Cooperating Structures Based on Semantic Schemas in Knowledge Representation

Nicolae ŢĂNDĂREANU

Faculty of Mathematics and Computer Science,
Department of Computer Science,
University of Craiova, Romania
ntand@rdslink.ro

Abstract. Based on the concept of semantic schema we describe in this paper three structures, each of them allowing the cooperation between several semantic schemas. The first structure is a master-slave system, the second structure is based on the maximal graphs in semantic schemas and the last structure is obtained by means of a binary operation named cross-product of two schemas. For each structure the semantical computations are defined and the concepts are exemplified on particular cases.
Keywords: labeled graph, semantic schema, cooperating system, interpretation, master schema, slave schema, maximal graph
AMS Subject Classification 2000: 68T30

1 Introduction

The systems of cooperating structures are more and more used to satisfy a common task. A cooperating system is a collection of devices such that based on some strategy of communication between them, a common task can be performed. Various cooperating systems were introduced: cooperating grammar systems ([2]), cooperating mobile robots ([12], [13]), master-slave systems ([14]), neural networks and so on.

In this paper we present three cooperating structures that are based on semantic schemas. The structure of the paper is the following. In Section 2 the basic concepts connected by semantic schemas are presented. In Section 3 we define the concepts of master-schema, slave-schema and the cooperation between them. In Section 4 we define the maximal graph of a semantic schema and the cooperation based on this structure. In Section 5 we introduce a binary operation between semantic schema named the cross-product of two semantic schemas. The last section contains two ideas to extend the subject presented in this paper.

2 Semantic schemas

We consider a finite and nonempty set A_0 and we denote by θ an operator symbol of arity 2. We denote by $\overline{A}_0$ *the Peano θ-algebra generated by* A_0,

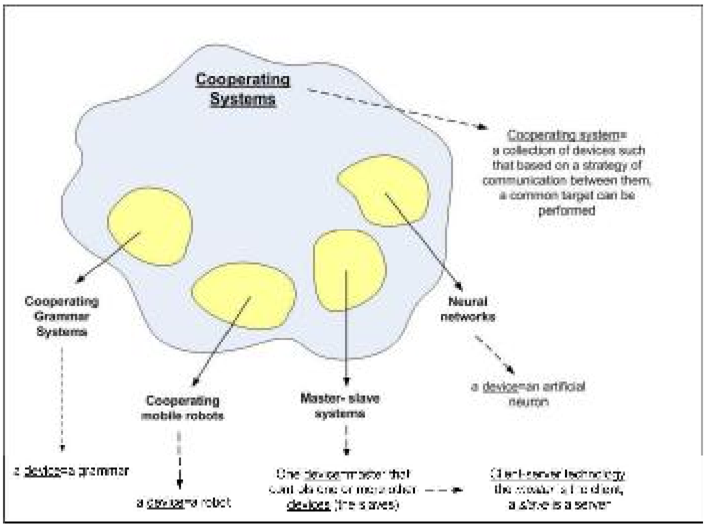

Fig. 1. Cooperating systems

therefore $\overline{A}_0 = \bigcup_{n\geq 0} A_n$ where A_n is defined recursively ([1], [8]) by

$$A_{k+1} = A_k \cup \{\ \theta(u,v)\ \ |\ \ u, v \in A_k\},\ \ k \geq 0$$

If we take

$$B_0 = A_0,\ \ B_{n+1} = A_{n+1} \setminus A_n \tag{1}$$

then $\overline{A}_0 = \bigcup_{n\geq 0} B_n$ and $B_i \cap B_j = \emptyset$ for $i \neq j$. For $u \in \overline{A}_0$ we write $length(u) = n$ if $u \in B_n$.

In what follows we recall the main results concerning the concept of θ-semantic schema introduced in [3] and developed in [4], [5], [6] and [7]. We mention in this section only those results that are used in this paper.

A **θ-semantic schema** (shortly, θ-schema) is a system $\mathcal{S} = (X, A_0, A, R)$, where

• X is a finite non-empty set of symbols and its elements are named *object symbols*

• A_0 is a finite non-empty set of elements named *label symbols* and $A_0 \subseteq A \subseteq \overline{A}_0$, where $\overline{A}_0$ is the Peano θ-algebra generated by A_0

• $R \subseteq X \times A \times X$ is a non-empty set which fulfills the following conditions:

$$\begin{aligned}(x, \theta(u,v), y) \in R \Longrightarrow \exists z \in X :\\ (x,u,z) \in R, (z,v,y) \in R\end{aligned} \tag{2}$$

$$\left.\begin{array}{l}\theta(u,v)\in A,\\(x,u,z)\in R,\\(z,v,y)\in R\end{array}\right\}\Rightarrow(x,\theta(u,v),y)\in R \tag{3}$$

$$u\in A\Longleftrightarrow\exists(x,u,y)\in R \tag{4}$$

We denote $R_0 = R \cap (X \times A_0 \times X)$.

Proposition 1. *If $\theta(u,v) \in A$ then $u \in A$ and $v \in A$.*

Proof. If $\theta(u,v) \in A$ then by (4) and (2) we deduce that there are $(x,u,y) \in R$ and $(y,v,z) \in R$. Using again (4) we obtain $u \in A$ and $v \in A$.

Let $\mathcal{S} = (X, A_0, A, R)$ be a θ-schema. If h is a symbol of arity 1 then we consider the set:

$$M = \Big\{h(x,a,y) \quad | \quad (x,a,y) \in R_0\Big\}$$

where we use the notation $h(x,a,y)$ instead of $h((x,a,y))$.

We consider a symbol σ of arity 2 and let $\mathcal{H}$ be *the Peano σ-algebra generated by M*.

We denote by Z the alphabet including the symbol σ, the elements of X, the elements of A, the left and right parentheses, the symbol h and comma. We denote by Z^* the set of all words over Z. We define the following binary relation on Z^*:

Let be $w_1, w_2 \in Z^*$.

- If $a \in A_0$ and $(x,a,y) \in R$ then $w_1(x,a,y)w_2 \Rightarrow w_1h(x,a,y)w_2$
- Let be $(x,\theta(u,v),y) \in R$. If $(x,u,z) \in R$ and $(z,v,y) \in R$ then

$$w_1(x,\theta(u,v),y)w_2 \Rightarrow w_1\sigma((x,u,z),(z,v,y))w_2$$

We denote by $\Rightarrow^*$ the reflexive and transitive closure of the relation $\Rightarrow$.

The **mapping generated** by $\mathcal{S}$ is the mapping $\mathcal{G}_\mathcal{S} : R \longrightarrow 2^\mathcal{H}$ defined as follows:

- $\mathcal{G}_\mathcal{S}(x,a,y) = \{h(x,a,y)\}$ for $a \in A_0$
- $\mathcal{G}_\mathcal{S}(x,\theta(u,v),y) = \{w \in \mathcal{H} \mid (x,\theta(u,v),y) \Rightarrow^* w\}$

We denote $\mathcal{F}(\mathcal{S}) = \bigcup_{(x,u,y)\in R} \mathcal{G}_\mathcal{S}(x,u,y)$.

An **interpretation** ([7]) of $\mathcal{S}$ is a system $\mathcal{I} = (Ob, ob, \{Alg_u\}_{u\in A})$, where

- Ob is a finite set of elements named the **objects** of $\mathcal{I}$
- $ob : X \rightarrow Ob$ is a bijective function
- $\{Alg_u\}_{u\in A}$ is a set of algorithms such that each algorithm has two input parameters and one output parameter.

Consider an interpretation $\mathcal{I} = (Ob, ob, \{Alg_u\}_{u\in A})$ of $\mathcal{S}$. The **output space** Y of $\mathcal{I}$ is the set $Y = \bigcup_{u\in A} Y_u$, where

$$Y_a = \{Alg_a(ob(x), ob(y)) | (x,a,y) \in R_0\}$$

if $a \in A_0$ and otherwise

$$Y_{\theta(u,v)} = \{Alg_{\theta(u,v)}(o_1, o_2) | o_1 \in Y_u, o_2 \in Y_v\}$$

We define recursively the **valuation mapping**

$$Val_{\mathcal{I}} : \mathcal{F}(\mathcal{S}) \longrightarrow Y$$

as follows:

- $Val_{\mathcal{I}}(h(x, a, y)) = Alg_a(ob(x), ob(y))$
- $Val_{\mathcal{I}}(\sigma(\alpha, \beta)) = Alg_{\theta(u,v)}(Val_{\mathcal{I}}(\alpha), Val_{\mathcal{I}}(\beta))$ if $\sigma(\alpha, \beta)$ is derived from an element of the form $(x, \theta(u, v), y) \in R$ (in fact this element is uniquely determined, [6]).

An intuitive representation of the computations is given in Figure 2.

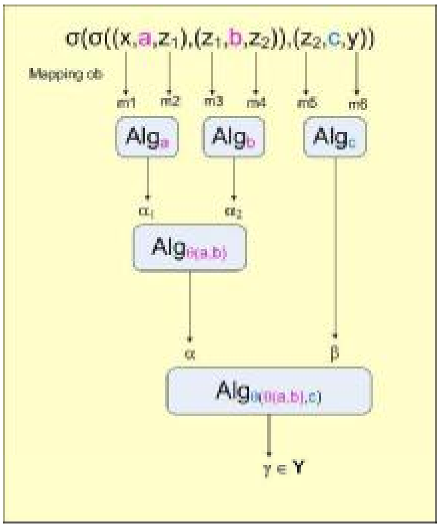

Fig. 2. Intuitive computations

We consider a finite and nonempty set A_0 and we denote by $\Theta = \{\theta_1, \ldots, \theta_n\}$ a finite set of operator symbols of arity 2. We denote by $\overline{A}_0$ *the Peano Θ-algebra generated by* A_0, therefore $\overline{A}_0 = \bigcup_{n \geq 0} A_n$ where A_k are defined recursively as follows ([3]):

$$A_{k+1} = A_k \cup \{ \ \theta(u, v) \ \ | \ \ \theta \in \Theta, \ \ u, v \in A_k\}, \tag{5}$$

In what follows we introduce the concept of Θ-schema as an extension of θ-semantic schema.

Definition 1. *A Θ-***semantic schema** *(shortly, Θ-schema) is a system $\mathcal{S} = (X, A_0, A, R)$, where*

- *X is a finite non-empty set of symbols and its elements are named object symbols*
- *A_0 is a finite non-empty set of elements named label symbols and $A_0 \subseteq A \subseteq \overline{A}_0$, where $\overline{A}_0$ is the Peano Θ-algebra generated by A_0*
- *$R \subseteq X \times A \times X$ is a non-empty set which fulfills the conditions (2), (3) and (4).*

Remark 1. Obviously a Θ-schema for $n = 1$ is a θ-schema, where $\Sigma = \{\theta\}$.

3 Master-slave cooperation

In this section we define a cooperating system of semantic schemas that uses the general description of a master-slave system. Such a system includes several distinct semantic schemas, each of them being a **slave**-schema. The system contains also a special structure named **master**-schema. The cooperation is described by the formal and semantic computations of the master-schema.

Definition 2. *([9]) We consider the binary symbols $\theta, \theta_1, \ldots, \theta_n$. The pair $MS_{n+1} = (\mathcal{S}, \{\mathcal{S}_i\}_{i=1}^n)$ is a* **master-slave system (of semantic schemas)** *with $n+1$ components if the following conditions are satisfied:*

- *For $i \in \{1, \ldots, n\}$ the entity $\mathcal{S}_i = (X_i, A_{0i}, A_i, R_i)$ is a θ_i-schema; we say that $\mathcal{S}_i$ is a* ***slave-schema*** *of MS_{n+1}.*
- *$\mathcal{S} = (X, A_0, A, R)$ is a θ-schema, named the* ***master-schema*** *of MS_{n+1}, such that $R_0 \subseteq \bigcup_{i=1}^n R_i$, where $R_0 = R \cap (X \times A_0 \times X)$.*

As a general notation, if $M \subseteq X_1 \times \ldots \times X_n$ and $i \in \{1, \ldots, n\}$ then

$$pr_i M = \{x \mid \exists (x_1, \ldots, x_{i-1}, x, x_{i+1}, \ldots, x_n) \in M\}$$

It is not difficult to observe that two arbitrary components of MS_{n+1} are distinct schemas because they use distinct binary symbols. Moreover, $A_0 = pr_2 R_0 \subseteq pr_2(\bigcup_{i=1}^n R_i) = \bigcup_{i=1}^n A_i$ because $pr_2 R_i = A_i$.

In order to exemplify this concept we consider the θ_1-schema $\mathcal{S}_1$ from Figure 3 and the θ_2-schema $\mathcal{S}_2$ from Figure 4 as slave schemas ([9]), where $p_1 = \theta_1(b_1, a_1)$, $p_2 = \theta_2(a_3, b_2)$, $p_3 = \theta_2(a_2, b_1)$, $p_4 = \theta_2(a_1, a_2)$, $p_5 = \theta_1(a_1, a_2)$ and $q_1 = \theta_1(a_2, a_1)$.

Take the set R_0 represented in Figure 5. We can obtain the master schema $\mathcal{S} = (X, A_0, A, R)$ if we take

- $X = pr_1 R_0 \cup pr_3 R_0$; $A_0 = pr_2 R_0$
- $R = R_0 \cup \{(z_1, \theta(a_1, p_1), z_3), (x_3, \theta(p_2, a_1), y_2), (z_1, \theta(a_1, p_3), z_4), (z_1, \theta(p_4, p_5), x_3), (x_3, \theta(p_2, p_4), x_1), (x_1, \theta(p_5, p_2), z_1), (x_1, \theta(\theta(p_5, p_2), a_1), y_2)\}$
- $A = pr_2 R$

We obtain the master-slave system $MS_3 = (\mathcal{S}, \{\mathcal{S}_1, \mathcal{S}_2\})$.

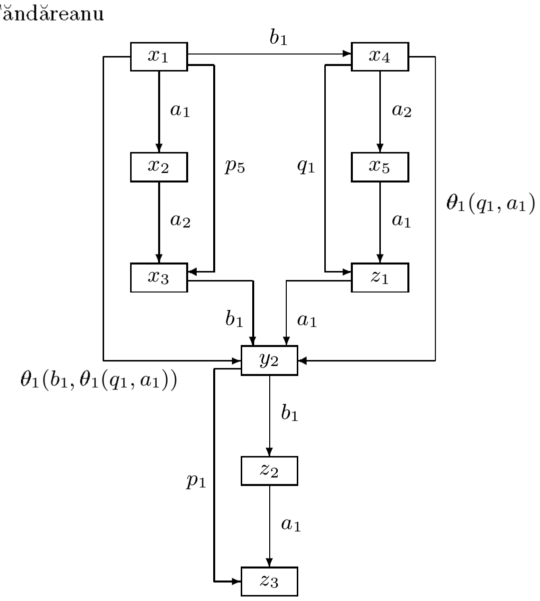

Fig. 3. θ_1-schema $\mathcal{S}_1$

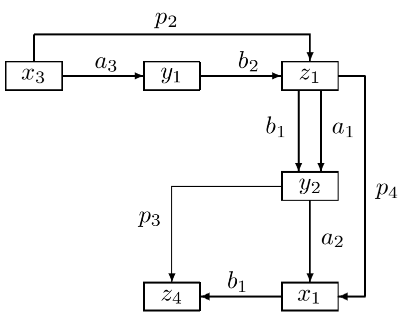

Fig. 4. θ_2-schema $\mathcal{S}_2$

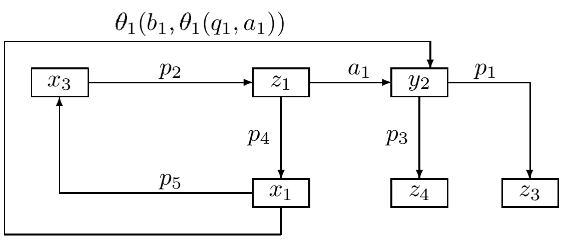

Fig. 5. The set R_0

Definition 3. *([9]) Consider a slave-schema $\mathcal{S}_i$, a symbol σ_i of arity 2 and a symbol h_i of arity 3. We denote $R_{0i} = R_i \cap (X_i \times A_{0i} \times X_i)$. We define the **direct derivation** $\Rightarrow_i$ as follows: if w_1 and w_2 are arbitrary words then*

- *for $(x, a, y) \in R_{0i}$ we write*

$$w_1(x, a, y)w_2 \Rightarrow_i w_1 h_i(x, a, y)w_2$$

- *for $(x, \theta_i(u, v), y) \in R_i$, $(x, u, z) \in R_i$ and $(z, v, y) \in R_i$ we write*

$$w_1(x, \theta_i(u, v), y)w_2 \Rightarrow_i w_1\sigma_i((x, u, z), (z, v, y))w_2$$

*We write $\alpha \Rightarrow_i^* \beta$ if either $\alpha = \beta$ or there are $\gamma_1, \ldots, \gamma_{k+1}$ such that $\gamma_1 = \alpha$, $\gamma_{k+1} = \beta$ and $\gamma_i \Rightarrow \gamma_{i+1}$ for $i \in \{1, \ldots, k\}$. If this is the case then the sequence $\gamma_1, \ldots, \gamma_{k+1}$ is a **derivation** of β from α in $\mathcal{S}_i$. If $M_i = \{h_i(x, a, y) \mid (x, a, y) \in R_{0i}\}$ and $\mathcal{H}_i$ is the Peano σ_i-algebra generated by M_i then we take*

$$\mathcal{F}(\mathcal{S}_i) = \{w \in \mathcal{H}_i \mid \exists (x, u, y) \in R_i : (x, u, y) \Rightarrow_i^* w\}$$

*A **formal computation** for $w \in \mathcal{F}(\mathcal{S}_i)$ is a derivation of w from an element of R_i.*

Remark 2. If $w \in \mathcal{F}(\mathcal{S}_i)$ then there is an element and only one $(x, \theta_i(u, v), y) \in R_i$ such that $(x, \theta_i(u, v), y) \Rightarrow_i^* w$. This property allows us to denote $type(w) = \theta_i(u, v)$.

The elements of the set $\mathcal{F}(\mathcal{S}_i)$ are abstract entities. In order to obtain the meaning of these elements we use a *semantical computation*. This computation is obtained by means of an **interpretation**.

Definition 4. *([9]) An **interpretation** $\mathcal{J}_i$ of $\mathcal{S}_i$ is a system*

$$\mathcal{J}_i = (Ob_i, ob_i, \{Alg_u^i\}_{u \in A_i})$$

where

- *Ob_i is a finite set of elements named* **objects***;*
- *$ob_i : X_i \longrightarrow Ob_i$ is a bijective function;*
- *$\{Alg_u^i\}_{u \in A_i}$ is a set of algorithms such that each algorithm has two input parameters and one output parameter.*

The **output space** Y_i of $\mathcal{J}_i$ is defined by $Y_i = \bigcup_{u \in A_i} Y_u^i$, where

- $Y_a^i = \{Alg_a^i(ob_i(x), ob_i(y)) \mid (x, a, y) \in R_{0i}\}$
- $Y_{\theta_i(u,v)}^i = \{Alg_{\theta_i(u,v)}^i(o_1, o_2) \mid o_1 \in Y_u^i, o_2 \in Y_v^i\}$

We define recursively the **valuation mapping**

$$Val_{\mathcal{J}_i} : \mathcal{F}(\mathcal{S}_i) \longrightarrow Y_i$$

as follows:

- $Val_{\mathcal{J}_i}(h_i(x,a,y)) = Alg_a^i(ob_i(x), ob_i(y))$
- $Val_{\mathcal{J}_i}(\sigma_i(\alpha,\beta)) = Alg_{\theta_i(u,v)}(Val_{\mathcal{J}_i}(\alpha), Val_{\mathcal{J}_i}(\beta))$

if $type(\sigma_i(\alpha,\beta)) = \theta_i(u,v)$.
The **meaning** of $w \in \mathcal{F}(\mathcal{S}_i)$ is $Val_{\mathcal{J}_i}(w)$.

The formal computation in a master-schema is based on some specific **derivation** and this concept is introduced in the next definition.

Definition 5. *([9]) Let $\mathcal{S} = (X, A_0, A, R)$ be a master-schema. Consider a symbol σ of arity 2. If $(x, \theta(u,v), y) \in R$, $(x,u,z) \in R$ and $(z,v,y) \in R$ then*

$$w_1(x, \theta(u,v), y)w_2 \mapsto w_1\sigma((x,u,z),(z,v,y))w_2$$

for any words w_1, w_2. We write $\alpha \mapsto^* \beta$ if either $\alpha = \beta$ or there are $\gamma_1, \ldots, \gamma_{m+1}$ such that $\gamma_1 = \alpha$, $\gamma_{m+1} = \beta$ and $\gamma_i \mapsto \gamma_{i+1}$ for $i \in \{1,\ldots,m\}$. If this is the case then the sequence $\gamma_1,\ldots,\gamma_{m+1}$ is a ***derivation*** *of β from α in $\mathcal{S}$.*

We denote by $\mathcal{H}$ the Peano σ-algebra generated by R_0. We define

$$\mathcal{F}(\mathcal{S}) = \{w \in \mathcal{H} \mid \exists (x,u,y) \in R : (x,u,y) \mapsto^* w\}$$

A **formal computation** for $w \in \mathcal{F}(\mathcal{S})$ is a derivation of w from an element of R.

Proposition 2. *In a master-slave system we have $\mathcal{F}(\mathcal{S}) \supseteq R_0$ and $X \subseteq \bigcup_{i=1}^n X_i$.*

Proof. $\mathcal{H}$ is generated by R_0, therefore $R_0 \subseteq \mathcal{H}$. Taking $(x,u,y) \in R_0$ we have $(x,u,y) \in \mathcal{H}$ and by the reflexivity of $\mapsto^*$ we have $(x,u,y) \mapsto^* (x,u,y)$. Thus $(x,u,y) \in \mathcal{F}(\mathcal{S})$. The second part is obvious.

Remark 3. If $w \in \mathcal{F}(\mathcal{S})$ then there is an element and only one $(x,\theta(u,v),y) \in R$ such that $(x,\theta(u,v),y) \mapsto^* w$. This property allows us to denote $type(w) = \theta(u,v)$.

An example of computation is described below for the master-slave system MS_3:

$$(x_1, \theta(\theta(\theta_1(a_1,a_2), \theta_2(a_3,b_2)), a_1), y_2) \mapsto$$
$$\sigma((x_1, \theta(\theta_1(a_1,a_2), \theta_2(a_3,b_2)), z_1), (z_1, a_1, y_2))$$

On the other hand

$$(x_1, \theta(\theta_1(a_1,a_2), \theta_2(a_3,b_2)), z_1) \mapsto \sigma((x_1, \theta_1(a_1,a_2), x_3), (x_3, \theta_2(a_3,b_2), z_1))$$

therefore

$$(x_1, \theta(\theta(\theta_1(a_1,a_2), \theta_2(a_3,b_2)), a_1), y_2) \mapsto^+$$
$$\sigma(\sigma((x_1, \theta_1(a_1,a_2), x_3), (x_3, \theta_2(a_3,b_2), z_1)), (z_1, a_1, y_2))$$

This computation proves that the element

$$\sigma(\sigma((x_1, \theta_1(a_1,a_2), x_3), (x_3, \theta_2(a_3,b_2), z_1)), (z_1, a_1, y_2))$$

belongs to $\mathcal{F}(\mathcal{S})$.

The semantical computations in a master-schema are defined taking into account the semantical computations in the slave-schemas of the system. First we introduce the concept of interpretation for a master-schema.

Definition 6. *([9])*
*Suppose that $(\mathcal{S}, \{\mathcal{S}_i\}_{i=1}^{n})$ is a master-slave system of semantic schemas, where $\mathcal{S} = (X, A_0, A, R)$ and $\mathcal{S}_i = (X_i, A_{0i}, A_i, R_i)$ for $i \in \{1, \ldots, n\}$. For each $i \in \{1, \ldots, n\}$ consider an interpretation $\mathcal{I}_i = (Ob_i, ob_i, \{Alg_u^i\}_{u\in A_i})$ for $\mathcal{S}_i$ such that if $x \in X_i \cap X_j$ for $i \neq j$ then $ob_i(x) = ob_j(x)$. An **interpretation** of $\mathcal{S}$ is a system $\mathcal{I} = (Ob, ob, \{Alg_u\}_{u\in A})$ such that $ob : X \longrightarrow Ob$ is defined by $ob(x) = ob_i(x)$ if $x \in X_i$ and Alg_u is an algorithm with two input arguments and one output argument.*

We define the space Y and the **valuation mapping**

$$Val_{\mathcal{I}} : \mathcal{F}(\mathcal{S}) \longrightarrow 2^Y$$

as follows:

- If $(x, a, y) \in \bigcup_{i=1}^{n}(R_{0i} \cap R_0)$ then

$$Val_{\mathcal{I}}(x, a, y) = \bigcup_{i=1}^{n}\{Alg_a^i(ob_i(x), ob_i(y))\}$$

Take $Y_a = \bigcup_{x,y} Val_{\mathcal{I}}(x, a, y)$.

- If $(x, \theta_i(u, v), y) \in R_i \cap R_0$, $(x, \theta_i(u, v), y) \Rightarrow_i^* \sigma_i(w_1, w_2)$ and $\sigma_i(w_1, w_2) \in \mathcal{F}(\mathcal{S}_i)$ then

$$Val_{\mathcal{I}}(x, \theta_i(u, v), y) = \{Val_{\mathcal{I}_i}(\sigma_i(w_1, w_2)\}$$

Take $Y_{\theta_i(u,v)} = \bigcup_{x,y} Val_{\mathcal{I}}(x, \theta_i(u, v), y)$.

- If $\sigma(\alpha, \beta) \in \mathcal{F}(\mathcal{S})$ and $type(\sigma(\alpha, \beta)) = \theta(u, v)$ then

$$Val_{\mathcal{I}}(\sigma(\alpha, \beta)) = \bigcup_{\substack{o_1 \in Val_{\mathcal{I}}(\alpha), \\ o_2 \in Val_{\mathcal{I}}(\beta),}} \{Alg_{\theta(u,v)}(o_1, o_2)\}$$

Take $Y_{\theta(u,v)} = \{Alg_{\theta(u,v)}(o_1, o_2) \mid o_1 \in Y_u, o_2 \in Y_v\}$ and $Y = \bigcup_{u\in A} Y_u$.

As an example of computation we consider the following cases for MS_3:

- $Val_{\mathcal{I}}(z_1, a_1, y_2) = \{Alg_{a_1}^1(ob_1(z_1), ob_1(y_2)), Alg_{a_1}^2(ob_2(z_1), ob_2(y_2))\}$
- $Val_{\mathcal{I}}(y_2, \theta_2(a_2, b_1), z_4) = \{Alg_{\theta_2(a_2,b_1)}(q_1, q_2)\}$, where $q_1 = Alg_{a_2}^2(ob_2(y_2), ob_2(x_1))$, $q_2 = Alg_{b_1}^2(ob_2(x_1), ob_2(z_4))$.
- If we denote $o_1 = Alg_{a_1}^1(ob_1(z_1), ob_1(y_2))$, $o_2 = Alg_{a_1}^2(ob_2(z_1), ob_2(y_2))$, $o_3 = Alg_{\theta_2(a_2,b_1)}(Alg_{a_2}^2(ob(y_2), ob_2(x_1)), Alg_{b_1}^2(ob_2(x_1), ob_2(z_4)))$ then based on the derivation

$$(z_1, \theta(a_1, \theta_2(a_2, b_1)), z_4) \mapsto \sigma((z_1, a_1, y_2), (y_2, \theta_2(a_2, b_1), z_4))$$

we obtain

$$Val_{\mathcal{I}}(z_1, \theta(a_1, \theta_2(a_2, b_1)), z_4) =$$

$$\{Alg_{\theta(a_1,\theta_2(a_2,b_1))}(o_1, o_3), Alg_{\theta(a_1,\theta_2(a_2,b_1))}(o_2, o_3)\}$$

Definition 7. *We consider a master-slave system* $MS_{n+1} = (\mathcal{S}, \{\mathcal{S}_i\}_{i=1}^n)$, *an interpretation* $\mathcal{I}$ *of* MS_{n+1} *and the interpretation* $\mathcal{I}_i$ *for* $\mathcal{S}_i$, $i \in \{1, \ldots, n\}$. *An* **interrogation** *of the master-slave system* MS_{n+1} *is a pair* (x, y) *of nodes from* $\mathcal{S}$. *The* **answer** *given by* $\mathcal{S}$ *is the entity*

$$Ans_{\mathcal{S}}(x, y) = \bigcup_{u \in A} \bigcup_{\sigma(\alpha,\beta) \in D(x,u,y)} Val_{\mathcal{I}}(\sigma(\alpha, \beta))$$

where $D(x, u, y) = \{\sigma(\alpha, \beta) \in \mathcal{H} \mid (x, u, y) \mapsto^* \sigma(\alpha, \beta)\}$

4 Cooperating based on maximal graphs

In this section we define another kind of cooperation, which is based on the concept of maximal graph of a semantic schema.

A *labeled graph* is a tuple $G = (S, L_0, T_0, f_0)$, where
- S is a finite set, an element of S is a *node* of G;
- L_0 is a set of elements named *labels*;
- T_0 is a set of binary relations on S;
- $f_0 : L_0 \longrightarrow T_0$ is a surjective mapping.

We use in what follows the *union* of two labeled graphs. In order to define this operation we consider the labeled graphs $G_1 = (S, L_0, T_0, f_0)$ and $G_2 = (Q, M_0, K_0, g_0)$, where $T_0 \subseteq 2^{S \times S}$ and $K_0 \subseteq 2^{Q \times Q}$. The union of G_1 and G_2 is the labeled graph $G_1 \cup G_2 = (S \cup Q, L_0 \cup M_0, W_0, h_0)$, where

$$h_0(\alpha) = \begin{cases} f_0(\alpha) \text{ if } \alpha \in \mathrm{L}_0 \setminus \mathrm{M}_0 \\ g_0(\alpha) \text{ if } \alpha \in \mathrm{M}_0 \setminus \mathrm{L}_0 \\ f_0(\alpha) \cup g_0(\alpha) \text{ if } \alpha \in \mathrm{L}_0 \cap \mathrm{M}_0 \end{cases}$$

Obviously we have $W_0 = h_0(L_0 \cup M_0)$.

For a θ-semantic schema $\mathcal{S} = (X, A_0, A, R)$ we can build the labeled graph $G_{\mathcal{S}} = (X, A, T, f)$, named the **labeled graph associated** to $\mathcal{S}$, where
- $f(\alpha) = \{(x, y) \in X \times X \mid (x, \alpha, y) \in R\}$
- $T = \{f(\alpha) \mid \alpha \in A\}$

We introduce now a partial relation on the component R of $\mathcal{S}$.

Definition 8. *For two elements* $(y_1, u_1, y_2) \in R$ *and* $(x_1, v_1, x_2) \in R$ *we write* $(y_1, u_1, y_2) \prec (x_1, v_1, x_2)$ *if one of the following conditions is verified:*

- $v_1 = \theta(u_1, u_2)$, $y_1 = x_1$, $(y_2, u_2, x_2) \in R$
- $v_1 = \theta(u_2, u_1)$, $y_2 = x_2$, $(x_1, u_2, y_1) \in R$

The transitive closure of $\prec$ is denoted by $\prec^+$. This means that $\alpha \prec^+ \beta$ if there are $\alpha_1, \ldots, \alpha_n \in R$ such that $\alpha = \alpha_1$, $\alpha_n = \beta$ and $\alpha_i \prec \alpha_{i+1}$ for every $i \in \{1, \ldots, n-1\}$.

Remark 4. Suppose $\alpha = (y_1, u, y_2)$ and $\beta = (x_1, v, x_2)$. If $\alpha \prec \beta$ then $length(u) < length(v)$. Consequently, if $\alpha \prec^+ \beta$ then $length(u) < length(v)$.

Proposition 3. *The relation $\prec^+$ is a strict partial order. In other words, for every $\alpha, \beta, \gamma \in R$ the following properties are satisfied:*
$\alpha \not\prec^+ \alpha$
$\alpha \prec^+ \beta \Rightarrow \beta \not\prec^+ \alpha$
$\alpha \prec^+ \beta, \beta \prec^+ \gamma \Rightarrow \alpha \prec^+ \gamma$

Proof. The first two conditions are verified by Remark 4. The last condition is verified by the transitivity of the relation $\prec^+$.

Definition 9. *An element $\alpha \in R$ is a* **maximal element** *if $\alpha \not\prec^+ \beta$ for all $\beta \in R$. We denote by R^{max} the set of all maximal elements of R.*

Definition 10. *([10]) If $\mathcal{S} = (X, A_0, A, R)$ is a θ-semantic schema then the labeled graph $G_{\mathcal{S}}^{max} = (Y, L, T, h)$ is the* **maximal graph** *associated to $\mathcal{S}$ if the following conditions are verified:*
- $Y = pr_1 R^{max} \cup pr_3 R^{max}$
- $L = pr_2 R^{max}$
- $h(\alpha) = \{(x, y) \mid (x, \alpha, y) \in R^{max}\}$ *for* $\alpha \in L$
- $T = \{h(\alpha) \mid \alpha \in L\}$

Definition 11. *([10]) A* **cooperating system of semantic schemas** *is a pair $(\{\mathcal{S}_i\}_{i=1}^n, E)$, where*
- $\mathcal{S}_i = (X_i, A_{0i}, A_i, R_i)$ *is a θ_i-semantic schema for $i \in \{1, \ldots, n\}$;*
- $E = (X, L_0, L, R)$ *is a θ-semantic schema such that*
  i) X and L_0 are the nodes and respectively the labels of the graph $\bigcup_{i=1}^n G_{\mathcal{S}_i}^{max}$
  ii) R satisfies the condition

$$(x, \theta(u, v), y) \in R, (x, u, z) \in R_i^{max}, (z, v, y) \in R_j^{max} \Rightarrow i \neq j \qquad (6)$$

At this point we emphasize an aspect concerning the formal computations performed in a semantic schema. Let us denote by $\mathcal{S} = (X, A_0, A, R)$ an arbitrary θ-semantic schema and $R_0 = R \cap (X \times A_0 \times X)$. If $R_0 = R$ then $A = A_0$ and in this case no deduction is modeled by $\mathcal{S}$. Such a schema can be used only to store the facts of a knowledge piece and to retrieve this information. In view of this remark one might say that a semantic schema $\mathcal{S} = (X, A_0, A, R)$ satisfying the property $A = A_0$ (or equivalently, $R = R_0$) is a **trivial** semantic schema.

The concept introduced in Definition 11 can be analyzed from various points of view. As a particular case we can consider a cooperating system containing only trivial semantic schemas. Obviously such a system becomes a θ-semantic

schema. In order to specify this case we consider the trivial schemas defined as follows:
• $\mathcal{S}_1 = (\{x, y, z_1\}, \{a, b\}, \{a, b\}, \{(x, a, y), (y, b, z_1)\})$
• $\mathcal{S}_2 = (\{x, y, z_2\}, \{a, b\}, \{a, b\}, \{(x, a, y), (y, b, z_2)\})$
Only two cooperating systems can be obtained by means of these schemas:

1. The trivial system given by $E = (\{x, y, z_1, z_2\}, \{a, b\}, \{a, b\}, R_0)$, where $R_0 = \{(x, a, y), (y, b, z_1), (y, b, z_2)\}$.
2. The non trivial cooperating system given by

$$E = (\{x, y, z_1, z_2\}, \{a, b\}, \{a, b, \theta(a, b)\}, R)$$

where $R_0 = \{(x, a, y), (y, b, z_1), (y, b, z_2)\}$ and $R = R_0 \cup \{(x, \theta(a, b), z_1), (x, \theta(a, b), z_2)\}$. The structure E is obviously a θ-schema.

Remark 5. If $\mathcal{S}_i = (X_i, A_{0i}, A_i, R_i)$ and $E = (X, L_0, L, R)$ then $X \subseteq \bigcup_{i=1}^{n} X_i$ and $L_0 \subseteq \bigcup_{i=1}^{n} A_i$. Really, if $G_{\mathcal{S}_i}^{max} = (Y_i, L_i, T_i, h_i)$ then by Definition 10 we have $Y_i = pr_1 R_i^{max} \cup pr_3 R_i^{max} \subseteq X_i$ and $L_i = pr_2 R_i^{max} \subseteq A_i$ for every $i \in \{1, \ldots, n\}$.

Proposition 4. *If $\mathcal{C} = (\{\mathcal{S}_i\}_{i=1}^{n}, E)$ is a cooperating system then either $n \geq 2$ or $\mathcal{C}$ is a trivial schema.*

Proof. We can write $L = \bigcup_{k \geq 0}(L \cap B_k)$, where B_k is defined as in (1). If $n = 1$ then (6) can not be applied, therefore $L \cap B_1 = \emptyset$. Using Proposition 1 we can verify by induction on k that $L \cap B_k = \emptyset$. It follows that $L = L \cap B_0 = L_0$ and $\mathcal{C}$ is a trivial schema.

In connection with Definition 11 we relieve the following aspects:

1. A cooperation system is based on several *distinct* semantic schemas because each schema $\mathcal{S}_i$ is built by means of a symbol θ_i and $\theta_i \neq \theta_j$ for $i \neq j$.
2. By Remark 5 we observe that L is a subset of the Peano θ-algebra generated by a finite set that contains some elements taken from the Peano θ_i-algebras of the schemas $\mathcal{S}_1$, …, $\mathcal{S}_n$.

Remark 6. The condition (6) was introduced because a cooperating system $(\{\mathcal{S}_i\}_{i=1}^{n}, E)$ is not able to extend the deduction of some component $\mathcal{S}_i$. As a matter of fact the task of E is to model the collaboration of its components.

We define now the computations in a cooperating system introduced in this section. We consider a cooperating system $\mathcal{C} = (\{\mathcal{S}_i\}_{i=1}^{n}, E)$, where $E = (X, L_0, L, R)$. In order to describe the computation in $\mathcal{C}$ we consider the symbols $\sigma, \sigma_1, \ldots, \sigma_n$ of arity 2. Two kinds of computations can be described in $\mathcal{C}$:
• A regular formal computation for the θ_i-schema $\mathcal{S}_i$. This computation was described in the first section for the general case of a semantic schema, with the remark that for $\mathcal{S}_i$ the symbol σ_i instead of σ is used.
• A proper formal computation for the θ-schema E. The derivation in E is given in the next definition.

Definition 12. *Suppose* $(x, \theta(u, v), y) \in R$. *If* $(x, u, z) \in R$ *and* $(z, v, y) \in R$ *then*

$$w_1(x, \theta(u, v), y)w_2 \vdash w_1\sigma((x, u, z), (z, v, y))$$

for every words w_1, w_2. *We denote by* $\vdash^*$ *the reflexive and transitive closure of* $\vdash$. *We denote by* $\mathcal{H}_E$ *the Peano* σ*-algebra generated by* $R_0 = R \cap (X \times L_0 \times X)$. *We define*

$$\mathcal{F}(E) = \{w \in \mathcal{H}_E \mid \exists (x, u, y) \in R : (x, u, y) \vdash^* w\}$$

Remark 7. Because $\mathcal{H}_E$ is generated by R_0 and $\vdash^*$ is a reflexive relation we have $\mathcal{F}(E) \supseteq R_0$. This inclusion is used further to define the valuation mapping of a cooperating system.

In order to exemplify this computation and other concepts which follow in this section we consider the semantic schemas $\mathcal{S}_1$ and $\mathcal{S}_2$ represented respectively in Figure 6 and Figure 7. We remark that (x_2, b, x_3) is a maximal element both in $\mathcal{S}_1$ and $\mathcal{S}_2$. In other words we have $R_1^{max} \cap R_2^{max} \neq \emptyset$.

Remark 8. The general case, $R_i^{max} \cap R_j^{max} \neq \emptyset$ for some $i \neq j$, implies some feature of the valuation mapping given in Definition 14.

The graph $G_1^{max} \cup G_2^{max}$ is represented in Figure 8. From this figure we deduce that the following entities are used to specify E:

- $X = \{x_1, x_2, x_3, x_4, y_1\}$
- $L_0 = \{b, \theta_1(a, a), \theta_1(b, a), \theta_2(a, b), \theta_2(b, b), \theta_2(b, a), \theta_2(a, \theta_2(a, b))\}$
- $R_0 = \{(x_1, \theta_1(a, a), x_2), (x_1, \theta_2(a, b), x_2), (x_2, \theta_2(b, a), y_1), (x_2, b, x_3), (x_3, \theta_1(b, a), y_1), (x_3, \theta_2(b, b), x_4), (y_1, \theta_2(a, \theta_2(a, b)), x_4)\}$

In order to finish the definition of E we take

$R \setminus R_0 = \{(x_1, \theta(\theta_1(a, a), b), x_3), (x_1, \theta(\theta(\theta_1(a, a), b), \theta_1(b, a)), y_1), (x_1, \theta(\theta_2(a, b), b), x_3), (x_1, \theta(\theta(\theta_2(a, b), b), \theta_1(b, a)), y_1), (x_1, \theta(\theta(\theta_2(a, b), b), \theta_2(b, b)), x_4)\}$

and therefore

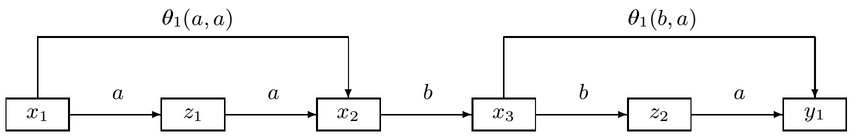

Fig. 6. Schema $\mathcal{S}_1$

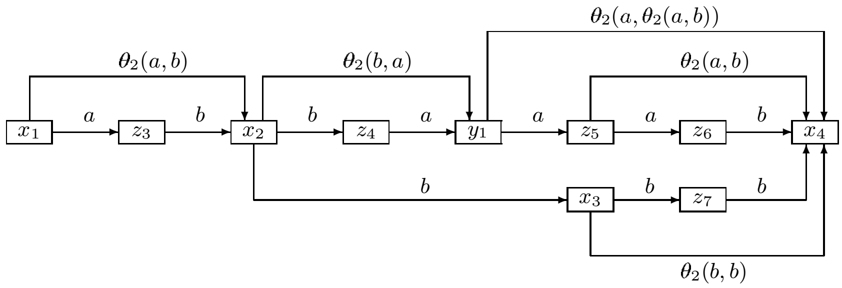

Fig. 7. Schema $\mathcal{S}_2$

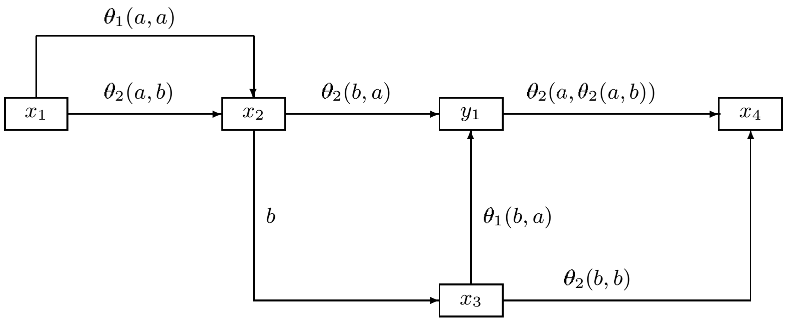

Fig. 8. $G_1^{max} \cup G_2^{max}$

$L \setminus L_0 = \{\theta(\theta_1(a,a),b), \theta(\theta(\theta_1(a,a),b),\theta_1(b,a)),\ \theta(\theta_2(a,b),b),$
$\theta(\theta(\theta_2(a,b),b),\theta_1(b,a)),\ \theta(\theta(\theta_2(a,b),b),\theta_2(b,b))\}$

We observe the condition (6) is satisfied by R. As an example of derivation in $\mathcal{C}$ we have the following sequence:

$$(x_1, \theta(\theta(\theta_1(a,a),b),\theta_1(b,a)), y_1) \vdash$$

$$\sigma((x_1, \theta(\theta_1(a,a),b), x_3), (x_3, \theta_1(b,a), y_1)) \vdash$$

$$\sigma(\sigma((x_1, \theta_1(a,a), x_2), (x_2, b, x_3)), (x_3, \theta_1(b,a), y_1))$$

By a similar computation we obtain also

$$(x_1, \theta(\theta(\theta_2(a,b),b),\theta_1(b,a)), y_1) \vdash$$

$$\sigma((x_1, \theta(\theta_2(a,b),b), x_3), (x_3, \theta_1(b,a), y_1)) \vdash$$

$$\sigma(\sigma((x_1, \theta_2(a,b), x_2), (x_2, b, x_3)), (x_3, \theta_1(b,a), y_1))$$

In order to define the valuation mapping of a cooperating system $\mathcal{C} = (\{\mathcal{S}_i\}_{i=1}^n, E)$ we denote $\mathcal{S}_i = (X_i, A_{0i}, A_i, R_i)$ and consider an interpretation $\mathcal{J}_i = (Ob_i, ob_i, \{Alg_u^i\}_{u \in A_i})$ of $\mathcal{S}_i$, $i \in \{1, \ldots, n\}$. We suppose that for $x, y \in X_i \cap X_j$ we have $x = y$ if and only if $ob_i(x) = ob_j(y)$.

Definition 13. *An* **interpretation** *of the cooperating system* $\mathcal{C}$ *is a system* $\mathcal{I} = (Ob, ob, \{Alg_u\}_{u \in A})$ *such that* $Ob = \bigcup_{i=1}^n ob_i(X \cap X_i)$*,* $ob(x) = ob_i(x)$ *if* $x \in X \cap X_i$*,* $ob : X \longrightarrow Ob$ *and* Alg_u *is an algorithm accepting two input arguments and one output argument.*

Proposition 5. *The mapping* $ob : X \longrightarrow Ob$ *is well defined and is bijective.*

Proof. If $x \in X \cap X_i \cap X_j$ for $i \neq j$ then $ob(x) = ob_i(x)$ and $ob(x) = ob_j(x)$ by the definition of ob. But $ob_i(x) = ob_j(x)$, therefore ob is well defined. If $y \in Ob$ then by Definition 13 there is i such that $y \in ob_i(X \cap X_i)$. Thus there is $x \in X \cap X_i$ such that $y = ob_i(x)$. But $ob(x) = ob_i(x)$, therefore $y = ob(x)$.

In what follows we consider the following decomposition of R: $R = D_0 \cup D_1 \cup D_2$, where $D_0 = R_0$, $D_1 = \{(x, \theta(u,v), y) \in R \mid u, v \in D_0\}$ and $D_1 = R \setminus (D_0 \cup D_1)$. We obtain a corresponding decomposition for $\mathcal{F}(E)$: $\mathcal{F}(E) = R_0 \cup \mathcal{F}_1(E) \cup \mathcal{F}_2(E)$, where $\mathcal{F}_1(E) = \{w \in \mathcal{F}(E) \mid \exists (x,u,y) \in D_1 : (x,u,y) \vdash^* w\}$ and $\mathcal{F}_2(E) = \{w \in \mathcal{F}(E) \mid \exists (x,u,y) \in D_2 : (x,u,y) \vdash^* w\}$.

Definition 14. *([10])* *The* **valuation mapping** *of the cooperating system* $\mathcal{C}$ *is the function* $Val_{\mathcal{I}} : \mathcal{F}(E) \longrightarrow 2^Y$*, where* Y *is the output space of the semantic schema* E*, defined as follows:*

- *If* $(x, a, y) \in D_0 \cap \left(\bigcup_{j=1}^n R_{0j}\right)$ *then*

$$Val_{\mathcal{I}}(x, a, y) = \bigcup_{i=1}^n \{Alg_a^i(ob_i(x), ob_i(y))\}$$

- $Val_{\mathcal{I}}(x, \theta_i(u, v), y) =$

$$\{Val_{\mathcal{I}_i}(\sigma_i(w_1, w_2)) \mid \sigma_i(w_1, w_2) \in \mathcal{F}(\mathcal{S}_i), (x, \theta_i(u, v), y) \Rightarrow_i^* \sigma_i(w_1, w_2)\}$$

- *Let be* $\sigma(\alpha, \beta) \in \mathcal{F}_1(E)$. *There is* $(x, \theta(u, v), y) \in D_1$ *such that* $(x, \theta(u, v), y) \vdash^* \sigma(\alpha, \beta)$. *We take*

$$Val_{\mathcal{I}}(\sigma(\alpha, \beta)) = \bigcup_{\substack{o_1 \in Val_{\mathcal{I}_i}(\alpha), \\ o_2 \in Val_{\mathcal{I}_j}(\beta), \\ i \neq j}} \{Alg_{\theta(u,v)}(o_1, o_2)\} \tag{7}$$

- *Let be* $\sigma(\alpha, \beta) \in \mathcal{F}_2(E)$. *There is* $(x, \theta(u, v), y) \in D_2$ *such that* $(x, \theta(u, v), y) \vdash^* \sigma(\alpha, \beta)$. *We take*

$$Val_{\mathcal{I}}(\sigma(\alpha, \beta)) = \bigcup_{\substack{o_1 \in Val_{\mathcal{I}}(\alpha), \\ o_2 \in Val_{\mathcal{I}}(\beta),}} \{Alg_{\theta(u,v)}(o_1, o_2)\}$$

Remark 9. The condition $i \neq j$ in (7) is connected by Remark 8.

5 Cooperating by cross-product of schemas

In this section we define a binary operation on the set of semantic schemas, which obtains from two semantic schemas another cooperating structure.

Definition 15. *We consider a symbol* $\theta \in \Sigma$.

- *An element* $(x, u, y) \in R_i^{max} \setminus R_j^{max}$ *is a* θ-**accepted element** *of type* (i, i) *and we denote type*$(x, u, y) = (i, i)$.
- *If* (x, u, y) *is a* θ*-accepted element, type*$(x, u, y) = (i, i)$ *and* (y, v, z) *is a* θ*-accepted element, type*$(y, v, z) = (j, j)$, $j \neq i$, *then* $(x, \theta(u, v), z)$ *is a* θ*-accepted element of type*$(x, \theta(u, v), z) = (i, j)$.
- *If* (x, u, y) *is a* θ*-accepted element of type*$(x, u, y) = (i, k_1)$, (y, v, z) *is a* θ*-accepted element of type*$(y, v, z) = (k_2, j)$ *and* $k_1 \neq k_2$ *then* $(x, \theta(u, v), z)$ *is* θ*-accepted element of type*$(x, \theta(u, v), z) = (i, j)$.

In order to exemplify this concept we consider the semantic schemas represented in Figure 9 and Figure 10.

We observe that:

- (x_1, a, x_2) is not an accepted element because it belongs to $R_1^{max} \cap R_2^{max}$.
- (x_2, b, x_3) is an accepted element of type $(1, 1)$, $(x_3, \theta_2(a, b), z_2)$ is an accepted element of type $(2, 2)$ and therefore $(x_2, \theta(b, \theta_2(a, b)), z_2)$ is an accepted element of type $(1, 2)$.

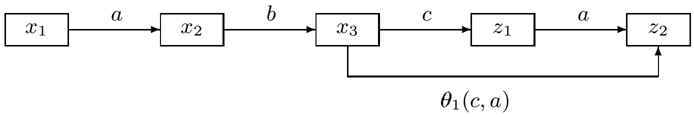

Fig. 9. Schema $\mathcal{S}_1$

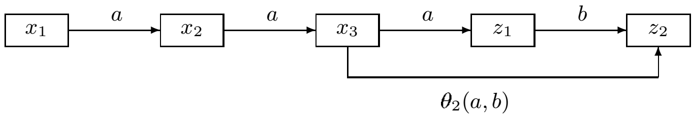

Fig. 10. Schema $\mathcal{S}_2$

- The element $(x_2, \theta(b, \theta_1(c, a)), y_2)$ is an accepted element of type $(2, 1)$.

Definition 16. *([11]) Consider a θ_1-schema $\mathcal{S}_1$ and a θ_2-schema $\mathcal{S}_2$. Suppose G_1^{max} and G_2^{max} are the corresponding maximal labeled graphs of $\mathcal{S}_1$, respectively $\mathcal{S}_2$. The θ-schema $\mathcal{S} = (Z, C_0, C, R)$ is a* **cross-product** *of $\mathcal{S}_1$ and $\mathcal{S}_2$ if the following conditions are satisfied:*

- *The labeled graph of $\mathcal{S}$ is $G_1^{max} \cup G_2^{max}$.*
- *R is a set of θ-accepted elements.*

We denote by $\mathcal{S}_1 \otimes_C \mathcal{S}_2$ this structure.

Remark 10. If $G_1^{max} = (X_1, A_0, T_1, f_0)$, $G_2^{max} = (X_2, B_0, T_2, h_0)$ and $\mathcal{S} = (Z, C_0, C, R)$ is a cross-product of $\mathcal{S}_1$ and $\mathcal{S}_2$ then

- $Z = X_1 \cup X_2$
- $C_0 = A_0 \cup B_0$
- $C = pr_2 R$

Remark 11. Obviously we have $\mathcal{S}_1 \otimes_C \mathcal{S}_2 = \mathcal{S}_2 \otimes_C \mathcal{S}_1$.

Let us consider the schema $\mathcal{S}_1$ from Figure 11, schema $\mathcal{S}_2$ from Figure 12. We can take the set R that contains the elements of R_0 taken from Figure 15 and the following θ-accepted elements:

$(z_1, \theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), x_3)$
$(z_1, \theta(\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), \theta_2(a_3, b_2)), z_1)$
$(z_1, \theta(\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), b_1), y_2)$

We obtain $C_0 = \{\theta_1(a_1, a_2), b_1, \theta_1(b_1, a_1), \theta_1(b_1, \theta_1(\theta_1(a_2, a_3), a_1)), \theta_2(a_3, b_2), b_1, \theta_2(a_2, b_1), \theta_2(a_1, a_2)\}$ and therefore $C \setminus C_0 = \{\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), \theta(\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), \theta_2(a_3, b_2)), \theta(\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), b_1)\}$.

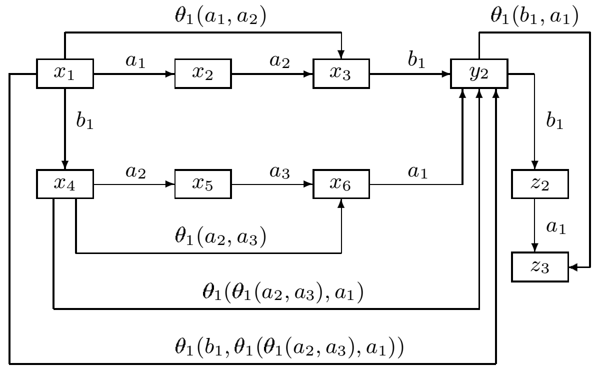

Fig. 11. Schema $\mathcal{S}_1$

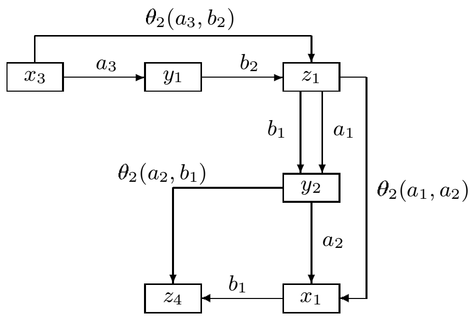

Fig. 12. Schema $\mathcal{S}_2$

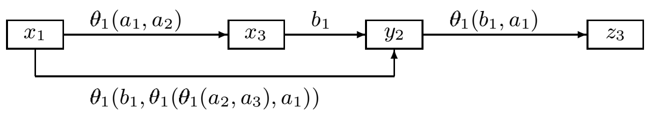

Fig. 13. G_1^{max}

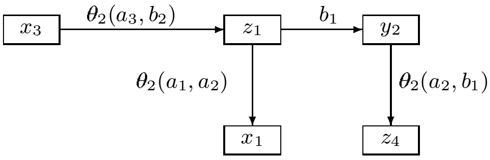

Fig. 14. G_2^{max}

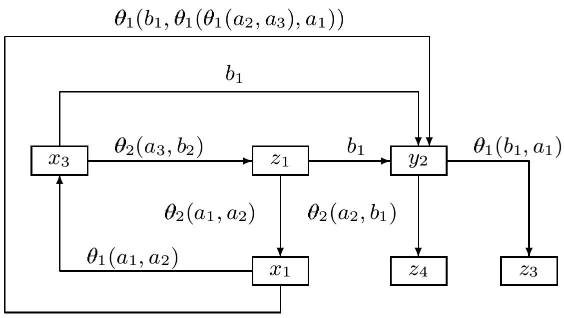

Fig. 15. $G_1^{max} \cup G_2^{max}$

Definition 17. *We suppose* $\mathcal{S} = (Z, C_0, C, R)$ *is a cross-product of the* θ_i*-schemas* $\mathcal{S}_i$*,* $i \in \{1, 2\}$*. If we substitute each element* $(x, \theta_i(u, v), y)$ *in* $\mathcal{S}$ *by all the paths generated in* $\mathcal{S}_i$ *by this element then we obtain a* Θ*-schema, where* $\Theta = \{\theta, \theta_1, \theta_2\}$*. This schema is named the* Θ*-schema* **associated** *to the cross-product* $\mathcal{S}$ *and is denoted by* $\mathcal{S}_\Theta$*.*

Definition 18. *([11]) A* **cross-deduction** *from* $\mathcal{S}_1$ *and* $\mathcal{S}_2$ *is a formal derivation in* $\mathcal{S}_\Theta$*. If* $(x, \theta(u, v), y) \Rightarrow^* w$ *and* $w \in \mathcal{F}_{comp}(\mathcal{S}_\Theta)$ *then* w *is a* **cross-conclusion**.

The following computation is an example of cross-deduction from $\mathcal{S}_1$ and $\mathcal{S}_2$:

$$(z_1, \theta(\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), \theta_2(a_3, b_2)), z_1) \Rightarrow$$

$$\sigma((z_1, \theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), x_3), (x_3, \theta_2(a_3, b_2), z_1))$$

But

$$(z_1, \theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), x_3) \Rightarrow \sigma((z_1, \theta_2(a_1, a_2), x_1), (x_1, \theta_1(a_1, a_2), x_3))$$

therefore

$$(z_1, \theta(\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), \theta_2(a_3, b_2)), z_1) \Rightarrow^*$$

$$\sigma(\sigma((z_1, \theta_2(a_1, a_2), x_1), (x_1, \theta_1(a_1, a_2), x_3)), (x_3, \theta_2(a_3, b_2), z_1))$$

On the other hand in $\mathcal{S}_1$ we have

$$(x_1, \theta_1(a_1, a_2), x_3) \Rightarrow^* \sigma(h(x_1, a_1, x_2), (h(x_2, a_2, x_3)))$$

and from $\mathcal{S}_2$ we obtain

$$(z_1, \theta_2(a_1, a_2), x_1) \Rightarrow^* \sigma(h(x_1, a_1, y_2), h(y_2, a_2, x_1))$$

and

$$(x_3, \theta_2(a_3, b_2), z_1)) \Rightarrow^* \sigma(h(x_3, a_3, y_1), h(y_1, b_2, z_1))$$

We obtain the following cross-deduction

$$(z_1, \theta(\theta(\theta_2(a_1, a_2), \theta_1(a_1, a_2)), \theta_2(a_3, b_2)), z_1) \Rightarrow^* \sigma(u, v)$$

where

$$u = \sigma(\sigma(h(x_1, a_1, y_2), h(y_2, a_2, x_1)), \sigma(h(x_1, a_1, x_2), (h(x_2, a_2, x_3)))$$

$$v = \sigma(h(x_3, a_3, y_1), h(y_1, b_2, z_1))$$

and thus we obtained a cross-conclusion.

6 Future research

The research line containing the cooperating systems based on semantic schemas can be developed. We relieve the following two ways by which we can develop this subject.

- Introduce a structure S that receives partial conclusions from two semantic schemas S_1 and S_2. A partial conclusion is obtained from a deductive path. Introduce a rule to combine the conclusion of a deductive path from S_1 with the conclusion of a deductive path from S_2. This rule can extend the cooperation based on maximal paths. This is a problem connected by the transfer of knowledge between semantic schemas.
- Study the transfer of knowledge between several semantic schemas and define a structure organized on levels such that the partial conclusions are transferred bottom up. This structure can be used to model distributed knowledge.

References

1. **V.Boicescu, A.Filipoiu, G.Georgescu, S.Rudeanu**, Lukasiewicz-Moisil Algebras, Annals of Discrete Mathematics 49, North-Holland, 1991
2. **J. Dassow**, Cooperating Grammar Systems (definition, Basic results, Open Problems), in Gheorghe Păun (Ed.), Artificial Life, Grammatical Models, (Black Sea University Press, Bucharest, Romania, 1995), 40-52.

3. **N. Ţăndăreanu**, Semantic Schemas and Applications in Logical Representation of Knowledge, Proceedings of the 10th Int. Conf. on CITSA, July 21-25, Orlando, Florida, Vol. III, p.82-87, 2004
4. **N. Ţăndăreanu** and **M.Ghindeanu**, A Three-Level Distributed Knowledge System Based on Semantic Schemas, 16th Int. Workshop on Database and Expert Systems Applications, Proceedings of DEXA'05, Copenhagen, p.423-427, 2005
5. **N. Ţăndăreanu**, Transfer of knowledge via semantic schemas, 9th World Multi-Conference on Systemics, Cybernetics and Informatics, July 10-13, Vol. IV, p.70-75, 2005
6. **N. Ţăndăreanu** and **M.Ghindeanu**, Properties of derivations in a Semantic Schema, Annals of University of Craiova, Math. Comp. Sci. Ser.,Vol.33, p.147-153, 2006
7. **N. Ţăndăreanu**, Semantic Schemas: The Least Upper Bound of Two Interpretations, 10th World Multiconference on Systemics, Cybernetics and Informatics, Orlando, USA, July 16-19, Vol. III, p.150-155, 2006
8. **N. Ţăndăreanu**, Lecture Notes on Universal Algebra, Basic Concepts of Peano Algebras and Lattices, Research Report in Artificial Intelligence 301, Universitaria Publishing House, 2006
9. **N. Ţăndăreanu**, Master-Slave Systems of Semantic Schemas and Applications, The 10th IASTED International Conference on Intelligent Systems and Control (ISC 2007), November 19-21, 2007, Cambridge, Massachusetts, USA, p.150-155, ISBN 978-0-88986-707-9
10. **N. Ţăndăreanu**, Cooperating Systems Based on Maximal Graphs in Semantic Schemas, Proceedings of the 11th WSEAS International Multiconference CSCC (Circuits, Systems, Communications, Computers), Vol. 4, p.517-522, Crete Island, Greece, July 23-28, 2007, ISSN: 1790-5117, ISBN: 978-960-8457-92-8
11. **N. Ţăndăreanu**, Cross-Deduction Based on Maximal Elements in Semantic Schemas, 11th World Multiconference on Systemics, Cybernetics and Informatics (WMSCI 2007), Orlando, USA, July 8-11, Vol.I, p. 97-102, 2007
12. Cooperating mobile robots, United States patent 6687571, http://www.patentstorm.us/patents, February, 2004.
13. **N. Tsuda, M. Ando, T. Nishino**, Development of Master-Slave system for Magnetic Levitation, http://www.robot.mach.mie-u.ac.jp/research/ms/ems.html.
14. **R. Bundgen, M. Gobel, W. Kuchlin**, A master-slave approach to parallel term rewriting on a hierarchical multiprocessor, Lecture Notes in Computer Science, Springer, Proceedings International Symposium DISCO'96, J. Calmet and C. Limongelli (Ed.), Vol.1128, 1996, 184-194.