the mean value of the variable in the group completes the description.

We present here the calculation of VT for continuous variables, for categorical variables see [Mor84]. Given a dataset containing n elements and a cluster k on the dataset containing n_k elements. Given a quantitative variable j, its average $E(n^j)$ and its variance $S^2(n^j)$ in the overall dataset. Given also the average $E(n_k^j)$ of j in the cluster k, the VT for the variable j in the cluster k is computed as follows :

$$VT(n_k^j) = \frac{(E(n_k^j) - E(n^j))}{\sqrt{(\frac{n-n_k}{n-1} \times \frac{S^2(n^j)}{n_k})}} \tag{1}$$

4. Analysis model

4.1. *Model overview*

Our goal is to describe, online or offline, what happens in a simulation at the cluster level. Our model can be described with several steps as illustrated in Fig. 1 :

1) Model Selection : what do we study ?

2) Data processing : what are the data ?

3) Clustering : can we find homogeneous groups ?

4) Cluster description : how can we describe them ?

5) Cluster evolution : how do they evolve ?

6) Simulation generation : is this reproducible ? In future work, we intend to use the most interesting (or user-selected) agent model (clusters) identified to generate new simulations with similar agents and thus test the clusters behavioral stability.

For a better understanding, we will describe each step with the application of our tool to an illustrative example.

4.2. *Model selection*

The first step is to choose the model to be studied. Our model can be applied both online (with NetLogo) or offline from logs (by simulating an online data stream).

We choose here the *Bank Reserves* model, provided with Netlogo, where financial agents either save or borrow money via loans (Fig. 2). This is a very simple model illustrating the effect of money creation via savings deposits/loans grants. There is only one Bank and $People$ (consumers) agents. Each agent begins with a random amount of money in its $wallet$ (between 0 and a parameter $richThreshold$). When an agent has a positive $wallet$, it deposits its money in the bank, which increases its $saving$ variable (and puts its $wallet$ at 0).

At each step, agents move randomly. When they meet someone they make a transaction, which is a simple transfer from one agent to the

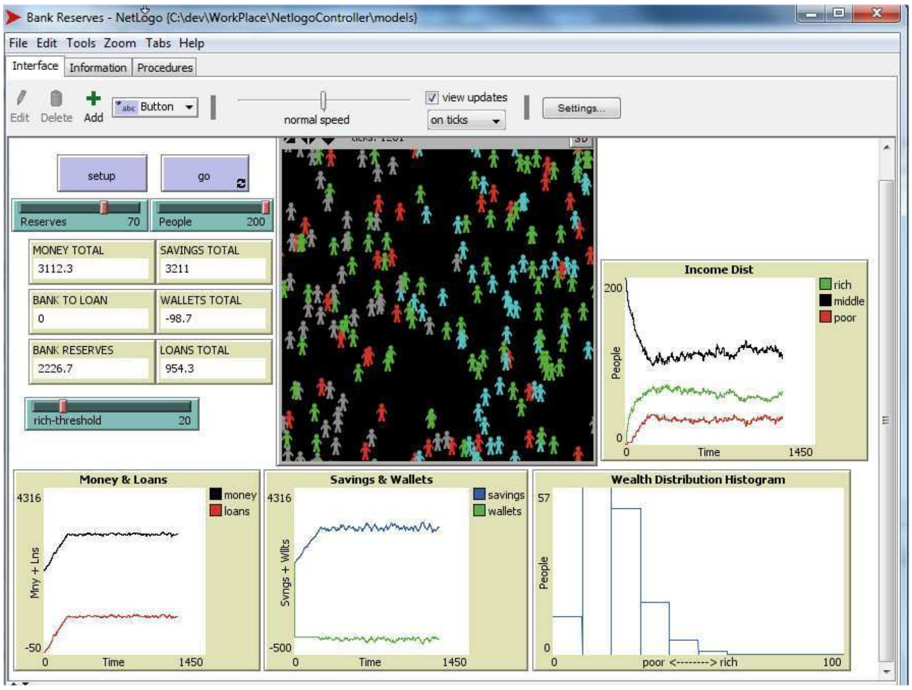


Figure 2 – Netlogo Bank Reserves Model

other. When the buyer agent has not enough money ($savings$ or $wallet$), he takes a $loan$. The bank grants loans (and creates money) unless the total amount of $loans$ reaches the total amount of deposits ($savings$) multiplied by a parameter (1-$Reserves$). In other words, the bank has to keep a $Reserve$ proportion of its deposits which can not be used for loans. When an agent receives money (via transactions), it uses it to pay back its loans if it has some. The $wealth$ of an agent is defined as $savings + wallet - loans$. For our illustrative experiment, we use $Reserves = 70$, $People = 200$ and $richThreshold = 20$.

NetLogo provides some tools to observe an experiment either at an individual or at a global level. For example, on Fig.2 some global variables are presented to give an overview of an experiment. The global amount of $loans$ and $money$ show an early increase, then a stabilization of the total $money$ when the maximum amount of $loans$ is reached. The $Income\ distribution$ graph gives an overview of the repartition of wealth between three fixed groups (negative $wealth$, $wealth$ higher

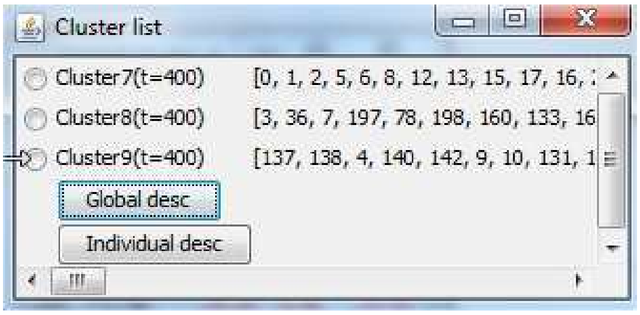


Figure 3 – The cluster list panel gives the identity number of the agents included in each cluster.

than $richThreshold$ and the rest). Even if these informations are interesting, a more detailed understanding of the model behavior can not be reached with such global/local observation. For example, it is difficult to answer the following questions : *who are the wealthy agents ? Do the rich stay rich ?* This would even be truer for more complex models, for which variable interactions are much more difficult to deduce than with such a simple toy simulation.

4.3. *Data processing*

A data matrix is generated every n steps. A line in the matrix represents one agent's state. Raw data from simulations are not the only interesting data for cluster's generation and analysis. Several filters or aggregators can also be used to process the data stream. We use two different aggregators to complete the initial matrix : i) the moving average (for each variable, we add a new variable computed as the average of the last five steps values) ; ii) the initial values for each agent (for each variable, we define a new variable corresponding to the value of the variable for this agent at the starting point of the simulation). These *initial values* variables are not used in the clustering but used latter for the description of the clusters.

4.4. *Clustering*

Clustering is performed on the final data in order to generate homogeneous agent groups (for now, X-Means is used, but any other clustering algorithm from Weka can be easily selected instead). Clusters are visualized in NetLogo (with colors), and their extension and description are presented. For example, in Fig. 3, three clusters are identified in t=400.

4.5. *Cluster's description*

Once the clusters are identified, it is possible to get an easy-to-read description by using VT (see sec. 3.2). The description with VT (Fig. 4) makes it easy to interpret and describe them. Positively (respectively negatively) significant variables are presented in blue (resp. red) : their average is significantly higher (resp. lower) than the global average.

For example, in $t = 400$, three clusters are identified. $Cluster7$, with 114 agents, is a "$poor$" cluster, whose agents have low $wealth$, $savings$, $wallet$ and the corresponding moving average variables ($MMsavings$ and $MMwealth$), and a higher amount of $loans$. Some significant variables are (probably) clustering artifacts ($YCor$) or random effects ($TOHeading$), and will justify our stability analysis.

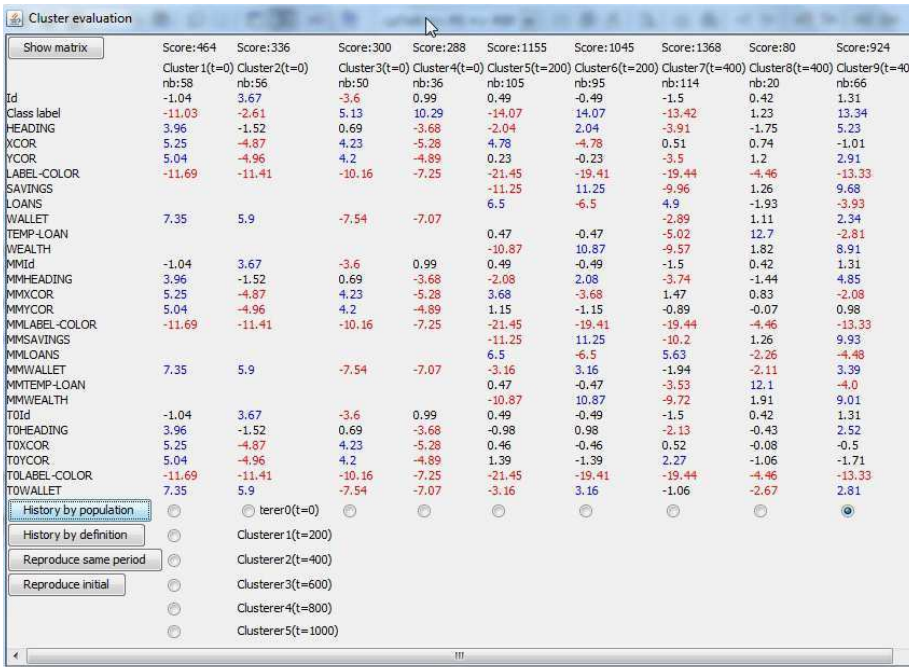
Similarly, $cluster9$ regroups the 66 "$wealthy$" people, with high $wealth$ and $savings$ and few $loans$. An interesting result is the significant $T0Wallet$ variable, corresponding to the $wallet$ value of agents at the beginning of the simulation. The $wealthy$ people were significantly richer than the average at the beginning of the simulation.

At the end of the simulation, a description of all the clusters obtained at each time step gives a global overview of the simulation (Fig. 5, with a selection of some results in Table 1). In our experiment, it is always possible to identify a $wealthy$ and a $poor$ cluster, and sometimes (like in $t = 400$) a $middle$ cluster. From their description, it is already possible to observe that the link between the $wealth$ and the initial wealth (the $T0Wallet$) is not significant anymore after $t = 400$. It may be related to the fact (observed with NetLogo global observation) that bank has

Global cluster d...

Show matrix	cluster7	cluster8	cluster9
	nb:114	nb:20	nb:66
Id	-1.5	0.42	1.31
Class label	-13.42	1.23	13.34
HEADING	-3.91	-1.75	5.23
XCOR	0.51	0.74	-1.01
YCOR	-3.5	1.2	2.91
LABEL-COLOR	-19.44	-4.46	-13.33
SAVINGS	-9.96	1.26	9.68
LOANS	4.9	-1.93	-3.93
WALLET	-2.89	1.11	2.34
TEMP-LOAN	-5.02	12.7	-2.81
WEALTH	-9.57	1.82	8.91
MMId	-1.5	0.42	1.31
MMHEADING	-3.74	-1.44	4.85
MMXCOR	1.47	0.83	-2.08
MMYCOR	-0.89	-0.07	0.98
MMLABEL-COLOR	-19.44	-4.46	-13.33
MMSAVINGS	-10.2	1.26	9.93
MMLOANS	5.63	-2.26	-4.48
MMWALLET	-1.94	-2.11	3.39
MMTEMP-LOAN	-3.53	12.1	-4.0
MMWEALTH	-9.72	1.91	9.01
T0Id	-1.5	0.42	1.31
T0HEADING	-2.13	-0.43	2.52
T0XCOR	0.52	-0.08	-0.5
T0YCOR	2.27	-1.06	-1.71
T0LABEL-COLOR	-19.44	-4.46	-13.33
T0WALLET	-1.06	-2.67	2.81

Figure 4 – Cluster Description at t=400 : listed variables are higher for the member of the cluster (blue positive value) of lower (red negative values) than in the global population. Black variables are not significantly different.



Show matrix	Score:464	Score:336	Score:300	Score:288	Score:1155	Score:1045	Score:1368	Score:80	Score:924
	Cluster1(t=0) nb:58	Cluster2(t=0) nb:56	Cluster3(t=0) nb:50	Cluster4(t=0) nb:36	Cluster5(t=200) nb:105	Cluster6(t=200) nb:95	Cluster7(t=400) nb:114	Cluster8(t=400) nb:20	Cluster9(t=40 nb:66
Id	-1.04	3.67	-3.6	0.99	0.49	-0.49	-1.5	0.42	1.31
Class label	-11.03	-2.61	5.13	10.29	-14.07	14.07	-13.42	1.23	13.34
HEADING	3.96	-1.52	0.69	-3.68	-2.04	2.04	-3.91	-1.75	5.23
XCOR	5.25	-4.87	4.23	-5.28	4.78	-4.78	0.51	0.74	-1.01
YCOR	5.04	-4.96	4.2	-4.89	0.23	-0.23	-3.5	1.2	2.91
LABEL-COLOR	-11.69	-11.41	-10.16	-7.25	-21.45	-19.41	-19.44	-4.46	-13.33
SAVINGS					-11.25	11.25	-9.96	1.26	9.68
LOANS					6.5	-6.5	4.9	-1.93	-3.93
WALLET	7.35	5.9	-7.54	-7.07			-2.89	1.11	2.34
TEMP-LOAN					0.47	-0.47	-5.02	12.7	-2.81
WEALTH					-10.87	10.87	-9.57	1.82	8.91
MMId	-1.04	3.67	-3.6	0.99	0.49	-0.49	-1.5	0.42	1.31
MMHEADING	3.96	-1.52	0.69	-3.68	-2.08	2.08	-3.74	-1.44	4.85
MMXCOR	5.25	-4.87	4.23	-5.28	3.68	-3.68	1.47	0.83	-2.08
MMYCOR	5.04	-4.96	4.2	-4.89	1.15	-1.15	-0.89	-0.07	0.98
MMLABEL-COLOR	-11.69	-11.41	-10.16	-7.25	-21.45	-19.41	-19.44	-4.46	-13.33
MMSAVINGS					-11.25	11.25	-10.2	1.26	9.93
MMLOANS					6.5	-6.5	5.63	-2.26	-4.48
MMWALLET	7.35	5.9	-7.54	-7.07	-3.16	3.16	-1.94	-2.11	3.39
MMTEMP-LOAN					0.47	-0.47	-3.53	12.1	-4.0
MMWEALTH					-10.87	10.87	-9.72	1.91	9.01
T0Id	-1.04	3.67	-3.6	0.99	0.49	-0.49	-1.5	0.42	1.31
T0HEADING	3.96	-1.52	0.69	-3.68	-0.98	0.98	-2.13	-0.43	2.52
T0XCOR	5.25	-4.87	4.23	-5.28	0.46	-0.46	0.52	-0.08	-0.5
T0YCOR	5.04	-4.96	4.2	-4.89	1.39	-1.39	2.27	-1.06	-1.71
T0LABEL-COLOR	-11.69	-11.41	-10.16	-7.25	-21.45	-19.41	-19.44	-4.46	-13.33
T0WALLET	7.35	5.9	-7.54	-7.07	-3.16	3.16	-1.06	-2.67	2.81

Figure 5 – Global Overview of clusters

reached its loan limit (the total amount of money stops to increase at around $t = 230$).

However, it is difficult to compare clusters at different time steps with this overview since they are different both in intension and in extension. In a more complex model, cluster may have a completely different meaning at different steps.

Tableau 1 – Selection of the "rich" clusters and some variables results from the global overview of a simulation (Fig 5)

VT	Cluster6	Cluster9	Cluster10	Cluster13
time	200	400	600	800
size	95	66	37	52
Savings	11,25	9,68	11,29	3,26
Loans	-6,5	-3,93	-2,63	-1,77
Wealth	10,87	8,91	9,72	3,27
TOWallet	3,16	2,81	0,46	-0,11

4.6. *Clusters evolution*

In order to describe the clusters' evolution, we consider two alternative hypothesis : either the extension in every cluster is considered as stable (we keep exactly the same agent population in the cluster), or the intension of every cluster is fixed (we keep the same definition of the cluster).

Evolution by extension/population

Once an interesting cluster is identified (for example the $wealthy$ agents of $t = 400$, $cluster9$), it is interesting to follow its evolution. To do it, the first way is to fix the extension (population) of the cluster.

Fig.6 describes the evolution of the $cluster9$ with fixed extension after $t = 400$ (see Table 2 for a selection of the most interesting variables). Initial parameters values are stable since the population does not change. Other variables may change (except for example Id since this variable is constant for every agent). This view clearly shows that all wealth-related differences with the other agents decrease : all the (ab-

Cluster evolution by population(Cluster9)

Show matrix

	Cluster9(t=0) nb:66	Cluster9(t=200) nb:66	Score:924 Cluster9(t=400) nb:66	Score:660 Cluster9(t=600) nb:66	Score:594 Cluster9(t=800) nb:66	Score:396 Cluster9(t=1000) nb:66	Score:396 Cluster9(t=1200) nb:66
Id			1.31	1.31	1.31	1.31	1.31
Class label			13.34	-4.62	-1.57	4.26	-0.77
HEADING			5.23	0.97	0.7	0.35	0.25
XCOR			-1.01	0.37	0.97	-0.68	-0.33
YCOR			2.91	-0.3	-0.04	0.41	2.52
LABEL-COLOR			-13.33	-13.33	-13.33	-13.33	-13.33
SAVINGS			9.68	7.26	4.87	4.15	4.0
LOANS			-3.93	-2.85	-2.39	-1.97	-1.04
WALLET			2.34	2.34	0.45	0.24	-0.13
TEMP-LOAN			-2.81	-1.97	-1.4	-1.04	-0.63
WEALTH			8.91	6.78	4.78	4.14	3.64
MMId			1.31	1.31	1.31	1.31	1.31
MMHEADING			4.85	4.34	4.01	1.24	0.76
MMXCOR			-2.08	-1.5	0.22	0.37	-0.01
MMYCOR			0.98	1.63	1.45	0.04	1.51
MMLABEL-COLOR			-13.33	-13.33	-13.33	-13.33	-13.33
MMSAVINGS			9.93	9.66	8.2	5.96	4.64
MMLOANS			-4.48	-4.13	-3.42	-2.66	-2.11
MMWALLET			3.39	3.02	2.69	2.04	0.19
MMTEMP-LOAN			-4.0	-3.72	-2.43	-1.64	-1.14
MMWEALTH			9.01	8.78	7.6	5.74	4.52
T0Id			1.31	1.31	1.31	1.31	1.31
T0HEADING			2.52	2.52	2.52	2.52	2.52
T0XCOR			-0.5	-0.5	-0.5	-0.5	-0.5
T0YCOR			-1.71	-1.71	-1.71	-1.71	-1.71
T0LABEL-COLOR			-13.33	-13.33	-13.33	-13.33	-13.33
T0WALLET			2.81	2.81	2.81	2.81	2.81

Figure 6 – Cluster evolution of Cluster9 (rich peoples of T=400) by extension (fixed population)

solute) VT values for $wealth$, $saving$ and $loans$ decrease. This mean that, in average, the wealthy people of $t = 400$ are becoming less and less wealthy. They are still significantly wealthier than the average at $t = 1200$, but their $loans$ are no more significantly lower in comparison with $t = 1000$. We can also check on this evolution that the clustering artifacts (like the $YCor$ variable) do not stay significant.

Evolution by definition

The second way is to fix the clusters intension (definition). Fig. 7 represents the description of the clusters identified at each step with the intension function of $t = 400$ (see Table 3 for a selection of interesting variables). All the variables considered in clustering are by definition roughly similar, since the intension of clusters is the same. However, the other variables may evolve (in our example, the initial parameters of the agents).

Tableau 2 – Evolution of Cluster9 (rich peoples of T=400) by extension : selection of some interesting variables

VT	Cluster9	Cluster9	Cluster9	Cluster9	Cluster9
time	400	600	800	1000	1200
size	66	66	66	66	66
Ycor	2,91	-0,3	-0,04	0,41	2,52
Savings	9,68	7,26	4,87	4,15	4
Loans	-3,93	-2,85	-2,39	-1,97	-1,04
Wealth	8,91	6,78	4,78	4,14	3,64
TOWallet	2,81	2,81	2,81	2,81	2,81

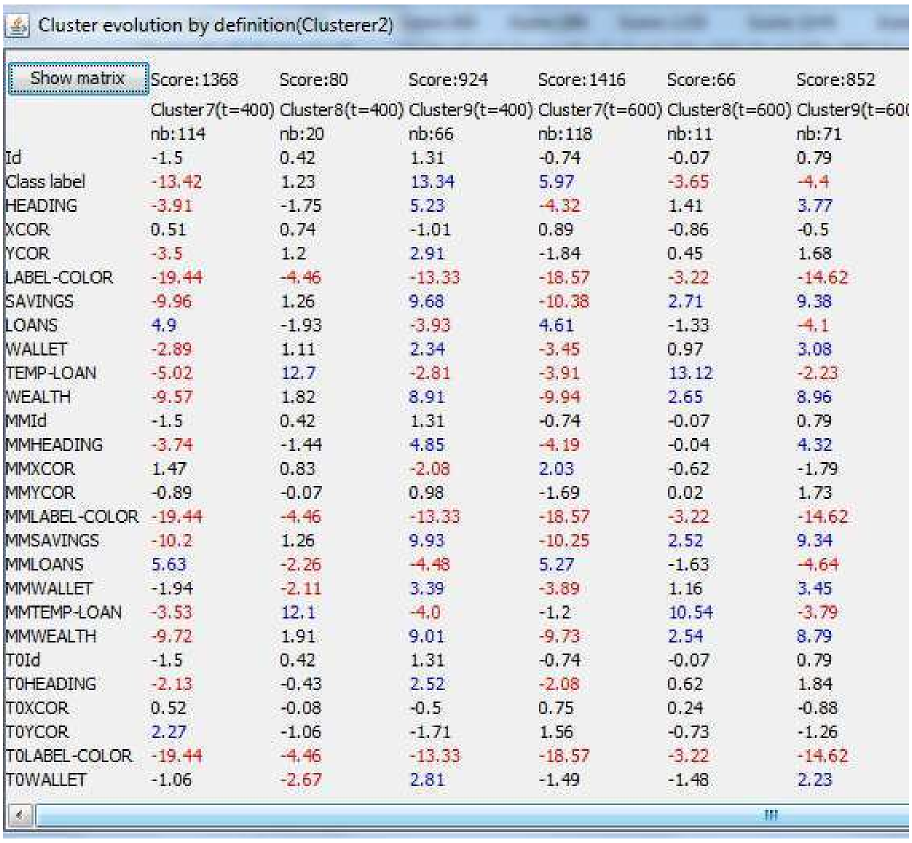
$Cluster9$, for example, regroup the wealthy agents at each step, but the number and the initial properties of these agents evolve. It is interesting to see that the number of wealthy agent stays approximatively constant (66, 71, 56, 58, 65). But the evolution of the initial parameters confirms the observation made with the global overview : the initial wealth of the agent ($TOWallet$) is not significant anymore after $t = 600$.

Tableau 3 – Evolution of Cluster9 (rich peoples defined at T=400) by definition : selection of some interesting variables

VT	Cluster6	Cluster6	Cluster6	Cluster6
time	400	600	800	1000
size	66	71	56	58
Savings	9,68	9,38	9,04	8,61
Wealth	8,91	8,96	8,52	8,49
T0Heading	2,52	1,84	1,47	1,24
T0Wallet	2,81	2,23	0,91	1,48

4.7. *Cluster evaluation*

The evaluation of the cluster is done by combining the size of the cluster and its ‘descriptivness’ measured by the number of significant VT. The objective is to identify clusters that are both big enough to be generic and descriptive enough to bring some interesting informations.



Cluster evolution by definition(Clusterer2)

Show matrix	Score:1368	Score:80	Score:924	Score:1416	Score:66	Score:852
	Cluster7(t=400)	Cluster8(t=400)	Cluster9(t=400)	Cluster7(t=600)	Cluster8(t=600)	Cluster9(t=60
	nb:114	nb:20	nb:66	nb:118	nb:11	nb:71
Id	-1.5	0.42	1.31	-0.74	-0.07	0.79
Class label	-13.42	1.23	13.34	5.97	-3.65	-4.4
HEADING	-3.91	-1.75	5.23	-4.32	1.41	3.77
XCOR	0.51	0.74	-1.01	0.89	-0.86	-0.5
YCOR	-3.5	1.2	2.91	-1.84	0.45	1.68
LABEL-COLOR	-19.44	-4.46	-13.33	-18.57	-3.22	-14.62
SAVINGS	-9.96	1.26	9.68	-10.38	2.71	9.38
LOANS	4.9	-1.93	-3.93	4.61	-1.33	-4.1
WALLET	-2.89	1.11	2.34	-3.45	0.97	3.08
TEMP-LOAN	-5.02	12.7	-2.81	-3.91	13.12	-2.23
WEALTH	-9.57	1.82	8.91	-9.94	2.65	8.96
MMId	-1.5	0.42	1.31	-0.74	-0.07	0.79
MMHEADING	-3.74	-1.44	4.85	-4.19	-0.04	4.32
MMXCOR	1.47	0.83	-2.08	2.03	-0.62	-1.79
MMYCOR	-0.89	-0.07	0.98	-1.69	0.02	1.73
MMLABEL-COLOR	-19.44	-4.46	-13.33	-18.57	-3.22	-14.62
MMSAVINGS	-10.2	1.26	9.93	-10.25	2.52	9.34
MMLOANS	5.63	-2.26	-4.48	5.27	-1.63	-4.64
MMWALLET	-1.94	-2.11	3.39	-3.89	1.16	3.45
MMTEMP-LOAN	-3.53	12.1	-4.0	-1.2	10.54	-3.79
MMWEALTH	-9.72	1.91	9.01	-9.73	2.54	8.79
T0Id	-1.5	0.42	1.31	-0.74	-0.07	0.79
T0HEADING	-2.13	-0.43	2.52	-2.08	0.62	1.84
T0XCOR	0.52	-0.08	-0.5	0.75	0.24	-0.88
T0YCOR	2.27	-1.06	-1.71	1.56	-0.73	-1.26
T0LABEL-COLOR	-19.44	-4.46	-13.33	-18.57	-3.22	-14.62
T0WALLET	-1.06	-2.67	2.81	-1.49	-1.48	2.23

Figure 7 – Cluster evolution of Cluster9 (rich peoples defined at T=400) by intension (fixed definition)

The score of a cluster c at time step t is calculated as the product of the number of significant variables (whose '$|VT|$ is greater than 2) : $score(c,t) = |V^S_{c,t}| \times n_{c,t}$ where $V^S_{c,t}$ is the set of significant variables of c and $n_{c,t}$ is the number of agents in c at the time t.

For example, in Fig. 6, the score of the cluster is initially relatively high (924) because the population is important and the number of significant VT is high. But since the cluster is not stable (the rich people don't stay rich), the number of significant variables decreases. The score decreases, reflecting the fact that the cluster becomes difficult to interpret (except by : "the ones that were poor at t=400").

5. Experiments

We tested our analysis tool on several simulation models, both online with NetLogo and offline with data logs. Since we have already described a NetLogo application in the previous section to illustrate the description tool, we will describe here an offline analysis application.

Model description

To illustrate how our model deals with more complex simulations, we chose a model following the KIDS approach [EM04] : the number of parameters and observed variables is kept high to be more descriptive and realistic rather than synthetic. The Rungis Wholesale Market simulation [CCB09a][CCB09b] was developed with the BitBang Framework [BMC06] and reproduces a Fruit and Vegetable wholesale market. One type of seller agent and 4 types of Buyer agents are considered in the simulation (with many variable parameters, 20 variables in average by agent type). The four type of buyer agents are : Restorators seeking efficiency, TimeFree seeking good opportunities, Barbes seeking low-quality and low price products and Neuilly buyers seeking high quality high prices products. The csv logs record one line for each couple day/agent, with 33 output variables (see below). We analyze here only the 60 Buyer agents during the first 10 days of the simulation.

The main observed variables of this model are the transaction time, the number of sellers per buyer, the quality and quantity of the products and the prices. There are four types of prices, the producer price (price paid by the sellers to the producers), the transaction price(price paid by the buyers to the sellers), the standard price (price given at the beginning of the transaction) and the final price (price paid by the consumers). In general $producerprice < transactionprice < standardprice < finalprice$.

Clusters description

Two clusters are identified at each step. They are easy to describe since many variables are significant (see Fig 8 where the high proportion of red or blue numbers illustrates the high number of significant variables, and a selection and description of interesting variables for the cluster 3 in Table 4). For example, at t=0 (and t=1) we identify the *"expansive"* cluster that is composed by the buyers of high-quality expansive products, and the *"cheaps"* cluster that is composed by the agents buying low-quality cheap products. Even if this could be anticipated from the buyers definition, one first quick analysis helps to validate the model behaviors : the *"expansive"* products are fresher (product Age), with higher prices, and the Buyer and Seller profit as well as profit rate higher than for the *"cheaps"*. Other significant variables give some new interesting informations : the buying time ($MoyHour$) is not significantly different, but the *expansive* buy significantly more products ($SumofQty$) to more Sellers ($NbSeller$) than the *cheaps* agents. These informations were not trivial to identify unless you knew you want to find it before the analysis.

Cluster's evolution

Following the identified clusters brings new interesting insights. For example, in Fig. 9 (and some selected variables in Table 5), we follow the *"expansive"* cluster identified at t=1. First, it is possible to check that the clusters has a stable behavior. The agents buying expansive and high-quality products still do the same in the following days. It is however possible to describe some cluster evolutions : the *MoyHour* variable, which measures the Average transaction time shows a constant decrease. *"expansive"* agents buy progressively sooner than the *"cheaps"* agents. Also, the $SumQty$ variables, which was significant when the cluster was created, becomes quickly not-significant : the

Cluster evaluation

Show matrix	Score:968	Score:704	Score:980	Score:1225	Score:874	Score:1380
	Cluster1(t=0) nb:22	Cluster2(t=0) nb:16	Cluster3(t=1) nb:20	Cluster4(t=1) nb:25	Cluster5(t=2) nb:19	Cluster6(t=2) nb:30
Day	0.0	0.0	0.0	0.0	0.0	0.0
Id	5.09	-5.09	-4.47	4.47	-5.19	5.19
MoyHour	-0.72	0.72	1.76	-1.76	1.3	-1.3
NbSeller	0.03	-0.03	-2.26	2.26	-1.28	1.28
Somme de Qty	0.45	-0.45	-2.43	2.43	-1.66	1.66
NbProd	0.03	-0.03	-2.26	2.26	-1.28	1.28
MoyTransPrice	5.95	-5.95	-5.89	5.89	-5.26	5.26
MoyFinPrice	5.78	-5.78	-6.1	6.1	-6.28	6.28
MoyQualID	3.82	-3.82	-5.38	5.38	-5.25	5.25
MoyStdPrice	5.86	-5.86	-5.72	5.72	-4.01	4.01
MoyProdPrice	3.33	-3.33	-3.09	3.09	-1.32	1.32
MoyAge	-5.84	5.84	5.59	-5.59	5.62	-5.62
MoyQuality	5.65	-5.65	-6.18	6.18	-5.98	5.98
MoyQty	0.5	-0.5	1.05	-1.05	-0.43	0.43
MoyQualAdmId	5.52	-5.52	-6.09	6.09	-5.95	5.95
MoyIDAdmin	1.26	-1.26	-3.11	3.11	-1.82	1.82
MoyAdminPrice	5.92	-5.92	-5.87	5.87	-5.76	5.76
Moysupply	5.45	-5.45	-5.3	5.3	-3.38	3.38
MoyHour2	-0.72	0.72	1.79	-1.79	1.35	-1.35
MoymargBuyer	-5.53	5.53	-3.76	3.76	-5.85	5.85
MoyNewOffer	5.45	-5.45	-5.83	5.83	-5.99	5.99
MoymargSeller	5.88	-5.88	-5.29	5.29	-4.39	4.39
MoyVmargBuyer	-4.84	4.84	-2.6	2.6	-5.55	5.55
MoyVmargSeller	5.25	-5.25	-4.52	4.52	-3.69	3.69
Moydem2	-3.35	3.35	4.65	-4.65	4.53	-4.53
Moysup2	5.13	-5.13	-4.31	4.31	-2.46	2.46
Moysup/dem	4.42	-4.42	-5.87	5.87	-5.49	5.49

Figure 8 – Global Overview of clusters after 10 days of Rungis Market Simulation

Tableau 4 – Variable selection and description for the cluster3 identified at t=1

Cluster	Variable description	4
Pop	Cluster size	25
MoyHour	Avg Time for transaction	-1,76
NbSeller	Nb of visited sellers	2,26
SumQty	Total Bought Qty	2,43
NbProd	Nb of product category	2,26
MoyTransPrice	Avg Transaction Price	5,89
MoyFinPrice	Avg Price for final consummer	6,1
MoyStdPrice	Avg Starting Price	5,72
MoyProdPrice	Avg Producer Price	3,09
MoyAge	Avg Age of the product	-5,59
MoyQuality	Avg Quality	6,18
MoyQty	Avg Qty for transactions	-1,05
MoymargBuyer	Avg Buyer margin rate	3,76
MoymargSeller	Avg Seller margin rate	5,29

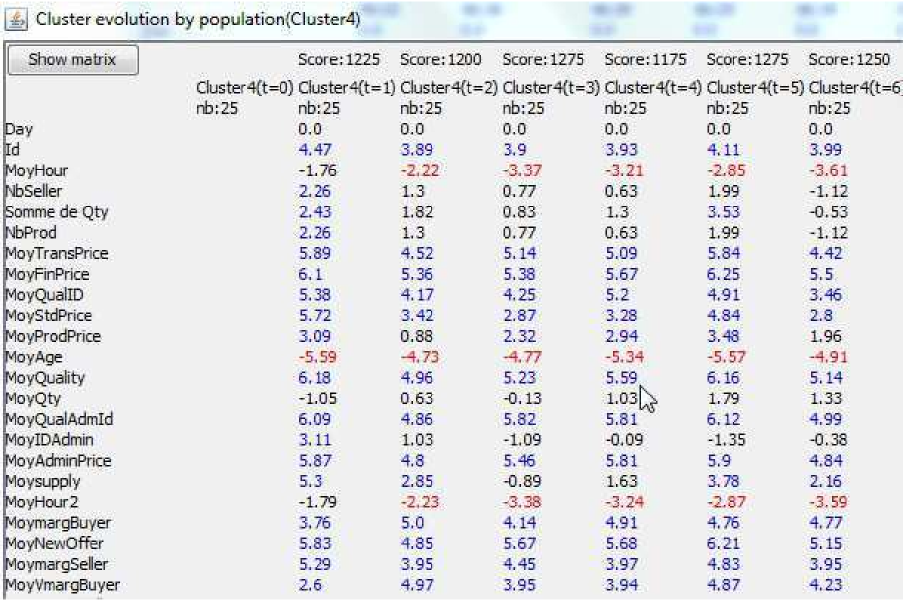
$expansive$ buyers do not buy more product thant the orthers, it was just an artifact at the cluster creation.

6. Conclusions

The framework for the observation of MAS simulation that we present here, provides the modeller with generic tools that allow him/her to get a synthetic descriptive view of MABS. Presently, it can be used to understand the dynamics of simulations and to ease their validation. In future work we intend to develop a mechanism that will allow the modeller to use the definitions of interesting clusters in new simulations as generic “agent models”. Indeed, cluster definition and population can be used to define a distribution function to generate agents profiles. It is easy to retrieve the average values and the standard deviation of every variable by using simple statistical analysis tools. Agents generated using the distribution functions could be reintroduced in simulations, clusters can be rebuilt and the global simulation variables can be

Tableau 5 – Evolution by extension of the *expansive* group detected at t=1 (cluster3)

Day	1	2	3	4	5	6
Pop	25	25	25	25	25	25
MayHour	-1,76	-2,22	-3,37	-3,21	-2,85	-3,61
NbSeller	2,26	1,3	0,77	0,63	1,99	-1,12
SumQty	2,43	1,82	0,83	1,3	3,53	-0,53
NbProd	2,26	1,3	0,77	0,63	1,99	-1,12
MoyTransPrice	5,89	4,52	5,14	5,09	5,84	4,42
MoyFinPrice	6,1	5,36	5,38	5,67	6,25	5,5
MoyStdPrice	5,72	3,42	2,87	3,28	4,84	2,8
MoyProdPrice	3,09	0,88	2,32	2,94	3,48	1,96
MoyAge	-5,59	-4,73	-4,77	-5,34	-5,57	-4,91
MoyQuality	6,18	4,96	5,23	5,59	6,16	5,14
MoyQty	-1,05	0,63	-0,13	1,03	1,79	1,33
MoymargBuyer	3,76	5	4,14	4,91	4,76	4,77
MoymargSeller	5,29	3,95	4,45	3,97	4,83	3,95


Cluster evolution by population(Cluster4)

Show matrix		Score:1225	Score:1200	Score:1275	Score:1175	Score:1275	Score:1250
	Cluster4(t=0) nb:25	Cluster4(t=1) nb:25	Cluster4(t=2) nb:25	Cluster4(t=3) nb:25	Cluster4(t=4) nb:25	Cluster4(t=5) nb:25	Cluster4(t=6 nb:25
Day		0.0	0.0	0.0	0.0	0.0	0.0
Id		4.47	3.89	3.9	3.93	4.11	3.99
MoyHour		-1.76	-2.22	-3.37	-3.21	-2.85	-3.61
NbSeller		2.26	1.3	0.77	0.63	1.99	-1.12
Somme de Qty		2.43	1.82	0.83	1.3	3.53	-0.53
NbProd		2.26	1.3	0.77	0.63	1.99	-1.12
MoyTransPrice		5.89	4.52	5.14	5.09	5.84	4.42
MoyFinPrice		6.1	5.36	5.38	5.67	6.25	5.5
MoyQualID		5.38	4.17	4.25	5.2	4.91	3.46
MoyStdPrice		5.72	3.42	2.87	3.28	4.84	2.8
MoyProdPrice		3.09	0.88	2.32	2.94	3.48	1.96
MoyAge		-5.59	-4.73	-4.77	-5.34	-5.57	-4.91
MoyQuality		6.18	4.96	5.23	5.59	6.16	5.14
MoyQty		-1.05	0.63	-0.13	1.03	1.79	1.33
MoyQualAdmId		6.09	4.86	5.82	5.81	6.12	4.99
MoyIDAdmin		3.11	1.03	-1.09	-0.09	-1.35	-0.38
MoyAdminPrice		5.87	4.8	5.46	5.81	5.9	4.84
Moysupply		5.3	2.85	-0.89	1.63	3.78	2.16
MoyHour2		-1.79	-2.23	-3.38	-3.24	-2.87	-3.59
MoymargBuyer		3.76	5.0	4.14	4.91	4.76	4.77
MoyNewOffer		5.83	4.85	5.67	5.68	6.21	5.15
MoymargSeller		5.29	3.95	4.45	3.97	4.83	3.95
MoyVmargBuyer		2.6	4.97	3.95	3.94	4.87	4.23

Figure 9 – Evolution by extension of the *expansive* group detected at t=1 (cluster3)

compared with their previous values. In that way, the clusters stability and their "expressiveness" can be measured over different simulations.

In order to allow the analysis of a wide number of different type of simulations we are currently adapting our framework both to consider qualitative and network variables and facilitate large simulations analysis. The latter will be done by integrating our framework to the OpenMole engine. That engine provides, among other facilities, the easy use of cluster and grid computing for simulations.

Références

[BMC06] T. Baptista, T. Menezes, and E. Costa. Bitbang : A model and framework for complexity research. In *ECCS 2006*, 2006.

[Cai10] Philippe Caillou. Automated multi-agent simulation generation and validation. In *PRIMA 2010*, page 16p. LNCS, 2010.

[CCB09a] Philippe Caillou, Corentin Curchod, and Tiago Baptista. Simulation of the rungis wholesale market : Lessons on the calibration, validation and usage of a cognitive agent-based simulation. In *IAT*, pages 70–73, 2009.

[CCB09b] Corentin Curchod, Philippe Caillou, and Tiago Baptista. Which Buyer-Supplier Strategies on Uncertain Markets ? A Multi-Agents Simulation. In *Strategic Managemtn Society*, Washington États-Unis d'Amérique, 2009.

[EHT07] Bruce Edmonds, Cesareo Hernandez, and Klaus Troitzsch, editors. *Social Simulation : Technologies, Advances and New Discoveries*, volume 1. Idea Group Inc., 2007.

[EM04] Bruce Edmonds and Scott Moss. From kiss to kids - an 'anti-simplistic' modelling approach. In *MABS*, pages 130–144, 2004.

[GKMP08] Francois Gaillard, Yoann Kubera, Philippe Mathieu, and Sebastien Picault. A reverse engineering form for multi agent systems. In *ESAW 2008*, pages 137–153, 2008.

[GQLH10] Javier Gil-Quijano, Thomas Louail, and Guillaume Hutzler. From biological to urban cells : lessons from three multilevel agent-based models. In *PRACSYS 2010 – First Pacific Rim workshop on Agent-based modeling and simulation of Complex Systems*, Kolkota, India, 2010. LNCS.

[GQPD07] J. Gil-Quijano, M. Piron, and A. Drogoul. Mechanisms of automated formation and evolution of social-groups : A multi-agent system to model the intra-urban mobilities of Bogotá city. In *Social Simulation : Technologies, Advances and New Discoveries*, chapter 12, pages 151–168. Idea Group Inc., 2007.

[KMP08] Yoann Kubera, Philippe Mathieu, and S ?bastien Picault. Interaction-oriented agent simulations : From theory to implementation. In Malik Ghallab, Constantine Spyropoulos, Nikos Fakotakis, and Nikos Avouris, editors, *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI'08)*, pages 383–387. IOS Press, 2008.

[Llo82] Stuart P. Lloyd. Least squares quantization in pcm. In *IEEE Transactions on Information Theory*, volume 28, pages 129–137, 1982.

[LM] Center For Connected Learning and Computer-Based Modeling. Netlogo : http ://ccl.northwestern.edu/netlogo/.

[LPM06] Ludovic Lebart, Marie Piron, and Alain Morineau. *Statistique exploratoire multidimeensionnelle : visualisation et inférence en fouilles de données*. Fourth edition, Dunod, Paris, 2006.

[LR] Mathieu Leclaire and Romain Reuillon. Simexplorer : http ://www.simexplorer.org/.

[Mor84] Alain Morineau. Note sur la caractérisation statistique d'une classe et les valeurs-tests. In *Bull. Techn. du Centre de Statistique et d'Informatique Appliquées*, volume 2, pages 20–27, 1984.

[NCV06] Michael J. North, Nicholson T. Collier, and Jerry R. Vos. Experiences creating three implementations of the repast agent modeling toolkit. In *ACM Transactions on Modeling and Computer Simulation*, volume 16, pages 1–5, 2006.

[Pha] Denis Phan. From agent-based computational economics towards cognitive economics. In *Cognitive Economics, Handbook of Computational Economics*.

[PM00] Dan Pelleg and Andrew Moore. Xmeans : Extending k-means with efficient estimation of the number of clusters. In *17th International Conference on Machine Learning*, pages 727–734, 2000.

[RLJ06] Steven F. Railsback, Steven L. Lytinen, and Stephen K. Jackson. Agent-based simulation platforms : Review and development recommendations. In *Simulation*, volume 82, pages 609–623, 2006.

[TDV10] Patrick Taillandier, Alexis Drogoul, and Duc-An Vo. Gama : a simulation platform that integrates geographical information data, agent-based modeling and multi-scale control. In *PRIMA 2010*, 2010.