Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 37(1), 2010, Pages 80–89
ISSN: 1223-6934

Algebraic properties of ω-trees (I)

Nicolae Ţăndăreanu and Cristina Zamfir

Abstract. This paper is a starting point for a possible research line to study the theoretical aspects of the answer function for a master-slave system based on semantic schemas. We define the concept of ω-labeled tree as a binary, ordered and labeled tree with several features concerning the labels and order between the direct descendants of a node. The labeling operation of the nodes is guided by the mapping ω which defines the splitting operation for labels. An embedding operation of an ω-tree into another ω-tree is introduced. We prove that this operation is performed by means of an injective mapping. Based on this operation some binary relation between ω-labeled trees is defined. This is a reflexive and transitive relation, but is not antisymmetric. All the results proved in this paper and in [15] constitute the algebraic background of a forthcoming paper as we mention in the last section.

2010 Mathematics Subject Classification. Primary 05C20; Secondary 18A10.
Key words and phrases. directed ordered graph, tree, ordered tree, partial order, embedding mapping.

1. Introduction

Research in artificial intelligence based on graphs theory, knowledge-based systems, neural networks and algebraic methods is very productive. An integrated approach with some graph-based heuristic working rules and neural networks-based practices to assist the assembly engineers in generating and predicting a best and most effective assembly sequence is described in [4]. The problem of verifying external adequacy in expert systems by means of the graph theory is treated in [16]. The use of attack graphs in network security can be found in [2]. Learning schemas based on decision forests were studied in [8]. The theory of conceptual graphs was first published by John Sowa ([7]). Since then various studies based on this concept were obtained such as nested conceptual graphs and conceptual graphs with negation ([3]). The coherence graphs, which give a graphical representation of the conditional lower previsions, were treated in [6]. The paper [5] develops a representation of multi-model based controllers using artificial intelligence techniques as graph theory, neural networks, genetic algorithms, and fuzzy logic. The structures known as labeled stratified graphs ([9], [10], [11]) and semantic schemas ([12], [13], [14]) use a labeled graph and contain an algebraic structure helping to knowledge representation.

This paper is the first in a series of papers which aims to study certain algebraic properties of the master-slave systems based on semantic schemas. The last papers of this series help us to introduce the concepts of semantic tree and syntactic tree. Both concepts help us to study the mechanism of the computations in a master-slave systems based on semantic schemas. Finally we intend to apply these results to design intelligent dialogue systems.

Received January 07, 2010. Revision received February 10, 2010.

In the remainder of this section we give a short description of this paper. In Section 2 we recall the basic concepts and notations used in the subsequent sections: directed ordered graph, ordered tree, arcs and paths in these structures. In Section 3 we introduce the concept of ω-labeled tree. This structure is a binary ordered tree whose nodes are labeled by the elements of a set L. Each node has a label and the labeling rule is given by the mapping ω. The set L is divided into two distinct subset: terminal and nonterminal labels. A node labeled by a terminal label is a leaf of the tree. For a given labeling mapping ω we consider the set $OBT(\omega)$ of all binary ordered trees whose nodes are labeled by means of ω. In Section 4 we define an embedding operation of a tree into another tree. This operation allow us to define a binary relation on the set $OBT(\omega)$. We prove that this is a reflexive and transitive relation, but is not an antisymmetric one. Based on this relation in [15] we introduce an equivalence relation on the set $OBT(\omega)$. These results allows to study the mechanism of formal computations in a master-slave system based on semantic schemas. The last section contains conclusions and future work.

2. Basic notions and notations

A *directed ordered graph* ([1]) is a pair $G = (A, D)$, where
- A is a finite set of elements called *nodes*;
- D is a finite set of elements of the form $[(i, i_1), \ldots, (i, i_n)]$, where $n \geq 1$ and $i, i_1, \ldots, i_n \in A$;
- D satisfies the following condition: if $[(i, i_1), \ldots, (i, i_n)] \in D$ and $[(j, j_1), \ldots, (j, j_s)] \in D$ then $i \neq j$.

We observe that for an element $[(i, i_1), \ldots, (i, i_n)] \in D$ we may have $i_j = i_k$ for some $j \neq k$. On the other hand, an element of D is a list and the *order* of its elements are taken into consideration. An element of a list is a *directed arc* and simply is named *arc*. This can explain why the concept is named *directed ordered graph.*

We can represent a directed ordered graph as follows. We represent, as usual, a node of the graph by a point. If $[(i, i_1), \ldots, (i, i_n)] \in D$ then we draw *an arc* from node i to node i_j for every $j \in \{1, \ldots n\}$. The elements $i_1, \ldots, i_n$ are called *the direct descendants* of i. We shall consider that all direct descendants of i are ordered linearly and the order is given by the place of i_j in the element $[(i, i_1), \ldots, (i, i_n)]$.

If $G = (A, D)$ is a directed ordered graph then we can associate to G a *directed graph* $G' = (A, D')$, where

$$D' = \{(i, j) \mid \exists[(i, i_1), \ldots, (i, i_n)] \in D, \exists r \in \{1, \ldots, n\} : j = i_r\}$$

An *ordered tree* is a directed ordered graph $G = (A, D)$ such that G' is a tree and the following property is satisfied:

$$[(i, i_1), \ldots, (i, i_n)] \in D, j, r \in \{1, \ldots, n\}, j \neq r \Rightarrow i_j \neq i_r \tag{1}$$

Two distinct ordered trees are represented in Figure 1. In the left part of this figure we have $[(c, d), (c, e), (c, f)] \in D$, whereas in the right part we have $[(c, f), (c, e), (c, d)] \in D$. They have the same set of nodes but the order is another.

A *path* in a directed ordered graph is a sequence $d = (n_0, n_1, \ldots, n_k)$ of nodes such that for every $i \in \{0, \ldots, k-1\}$ we have an arc from n_i to n_{i+1}. The number k is the *length* of d. We denote by $Path(G)$ the set of all paths in G.

A *binary tree* is a tree such that every node has exactly zero or two direct descendants. The *root* is a node that is not a direct descendant of any other node. A tree has a single root. Every node that is not the root in the binary tree is reachable from

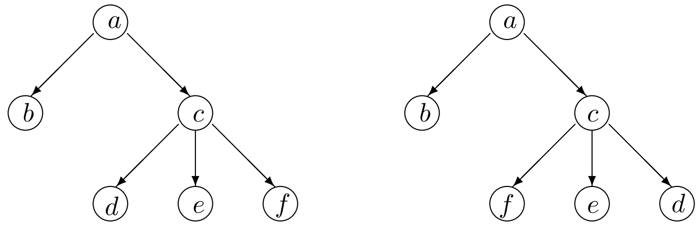


FIGURE 1. Two distinct ordered trees

the root node by a unique path. A node with neither a left descendant nor a right descendant is called a *leaf*. By an abuse of language we shall use the concepts of arc and path in an ordered tree t and in this case we suppose that these concepts are applied to the graph associated to t. Moreover, for an ordered tree t we denote by $Path(t)$ the set of all paths of t.

3. The set OBT(ω) of ordered binary trees

We consider a finite set L and a decomposition $L = L_N \cup L_T$, where $L_N \cap L_T = \emptyset$. The elements of L_N are called *nonterminal labels* and those of L_T are called *terminal labels*. The elements of L are called *labels*.

Definition 3.1. *Let $L = L_N \cup L_T$ be a set of labels. A* **split mapping** *on L is a function $\omega : L_N \longrightarrow L \times L$. For each $x \in L_N$ we denote $\omega(x) = (\omega_1(x), \omega_2(x))$. The entity $\omega_1(x)$ is named the* ***left component*** *and $\omega_2(x)$ is the* ***right component*** *of $\omega(b)$.*

Definition 3.2. *Let $\omega : L_N \longrightarrow L \times L$ be a split mapping on L. An ω-**tree** is a triple $t = (A, D, h)$, where*
- *(A, D) is an ordered tree and every element of D is of the form $[(i, i_1), (i, i_2)]$;*
- *$h : A \longrightarrow L$ is a mapping such that*

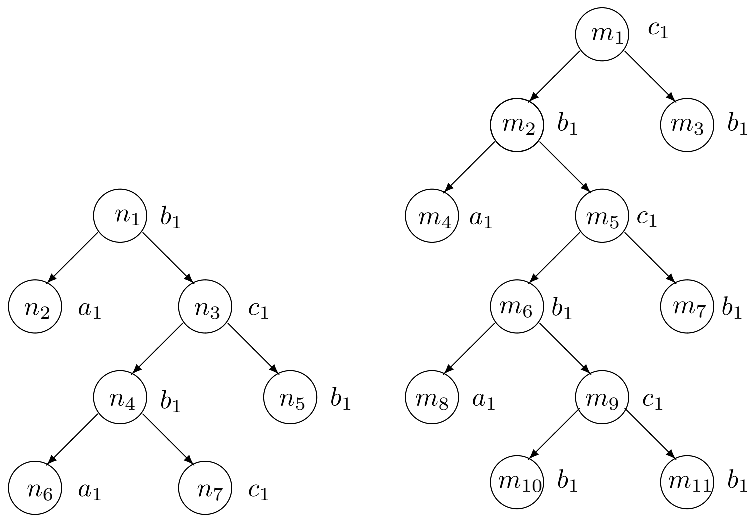
$$[(i, i_1), (i, i_2)] \in D \Rightarrow h(i) \in L_N \ \ \& \ \ \omega(h(i)) = (h(i_1), h(i_2)) \qquad (2)$$

For each $i \in A$ the element $h(i)$ is called the ***label*** *of the node i. The mapping h is named the* ***labeling mapping*** *of t. By $OBT(\omega)$ we denote the set of all ω-trees.*

Remark 3.1. *In an ω-tree we have the following property: if a node n has direct descendants then the label $h(n)$ of n is an element of L_N. Moreover, in this case the left (right) descendant of n is labeled by the left (right) component of $\omega(h(n))$. A leaf of an ω-labeled tree may be labeled by an element of L_N, but a node labeled by an element of L_T is a leaf.*

In order to exemplify these concepts we consider the following case:
- $L = \{a_i, b_i, c_i\}_{i \geq 1}$, $L_N = \{b_i, c_i\}_{i \geq 1}$, $L_T = \{a_i\}_{i \geq 1}$;
- $\omega(b_i) = (a_i, c_i)$ and $\omega(c_i) = (b_i, b_i)$ for $i \geq 1$.

In Figure 2 we represented the following two ω-trees $t_1 = (A_1, D_1, h_1)$ and $t_2 = (A_2, D_2, h_2)$, where:



$t_1 = (A_1, D_1, h_1)$ $\qquad$ $t_2 = (A_2, D_2, h_2)$

FIGURE 2. Two ω-trees

- $A_1 = \{n_j\}_{j=1,\dots,7}$; $A_2 = \{m_j\}_{j=1,\dots,11}$;
- $h_1(n_i) = b_1$ for $i \in \{1, 4, 5\}$; $h_1(n_2) = h(n_6) = a_1$; $h_1(n_3) = h_1(n_7) = c_1$;
- $h_2(m_4) = h_2(m_8) = a_1$; $h_2(m_i) = b_1$ for $i \in \{2, 3, 6, 7, 10, 11\}$; $h_2(m_1) = h_2(m_5) = h_2(m_9) = c_1$;
- $D_1 = \{[(n_1, n_2), (n_1, n_3)], [(n_3, n_4), (n_3, n_5)], [(n_4, n_6), (n_4, n_7)]\}$
- $D_2 = \{[(m_1, m_2), (m_1, m_3)], [(m_2, m_4), (m_2, m_5)], [(m_5, m_6), (m_5, m_7)], [(m_6, m_8), (m_6, m_9)], [(m_9, m_{10}), (m_9, m_{11})]\}$.

Remark 3.2. *In Figure 2 we remark that if we make abstraction of node names and translate t_1 such that n_1 overlaps m_2 then t_1 becomes a part of t_2. This part can be viewed as an "image" of t_1 into t_2. We see that the image of t_1 is not a subtree of t_2 because the nodes m_{10} and m_{11} do not belong to this image.*

4. Embedding mappings

Let $t_1 = (A_1, D_1, h_1)$ and $t_2 = (A_2, D_2, h_2)$ be two elements of $OBT(\omega)$ and an arbitrary mapping $\alpha : A_1 \longrightarrow A_2$. For every $u = [(i, i_1), (i, i_2)] \in D_1$ we denote

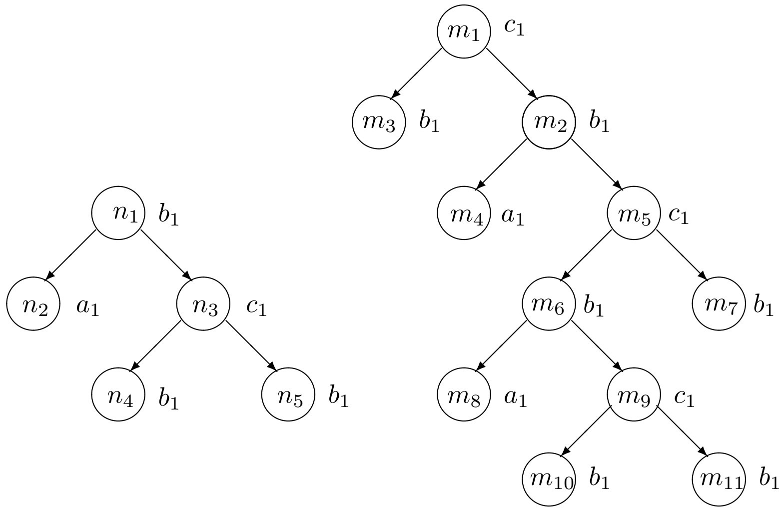
$$\overline{\alpha}(u) = [(\alpha(i), \alpha(i_1)), (\alpha(i), \alpha(i_2))]$$

If $t = (A, D, h)$ is an ω-tree then we denote by $root(t)$ the element of A designated by the root of t.

Definition 4.1. *If $t_1 = (A_1, D_1, h_1) \in OBT(\omega)$ and $t_2 = (A_2, D_2, h_2) \in OBT(\omega)$ then we define the relation $t_1 \preceq t_2$ if and only if there is a mapping $\alpha : A_1 \longrightarrow A_2$ such that:*

$$u \in D_1 \Rightarrow \overline{\alpha}(u) \in D_2 \tag{3}$$

$$h_1(root(t_1)) = h_2(\alpha(root(t_1))) \tag{4}$$



$t_1 = (A_1, D_1, h_1)$ $\qquad$ $t_2 = (A_2, D_2, h_2)$

FIGURE 3. Two ω-trees

Such a mapping α is an ***embedding mapping*** *of t_1 into t_2.*

If we consider the trees from Figure 2 then we can verify immediately that $t_1 \preceq t_2$. Really, the mapping α defined by $\alpha(n_1) = m_2$ and $\alpha(n_j) = m_{j+2}$ for $j = 2, \ldots, 7$ satisfies Definition 4.1 and thus α is an embedding mapping.

If $t_1 \preceq t_2$ then the embedding mapping of t_1 into t_2 is not uniquely determined. In order to see such a case we consider the elements represented in Figure 3. The translations given by the mappings $h_1(n_1) = m_1$, $h_2(n_1) = m_2$ and $h_3(n_1) = m_6$ give three distinct embedding mappings of t_1 into t_2.

The next proposition shows that the condition (4) is extended to all nodes of t_1.

Proposition 4.1. *Let be $t_1 = (A_1, D_1, h_1) \in OBT(\omega)$ and $t_2 = (A_2, D_2, h_2) \in OBT(\omega)$. If $t_1 \preceq t_2$ then $h_1(i) = h_2(\alpha(i))$ for every $i \in A_1$, where α is an embedding mapping.*

Proof. In a regular tree t for every node $n \neq root(t)$ there is a path and only one from $root(t)$ to n. We shall verify by induction on k the following property $P(k)$: *for every $i \in A_1$ such that the path from $root(t_1)$ to i has $k \geq 0$ arcs we have $h_1(i) = h_2(\alpha(i))$.*
For $k = 0$ the property $P(k)$ is true because in this case we have $i = root(t_1)$ and $P(0)$ is true in virtue of (4).
We suppose the property $P(k)$ is true and let us verify that $P(k+1)$ is true. We consider a node $j \in A_1$ for which the path $(root(t_1), n_1, \ldots, n_k, j)$ from $root(t_1)$ to j has $k+1$ arcs. The node j is a direct descendant of n_k. In order to make a choice we suppose that j is a left direct descendant of n_k. In other words, there is $r \in A_1$ such that $u = [(n_k, j), (n_k, r)] \in D_1$. From (3) we have $\overline{\alpha}(u) \in D_2$. But $\overline{\alpha}(u) = [(\alpha(n_k), \alpha(j)), (\alpha(n_k), \alpha(r))]$. From (2) we have

$$\omega(h_2(\alpha(n_k))) = (h_2(\alpha(j)), h_2(\alpha(r))) \tag{5}$$

Analogous, because $[(n_k, j), (n_k, r)] \in D_1$ we have

$$\omega(h_1(n_k)) = (h_1(j), h_1(r)) \tag{6}$$

By the inductive assumption for $P(k)$ we have $h_1(n_k) = h_2(\alpha(n_k))$. From (5) and (6) we deduce $h_1(j) = h_2(\alpha(j))$ and thus the proposition is proved. ■

Proposition 4.2. *Suppose that $t_1 = (A_1, D_1, h_1) \in OBT(\omega)$, $t_2 = (A_2, D_2, h_2) \in OBT(\omega)$ and $t_1 \preceq t_2$. Let us denote by $\alpha : A_1 \longrightarrow A_2$ an embedding mapping of t_1 into t_2.*

(1) *If (m, n) is an arc in t_1 then $(\alpha(m), \alpha(n))$ is an arc in t_2;*

(2) *If $d = (n_0, n_1, \ldots, n_k) \in Path(t_1)$ then $\alpha(d) = (\alpha(n_0), \alpha(n_1), \ldots, \alpha(n_k)) \in Path(t_2)$.*

Proof. If $(m, n))$ is an arc in t_1 then there is $u \in D_1$ such that $u = [(m, n), (m, p)]$ or $u = [(m, p), (m, n)]$. But $\overline{\alpha}(u) \in D_2$, therefore $[(\alpha(m), \alpha(n)), (\alpha(m), \alpha(p))] \in D_2$ or $[(\alpha(m), \alpha(p)), (\alpha(m), \alpha(n))] \in D_2$. It follows that $(\alpha(m), \alpha(n))$ is an arc in t_2. The second property is immediate from the first property. ■

Proposition 4.3. *An embedding mapping is injective.*

Proof. Suppose that $t_1 = (A_1, D_1, h_1) \in OBT(\omega)$ and $t_2 = (A_2, D_2, h_2) \in OBT(\omega)$. Consider that $t_1 \preceq t_2$ and $\alpha : A_1 \longrightarrow A_2$ is an embedding mapping of t_1 into t_2. Suppose that $n_1, n_2 \in A_1$ and $n_1 \neq n_2$. We prove that $\alpha(n_1) \neq \alpha(n_2)$.
We consider the following sets of nodes:

$$\begin{cases} M_0 = \{root(t_1)\} \\ M_{k+1} = \{n \in A_1 \mid \exists(root(t_1), n_1, \ldots, n_k, n) \in Path(t_1)\}, k \geq 0 \end{cases} \tag{7}$$

Obviously there is a natural number n such that $M_n = M_{n+i}$ for every $i \geq 1$, so we can write

$$A_1 = \bigcup_{i \geq 0} M_i \tag{8}$$

We prove that for every $k \geq 1$ the following property $P(k)$ is true: *if $n_1, n_2 \in \bigcup_{i=0}^{k} M_i$ and $n_1 \neq n_2$ then $\alpha(n_1) \neq \alpha(n_2)$.* Based on this property and the relation (8) we obtain our proposition.
Take $n_1, n_2 \in M_1$ such that $n_1 \neq n_2$. If $n_1 = root(t_1)$ then there is an arc from n_1 to n_2 in t_1, therefore from Proposition 4.2 there is an arc from $\alpha(n_1)$ to $\alpha(n_2)$ in t_2, therefore $\alpha(n_1) \neq \alpha(n_2)$. The case $n_2 = root(t_1)$ is a similar one. Let us suppose now that $[(root(t_1), n_1), (root(t_1), n_2)] \in D_1$. In this case we have $[(\alpha(root(t_1)), \alpha(n_1)), (\alpha(root(t_1)), \alpha(n_2))] \in D_2$. From (1) it follows that $\alpha(n_1) \neq \alpha(n_2)$. Therefore the property $P(1)$ is true.
Let us suppose that $P(k)$ is true and we prove now that $P(k+1)$ is also true. We take $n_1, n_2 \in \bigcup_{i=0}^{k+1} M_i$ such that $n_1 \neq n_2$. We have the following three cases:

a) Suppose that $n_1, n_2 \in \bigcup_{i=0}^{k} M_i$. In this case the property is true following the inductive assumption $P(k)$.

b) Suppose that $n_1, n_2 \in M_{k+1}$. There are the arcs (p_1, n_1) and (p_2, n_2) in t_1. We have $p_1, p_2 \in \bigcup_{i=0}^{k} M_i$.

- If $p_1 = p_2$ then taking into account the fact that (p_1, n_1) and (p_2, n_2) are arcs in t_1 we deduce that $[(p_1, n_1), (p_1, n_2)] \in D_1$ or $[(p_1, n_2), (p_1, n_1)] \in D_1$. From Proposition 4.2 we deduce that $[(\alpha(p_1), \alpha(n_1)), (\alpha(p_1), \alpha(n_2))] \in D_2$ or $[(\alpha(p_1), \alpha(n_2)), (\alpha(p_1), \alpha(n_1))] \in D_2$. From (1) we have $\alpha(n_1) \neq \alpha(n_2)$.

- If $p_1 \neq p_2$ then by the inductive assumption we have $\alpha(p_1) \neq \alpha(p_2)$. But (p_1, n_1) and (p_2, n_2) are arcs in t_1. From Proposition 4.2 we deduce that $(\alpha(p_1), \alpha(n_1))$ and

$(\alpha(p_2), \alpha(n_2))$ are arcs in t_2. If $\alpha(n_1) = \alpha(n_2)$ then t_2 is not a tree. It follows that $\alpha(n_1) \neq \alpha(n_2)$.

c) Suppose that $n_1 \in \bigcup_{i=0}^{k} M_i$ and $n_2 \in M_{k+1}$. We have two cases:

- Suppose that $n_1 \in M_0$. In other words we have $n_1 = root(t_1)$. There is a path $d = (n_1, \ldots, n_2)$ of length $k+1$ in t_1. From Proposition 4.2 we deduce that $\alpha(d)$ is a path from $\alpha(n_1)$ to $\alpha(n_2)$ of length $k+1$. It follows that a $\alpha(n_1) \neq \alpha(n_2)$, otherwise the path $\alpha(d)$ is a circuit in t_2.

- Suppose that $n_1 \in M_r$, where $1 \leq r \leq k$. There is a sequence $(root(t_1), q_1, \ldots, q_{r-1}, n_1) \in Path(t_1)$. There is also a sequence $(root(t_1), s_1, \ldots, s_k, n_2) \in Path(t_1)$. From Proposition 4.2 we deduce that $d_1 = (\alpha(root(t_1)), \alpha(q_1), \ldots, \alpha(q_{r-1}), \alpha(n_1)) \in Path(t_2)$. Similarly we have $d_2 = (\alpha(root(t_1)), \alpha(s_1), \ldots, \alpha(s_k), \alpha(n_2)) \in Path(n_2)$. The length of d_1 is r, the length of d_2 is $k+1$ and $r < k+1$. If $\alpha(n_1) = \alpha(n_2)$ then there are two distinct paths from $\alpha(root(t_1))$ to $\alpha(n_1)$. This contradicts the fact that t_2 is a tree. Therefore $\alpha(n_1) \neq \alpha(n_2)$. ■

If $t = (A, D, h) \in OBT(\omega)$ and $\beta : A \longrightarrow B$ is an injective mapping then we consider the triple $t^\beta = (A^\beta, D^\beta, h^\beta)$, where the components of t^β are defined as follows:

$A^\beta = \beta(A)$

$D^\beta = \{ [(\beta(i), \beta(i_1)), (\beta(i), \beta(i_2))] \mid [(i, i_1), (i, i_2)] \in D\}$

$h^\beta : A^\beta \longrightarrow L$, $h^\beta(\beta(i)) = h(i)$

Proposition 4.4. *If $t \in OBT(\omega)$ then $t^\beta \in OBT(\omega)$ and $root(t^\beta) = \beta(root(t))$.*

Proof. We verify that $G^\beta = (A^\beta, D^\beta)$ is an ordered tree.

a) G^β is a directed ordered graph.

In order to verify this property we consider two elements $[(\beta(i), \beta(i_1)), (\beta(i), \beta(i_2))] \in D^\beta$ and $[(\beta(j), \beta(j_1)), (\beta(j), \beta(j_2))] \in D^\beta$. From the definition of D^β we obtain $[(i, i_1), (i, i_2)] \in D$ and $[(j, j_1), (j, j_2)] \in D$. But (A, D) is a directed ordered graph, therefore $i \neq j$. The mapping β is injective, therefore $\beta(i) \neq \beta(j)$.

b) Take $[(j, j_1), (j, j_2)] \in D^\beta$. There is $[(i, i_1), (i, i_2)] \in D$ such that $[(\beta(i), \beta(i_1)), (\beta(i), \beta(i_2))] = [(j, j_1), (j, j_2)]$. But (A, D) is an ordered tree, therefore $i_1 \neq i_2$. It follows that $\beta(i_1) \neq \beta(i_2)$ because β is an injective mapping. Thus $j_1 \neq j_2$.

c) The associated graph of G^β is a tree:

- Obviously (i, i_1) is an arc in the associated graph of (A, D) if and only if the pair $(\beta(i), \beta(i_1))$ is an arc in the associated graph of G^β. The sequence $(i_1, \ldots, i_n)$ is a path in the associated graph of (A, D) if and only if $(\beta(i_1), \ldots, \beta(i_n))$ is a path in the associated graph of G^β.

- We have $root(t^\beta) = \beta(root(t))$. There is no predecessor of $\beta(root(t))$ in the associated graph of G^β because if $(\beta(j), \beta(root(t)))$ is an arc then $(j, root(t))$ is an arc in the associated graph of (A, D) and this property contradicts the property of $root(t)$. For every $j \in A^\beta$ there is a path and only one from $root(t^\beta)$ to j. If $j \in A^\beta$ then there is $i \in A$ such that $j = \beta(i)$. There is a path $(root(t), i_1, \ldots, i_n, i)$ in the associated graph of (A, D), therefore $(root(t^\beta), \beta(i_1), \ldots, \beta(i_n), \beta(i))$ is a path in the associated graph of G^β from $root(t^\beta)$ to j. Suppose that $(root(t^\beta), j_1, \ldots, j_s, j)$ and $(root(t^\beta), p_1, \ldots, p_k, j)$ are two paths in the associated graph of G^β. There are $i_1, \ldots, i_s, q_1, \ldots, q_k \in A$ such that

$\beta(i_1) = j_1, \ldots, \beta(i_s) = j_s$

$\beta(q_1) = p_1, \ldots, \beta(p_k) = q_k$

$(root(t), i_1, \ldots, i_s, i)$ and $(root(t), q_1, \ldots, q_k, i)$ are paths in the associated graph of (A, D).

It follows that $s = k$ and $i_1 = q_1, \ldots, i_s = q_s$. Thus $j_1 = p_1, \ldots, i_s = p_s$ and $(root(t^\beta), j_1, \ldots, j_s, j) = (root(t^\beta), p_1, \ldots, p_k, j)$.
Let us verify the condition (2). Suppose that $j \in A^\beta$ and $[(j, j_1), (j, j_2)] \in D^\beta$. There is $[(i, i_1), (i, i_2)] \in D$ such that $[(\beta(i), \beta(i_1)), (\beta(i), \beta(i_2))] = [(j, j_1), (j, j_2)]$. The condition (2) is satisfied by t. It follows that $h(i) \in L_N$ and $\omega(h(i)) = (h(i_1), h(i_2))$. But $\omega(h^\beta(j)) = \omega(h^\beta(\beta(i))) = \omega(h(i))$ and $\omega(h(i)) \in L_N$, therefore $\omega(h^\beta(j)) \in L_N$. But $\omega(h(i)) = (h(i_1), h(i_2))$, therefore $\omega(h^\beta(j)) = (h(i_1), h(i_2))$. On the other hand $h^\beta(j_1) = h^\beta(\beta(i_1)) = h(i_1)$ and $h^\beta(j_2) = h^\beta(\beta(i_2)) = h(i_2)$. It follows that $\omega(h^\beta(j)) = (h^\beta(j_1), h^\beta(j_2))$. ∎

Proposition 4.5. *If* $t_1 \preceq t_2$ *then* $t_1 \preceq t_1^\beta \preceq t_2$.

Proof. Consider $t_1 = (A_1, D_1, h_1)$ and $t_2 = (A_2, D_2, h_2)$. Let us denote by α the embedding mapping of t_1 into t_2. This means that the following conditions are satisfied:

$$u \in D_1 \Rightarrow \overline{\alpha}(u) \in D_2 \tag{9}$$

$$h_2(\alpha(root(t_1))) = h_1(root(t_1)) \tag{10}$$

We consider the mapping $\gamma : A_1^\beta \longrightarrow A_2$ defined by

$$\gamma(\beta(i)) = \alpha(i) \tag{11}$$

for every $i \in A_1$.
In order to prove that $t_1^\beta \preceq t_2$ we have to verify the conditions (3) and (4) for these ω-trees.
First we verify the condition (3). Let us suppose that $v \in D_1^\beta$. From the definition of D_1^β we deduce that there is $u = [(i, i_1), (i, i_2)] \in D_1$ such that $v = [(\beta(i), \beta(i_1)), (\beta(i), \beta(i_2))]$. Based on this property we deduce that $\overline{\gamma}(v) = [(\gamma(\beta(i)), \gamma(\beta(i_1))), (\gamma(\beta(i)), \gamma(\beta(i_2)))] = [(\alpha(i), \alpha(i_1)), (\alpha(i), \alpha(i_2))]$. From (9) we deduce $\overline{\alpha}(u) \in D_2$. In conclusion the implication:

$$v \in D_1^\beta \Rightarrow \overline{\gamma}(v) \in D_2 \tag{12}$$

is true.
Now we verify (4). Based on the Proposition 4.4 we have $root(t_1^\beta) = \beta(root(t_1))$, therefore we can write $h_1^\beta(root(t_1^\beta)) = h_1^\beta(\beta(root(t_1)))$. According to the definition of h_1^β we have $h_1^\beta(\beta(i)) = h_1(i)$, therefore

$$h_1^\beta(root(t_1^\beta)) = h_1(root(t_1)) \tag{13}$$

Based on the same proposition we have $h_2(\gamma(root(t_1^\beta))) = h_2(\gamma(\beta(root(t_1))))$, therefore in virtue of (11) we obtain

$$h_2(\gamma(root(t_1^\beta))) = h_2(\alpha(root(t_1))) \tag{14}$$

From (10), (13) and (14) we obtain

$$h_1^\beta(root(t_1^\beta)) = h_2(\gamma(root(t_1^\beta))) \tag{15}$$

From (12) and (15) we deduce that $t_1^\beta \preceq t_2$ and moreover, γ is an embedding mapping of t_1^β into t_2.
Directly from definition of t_1^β we have

$$v \in D_1 \Rightarrow \overline{\beta}(v) \in D_1^\beta$$

and from (13) we have $h_1^\beta(\beta(root(t_1))) = h_1(root(t_1))$. Thus we verified that $t_1 \preceq t_1^\beta$. ■

Corollary 4.1. *If* $t_1 \preceq t_2$ *and* α *is an embedding mapping of* t_1 *into* t_2 *then* $t_1 \preceq t_1^\alpha \preceq t_2$.

Proposition 4.6. *The relation* $\preceq$ *is reflexive and transitive, but is not antisymmetric.*

Proof. Suppose that $t_1 = (A_1, D_1, h_1) \in OBT(\omega)$. Taking the identity mapping $\alpha : A_1 \longrightarrow A_1$, $\alpha(x) = x$ for every $x \in A_1$ we find that $t_1 \preceq t_1$ because from Corollary 4.1 we have $t_1 \preceq t_1^\alpha$ and $t_\alpha = t_1$ for our choice of α. Therefore the relation $\preceq$ is reflexive.
Now we suppose that $t_1 = (A_1, D_1, h_1)$, $t_2 = (A_2, D_2, h_2)$ and $t_3 = (A_3, D_3, h_3)$ are three ω-labeled trees such that $t_1 \preceq t_2$ and $t_2 \preceq t_3$. There is an embedding mapping $\alpha : A_1 \longrightarrow A_2$ of t_1 into t_2 and an embedding mapping $\beta : A_2 \longrightarrow A_3$ of t_2 into t_3. It follows that the following conditions are satisfied:

$$u \in D_1 \Rightarrow \overline{\alpha}(u) \in D_2 \tag{16}$$

$$u \in D_2 \Rightarrow \overline{\beta}(u) \in D_3 \tag{17}$$

$$h_1(root(t_1)) = h_2(\alpha(root(t_1))) \tag{18}$$

We consider the mapping $\alpha \circ \beta : A_1 \longrightarrow A_3$ defined by $\alpha \circ \beta(x) = \beta(\alpha(x))$. From (16) and (17) we obtain

$$u \in D_1 \Rightarrow \overline{\alpha \circ \beta}(u) \in D_3 \tag{19}$$

because $\overline{\alpha \circ \beta}(u) = \overline{\beta}(\overline{\alpha}(u))$.
If we use Proposition 4.1 then we obtain $h_3(\beta(j)) = h_2(j)$ for every $j \in A_2$. It follows that $h_3(\alpha \circ \beta(root(t_1))) = h_3(\beta(\alpha(root(t_1)))) = h_2(\alpha(root(t_1)))$. Now if we use (18) then we obtain

$$h_3(\alpha \circ \beta(root(t_1))) = h_1(root(t_1)) \tag{20}$$

From (19) and (20) we obtain $t_1 \preceq t_3$ and thus the relation $\preceq$ is transitive.
If β is not the identity mapping then applying Proposition 4.5 for the case $t \preceq t$ we obtain $t \preceq t^\beta$ and $t^\beta \preceq t$. But $t \neq t^\beta$ because β is not the identity mapping. Thus the relation $\preceq$ is not antisymmetric. ■

Remark 4.1. *The relation* $\preceq$ *is not based on the concept of subtree of a tree. More precisely, if* $t_1 \preceq t_2$ *and* α *is an embedding mapping of* t_1 *into* t_2 *then* t_1^α *is not necessarily a subtree of* t_2*. This can be viewed if we consider again the case presented in Figure 2. It is easy to see that* t_1^α *is not a subtree of* t_2 *because the node* m_9 *has two direct descendants in* t_2.

5. Conclusions and future work

In this paper we considered a set L of elements such that $L = L_N \cup L_T$, where $L_N \cap L_T = \emptyset$ and for a given mapping $\omega : L_N \longrightarrow L \times L$ we defined the set $OBT(\omega)$ of ω-trees. An element of $OBT(\omega)$ is a binary tree, an ordered tree and a labeled tree. The labels of such a structure are assigned to nodes by a rule defined as follows: if n is a node labeled by $k \in L_N$ and n_1, n_2 are the left descendant and respectively the right descendant of n then the label of n_1 is k_1 and the label of n_2 is k_2 if $\omega(k) = (k_1, k_2)$. We defined the embedding operation of an ω-tree into another ω-tree. This operation allows to define a binary relation on $OBT(\omega)$, such that this is a reflexive and transitive relation, but is not an antisymmetric one. Several algebraic properties are proved and the concepts are exemplified.

The subject presented in this paper will be continued in [15]. Based on the relation $\preceq$ we shall define an equivalence relation on the set $OBT(\omega)$. The factor set of equivalence classes is organized as a partially ordered set and the maximal elements of this set are characterized in [15]. An equivalence class can be viewed as a template and the maximal templates are used in a forthcoming paper to relieve several algebraic properties of the formal computations in a master-slave system based on semantic schemas.

References

[1] A.V.Aho and J.D.Ullman, *The Theory of Parsing, Translation, and Compiling*, Prentice-Hall, Vol. I, 1972.

[2] S. Bhattacharya and S. K. Ghosh, An Artificial Intelligence Based Approach for Risk Management Using Attack Graph, *Proceedings of the 2007 International Conference on Computational Intelligence and Security* (2007), 794–798.

[3] M. Chein and M.L. Mugnier, Graph-based Knowledge Representation-Computational Foundations of Conceptual Graphs, Series: *Advanced Information and Knowledge Processing* (2008).

[4] Y. Y. Hsu, W. C. Chen, P. H. Tai and M. D. Tsai, An Enhanced Graph and Artificial Intelligence-based Approach for Assembly Sequence Planning, *Proceedings of the 35th International MATADOR Conference* (2007), 49–52.

[5] A. Ibeas and M. de la Sen, Artificial Intelligence and Graph Theory Tools for Describing Switched Linear Control Systems, *Applied Artificial Intelligence* **20** (2006), no. 9, 703–741.

[6] E. Miranda and M. Zaffalon, Coherence graphs, *Artificial Intelligence* **173** (2009), no. 1, 104–144.

[7] J.F.Sowa, Conceptual graphs for a database interface, *IBM Journal of Research and Development* **20** (1976), no. 4, 336–357.

[8] P.J.Tan and D.L.Dowe, Decision Forests with Oblique Decision Trees, *MICAI 2006: Advances in Artificial Intelligence, Lecture Notes in Computer Science* **4293** (2006), 593–603.

[9] N. Ţăndăreanu, Collaborations between distinguished representatives for labelled stratified graphs, *Annals of the University of Craiova, Mathematics and Computer Science Series* **30** (2003), no. 2, 184–192.

[10] N. Ţăndăreanu, Distinguished Representatives for Equivalent Labelled Stratified Graphs and Applications, *Discrete Applied Mathematics* **144** (2004), no. 1-2, 183–208.

[11] N. Ţăndăreanu, Knowledge representation by labeled stratified graphs, *Proceedings of the 8th World Multi-Conference on Systemics, Cybernetics and Informatics* **5** (2004), 345–350.

[12] N. Ţăndăreanu, Semantic Schemas and Applications in Logical Representation of Knowledge, *Proceedings of the 10th Int. Conf. on CITSA* **III** (2004), 82–87.

[13] N. Ţăndăreanu, Master-Slave Systems of Semantic Schemas and Applications, *The 10th IASTED International Conference on Intelligent Systems and Control* (2007), 150–155.

[14] N. Ţăndăreanu, An Extension of Semantic Schemas to Represent Multiple Meanings, *International Multi-Conference on Engineering and Technological Innovation* **I** (2008), 143–148.

[15] N. Ţăndăreanu and C.Zamfir, Algebraic properties of ω-trees (II), *Annals of the University of Craiova, Mathematics and Computer Science Series* **37** (2010), no. 2 (to appear).

[16] G.Valiente, Using layered support graphs for verifying external adequacy in rule-based expert systems, *ACM SIGART Bulletin* **3** (1992), no. 1, 20–24.

(Nicolae Ţăndăreanu and Cristina Zamfir) Department of Informatics, University of Craiova, Al.I. Cuza Street, No. 13, Craiova RO-200585, Romania, Tel. & Fax: 40-251412673

E-mail address: ntand@rdslink.ro, cristina.zamfir@star-storage.ro