AN AUTOMATON DURING ITS OPERATION

Gerasimos G. Massouros
54, Klious st., 155 61 Cholargos, Athens, Greece

(Dedicated to the memory of GEORGE GH. MASSOUROS)

Abstract

In this paper a hypergroup is being presented which describes an automaton during its operation. Using it and through a certain procedure which is being developed here, among others, all the states at which the automaton can possibly be found, at any given time $t$, are being determined.
1. **Introduction.**

The attached hypergroup of an automaton, a notion which derives from the attempt of expressing and solving problems of the Theory of Automata and Languages with use of the Theory of the Hypercompositional Structures, has been introduced in [2]. There two kinds of attached hypergroups appear, the *attached order hypergroups* which generally are several types of canonical hypergroups and the *attached grade hypergroup*, which is either join polysymmetrical hypergroup or fortified join hypergroup, when the automaton is minimized. Apart from other applications [3],[4], these attached hypergroups have been used for the minimization of the automaton. Moreover, in [3], another kind of attached hypergroup has been introduced, the *attached hypergroups of the paths*. But all these attached hypergroups deal with the set of states of an automaton and describe its structure. However the operation of an automaton involves the factor of time. Therefore in the following we shall search for a hypergroup which will somehow describe the automaton during its operation.

2. **The attached hypergroup of the operational paths of an automaton.**

An automaton is a mathematical model for a machine that accepts a particular set of words over some alphabet $A$ [5]. Let's consider an automaton $(A, S, s_0, \delta, F)$, the one of figure 1 for instance:

![Diagram](image)

*Figure 1.*
Definition 2.1. We say that the state $s_j$ is successive to $s_i$ if there exists $\chi \in A$ such that $\delta(s_i, \chi) = s_j$.

Obviously the fact that $s_j$ is successive to $s_i$ does not imply that $s_i$ is successive to $s_j$ as well (without excluding it though). Thus the following diagram presents the successive states of the above automaton.

![Diagram of successive states]

Figure 2.

Definition 2.2. We say that the state $s_j$ is connected with $s_i$ if there exists a word $\chi \in A^*$ such that $\delta^*(s_i, \chi) = s_j$.

The automaton always starts its operation from the start state $s_0$ and while reading one letter at every moment (clock pulse) it moves to the next successive state. It is possible though, during its operation, to pass from the same state at different moments. Thus for instance the automaton of figure 1 can be found on state $s_1$ with the word $\alpha$, with the word $\beta\alpha$, with the word $\alpha\beta\alpha$, etc. This means that it can be found on state $s_1$ during the first moment of its operation, during the second moment, the third etc.

Therefore if we wish to describe the operation of an automaton, we can consider the cartesian product $S \times N$ where $S$ the set of states of the
automaton and \( N \) the set of the non negative integers. We will use the notation \( s_i^k \) to represent the ordered pair \((s_i, k)\) which shows the state \( s_i \) at the moment \( k \). The graph of these ordered pairs can be drawn in the same manner as the following one, which refers to the automaton of figure 1. In this graph, where the horizontal axis, which is the set \( N \), represents the clock pulses, and the vertical axis represents the states, we have connected all the successive states up to the tenth clock pulse of the automaton's operation.

![Graph showing automaton states](image)

From the graph we see that in fact we are interested in the states on which the automaton can possibly be found. Hence we introduce the definition:

**Definition 2.3.** An element \( s_i^t \) of the cartesian product \( S \times N \) is called **activated**, if, after \( t \) clock pulses, the automaton can be found on the state \( s_i \).

When we want to especially emphasize the moment of the activation, then we speak for the **t-activated** element.

Next let's denote with \( A \) the set of the activated elements. In this set we generalize the notions of the functions \( \delta \) and \( \delta^* \) in the following way:

\[
\delta_A(s_i^t, x) = s_j^{t+1}, \quad \text{where } s_j = \delta(s_i, x) \quad \text{and}
\]

\[
\delta^*_A(s_i^t, w) = s_j^{t+|w|}, \quad \text{where } s_j = \delta^*(s_i, w) \quad \text{and}
\]

\(|w|\) is the length of the word \( w \)(that is the number of \( w \)'s letters [2]).
Using simpler notations the above can be written as: \( \delta_A(s_i^j, x) = \delta(s_i, x)^{t+1} \) and \( \delta_A^*(s_i^j, x) = \delta^*(s_i, x)^{t+1} \).

**Definition 2.4.** The elements \( s_i^j \) and \( s_i^l \) will be called **successive** if the state \( s_j \) is successive to the state \( s_i \) and \( r = t + 1 \). Also the element \( s_i^j \) is called **connected** with the \( s_i^l \) if the state \( s_j \) is connected with the state \( s_i \) and \( t < r \).

Thus it is obvious that in the case of a deterministic automaton two \( t \)-activated elements cannot be connected.

In the set of the activated elements we observe that two elements can be:

(a) not connected (e.g. the \( s_2^2 \) and \( s_1^1 \) of fig. 3).

(b) connected one with the other by one word (e.g. the \( s_3^3 \) and \( s_1^1 \) of fig. 3).

(c) connected one with the other by more than one word (e.g. the \( s_3^4 \) and \( s_0^0 \) of fig. 3).

These remarks lead to the introduction of a hypercomposition "+" on the set \( A \) of the activated elements of the cartesian product \( S \times N \):

\[
s_i^m + s_j^n = \begin{cases} 
\{ \delta_A^*(s_i^m, x) \mid x \in \text{Prefix}(r), \delta_A^*(s_i^m, r) = s_j^n \}, & \text{if } s_j^n \text{ is connected with } s_i^m, \\
\{ s_i^m, s_j^n \}, & \text{if } s_j^n \text{ is not connected with } s_i^m.
\end{cases}
\]

This hypercomposition is associative. Indeed let the element \( s_i^m \) be connected with the \( s_j^n \), \( s_k^p \) and let the \( s_j^n \) be connected with \( s_k^p \). Also let \( m < n < p \). Then

\[
(s_i^m + s_j^n) + s_k^p = \{ \delta_A^*(s_i^m, x) \mid z \in \text{Prefix}(r), \delta_A^*(s_i^m, r) = s_j^n \} + s_k^p = \\
= \{ \delta_A^*(s_i^m, x), y \} \mid x \in \text{Prefix}(r), \delta_A^*(s_i^m, r) = s_j^n, \\
y \in \text{Prefix}(q), \delta_A^*(s_m^n, z), q = s_k^p \} = \\
= \{ \delta_A^*(s_i^m, v) \mid v \in \text{Prefix}(w), \delta_A^*(s_i^m, w) = s_k^p \}.
\]

Moreover,

\[
s_i^m + (s_j^n + s_k^p) = \{ \delta_A^*(s_i^m, x) \mid x \in \text{Prefix}(r), \delta_A^*(s_i^m, r) = s_k^p \} = \\
= \{ \delta_A^*(s_i^m, z) \mid z \in \text{Prefix}(u), \delta_A^*(s_i^m, u) = s_k^p \text{ or} \\
\delta_A^*(s_i^m, v) = \delta_A^*(s_j^n, x), x \in \text{Prefix}(r), \delta_A^*(s_j^n, r) = s_k^p \} = \\
= \{ \delta_A^*(s_i^m, v) \mid v \in \text{Prefix}(w), \delta_A^*(s_i^m, w) = s_k^p \}.
\]
Next let the elements $s_i^m, s_j^n$ and $s_i^m, s_k^p$ be connected one with the other, while $s_j^m, s_k^p$ are not connected. Then

$$(s_i^m + s_j^n) + s_k^p = \{\delta_A(s_i^m, x) \mid x \in \text{Prefix}(r), \delta_A(s_i^m, r) = s_j^n\} + s_k^p = (s_i^m + s_j^n) + (s_i^m + s_k^p).$$

Moreover $(s_i^m + s_j^n) + s_k^p = s_i^m + \{s_j^n, s_k^p\} = (s_i^m + s_j^n) + (s_i^m + s_k^p)$.

Now if the elements $s_i^m, s_j^n$ are connected, while the $s_k^p$ is not connected with none of the other two, then

$$(s_i^m + s_j^n) + s_k^p = \{\delta_A(s_i^m, x) \mid x \in \text{Prefix}(r), \delta_A(s_i^m, r) = s_j^n\} \cup \{s_k^p\}.$$  

But the element $s_k^p$ is not connected with none of the $\delta_A(s_i^m, x)$ since if it were, then $s_i^m$ would have been connected with $s_k^p$, which is absurd. Thus we have:

$$(s_i^m + s_j^n) + s_k^p = \{\delta_A(s_i^m, x) \mid x \in \text{Prefix}(r), \delta_A(s_i^m, r) = s_j^n\} \cup \{s_k^p\}.$$  

Moreover

$$s_i^m + (s_j^n + s_k^p) = s_i^m + \{s_j^n, s_k^p\} = (s_i^m + s_j^n) + (s_i^m + s_k^p) = \{\delta_A(s_i^m, x) \mid x \in \text{Prefix}(r), \delta_A(s_i^m, r) = s_j^n\} \cup \{s_k^p\}.$$  

Finally if the elements $s_i^m, s_j^n, s_k^p$ are not connected, then

$$(s_i^m + s_j^n) + s_k^p = (s_i^m, s_j^n) + s_k^p = (s_i^m + s_k^p) \cup (s_j^n + s_k^p) = \{s_i^m, s_j^n, s_k^p\}.$$  

Similarly, $s_i^m + (s_j^n + s_k^p) = \{s_i^m, s_j^n, s_k^p\}$. For every element $s_i^m$ from $A$, we have $s_i^m + A = A + s_i^m = A$ and so we have the proposition:

**Proposition 2.1.** The set $A$ of the activated elements of the cartesian product $S \times N$, where $S$ is the set of the states of an automaton, becomes a hypergroup if we introduce in it the above defined hypercomposition.

This hypercomposition is not commutative and therefore we have the following definitions of the two induced hypercompositions "$\cdot m$" and "$..n$", [1]

$$s_i^m : s_j^n = \begin{cases} 
  \{s_k^p\} & \text{if } x \in A^* \text{ such that } \delta_A(s_k^p, x) = s_i^m, \\
  \{s_i^m\} & \text{if } m < n \text{ and } s_i^m, s_j^n \text{ are connected,} \\
  \{s_i^m, s_j^n\} & \text{if } s_i^m, s_j^n \text{ are not connected;} 
\end{cases}$$
$$s_i^m \cdot s_j^n = \begin{cases} \{s_k^p \mid z \in A^* \text{ such that } \delta_A^e(s_j^n, x) = s_k^p \}, & \text{if } m < n \text{ and } s_i^m, s_j^n \text{ are connected,} \\ \{s_i^m\}, & \text{if } s_i^m, s_j^n \text{ are not connected.} \end{cases}$$

Hence the hypergroup \((A, +)\) does not satisfy the join axiom.

3. A method for calculating the results of the hypercomposition and giving information for the operation of the automaton.

In the last part of this work we present a method with which we can find the results of the hypercomposition defined above, that is we can see if \(s_i^t\) is a \(t\)-activated element and at the same time we can determine all the intermediate states that the automaton passes from until it reaches the \(s_i^t\). The procedure is the following:

Initially the "matrix of the possible states" \(\Pi \delta\) is defined as an \(n \times n\)-matrix with rows and columns the \(n\) states of the automaton. Every element \(\Pi \delta_{ij}\) of this matrix is calculated according to the rule:

\[
\Pi \delta_{ij} = \begin{cases} 
\text{the state } s_j, \text{ if it is possible for the automaton to reach this state during the next clock pulse, starting from state } s_i, \\
0 \text{ in any other case.}
\end{cases}
\]

Next the "matrix of the first clock pulse" \(\Pi[1]\) is defined as an \(n \times n\)-matrix with its columns the elements \(s_i^1, i = 1, 2, \ldots, n\) and rows the elements \(s_i^0, i = 1, 2, \ldots, n\), that is the elements of the previous clock pulse. Every entry \(\Pi_{ij}\) is being defined from the rule:

\[
\Pi_{ij} = \begin{cases} 
\sigma_0 s_i^1, \text{ if the automaton can reach state } s_j \text{ at the first clock pulse,} \\
0 \text{ in any other case.}
\end{cases}
\]

In matrix \(\Pi[1]\) the row corresponding to the element \(s_0^0\) is expected to have entries different from zero, since at this phase the automaton is obviously starting from the start state \(s_0^0\).

Finally the matrix of every subsequent clock pulse derives from the matrix of the previous clock pulse with the use of \(\Pi \delta\). Thus for the \(i\) row of the "matrix of the \(t\) clock pulse" \(\Pi[t]\), [which has \(n\) columns (the elements \(s_i^t, i = 1, 2, \ldots, n\)) and \(n\) rows (the elements \(s_i^{t-1}, i = 1, 1, \ldots, n\))] we have

- It comes from the \(i\) column of the matrix \(\Pi[t - 1]\) and the \(i\) row of \(\Pi \delta\).
- It has zero everywhere if the entire row \( \Pi \delta \) or the entire column of \( \Pi[t-1] \) is zero, otherwise:

- It has zero at all the places where the row of \( \Pi \delta \) has zero and

- In every one of the rest places we write the set of the paths that derives if at the end of each path which appears in the entire column of the matrix \( \Pi[t-1] \), (regardless of the row) we add the element that comes from the state which is written at the respective place of the matrix \( \Pi \delta \), at the clock pulse \( t \).

As it seems, from the procedure of its construction, at every position of \( \Pi[t] \) there may be more than one entries (paths). Also the position at which every set of the paths that derives as described above will be written is determined from the respective position of the state of \( \Pi \delta \), which will consist the last element of the path.

Additionally we remark that:

1. The corresponding element \( s^t_i \) to the column of matrix \( \Pi[t] \) with zero entries everywhere is a non activated element.

2. The corresponding element \( s^t_j \) to the row of matrix \( \Pi[t] \) with zero entries everywhere is a non activated element.

3. The corresponding to the zero entry elements of matrix \( \Pi[t] \) are not connected elements.

Applying the above to the automaton we have described in fig. 1 we have:

\[
\Pi \delta = \begin{pmatrix}
  s_0 & s_1 & s_2 & s_3 \\
  s_0 & 0 & s_1 & s_2 & 0 \\
  s_1 & 0 & 0 & s_2 & s_3 \\
  s_2 & 0 & s_1 & 0 & s_3 \\
  s_3 & 0 & 0 & 0 & s_3
\end{pmatrix}
\]

Also the first 4 matrices are:

\[
\Pi[1] = \begin{pmatrix}
  s_0^1 & s_1^1 & s_2^1 & s_3^1 \\
  s_0^0 & 0 & s_0^1 & s_0^1 & 0 \\
  s_1^0 & 0 & 0 & 0 & 0 \\
  s_2^0 & 0 & 0 & 0 & 0 \\
  s_3^0 & 0 & 0 & 0 & 0
\end{pmatrix}
\]
Interpreting the results of matrix $\Pi[4]$ for instance, we see that, at the fourth moment of its operation, the automaton of the example can be found:

- on state $s_1$ having followed the path $s_0 s_2 s_1 s_2 s_1$,
- on state $s_2$ having followed the path $s_0 s_1 s_2 s_1 s_2$,
- on state $s_3$ having followed one of the paths: $s_0 s_1 s_2 s_1 s_3$, or $s_0 s_2 s_1 s_2 s_3$, or $s_0 s_2 s_1 s_3 s_3$, or $s_0 s_1 s_2 s_3 s_3$.

So the 4-activated elements form the set $\{s_1^4, s_2^4, s_3^4\}$, and the result of the hypercomposition, say $s_0^4 + s_1^4$, is: $s_0^4 + s_1^4 = \{s_0^0, s_1^1, s_2^2, s_3^3, s_4^4\}$, while $s_2^2 + s_3^3 = \{s_2^2, s_2^3, s_3^4\}$. 
REFERENCES


