Control on System Diffusion Using Genetic Automata

Luaï Jaff[(1)], Cyrille Bertelle[(1)], Gérard H.E. Duchamp[(2)]

[(1)]LITIS - University of Le Havre, 25 rue Philippe Lebon, BP 540, 76058 Le Havre Cedex, France
email: luai.jaff@gmail.com, cyrille.bertelle@litislab.eu

[(2)]LIPN - University of Paris XIII, 99 avenue Jean-Baptiste Clément, 93430 Villetaneuse, France
email: ghed@lipn.univ-paris13.fr

KEYWORDS
Self-organization; Genetic Automata; Diffusion; Control

ABSTRACT

Complex systems (LeMoigne 1999) are models which describe dynamic organizations or systems where the internal interaction network between their components does not allow the control of individual components with efficience. In a lot of cases, the individual control leads to break the whole structure itself. In other cases, some complex systems exhibit resilience properties which lead them to recover their initial state after small perturbations or individual controls. Controlling complex systems need to not trying to manage the individual controls but to manage the global control, guiding the whole system without directly act on individuals themselves. The study presented in this paper explains how to control a high level system property, the diffusive behavior, by spectral analysis over genetic automata. First experiments are presented using MuPad implementation.

COMPLEX SYSTEMS AND FEED-BACKS FOR ECONOMY

We describe the systemic approach of economical system modelling, especially the feed-back concept which can be considered as a global control over the system components.

Complexity in Economy

We can classify the economical systems in two main classes (Marshall 1920):

- The classical approaches based on a rational choice of sets of equations representing the state of the economy. The drawbacks from this approach are: the unrealistic assumption of rational choices and perfect information; assumptions on homogeneity and lack of distinction between the agent and the aggregate level; inability to account for the emergence of new kinds of entities, patterns, structures, etc.
- Complexity perspective approach. This approach is more recent as a view of economy. It focusses on viewing the economy as an evolving complex system, using Holland's complex adaptive systems paradigm (Holland 1975).

With the complexity approach, the aim is to understand how global economic phenomena arise purely from the local interactions and local knowledge of the agents. This stems from the recognition that there is no central or global control in an economy and that the global regularities which arise are purely due to the local interactions of adaptive autonomous agents. The agents within an economic network may be individuals or institutions. We can consider an institution agent as being composed of many individual agents.

Feed-Backs Modelling

W. Brian Arthur (Arthur 1999), and other economic theorists have been interested to a view of the economy based on feedbacks, more precisely, positive feedback mechanisms. We can say that modern economies can be divided into two interrelated branches :

- Diminishing Returns and Negative Feedback.

  Diminishing returns refers to the notion that the return that a company receives for additional effort decreases as the number of units (outputs) increases. Diminishing returns explains why industrial companies become more inefficient once they grow over a certain size.

  Negative feedback is the process of feeding back to the input a part of a system's output, so as to reverse the direction of change of the output. This tends to keep the output from changing, so it is stabilizing and attempts to maintain constant functionning conditions.

Diminishing returns assumptions are the basis of the conventional economic theory. The actions of the economy induces a negative feedback which leads to predictable equilibrium for prices and market share. Negative feedback stabilizes the economy because any major change will be cancelled out by the reaction they generate. Diminishing returns imply a unique equilibrium point for the economy.

Alfred Marshall (Marshall 1920) believed that we can not applied the increasing returns everywhere in the economy. His remark leads W. Brian Arthur to observe that the part of the economy that are resource-based (the traditional part) are still subject to diminishing returns, on one hand. And the part of the economy that are knowledge-based (the newer part) are still subject to increasing returns, on the other hand as defined in the following.

- Increasing Returns and Positive Feedback

Increasing returns refer to the notion that the greater the size of the network, the greater the advantage of each participant of the network (network effects). Each participant of the network brings value to the overall network. This is in contrast to diminishing returns which refers to the greater the size (number of users) the less each participant can benefit from participation. Increasing returns generate instability.

We call a feedback mechanism positive if the resulting action goes in the same direction as the condition that triggers it. Positive feedback is an open system contain many types of regulatory systems, among which are systems that involve positive feedback and its relative negative feedback.

Increasing returns make for multiple solutions (multiple equilibrium points), and it generate instability or criticality stability following Per Bak (Bak 1996).

After describing the scientific environment of complexity, we give in the next section formal tools allowing to model some auto-regulation processes leading to implement the feed-back mechanisms.

AUTOMATA-BASED SYSTEM MODELLING

We focus our attention on a special kind of automata with outputs which are efficient in operational way (Hopcroft et al. 2001). This automata with output are called automata with multiplicities and they are defined in the following.

Automata with multiplicites

An automaton with multiplicities is based on the fact that the output data of the automata with output belong to a specific algebraic structure, a semi-ring. In that way, we will be able to build effective operations on such automata, using the power of the algebraic structures of the output data. And we are also able to describe this automata in matrix representation with all the power of the linear algebra.

Definition 0.1 *An* **automaton with multiplicities** *over an alphabet* $\sum$ *and a semi-ring* K *is the 5-uple* Σ, Q, I, T, δ *where*

- $Q = \{S1, S2, ..., Sn\}$ *is the finite set of state*
- $I : Q \rightarrow K$ *is a function over the set of initial states, which associates to each initial state a value of K, called entry cost, and to non-initial state a null value*
- $T : Q \rightarrow K$ *is a function over the set of the final states, which is associated to each final state a value of K, called final cost, and to non-final state a null value*
- δ *is the transition function, that is* $\delta : Q \times \Sigma \times Q \rightarrow K$ *which from a state S1 , a letter a and a state S2 go to a value z of K if it exist a transition labeled with a from the state S1 to the state S2 and with the output z.*

We remark that

- automata with multiplicities is a generalization of finite automata. In fact, finite automata can be considered as automata with multiplicities with for the semi-ring K, the boolean set $B = \{0, 1\}$. To each transition we affect 1 if it exists and 0 if not.
- We have not yet, on purpose, defined what a semi-ring is. Roughly it is the least structure $(K, +, x)$ that allows matrix computation with units (one can think of a ring without the **minus** operation)

Linear Representation

The previous automata with multiplicities can be expressed by a linear representation which is a triplet

$$p = (\lambda, \mu, \gamma)$$

with $\lambda \in K^{1\times n}$ is a row-vector which coefficients are $\lambda_i = I(S_i)$, $\gamma \in K^{n\times 1}$ is a row-vector which coefficients are $\gamma_i = T(S_i)$, and $\mu : \Sigma \rightarrow K^{n\times n}$ is a morphism of monoids such that $\forall a \in \Sigma$, the coefficient on the i-th row and j-th column of all transitions labeled with a is $\mu(a)_{ij} = \delta(S_i, A, S_j)$.

In the following, we describe the linear representation of the automata with multiplicities which corresponds to the the following figure.

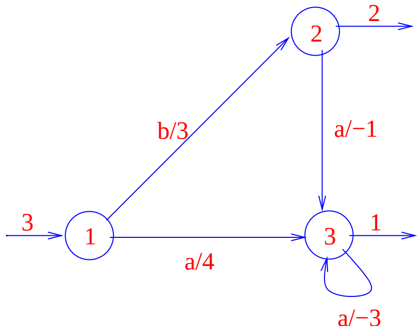

Figure 1: Automata with multiplicities

In this figure, the input states are characterized with input arrows from nothing and the output states are characterized with output arrow to nothing. On each transition, we indicate before the semi-column, the input data and after the semi-column, the output associated for each input data.

The linear representation is the following

$$\lambda = \begin{pmatrix} 3 & 0 & 0 \end{pmatrix}$$

$$\gamma^t = \begin{pmatrix} 0 & 2 & 1 \end{pmatrix}$$

$$\mu(a) = \begin{pmatrix} 0 & 0 & 4 \\ 0 & 0 & -1 \\ 0 & 0 & -3 \end{pmatrix}$$

$$\mu(b) = \begin{pmatrix} 0 & 3 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

GENETIC AUTOMATA

Genetic algorithms play an important role in the studies of complex adaptive systems, ranging from adaptive agents in economic theory to the use of machine learning techniques. We describe the principles of these algorithms and then their application to genetic automata.

Genetic Algorithms and Operations

The major aspect of genetic algorithms is the adaptation property. This property can be considered as the search of the optimum of a specific function. Genetic algorithms work on a population of individuals. The individuals are represented with chromosomes which are composed with primitive information called alleles. In term of computable formalization, chromosomes are generally strings or sequences of information over a finite alphabet. To begin the algorithm, we generate a population of chromosomes.

The algorithm is based on a variation-selection process:

- The variation step concerns the basic genetic operators on the individual level and so this step acts on the chromosomes. These basic operators are composed of reproduction/duplication, crossing-over and mutation and we describe how implement them for genetic automata in the following section. The result gives another population with a greater number of chromosomes than the initial one
- The selection step concerns the population level inside which a selection function modifies the whole population constitution. This step leads to keep only some of the chromosomes that have been generated during the variation step and which satisfy to specific constraints

Automata Chromosome Coding and Genetic Operators

We define the chromosome for each automata with multiplicities as the sequence of all the matrices associated to each letter from the (linearly ordered) alphabet. The chromosomes are composed with alleles which are here the lines of the matrices.

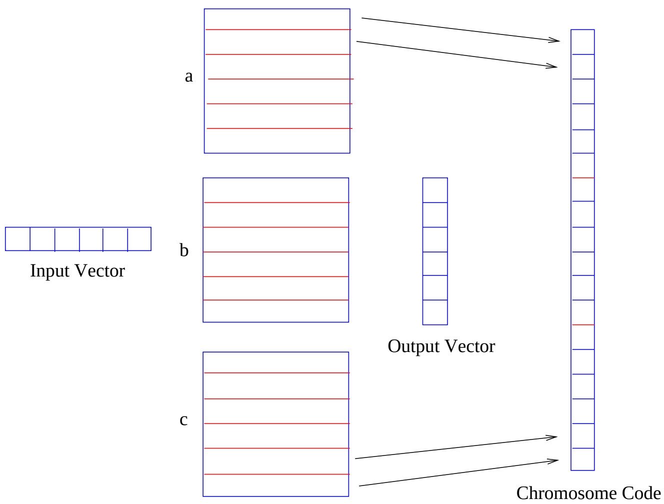

Figure 2: Chromosome code

In the processus, genetic algorithms generate new automata containing possibly new transitions from the ones included in the initial automata. The genetic operators are applied as described in the following:

- Reproduction/duplication: this operator simply copy the automaton chromosome.

- Crossing-over: From 2 chromosomes, it consists in cutting of the initial chromosomes in the same place chosen at random and in permuting the two parts of the matrices lines chains.

- Mutation : In this step, we chose a few chromosomes at random to be candidate for the mutation process. If selected, we modify one line of the matrix lines chromosome. If the matrix lines have to respect some properties (like probabilistic matrix) we constraint the new values to respect them.

EXPERIMENTS ON DIFFUSIVE PROPERTY CONTROL

We present in the following some experiments based on the package MuPAD-Combinat which implements many data structures related to combinatorics (Hivert and Thierry 2004). The genetic operators are developped on the predefined domain called Weighted Automaton from this package.

Our experiment consists to define chromosome code for probablitistic automata which are specific automata with multiplicities satisfying the condition that the sum of each lines coefficient must be equal to 1.

Fitness Function and Auto-Regulation Process Based on Spectral Analysis

We define a population of genetic probabilistic automata generated at random and we compute for each matrix, the secondary greatest eigenvalue module (the first one is always equal to 1, because of the probablitistic matrices used, respecting Perron-Frobenius Theorem). The genetic algorithm is described in 1.

Experiment on non Diffusive Systems Control

In the figure 3, we represent a series of histograms computed at different steps where we indicate the number of automata according to their secondary greatest eigenvalue module. With the evolution of the population, we observe that the selection will generate population with chromosome of high secondary greatest eigenvalue module.The x-scale is adaptive to the interval containing all the values of this secondary greatest eignevalue for the automata population.

Algorithm 1: Auto-regulation algorithm for genetic automata population

```
Generation of P the initial population of probabilistic automata chromosome;
while not end of iterations do
    for each chromosome c ∈ P do
        Duplicate c;
        Update P;
    for each couple of chromosomes (c1, c2) ∈ P do
        Make crossing-over on (c1, c2);
        Make mutation on c1 and c2 with low random probability;
        Update P;
    for each chromosome c ∈ P do
        Compute the fitness of c as the secondary greatest eigenvalue module;
    Select from P the half higher fitness population and update P with this selection;
```

Results analysis and interpretation

As expected by the fitness function, we generate successive populations which are characterized by the convergence process which leads to obtain a population where the greatest secondary eigenvalue module of all automata with multiplicities is equal 1.

This processus is an auto-regulation processus on the population itself, managed by a feed-back of the result control on the system itself.

By controlling the generating automata population with eigenvalues module equals 1, we constraint the system to not be dissipative. The spectral analysis and the eigenvalues module give some indications on the dissipative aspect during the evolution. If the eigenvalue modules are less than 1, then we can conclude that a dissipation exists.

The non dissipative evolution process over automata with multiplicities proposed here, is a tool based on the power of the algebraic representation of the automata with multiplicities, using some computation on their matrix representation. So we deal on a dynamic data structure according to the complexity of dissipative process over some self-organized systems. Without managing each matrix coefficient value, we give a kind of control tool according to dissipative process increasing or decreasing.

CONCLUSION AND PERSPECTIVES

In this paper, we present a methodology which allows to control a complex system by a global function which is based on the use of a genetic probabilistic automata population. We scan this population by means of a spectral analysis and control the system by reducing its diffusive properties. The applications to economy can be relevant and we propose by this method, a practical

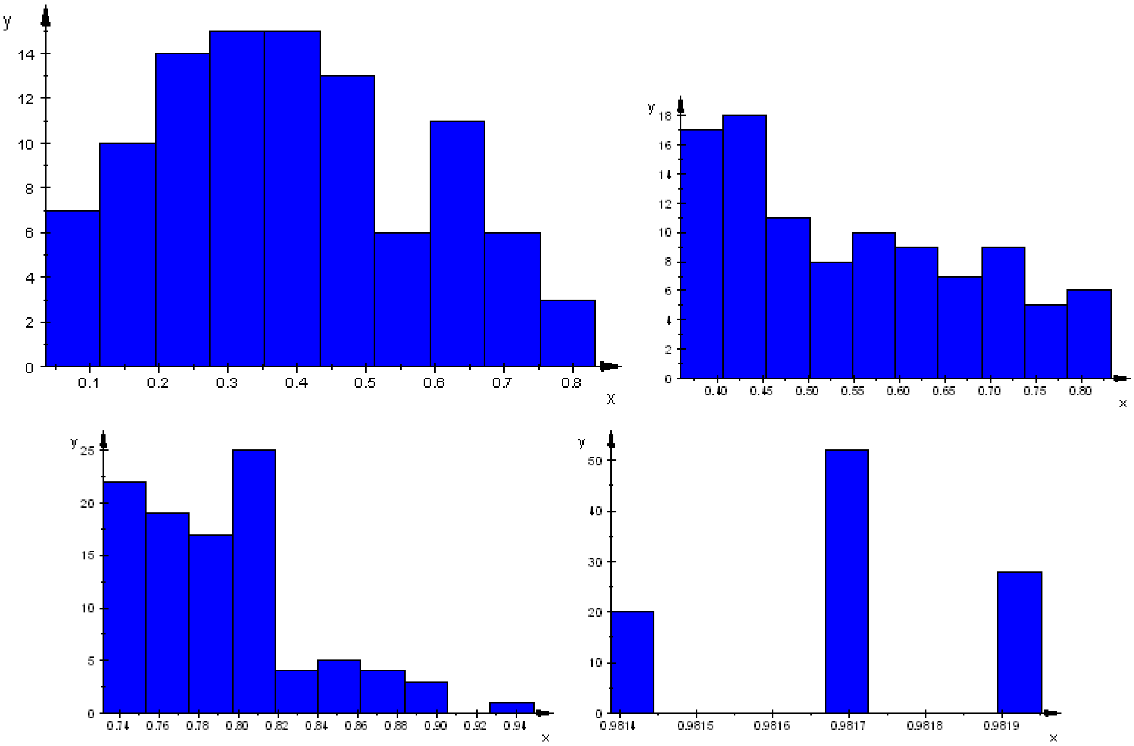

Figure 3: Histograms of the evolution of automata population according to their secondary greatest eignevalue module at iteration 1, 10, 20, 30 and 40

and computable way to implement the feed-back mechanisms over systems in economy.

REFERENCES

Arthur B., 1999. *Complexity and the Economy. Science*, 284, 107–109.

Bak P., 1996. *How nature works - the science of self-organized criticality.* Springer Verlag.

Hivert F. and Thierry N.M., 2004. *MuPAD - Combinat, an open-source package for research in algebraic combinatorics. Seminaire Lotharingien de Combinatoire*, 51.

Holland J., 1975. *Hidden Order - How adaptation builds complexity.* Helix Book.

Hopcroft J.; Motwani R.; and Ullman J., 2001. *Introduction to automata theory, languages and computation.* Addison-Wesley.

Jaff L., 2007. *Dynamic Data Structures for Complex Systems.* Ph.D. thesis, University of Le Havre.

Kline D., 2001. *Positive feedback, lock-in and environmental policy. Policy Sciences*, 34, 95–107.

LeMoigne J.L., 1999. *La modélisation des systèmes complexes.* Dunod.

Marshall A., 1920. *Principles of Economics.* MacMillan.

Schutzenberger M., 1961. *On the definition of a family of automata. Information and Control*, 4, 245–270.