Scientific Research of the Institute of Mathematics and Computer Science, 1(10) 2011, 205-213

THE ASSOCIATION AUTOMATA AS A MECHANISM FOR MEMORY ORGANIZATION

Henryk Piech

Czestochowa University of Technology, e-mail: hpiech@adm.pcz.czest.pl

Abstract. The association process is one of man's attitudes of mind and is connected with its functions. It is connected with retaining, remembering and forgetting. It also refers to the stages of attitude of mind. The sequence of associations is adequate for the process of analysing different possibilities of finding a solution. We can find many analogies to practical situations regarding current conditions and limits. We want to propose a mechanism for automatic reactions to environment changes. In this case, our automaton should react not only upon a known situation structure but also create new associations with their properties (attributes). We assume that all perceptions can be sent to all so-called nodes of association. Every association is represented by its name. Every association is evoked by a set of generators based on different sets of attributes. We introduce several new notations such as background, depth, domination or profile associations. The proposed automata can be implemented in transporting, routing, military and security strategies.

Introduction

Association is a function of retaining, remembering and forgetting. Let us assume that we return to our experiences, connected with current observations, leading us to find in our memory past events, objects, relations, feelings etc. [1, 2]. We say that a current observation associates with past situations, seen objects. These elements are the so-called background of current associations. We cannot say which objects exactly (it is usually a set of objects) generate a given association. Hence, we only assume in the proposed simplified model that it will be a conjunction of perceived elements. Hence the remembering function (intentional or unintentional searching in our memory) would be treated as a conjunction result of present perceptions. They consist of current observations and "pictures" accumulated in the background (current remembered elements):

$$Pa = Pc \cup PB \tag{1}$$

where: Pc - current observations set, PB - accumulated perceptions and associations in the background (in close and small auxiliary memory).

Obviously, the association background can be enriched with new associations, which cannot be current observations and simultaneously improver is had by forgotten elements

$$PB = PB \cup Pat \cup NS / ZS(2) \tag{2}$$

where: $NS = Pp \cup Pg$ - new associations set, *Pat* - perception attributes set, *Pp* - perception association set, *Pg* - generated associations set, *ZS* - forgotten associations set.

Let us assume that association consists of attributes and a name (or set of names) $S(Nm) = \cup At \cup Nm$, where: *At* - set of association attributes, *Nm* - set of association names (sometimes used interchangeably with given association). As we show, names (associations) can be treated as attributes.

Generally, a set of attributes leads to the generation of the association that means to join them with the association names. For example, a set of attributes (unlock, cipher, signature) can generate the association (key) which we write (unlock, cipher, signature) →(key). As a rule, several or even many sets of attributes generate the same association. A full set of attributes used to generate a given association can be used to create subsets being the base for evoking the name(s) of an association. If a new association is introduced, all its attributes and names are sent to the background and then to the main memory (process of remembering). When the name of an association and if the attributes connected with in the generator structure are remembered, retaining or refreshing processes related to it can be exploited. In natural memory systems, what is especially annoying is the life of associations in the main memory (or troubles retaining). The associations life depends on the retaining and refreshing frequency and also indirectly on the numbers of attributes and generators. One of the main disadvantages (in the organizing point of view it can be treated as an advantage) of natural memory systems is the domination of one association which effectively depreciates the others (diminished recognition of them) in a given moment. In this case, we can use the notation "concentration" which means the possibility of avoiding the other association influences.

In the practice of modeling, we can propose an approach connected with the capacity of memory based on providing a stable (approximatelly constant) level of present perceptions, the number and maximal depth of associations (number of stages of joining associations):

$$MP = Pa * DSmax = const \tag{3}$$

where: *MP* - stable product, *DSmax* - maximal depth of association.

In the related association model, we propose concentration on the dominant association, that means the one with maximal depth or maximal number of generators.

1. Association automaton structure

Association automaton is a net which can be described by a 3-tuple

$$AS = \{S,P,B\}$$

where: *WS* - association nodes set, *P* - perceptor, *B* - association background.

The net has connections between the preceptor and all the nodes and the background. Every node can be represented by a 3-tuple [3, 4] describing its structure and states of life:

$$WS(i) = \{\Sigma(i), G(i), T(i)\} \tag{4}$$

where: $\Sigma(i)$ - alphabet of i-th association attributes, $G(i)$ - association generators, $T(i)$ - time function of association disappearance.

The idea scheme of an association automaton is presented in Figure 1. The situation where a preceptor is connected in a direct way and the background is connected in an indirect way with all the association nodes is difficult to clearly present. Obviously, the nodes are connected among them an indirect way. Hence, we present this complicated system of connections with overlapping areas representing specified structure elements (layers). The direction of activation of these layers is described by the following implication form:

$$P \rightarrow B,\ P \rightarrow WS(i),\ B \rightarrow WS(i),\ WS(i) \rightarrow WS(j),\ i,j = 1,2,\ldots n \tag{5}$$

The overlapping layers symbolize mutual connection among the elements belonging to these areas.

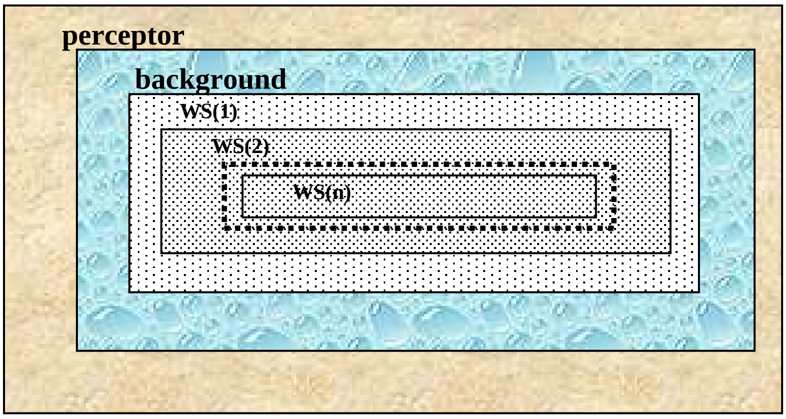

Fig. 1. Idea scheme of association automata (dotted zones contain elements with undirected communication with elements on setting zone)

The alphabet of a given association is a set containing attributes and names of associations which play the role of attributes for this association: $\Sigma = \cup At \cup Nm = \{at(1), at(2), \ldots, at(r(i)), nm(1), nm(2), \ldots, nm(s(i))\}$, where: $r(i)$ - attributes number, and $s(i)$ - number of all associated names participating in generation of given association. Simultaneously, the alphabet is a set of arguments for the longest generator (according to its structure) for a given association. In short, generators with a different length generate a defined association, for instance:

$gg(1)=\{at(4)\wedge\ nm(3)\wedge\ nm(7)\wedge\ nm(12)\}$; $gg(2)=\{at(22)\wedge\ nm(3)\}$ and we add to them descriptions $g(1)=\{desert\wedge\ wind\wedge\ Jerycho\ \wedge\ China\}$, $g(2)=\{fragrance\wedge\ wind\}$ which evoke the same association: {rose}. Therefore, we have two generators:

1) $g(1)=\{desert \wedge wind \wedge Jerycho \wedge China\}\rightarrow$\{rose\},
2) $g(2)=\{fragrance \wedge wind\}\rightarrow$\{rose\}.

The field of association is a product of the alphabet element number and generator number. Its value influences the domination coefficient of a given association (Fig. 2):

$$FS(i) = |\Sigma(i)|*\lg(i)$$

where: $|\Sigma(i)|$ - alphabet elements number, $lg(i)$ - generator number for i-th association.

Association arguments are activated as a result of the last preceding association and recent perceptions. It is supplemented by elements from the background. In Figure 2 in the right box only the third generator (in third row) evoked a given association.

We usually determine the domination coefficient *DC*. According to our assumption, the motivations for introducing this coefficient are as follows:

– to find the most important according to depth association
– to determine the moment of the stop simulation process

To define *DC*, we can use parameter *MP*. Adding auxiliary parameters creates threshold *TDC*, which when exceeded stops the simulation process.

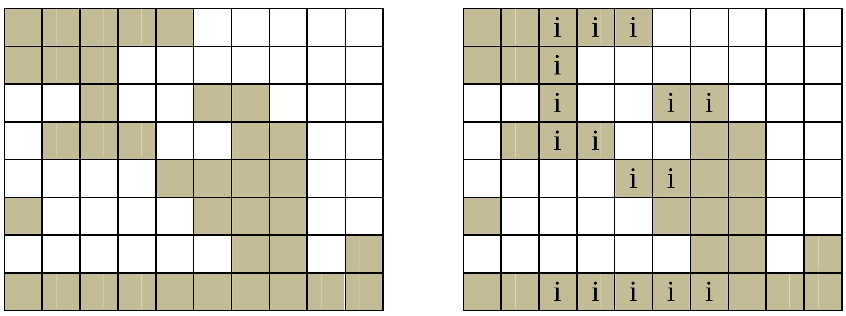

Fig. 2. Association field, where rows are equal to generators, columns (shadowed fields) to generation arguments (attributes and previous associations). The letter „i" denotes current arguments activation. Last row refers to "the longest" generator based on full alphabet

Coming back to generators, we should say that they realized a common conjunction of based elements chosen from the alphabet. Hence, the problem of ordering these elements does exist. Sometimes even a single argument creates a full base for one generator. The strength of the argument in a single generator can be defined as $1/la(i,j)$, where $la(i,j)$ - number of argument in j-th generator for i-th association. Argument activations take place when they are generated as previous associations (come from association background) or they are sent from the preceptor: $\Sigma(1),G(1), NS(1) \rightarrow \Sigma(2),G(2),NS(2) \rightarrow \ldots \rightarrow \Sigma(i),G(i),NS(i) \rightarrow \ldots \rightarrow \Sigma(d),G(d), NS(d)$, where *NS* - new association, *d* - association depth.

2. Life of association - time is its evanescence

The interpretation of the forgetting processes in natural memory systems are treated differently. In the proposed model of association automaton, we differentiate two kinds of memory: short and long-term. Short term memory is connected with recent perceptions (present situation) and is located in the background. Long-term memory refers to nodes of associations. The time of association evanescence depends on the amount of memory refreshing and retention. How do we treat both of these notions? The main feature can be described as follows:

- intentional or unintentional character of refreshing operations and only intentional of retaining,
- short time distances between refreshing operations (or between perception and refreshing) and relatively long distances between retaining operations (or between remembering and retaining),
- refreshing subject is still contained in association background, and retaining is beyond the background.

We assume, maybe not correctly (because we do not know the real character of mining association retention processes), that the refreshing operation increases the association life and retaining decreases it. Hence, our proposition of a time function figure is as follows [5, 6]:

$$T(i) = c1 * e^{-d1*t} + \sum_{i=1}^{lo} c2 e^{-d2*t} - \sum_{i=1}^{lp} c3 e^{-d3*t} \qquad (6)$$

where: $c1,c2,c3,d1,d2,d3$ - coefficient of association in the case without refreshing or retaining (with number 1), only with refreshing (with number 2), and only with retaining (with number 3), *lo* - refreshing number, *lp* - retaining number.

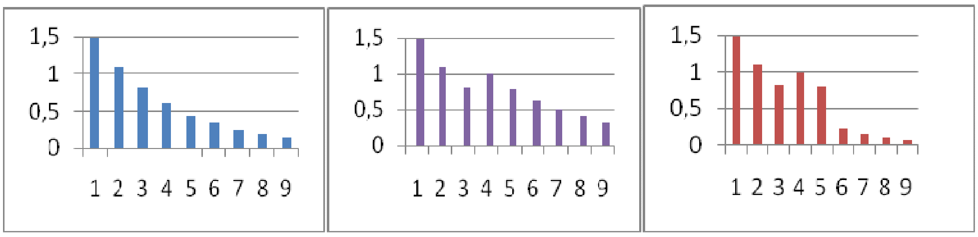

Fig. 3. Refreshing influence, in moment t = 4 (center diagram), and following after it retaining, in moment t = 6 (right diagram) on association life

3. Association evolution

Evolution usually has a quantitative and qualitative character. Perceptions and associations evoke new associations. The minimal number of elements creating a new association equals two: the attribute and name of the association. The attribute is the atomic part of the alphabet and generator. In our conception as in natural memory systems, the association name can play the role of an attribute. Quality evolution consists in adding new attributes to an existing association. It leads to the creation of new generators on the base of an enriched alphabet. By adding one attribute to every generator set, we obtain $m + 1$ new generators, according to the presentation in Figure 4, where: m - number of hitherto existing generators. In a simple variant, adding k - attributes, the number of generators increases by $dG = (k+1)m$.

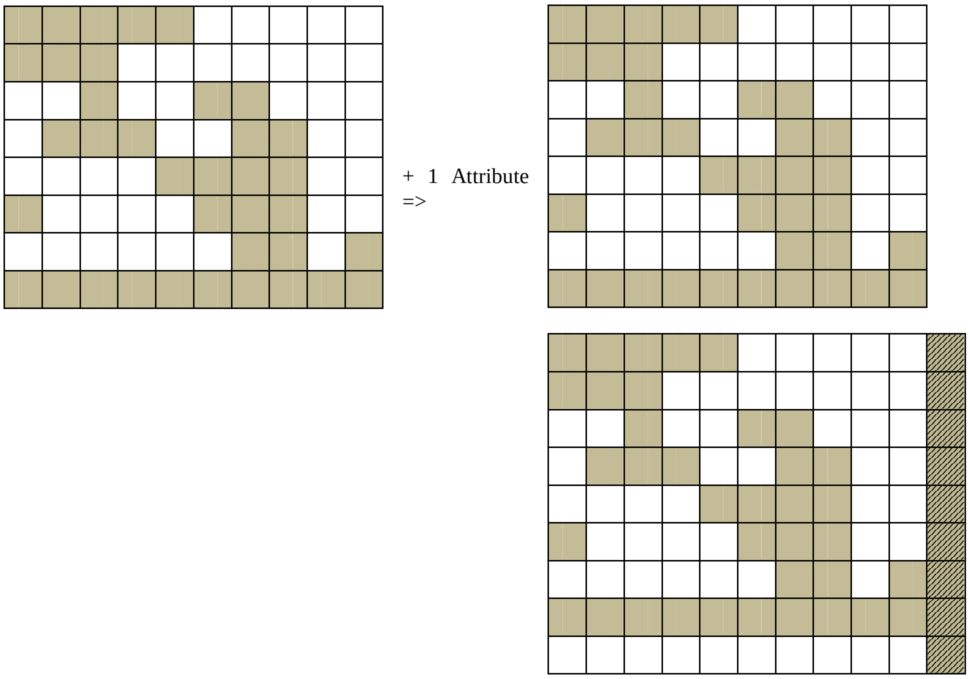

Fig. 4. Illustration of effect of adding one attribute to an association in aspect of generators structure. Existing generators remained without change but new generators are built by adding a new attribute to set of arguments. Last generator has only new attribute argument

A number generator increase can lead to the enlargement of the depths of associations and in the same case also the maximum depth of association. It can also lead to the development of the association domination and sometimes to increasing the number of them. The consequences of such phenomena lead to distraction, accidental inferring, chaos in the inferring mechanism or in short, to a wool-

-gathering state. Obviously, it causes changes in the probability distribution of association appointments [7].

This proposition is not full (in fact, it is a simplified variant) because we can create combination structures [8] of hitherto arguments (or association alphabet elements) and a new attributes.

Axiom: *An added attribute generating a new association also generates it with all the combinations with hitherto existing generator attributes of this association:*

$$\begin{gathered} komb\binom{n}{1} + \text{added attribute,} \\ komb\binom{n}{2} + \text{added attribute,} \\ \cdots\cdots\cdots\cdots \\ komb\binom{n}{n} + \text{added attribute,} \end{gathered} \tag{7}$$

where $komb\binom{n}{i}$ - *given association attributes set with length i.*

The number of generators enlarges on:

$$dG'(k=1) = \binom{n}{1} + \binom{n}{2} + ,\ldots, + \binom{n}{n} = \sum_{j=1}^{n} \binom{n}{j}, \tag{8}$$

or for $k > 1$

$$\begin{aligned} dG'(k>1) = &\binom{n}{1} + \binom{n}{2} + ,\ldots, + \binom{n}{n} + \binom{n+1}{1} + \binom{n+1}{2} + ,\ldots, + \binom{n+1}{n+1} +, \\ &\cdots\cdots\cdots\cdots \\ &+ \binom{n+k-1}{1} + \binom{n+k-1}{2} + ,\ldots, + \binom{n+k-1}{n+k-1} = \sum_{i=1}^{k} \sum_{j=1}^{n+i-1} \binom{n+i-1}{j} \end{aligned} \tag{9}$$

We should say that a single attribute can but does not have to determine the base of a new generator. Hence, we can treat *DG* as an interval variable *dGI* = [*kn*,*DG*], where *kn* - lower boundary of generator number increasing. The difference between the lower and upper boundary is relatively large and we should make an individual decision about a new generator structure. To simplify the problem of association generators structure automatization, we propose to exploit only the lower boundary of interval variable *dGI* = *kn* (Fig. 5) the more so as it does not collide with the above-defined axiom.

The automatization of quality evolution means enlarging the attribute number for a given association which requires selecting and defining a strictly defined convention. If we chose an interactive mode with an individual regarding every generator structure creation, it leads to diverging away from the automata conventions. In this case, we can attain perception dispersal, leading to attribute separation. Each of them would be adopted to the existing generators individually [9].

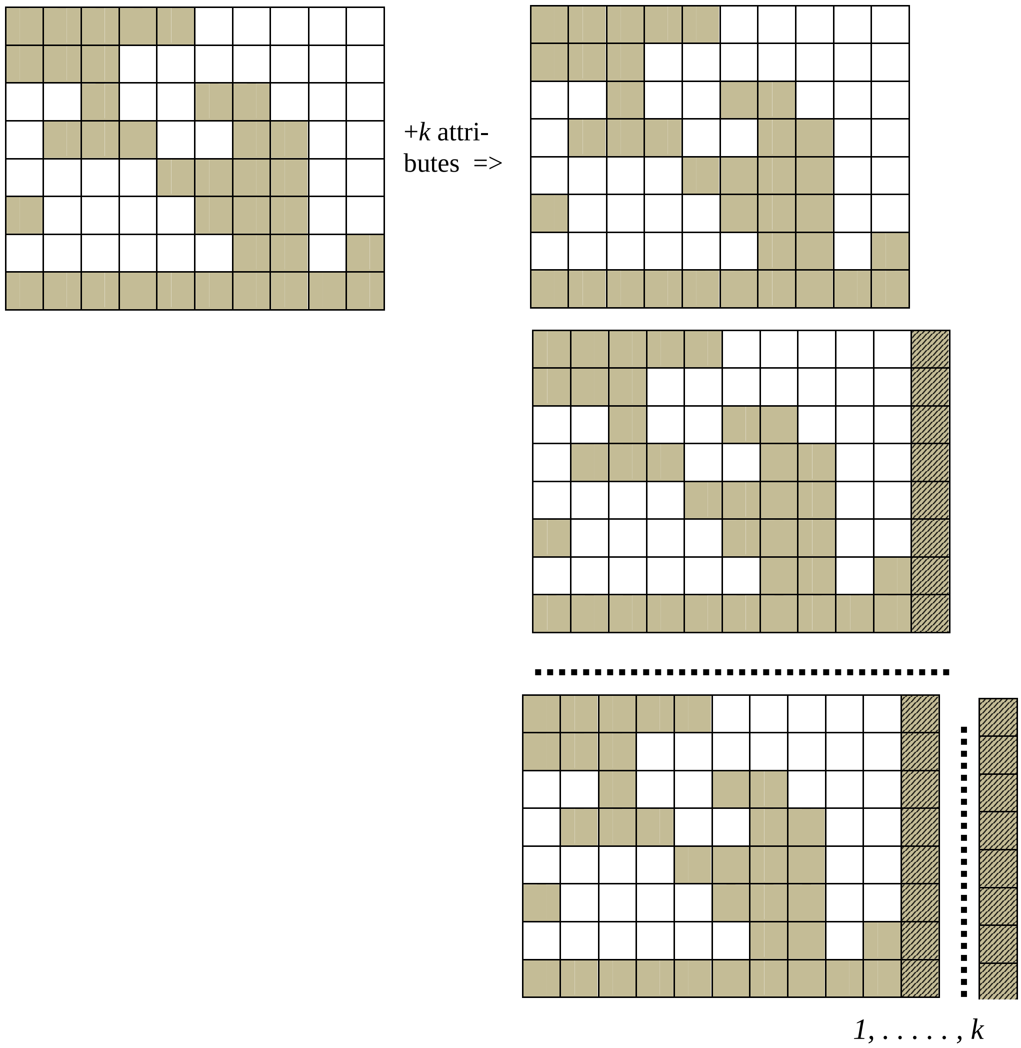

Fig. 5. Illustration of effect of adding *k* attributes to association in aspect of generators structure and to lower boundary of interval variable of number generator enlarging $dGI = kn$

Otherwise, if we left perception structures as sets trying to automatize their modifications, we should add information about the distribution probability of associations referring to the location in perception and background environment. Such probabilities can be defined on the basis of single events probabilities (connected with appearing single attributes): $pa(i)$, $i = 1,2,\ldots,la$, where la is the number of attributes. In this case we can calculate the probability of events sum:

$$pas(i(1)\cup i(2)\cup \ldots \cup i(l)) =$$

$$= \sum_{i_1=1}^{l} pa(i_1) - \sum_{i_1=1}^{l-1} \sum_{i_2=i_1+1}^{l} pa(i_1)\, pa(i_2) + \ldots + (-1)^{l-1} \sum_{i_1=1}^{1} \sum_{i_2=i_1+1}^{2} \ldots \sum_{i_l=i_{l-1}+1}^{l} \tag{10}$$

$$pa(i_1)\; pa(i_2) \ldots pa(i_l),$$

or probability of events product:

$$pap(i(1)\cap \mathrm{i}(2)\cap \ldots \cap i(la)) = pa(i_1)\; pa(i_2) \ldots pa(i_l), \tag{11}$$

where l - attribute number in association background.

In a sequence variant of automaton modeling, the probability distribution of single attributes is used. The distribution of single attributes and the set of them are regarded in association distribution referring to a set of generators.

Conclusions

Association automata work on the general rules of the attitude of the mind. In fact, we chose only thinking processes connected with retaining, remembering and forgetting. We add to them the unconscious process of refreshing our memory. They are processes dependent on conception learning and structural exploitation of its effects.

Association automata can be exploited for the recognition of situations and to activate specific reactions relating to it based on cause-effect mechanisms. The proposed automats can be systematically structurally developed and enriched with new attributes inside the nodes. The probabilistic characteristics of particular parts (background, nodes etc.) or elements (attributes, associations) dynamically changes during the exploitation process.

In the aim to refer to natural memory and association systems [3], the parameters of association life and the external factors influencing them are introduced.

References

[1] Kiesler C.A., The Psychology and Commitment, Academic Press, New York 1971.
[2] Motwani R., Raghavan P., Randomized Algorithms, Cambridge University Press 1994.
[3] Bowart W., Operation Mind Control, Fontana, London 1978.
[4] Kozen D.C., Automata and Computability, Springer, New York 1997.
[5] Ambainis A., Watrous J., Two-way 4nite automata with quantum and classical states, Theoret. Comput. Sci. 2002, 287, 299-311.
[6] Paz A., Introduction to Probabilistic Automata, Academic Press, New York 1971.
[7] Piech H., Sequence Automata for Researching Concencuse Levels, Lecture Networks in Artificial Intelligence, 6682, Springer Verlag 2011, 251-261.
[8] Thierrin G., Permutation automata, Mathematical Systems Theory 1968, 2(1), 83-90.
[9] Shukla W., Srivastava A.K., A topology for automata: a note, Inf. Control 1976, 32, 163-168.