

K. Khatatneh, R. Ghneamat, S. Oqeili Al-Balqa'  
Applied University - Al-Salt, Jordan

# AUTOMATA-BASED ADAPTIVE BEHAVIOR FOR ECONOMICAL MODELING USING GAME THEORY

*C. Bertelle*

LIH - University of Le Havre - France

*G. H. E. Duchamp*

LIPN - University of Paris XIII - France

## ABSTRACT

todo ?

### 1. Introduction

In this paper, we deal with some specific domains of applications to game theory. The latter is one of the major class of models in the new approaches of modelling in the economical domain. First, we explain how adaptive strategies are useful concerning these domains in the next section. Then, we present in section 3, some efficient algebraic structures, the automata with multiplicities, which allow to implement powerful operators. Such algebraic structures are used as the basis of evolutionary algorithms and we explain in the section 4 how genetic automata are well-adapted for adaptive strategies modeling. In section 5, we focus our attention on the "iterated prisoner dilemma" and we build an original evolutive probabilistic automaton for strategy modeling.

### 2. Adaptive behaviour modeling for game theory

Game theory has become since the five last decades a major aspect in economical sciences modelling and in a great number of domains where strategical aspects has to be involved. Game theory is usually defined as a mathematical tool allowing to analyse strategical interactions between individuals. Initially funded by mathematical researchers, J. von Neumann, E. Borel or E. Zermelo in 1920s, game theory increased in importance in the 1940s with a major work from J. von Neumann and O. Morgenstern and then with the works of John Nash in the 1950s. John Nash has proposed an original equilibrium obtained with adaptive process. In game theory, the Nash equilibrium is a kind of optimal strategy for games involving two or more players, whereby the players reach an outcome to mutual advantage. If there is a set of strategies for a game with the property that no player can benefit by changing his strategy while the other players keep their strategies unchanged, then this set of strategies and the corresponding payoffs constitute a Nash equilibrium. We can understand easily that the modelization of player behavior needs some adaptive properties. The computable model corresponding

to genetic automata are in that way a good tool to modelize such adaptive strategy.

(to be completed with augmented references: R. Axelrod, Ph. Mathieu from LIL-France for exemple)

### 3. Automata from boolean to multiplicities theory (Automata with scalars)

(Here we have to talk about the power of semiring as scalars operators ... )

Automata are initially considered as theoretical tools. They are created in the 1950's following the works of A. Turing who previously deals with the definition of an abstract "machine". The aim of the Turing machines is to define the boundaries of what a computing machine could do and what it could not do.

The first class of automata, called finite state automata corresponds to simple kinds of machines. They are studied by a great number of researchers as abstract concepts for computable building. In that aspect, we can recall the works of some linguist researchers, for example N. Chomsky who defined the study of formal grammars.

In many works, finite automata are associated to an operator which allow to describe a language. In such works, the condition of a transition is simply a symbol from an alphabet. From a specific state  $S$ , the reading of a symbol  $a$  allows to make the transition which is labeled by  $a$  and come from  $S$ . A whole automaton is in that way, associated to a language, the recognized language, which is the set of words. These recognized words are composed of the sequence of letters of the alphabet which allows to go from a specific state called initial state, to another specific state, called final state.

A first classification is based on the geometric aspect : DFA (Deterministic Finite Automata) and NFA (Nondeterministic Finite Automata)

- In deterministic automata, for each state there is almost one transition for each possible input and only one initial state.
- In non-deterministic automata, there can be none or more than one transition from a given state for a given possible input.

Besides the classical aspect of automata as machines allowing to recognize languages, another approach con-

sists in associate to the automata a functional goal. In addition of accepted letter from an alphabet as condition of transition, we add for each transition an information which can be considered as an output data of the transition, the read letter is now called input data. We define in such a way an automaton with outputs or weighted automaton.

Such automata with outputs give a new classification of machines. Transducers are some kind of machines, they generate outputs based on a given input and/or a state using actions. They are currently used for control applications. Moore machines are also such machines where output depends only on a state, i.e. the automaton uses only entry actions. The advantage of the Moore model is a simplification of the behaviour.

Finally, we focus our attention on a special kind of automata with outputs which are efficient in operational way. This automata with output are called automata with multiplicities. 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 semiring. 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 automaton by means of a matrix representation with all the power of linear algebra.

**Definition (Automaton with multiplicities):** An automaton with a multiplicities over an alphabet  $A$  and a semiring is the 5-uple  $(A, Q, I, T, K)$  where

- $Q = \{S_1, S_2 \dots S_n\}$  is the finite set of state;
- $I : Q \mapsto K$  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 zero value ;
- $F : Q \mapsto K$  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 zero value;
- $T$  is the transition function, that is  $T : Q \times A \times Q \mapsto K$  which to a state  $S_i$ , a letter  $a$  and a state  $S_j$  associates a value  $z$  of  $K$  if it exist a transition labelled with  $a$  from the state  $S_i$  to the state  $S_j$  and and zero otherwise.

**Remark 1** *i) Automata with multiplicities is a generalisation of finite automata. In fact, finite automata can be considered as automata with multiplicities with for the semiring  $K$ , the boolean set  $B=0,1$ . To each transition we affect 1 if it exists and 0 if not.*

*ii) We have not yet, on purpose, defined what a semiring is. Roughly it is the least structure that allows matrix computation with units (one can think of a ring without the "minus" operation).*

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

- $\lambda \in K^{1 \times Q}$  which is a row-vector which coefficients are  $\lambda_i = I(S_i)$ ,
- $\gamma \in K^{Q \times 1}$  is a row-vector which coefficients are  $\gamma_i = F(S_i)$ ,
- $\mu : A^* \mapsto K^{Q \times Q}$  is a morphism of monoids (indeed  $K^{Q \times Q}$  is endowed with the product of matrices) such that the coefficient on the  $q_i$ th row and  $q_j$ th column of  $\mu(a)$  is  $T(q_i, a, q_j)$

( to be continued : add a sample of automaton with associated linear representation)

#### 4. Topological considerations

If  $K$  is a field, one sees that the space  $\mathcal{A}_{(n)}$  of automata of dimension  $n$  (with multiplicities in  $K$ ) is a  $K$ -vector space of dimension  $k.n^2 + 2n$  ( $k$  is here the number of letters). So, in case the ground field is the field of real and complex numbers, one can take any vector norm (usually one takes one of the Hölder norms  $\|(x_i)_{i \in I}\|_\alpha := (\sum_{i \in I} |x_i|^\alpha)^{\frac{1}{\alpha}}$  for  $\alpha \geq 1$ , but any norm will do) and form the distance in the classical way by

$$d(\mathcal{A}_1, \mathcal{A}_2) = \text{norm}(V(\mathcal{A}_1) - V(\mathcal{A}_2)) \quad (1)$$

one has then the result of Theorem 1. Assuming that  $K$  is the field of real or complex numbers, we endow the space of series/behaviours with the topology of pointwise convergence (Topology of F. Treves [16]).

**Theorem 1** *Let  $(\mathcal{A}_n)$  be a sequence of automata with limit  $\mathcal{L}$  ( $\mathcal{L}$  is an automaton), then one has*

$$\text{Behaviour}(\mathcal{L}) = \lim_{n \rightarrow \infty} \text{Behaviour}(\mathcal{A}_n) \quad (2)$$

where the limit is computed for the topology of Treves.

**Remark 2** *i) The definition of relevant distances over spaces of automata is crucial for the study of evolution and for probabilistic description [19]. We will return to this notion in a forthcoming study.*

*ii) The reader should be aware that, even if the alphabet is finite (non void however), the space of series endowed with Treve's topology is a metric (Fréchet) space which cannot be normed.*

#### 5. Genetic automata as efficient operators

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

In the following, genetic algorithms are going to generate new automata containing possibly new transitions from the ones included in the initial automata.

The genetic algorithm over the population of automata with multiplicities follows a reproduction iteration broken up in three steps:

- Duplication where each automata generates a clone of itself;
- Crossing-over concerns a couple of automata. Over this couple, we consider a sequence of lines of each matrix for all . For each of these matrices, a permutation on the lines of the chosen sequence is made between the analogue matrices of this couple of automata.
- Mutation where a line of each matrix is randomly chosen a sequence of new values is given for this line.

Finally the whole genetic algorithm scheduling for a full process of reproduction over all the population of automata is the evolutionary algorithm:

1. For all couple of automata, two children are created by duplication, crossover and mutation mechanisms;
2. The fitness for each automata is computed;
3. For all 4-uple composed of parents and children, the performless automata, in term of fitness computed in previous step, are suppressed. The two automata, still living, result from the evolution of the two initial parents.

**Remark 3** *The fitness is not defined at this level of abstract formulation, but it's defined corresponding to the context for which the automata is a model.*

## 6. Applications to competition-cooperation modeling using prisoner dilemma

### 6.1. From adaptive strategies to probabilistic automata

The prisoner dilemma is a two-players game where each player has two possible actions: cooperate (C) with its adversary or defect it (D). So, four outputs are possible for the global actions of the two players. A relative payoff is defined relatively to these possible outputs, as described in the following table where the rows correspond to one player behaviour and the columns to the other player one.

In iterative version of the prisoner dilemma, successive steps can be defined. Each player do not know the action of its adversary during the current step but he knows it for the preceding step. So different strategies can be defined for a player behaviour, the goal of each one is to obtain maximal payoff for himself. In the illustrations 1 and 2, we describe two strategies with transducers. Each transition is labeled by the input corresponding to the player perception which is the precedent adversary action and the output corresponding to the present player action. The only initial state is the state 1, recognizable by the incoming arrow labeled only by the output. The final states are the states 1 and 2, recognizable with the double circles. In strategy of the illustration 1, the player has systematically the same behaviour as its adversary at the previous step. In strategy

of the illustration 2, the player chooses definitively to defect as soon as his adversary does once. The previous automata represent static strategies and so they are not well adapted for the modelization of evolutive strategies. For this purpose, we propose a model based on a probabilistic automaton described by the illustration 3.

This automaton represents all the two-states strategies for cooperation and competitive behaviour of one agent against another in prisoner dilemma. The transitions are labeled in output by the probabilities  $p_i$  of their realization. The first state is the state reached after cooperation action and the second state is reached after defection.

For this automaton, the associated linear representation, as described previously, is:

$$I = (p_1 \quad 1 - p_1); F = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad (3)$$

$$T(C) = \begin{pmatrix} p_2 & 1 - p_2 \\ p_3 & 1 - p_3 \end{pmatrix}; T(\bar{C}) = \begin{pmatrix} p_4 & 1 - p_4 \\ p_5 & 1 - p_5 \end{pmatrix}$$

### 6.2. From probabilistic automata to genetic automata

With the linear representation of the automata, we can compute genetic automata as described in previous section. Here the chromosomes are the sequences of all the matrices associated to each letter. We have to define the fitness in the context of the use of these automata. The fitness here is the value of the payoff.

### 6.3. General GA process for genetic automata

A population of automata is initially generated. These automata are playing against a predefined strategy, named SO.

Each automaton makes a set of plays. At each play, we execute the probabilistic automaton which gives one of the two outputs: C or D. With this output and the SO's output, we compute the payoff of the automaton, according with the payoff table.

At the end of the set of plays, the automaton payoff is the sum of all the payoffs of each play. This sum is the fitness of the automaton. At the end of these set of plays, each automaton has its own fitness and so the selection process can select the best automata. On the end of these selection process, we obtain a new generation of automata.

This new generation of automata is the basis of a new computation of the 3 genetics operators.

(to be completed ... we can add a discussion about prisoner dilemma and how that can be used in economical modeling : firm concurrence, cold war, ...)

## 7. Conclusion

(to be continued)

## 8. References

- [1] David E. Goldberg, *Genetic Algorithms*, Addison-Wesley, 1989
- [2] Melanie Mitchell, *An introduction to Genetic Algorithms*, The MIT Press, 1996
- [3] John H. Holland, *Hidden Order - How adaptation builds complexity*, publisher 1995
- [4] J.E. Hopcroft, R. Motwani, J.D. Ullman, *Introduction to automata theory, Languages and Computation*, Addison-Wesley, 2001
- [5] R. Axelrod, *The complexity of cooperation*, Princeton University Press, 1997
- [6] I. Rechenberg, *Evolution strategies*, Fromman-Holzboog, 1973
- [7] L.J. Fogel, A.J. Owens, M.J. Welsh, *Artificial intelligence through simulated evolution*, John Wiley, 1966
- [8] S. Eilenberg, *Automata, languages and machines*, Vol.A et B, Academic press, 1976
- [9] R.P. Stanley, *Enumerative combinatorics*, Cambridge University Press, 1999
- [10] J.S. Golan, *Power algebras over semirings*, Kluwer Academic Publishers, 1999
- [11] J. Berstel and G. Reutenauer, *Rational series and their language*, EATCS, 1988
- [12] John Koza, *Genetic programming*, Encyclopedia of Computer Sciences and Technology, 1997
- [13] C. Bertelle, M. Flouret, V. Jay, D. Olivier, and J.-L. Ponty "Adaptive behaviour for prisoner dilemma strategies based on automata with multiplicities." In *ESS 2002 Conf., Dresden* (Germany), October 2002.
- [14] C. Bertelle, M. Flouret, V. Jay, D. Olivier, and J.-L. Ponty "Genetic algorithms on automata with multiplicities for adaptive agent behaviour in emergent organizations" In *SCI'2001*, Orlando, Florida, USA, 22-25th July 2001
- [15] M.P. Schutzenberger, "On the definition of a family of automata", *Information and Control* 4, 245- 270 (1961)
- [16] F. Trèves, *Topological Vector Spaces, Distributions and Kernels*, Acad. Press 1967. rajouter les refs GHED
- [17] G. Duchamp, Hatem Hadj-Kacem and Eric Laugerotte, "Algebraic elimination of  $\epsilon$ -transitions", *DMTCS, Volume 7 n 1* (2005), pp. 51-70
- [18] G. Duchamp, J-M Champarnaud, *Derivatives of rational expressions and related theorems*, *Theoretical Computer Science* **313** (2004).
- [19] Duchamp G., Flouret M., Laugerotte É., Luque J-G., *Direct and dual laws for automata with multiplicities* T.C.S. **267**, 105-120 (2001).