Research Center for Artificial Intelligence
Department of Computer Science
University of Craiova, Romania

N. Ţăndăreanu

Knowledge Representation by Semantic Schemas

Technical report

2006

Table of Contents

1. From semantic networks to semantic schemas

1.1 Introduction

In this chapter we present the concept of semantic schema from an intuitive point of view. The concept of semantic schema was introduced to extend that of semantic network. This idea explains why Section 1.2 deals with an intuitive description for semantic networks. In Section 1.3 we prepare the formalism given in the subsequent sections of the next chapter.

The aim of this chapter is to give an intuitive description of the transition from semantic networks to semantic schemas.

1.2 Semantic networks: an intuitive description

In general a semantic network is a graph structure which uses its nodes to represent concepts and its arcs to represent relations among concepts. We remark that such a structure represents the relationships between the concepts in some specific domain of knowledge. There are different kinds of relationships that are represented in a semantic network. The most common kinds are the relationships *ako*, *isa* and *has*. The abbreviations *ako* and *isa* mean *a kind of* and *is a* respectively.

In order to take an example we consider the relationships *isa*, *eako*, *eis* and *ehas* whose meanings are specified in Table 1.1.

Relationship	**Meaning**
$(x, y) \in isa$	*x is a y*
$(x, y) \in eako$	*every x is a kind of y*
$(x, y) \in eis$	*every x is y*
$(x, y) \in ehas$	*every x has y*
$(x, y) \in has$	*x has y*

Table 1.1.

Let us consider the following knowledge piece KP_1:
Bob is a bird. Every bird is a kind of animal. Every bird has wings. Every animal is alive.

It is not difficult to represent KP_1 as in Figure 1.1.

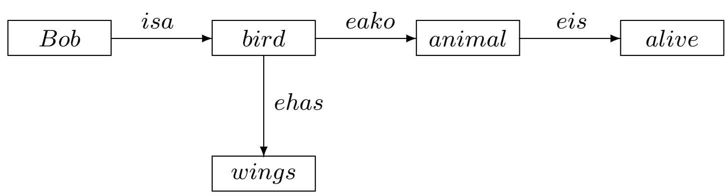

Fig. 1.1. Semantic network for KP_1

In this chapter we consider that a semantic network has the following features:

- the nodes designate some objects of a real world;
- the arc labels have arbitrary meanings, not only the meanings specified above;
- the reasoning process is based on the concept of path; because two nodes can be connected by several arcs, a path is viewed as a pair $([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$, where $x_1, \ldots, x_{n+1}$ are nodes and $a_1, \ldots, a_n$ are labels such that for every $i \in \{1, \ldots, n\}$ there is an arc labeled by a_i leaving the node x_i and entering x_j;
- the conclusion of the reasoning is a sentence in a natural language.

A representation of a semantic network includes:

- a labeled graph;
- a rule which specifies the combination of the arc labels; this rule can be defined by means of a partial operation between the arc labels; more precisely, if the arc labels u and v can be combined and the result of the combination is w then we denote this fact by $\varphi(u, v) = w$;
- the meaning of each arc label.

A semantic network can be used not only to *represent* knowledge but also to *process* the knowledge. This means that some *reasoning* can be performed. In essence, two nodes n and m must be specified and a path from n to m is searched. Step by step two consecutive labels of the path are combined and the result replaces these labels. Finally only one label is obtained and this label specifies some property linking the initial and the final nodes of the path. For example, if we compose *isa* and *ehas* we obtain *has*. The conclusion obtained from the path $([Bob, bird, wings], [isa, ehas])$ is the sentence *Bob has wings*. We observe the *output* is a sentence in a natural language.

The reasoning performed in a semantic network is a path-driven one. Two kinds of reasoning can be developed in a semantic network:

- a *direct* reasoning
- a *confluent* reasoning

Suppose that x and y are two nodes of the labeled graph and

$$p = ([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$$

is a path from x to y, that is $x_1 = x$ and $x_{n+1} = y$. We define the concept of *virtual path* generated by p as an element of the greatest set $V(d)$ satisfying the following conditions:

- $([x_1, \ldots, x_{n+1}], [a_1, \ldots , a_n]) \in V(d)$;
- if $([z_{i_1}, \ldots, z_{i_{k+1}}], [b_{i_1}, \ldots, b_{i_k}]) \in V(d)$ and $\varphi(b_{i_r}, b_{i_{r+1}}) = c$ for some $r \in \{1, \ldots, k-1\}$ then

$$([y_{j_1}, \ldots, y_{j_k}], [c_{j_1}, \ldots, c_{j_{k-1}}]) \in V(d)$$

  where $([y_{j_1}, \ldots, y_{j_k}]$ is obtained from $([z_{i_1}, \ldots, z_{i_{k+1}}]$ by removing $z_{i_{r+1}}$ and $[c_{j_1}, \ldots, c_{j_{k-1}}]$ is obtained from the list $[b_{i_1}, \ldots, b_{i_k}]$ by removing the elements b_{i_r}, b_{i_r+1} and introducing the element c on the place r.

An element of the form $([x, y], [e]) \in V(d)$ is called a *final virtual path* from x to y. The computation performed to obtain a final virtual path is a *direct reasoning* and the conclusion assigned to this computation is the meaning of the label e.

In order to exemplify this situation we consider the following knowledge piece KP_2: *Bob is the son of Helen and George is the son of Peter. Peter is the brother of Susan and Helen is the sister of Susan. Bob plays tennis with Helen. Helen*

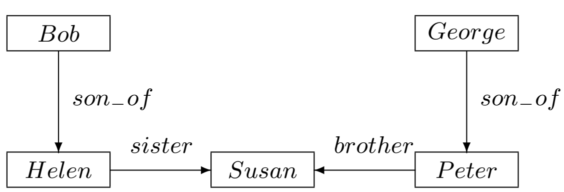

Fig. 1.2. Semantic network for KP_2

We say that two distinct paths $([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$ and $([y_1, \ldots, y_{k+1}], [b_1, \ldots, b_k])$ are *confluent* if $x_{n+1} = y_{k+1}$. For example, in Figure 1.2 the paths

$$([Bob, Helen, Susan], [son\_of, sister])$$

$$([George, Peter, Susan], [son\_of, brother])$$

are confluent paths. The confluence node is *Susan*. The confluent paths allow to perform additional deduction. For example, from the above representation we can deduce that *Bob is George's cousin*, *Helen is Peter's sister* and so on.

1.3 An intuitive extension of semantic networks

We consider the finite and nonempty sets X and A_0. Let θ be a symbol for a binary algebraic operation. We denote by $\overline{A}_0$ the Peano θ–algebra generated by A_0, therefore $\overline{A}_0 = \bigcup_{n\geq 0} M_n$ where M_n are defined recursively as follows (Rudeanu (1991)):

$$\begin{cases} M_0 = A_0 \\ M_{n+1} = M_n \cup \{ \ \theta(u,v) \ \mid \ u,v \in M_n\}, \quad n \geq 0 \end{cases}$$

Let us consider the graphical representation from Figure 1.3. Let us consider

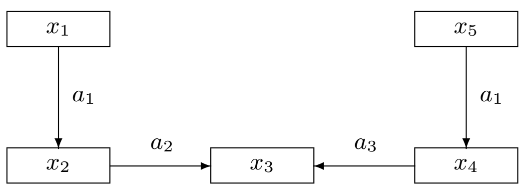

Fig. 1.3. An abstract representation

the following entities, which can be identified in an obvious manner from this figure:

- $X = \{x_1, x_2, x_3, x_4, x_5\}$
- $A_0 = \{a_1, a_2, a_3\}$
- $R_0 = \{(x_1, a_1, x_2), (x_2, a_2, x_3), (x_5, a_1, x_4), (x_4, a_3, x_3)\}$

In addition we consider the following sets:

- $A = A_0 \cup \{\theta(a_1, a_2), \theta(a_1, a_3)\}$
- $R = R_0 \cup \{(x_1, \theta(a_1, a_2), x_3), (x_5, \theta(a_1, a_3), x_3)\}$

We observe that X is the set of the nodes and A_0 is the set of arc labels. Moreover, we have $A_0 \subseteq A \subseteq \overline{A}_0$ and the set R_0 is the set of all labeled arcs of the graph. Every element of R is a triple such that two of its components represent some nodes of the graph. The first component of a triple is *the initial node* and the last is *the final node*.

We observe also that $R \subseteq X \times A \times X$ and the following properties establish a connection between R and A:

• If $(x, \theta(u, v), y) \in R$ then there is $z \in X$ such that $(x, u, z) \in R$ and $(z, v, y) \in R$
• Let be $\theta(u, v) \in A$. For all $x, y, z \in X$ if $(x, u, z) \in R$ and $(z, v, y) \in R$ then $(x, \theta(u, v), y) \in R$
• $pr_2 R = A$

where $pr_2 R = \{\alpha \mid \exists x, y \in X : (x, \alpha, y) \in R\}$.

These properties can be interpreted intuitively as follows:

- Every element of $R \setminus R_0$ can be broken into two "connected" entities of R; the connection is specified by the fact that the final node of the first entity is the initial node of the second entity.
- Two elements $(x, u, z) \in R$ and $(z, v, y) \in R$ are presumed to give some element of R because they are connected by the node z. If $\theta(u, v) \in A$ then these elements give really the element $(x, \theta(u, v), y)$ from R.
- Because $R \subseteq X \times A \times X$ it follows that $pr_2 R \subseteq A$. The condition $pr_2 R \supseteq A$ gives an additional information: the set A does not contain "foreign" elements. In other words, each element of A contributes to some element of R.

We consider a function symbol h of arity 1 and take the set M of all elements of the form $h(z)$, where $z \in R_0$. We obtain

$$M = \{h(x_1, a_1, x_2), h(x_2, a_2, x_3), h(x_5, a_1, x_4), h(x_4, a_3, x_3)\}$$

where in order to use a shorter notation we substituted the element $h((x, a, y))$ by $h(x, a, y)$.

We consider a symbol σ and denote by $\mathcal{H}$ the Peano σ-algebra generated by M. It follows that $\mathcal{H} = \bigcup_{n \geq 0} M_n$, where M_n are defined recursively as follows (Rudeanu (1991)):

$$\begin{cases} M_0 = M \\ M_{n+1} = M_n \cup \{\ \sigma(u, v) \ \mid \ u, v \in M_n\}, \quad n \geq 0 \end{cases}$$

We can introduce a derivation process as follows:

- At every step an element $(x, a, y) \in R_0$ is derived into $h(x, a, y)$; thus each element $z \in R_0$ introduces a letter h. The next step we can apply again the same rule and the element $h(x, a, y)$ derives the element $hh(x, a, y)$. This process is an infinite one and we specify each step of this computation as follows:

  $$(x, a, y) \Rightarrow h(x, a, y) \Rightarrow hh(x, a, y) \Rightarrow \ldots$$

  We observe that only in the first step we obtain an element of $\mathcal{H}$, the other elements do not belong to this structure.

- If we start with an element $(x, \theta(u, v), y) \in R \setminus R_0$ then based on the properties of $R \setminus R_0$ we know that there is an element $z \in X$ such that $(x, u, z) \in R$ and $(z, v, y) \in R$. These elements are used in the derivation process and for one step we write

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

We denote by $\Rightarrow^*$ the reflexive and transitive closure of the relation $\Rightarrow$. This means that we have $w \Rightarrow^* \omega$ if and only if one of the following condition is satisfied:

- $w = \omega$
- There are $w_1, \ldots w_n$ such that $w_1 = w$, $w_n = \omega$ and $w_i \Rightarrow w_{i+1}$ for every $i \in \{1, \ldots, n-1\}$

As an example of computations we obtain for our case:

$$(x_1, a_1, x_2) \Rightarrow h(x_1, a_1, x_2) \Rightarrow hh(x_1, a_1, x_2) \Rightarrow \ldots$$
$$(x_1, \theta(a_1, a_2), x_3) \Rightarrow \sigma((x_1, a_1, x_2), (x_2, a_2, x_3))$$

If we denote by h^n the word of length n of the form $h^n = h \ldots h$ then from the above computations we obtain

$$(x_1, a_1, x_2) \Rightarrow^* h^k(x_1, a_1, x_2)$$
$$(x_2, a_2, x_3) \Rightarrow^* h^m(x_2, a_2, x_3)$$
$$(x_1, \theta(a_1, a_2), x_3) \Rightarrow^* \sigma(h^k(x_1, a_1, x_2), h^m(x_2, a_2, x_3))$$

for every $k \geq 1$ and $m \geq 1$. We remark that only the elements for $k = m = 1$ belong to $\mathcal{H}$.

In general we can extract from $\mathcal{H}$ those elements which can be derived from the elements of R. For the particular case presented in Figure 1.3 we obtain the following set:

$$T = \{h(x_1, a_1, x_2), h(x_2, a_2, x_3), h(x_5, a_1, x_4), h(x_4, a_3, x_3),$$
$$\sigma(h(x_1, a_1, x_2), h(x_2, a_2, x_3)), \sigma(h(x_5, a_1, x_4), h(x_4, a_3, x_3))\}$$

The elements obtained are abstract entities. Our task is to assign a meaning for each of them. In order to perform this process we proceed as follows:

- In the first step we have to interpret the nodes from Figure 1.3. To perform this task we consider a set Ob of 5 objects and a bijective mapping $ob : X \longrightarrow Ob$. For our case we take
  - ⊡ $Ob = \{Bob, Helen, Susan, Peter, George\}$
  - ⊡ $ob(x_1) = Bob$; $ob(x_2) = Helen$; $ob(x_3) = Susan$; $ob(x_4) = Peter$; $ob(x_5) = George$
- In the second step we interpret the elements of R_0. This can be performed by means of a mapping $J_h : T_0 \longrightarrow Y$, where $T_0 = \{h(x, a, y) \mid (x, a, y) \in R_0\} \subseteq T$ and Y is the "output" space or the "semantic" space. For our case we consider:

⊡ Y is the following set of sentences, where x and y are arbitrary elements of Ob:

$p_1(x, y) =$ "*x is the son of y*"
$p_2(x, y) =$ "*x is the sister of y*"
$p_3(x, y) =$ "*x is the brother of y*"
$p_4(x, y) =$ "*x is the nephew of y*"

⊡ The mapping J_h is defined as follows:

$J_h(h(x, a_1, y)) = p_1(ob(x), ob(y))$
$J_h(h(x, a_2, y)) = p_2(ob(x), ob(y))$
$J_h(h(x, a_3, y)) = p_3(ob(x), ob(y))$

- In the third step we consider a partial binary operation $J_\sigma : Y \times Y \longrightarrow Y$ and for our case we consider the mapping:
  $J_\sigma(p_1(x, y), p_2(y, z)) = p_4(x, z)$
  $J_\sigma(p_1(x, y), p_3(y, z)) = p_4(x, z)$
  where x, y and z are arbitrary elements in Ob.
- In the last step we consider the mapping

$$J : T \longrightarrow Y$$

as follows:

$$\begin{cases} J(h(x, a, y)) = J_h(h(x, a, y)) \text{ if } h(x, a, y) \in T_0 \\ \\ J(\sigma(u, v)) = J_\sigma(J(u), J(v)) \end{cases}$$

The valuation computation for our case is described below:

- $J(h(x_1, a_1, x_2)) = p_1(Bob, Helen)$
- $J(h(x_2, a_2, x_3)) = p_2(Helen, Susan)$
- $J(h(x_5, a_1, x_4)) = p_1(George, Peter)$
- $J(h(x_4, a_3, x_3)) = p_3(Peter, Susan)$
- $J(\sigma(h(x_1, a_1, x_2), h(x_2, a_2, x_3))) =$
  $J_\sigma(J(h(x_1, a_1, x_2)), J(h(x_2, a_2, x_3))) =$
  $J_\sigma(p_1(Bob, Helen), p_2(Helen, Susan)) = p_4(Bob, Susan) =$
  "*Bob is the nephew of Susan*"
- $J(\sigma(h(x_5, a_1, x_4), h(x_4, a_3, x_3))) =$
  $J_\sigma(J(h(x_5, a_1, x_4)), J(h(x_4, a_3, x_3))) =$
  $J_\sigma(p_1(George, Peter), p_3(Peter, Susan)) =$
  $p_4(George, Susan) =$ "*George is the nephew of Susan*"

Remark 1.3.1. It is easy to observe the connection between the semantic network from Figure 1.2 and the structure represented in Figure 1.3. It is not difficult to read the information represented in Figure 1.2 because both the nodes and the arcs are directly represented in an intuitive manner. The situation is not the same for the representation from Figure 1.3. This is due to the fact that the structure from Figure 1.3 is an abstract one. For a particular choice of all nodes and arcs from Figure 1.3, as we proceeded in this section, we recovered the knowledge from Figure 1.2. For other choice of the same entities from Figure 1.3 we can obtain a knowledge representation which can not be obtained by semantic networks. For example, this is the case of the output space containing images, as we see in the next chapters.

Remark 1.3.2. The mapping J_σ is a partial one. The mapping J has the same property. In this vision we read the equality $J(\sigma(u, v)) = J_\sigma(J(u), J(v))$ as follows: $J(\sigma(u, v))$ is defined if $J(u)$ and $J(v)$ are defined and in addition, J_σ is defined in $(J(u), J(v))$.

Remark 1.3.3. The space Y is a set of sentences. Because x and y are arbitrary elements in *Ob* some sentences are "useless" as for example: *Helen is the son of George*, *George is the sister of Susan* and so on. As we observe the elements of Y are selected by the mapping *ob*, the meanings of the label arcs and the mapping J.

2. Semantic schemas

2.1 Overview

The concept of semantic schema was introduced in (Ţăndăreanu (2004d)) and this structure extends the concept of semantic network (Ţăndăreanu (2004f)). A semantic schema is an abstract structure, which can represent knowledge by means of an appropriate interpretation. Such a structure $\mathcal{S}$ is a tuple of four entities, $\mathcal{S} = (X, A_0, A, R)$, each of which specifying some features of the representation process. For a given semantic schema $\mathcal{S}$ an interpretation $\mathcal{I}_1$ represents a knowledge piece KP_1. If we change $\mathcal{I}_1$ by $\mathcal{I}_2$ then $\mathcal{S}$ represents other knowledge piece KP_2. Various interpretations can be used for the same semantic schema.

The concepts and results based on semantic schemas were applied in a client-server technology trying to model some aspects concerning the use of this structure in the domain of logic programming with constraints (Ţăndăreanu (2004d)), knowledge management (Ţăndăreanu (2005b)), distributed knowledge and reasoning by analogy (Ţăndăreanu (2005a)). These applications are treated in a separate chapter.

Two aspects are relieved in connection with a semantic schema $\mathcal{S}$:

1) A *formal aspect* in $\mathcal{S}$ by which some formal computations in a Peano σ-algebra are obtained. The computations are based on the concept of *derivation* (Ţăndăreanu and Ghindeanu (2006a)) and the set of the results is denoted by $\mathcal{F}_{comp}(\mathcal{S})$ (Figure 2.1). In this chapter we introduce the concept of *sort* for the formal entities of $\mathcal{F}_{comp}(\mathcal{S})$. A sort is an element of A, which is a subset of the Peano θ-algebra generated by A_0. Based on this concept we divide $\mathcal{F}_{comp}(\mathcal{S})$ into equivalence classes. An equivalence class includes all the elements of the same sort.

2) An *evaluation aspect* with respect to some interpretation. The entities obtained in the previous step get values from the semantic space Y. Every entity from an equivalence class $[u]_{\mathcal{F}}$, where u is a sort, is transformed to obtain its semantics. By such a transformation, a subset Y_u of Y is obtained and each object of Y_u has the *class* u. The space Y becomes the union of *classes of objects* (Figure 2.1).

In Figure 2.1 we represented on the first level the set R, which is a component of a semantic schema $\mathcal{S}$. An element of R is a triple (x, u, y), where $u \in A$.

The formal computations, based on derivation, give the equivalence class $[u]_{\mathcal{F}}$, each element of this set has the sort u and is transformed by means of an interpretation in an object of class u. In this manner, an element of A is a sort of a formal entity and a class of an object from the semantic space.

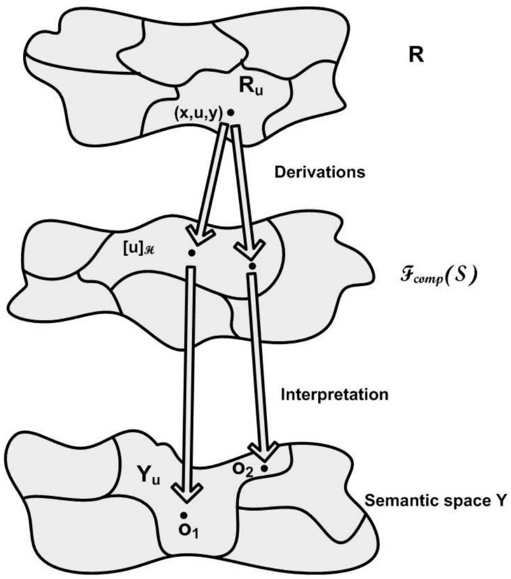

Fig. 2.1. Overview of the computations

2.2 Basic concepts

Consider a symbol θ of arity 2 and a finite non-empty set A_0. We denote by $\overline{A}_0$ *the Peano θ-algebra* (Rudeanu (1991)) *generated by* A_0, therefore $\overline{A}_0 = \bigcup_{n\geq 0} A_n$ where A_n are defined recursively as follows (Rudeanu (1991)):

$$A_{n+1} = A_n \cup \{\ \theta(u,v)\ \mid\ u,v \in A_n\},\quad n \geq 0 \tag{2.1}$$

For every $\alpha \in \overline{A_0}$ we define $trace(\alpha)$ as follows: if $\alpha \in A_0$ then $trace(\alpha) =< \alpha >$; if $\alpha = \theta(u,v)$ then $trace(\alpha) =< p,q >$, where $trace(u) = < p >$ and $trace(v) =< q >$.

Definition 2.2.1. *(Ţăndăreanu (2004d)) A* **semantic θ-schema** *(shortly,* **semantic 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 (Rudeanu (1991))*
- *$R \subseteq X \times A \times X$ is a non-empty set which fulfills the following conditions:*

$$(x, \theta(u, v), y) \in R \Longrightarrow \exists z \in X : (x, u, z) \in R, (z, v, y) \in R \tag{2.2}$$

$$\theta(u, v) \in A, (x, u, z) \in R, (z, v, y) \in R \Longrightarrow (x, \theta(u, v), y) \in R \tag{2.3}$$

$$u \in A \Longleftrightarrow \exists(x, u, y) \in R \tag{2.4}$$

We denote:

$$R_0 = R \cap (X \times A_0 \times X) \tag{2.5}$$

A basic property for A is described in the next proposition.

Proposition 2.2.1. *Let $\mathcal{S} = (X, A_0, A, R)$ be a semantic schema. The set A satisfies the following property:*

$$\theta(u, v) \in A \Longrightarrow u \in A, v \in A$$

Proof. Suppose $\theta(u, v) \in A$. Using (2.4) we deduce that there are $x, y \in X$ such that $(x, \theta(u, v), y) \in R$. From (2.2) it follows that there is $z \in X$ such that $(x, u, z) \in R$ and $(z, v, y) \in R$. Using again (2.4) we obtain $u \in A$ and $v \in A$. ■

A semantic schema can be represented as a labeled graph as follows:

• The elements of X are the nodes of the graph and they are graphically represented by rectangles.

• We draw an arc from x to y, which is labeled by α if and only if $(x, \alpha, y) \in R$.

If a semantic schema is represented as a labeled graph then the satisfiability of the conditions (2.2), (2.3) and (2.4) can be easily verified. As an example, the graph drawn in Figure 2.2 is based on the semantic schema

$$(\{x_1, x_2, x_3\}, \{a, b\}, \{a, b, \theta(a, b)\}, \{(x_1, a, x_2),$$
$$(x_2, b, x_3), (x_1, \theta(a, b), x_3)\})$$

whereas the graph drawn in Figure 2.3 can not represent any semantic schema. Really, we have $(x_3, a, y_1) \in R$, $(y_1, b, y_2) \in R$, $\theta(a, b) \in A$ and (2.3) is not satisfied.

Based on these properties we remark that a graphical representation of a semantic schema is a pair (G, A) where

- G is a graphical representation of a labeled graph, which relieves the set X of nodes and the set A_0 of labels.
- A is a set such that $A_0 \subseteq A \subseteq \overline{A_0}$ and A satisfies also the condition: if $\theta(u, v) \in A$ then $u \in A$ and $v \in A$.

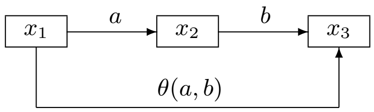

Fig. 2.2. A graph representing a semantic schema

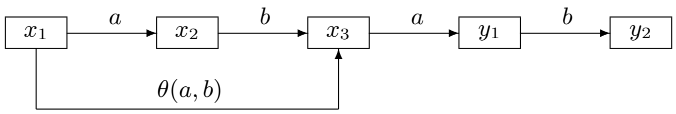

Fig. 2.3. A graph, which can not represent a semantic schema

If we accept this point of view then the set R is obtained immediately:

- R_0 is obtained directly from the graphical representation G. The other elements of R are obtained by (2.3).

Let us consider the graphical representation of G from Figure 2.4.

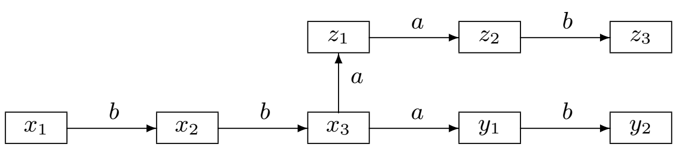

Fig. 2.4. Graphical representation of G

From this representation we deduce:

- $X = \{x_1, x_2, x_3, y_1, y_2, z_1, z_2, z_3\}$
- $A_0 = \{a, b\}$

Now, if we take

$$A = A_0 \cup \{\theta(a, b), \theta(b, a), \theta(b, \theta(b, a))\}$$

then we obtain:

$$R_0 = \{(x_1, b, x_2), (x_2, b, x_3), (x_3, a, y_1), (y_1, b, y_2), (x_3, a, z_1),$$
$$(z_1, b, z_2), (z_2, a, z_3)\}$$
$$R \setminus R_0 = \{(x_3, \theta(a, b), y_2), (x_3, \theta(a, b), z_2), (x_2, \theta(b, a), y_1),$$
$$(x_2, \theta(b, a), z_1), (z_1, \theta(b, a), z_3), (x_3, \theta(a, \theta(b, a)), z_3)\}$$

2.3 The join semilattice of schemas

In this section we introduce a partial order between two θ-schemas and we show that there is $sup\{\mathcal{S}_1, \mathcal{S}_2\}$ for arbitrary schemas $\mathcal{S}_1$ and $\mathcal{S}_2$. Moreover, this element can be effectively obtained in a finite number of steps from the components of $\mathcal{S}_1$ and $\mathcal{S}_2$.

Based on (2.1) we observe that if A is a set such that $A_0 \subseteq A \subseteq \overline{A}_0$ the we can write

$$A = A_0 \cup \bigcup_{i \geq 1} (A \cap A_i) \tag{2.6}$$

Remark 2.3.1. In general, if $X \subseteq Y_1 \times \ldots \times Y_n$ and $i \in \{1, \ldots, n\}$ then we denote $pr_i X = \{x \mid \exists (y_1, \ldots, x, \ldots, y_n) \in X\}$.

The following proposition is useful in what follows:

Proposition 2.3.1.

$$\overline{A}_0 \cup \overline{B}_0 \subseteq \overline{A_0 \cup B_0} \tag{2.7}$$

Proof. Applying (2.1) we obtain

$$B_{n+1} = B_n \cup \{\ \theta(u, v) \ \mid \ u, v \in B_n\}, \quad n \geq 0 \tag{2.8}$$

$$\begin{cases} C_0 = A_0 \cup B_0 \\ \\ C_{n+1} = C_n \cup \{\ \theta(u, v) \ \mid \ u, v \in C_n\}, \quad n \geq 0 \end{cases} \tag{2.9}$$

and $\overline{A_0 \cup B_0} = \bigcup_{n \geq 0} C_n$.
We prove by induction on n that

$$A_n \cup B_n \subseteq C_n \tag{2.10}$$

For $n = 0$ the relation (2.10) is true. Suppose (2.10) is true for n and we verify (2.10) for $n + 1$. Take an arbitrary element $x \in A_{n+1} \cup B_{n+1}$. Suppose $x \in A_{n+1}$. If $x \in A_n$ then $x \in C_n$ by (2.10). But $C_n \subseteq C_{n+1}$, therefore $x \in C_{n+1}$. If $x \in A_{n+1} \setminus A_n$ then $x = \theta(u, v)$ for some $u, v \in A_n$. But $A_n \subseteq C_n$, therefore $x = \theta(u, v)$ for some $u, v \in C_n$. Thus, $x \in C_{n+1}$. ■

Definition 2.3.1. *Consider the θ-schemas $\mathcal{S} = (X, A_0, A, R)$ and $\mathcal{P} = (Y, B_0, B, Q)$. We define the relation $\mathcal{S} \sqsubseteq \mathcal{P}$ if $X \subseteq Y$ and $R \subseteq Q$.*

Proposition 2.3.2. *Consider the θ-schemas $\mathcal{S} = (X, A_0, A, R)$ and $\mathcal{P} = (Y, B_0, B, Q)$. If $\mathcal{S} \sqsubseteq \mathcal{P}$ then $A_0 \subseteq B_0$ and $A \subseteq B$.*

Proof. We have $A = pr_2 R$, $B = pr_2 Q$ and $R \subseteq Q$ therefore $A \subseteq B$. On the other hand, using (2.6) we obtain $A = A_0 \cup \bigcup_{i \geq 1} (A \cap A_i)$, $B = B_0 \cup \bigcup_{i \geq 1} (B \cap B_i)$. But $A \subseteq B$ and $A_0 \subseteq A$ therefore $A_0 \subseteq B_0 \cup \bigcup_{i \geq 1} (B \cap B_i)$. But $A_0 \cap B_i = \emptyset$ for $i \geq 1$ because there is no element in A_0 of the form $\theta(u, v)$. It remains that $A_0 \subseteq B_0$. ■

Proposition 2.3.3. *The relation $\sqsubseteq$ is reflexive, antisymmetric and transitive, therefore it is a partial order.*

Proof. Obviously the relation is reflexive and transitive. If $\mathcal{S} = (X, A_0, A, R)$, $\mathcal{P} = (Y, B_0, B, Q)$ $\mathcal{S} \sqsubseteq \mathcal{P}$ and $\mathcal{P} \sqsubseteq \mathcal{S}$ then $X \subseteq Y$, $Y \subseteq X$, $R \subseteq Q$ and $Q \subseteq R$. It follows that $X = Y$ and $R = Q$. By Proposition 1 we have $A_0 = B_0$ and $A = B$, therefore $\mathcal{S} = \mathcal{P}$. ■

Proposition 2.3.4. *Consider the θ-schemas $\mathcal{S}_1 = (X, A_0, A, R)$ and $\mathcal{S}_2 = (Y, B_0, B, Q)$. We define recursively the sets:*

$$\begin{cases} Z_0 = R_0 \cup Q_0 \\ \\ Z_{i+1} = Z_i \cup V_i, \quad i \geq 0 \end{cases} \tag{2.11}$$

where $V_i = \{(x, \theta(u, v), y) \in (X \cup Y) \times (A \cup B) \times (X \cup Y) \mid \exists z : (x, u, z) \in Z_i, (z, v, y) \in Z_i\}$ and R_0, Q_0 are defined as in (2.5).

The sequence $\{Z_i\}_{i \geq 0}$ satisfies the following properties:

i) There is a natural number n_0 such that

$$Z_0 \subset Z_1 \subset \ldots \subset Z_{n_0} = Z_{n_0+1} = \ldots$$

ii) If we denote

$$\mathcal{S}_1 \vee \mathcal{S}_2 = (X \cup Y, A_0 \cup B_0, A \cup B, Z_{n_0}) \tag{2.12}$$

then $\mathcal{S}_1 \vee \mathcal{S}_2$ is a θ-schema. Moreover, $\mathcal{S}_1 \vee \mathcal{S}_2 = sup\{\mathcal{S}_1, \mathcal{S}_2\}$.

Proof. We verify by induction on n the following inclusion:

$$Z_n \subseteq (X \cup Y) \times (A \cup B) \times (X \cup Y) \tag{2.13}$$

From (2.11) we have $Z_0 = R_0 \cup Q_0 \subseteq (X \times A_0 \times X) \cup (Y \times B_0 \times Y) \subseteq (X \cup Y) \times (A_0 \cup B_0) \times (X \cup Y)$, therefore (2.13) is true for $n = 0$. Suppose (2.13) is true for n. We have $Z_{n+1} = Z_n \cup (Z_{n+1} \setminus Z_n)$. From the second line of (2.11) we have $Z_{n+1} \setminus Z_n \subseteq (X \cup Y) \times (A \cup B) \times (X \cup Y)$. By inductive assumption we have (2.13), therefore (2.13) is true for $n+1$. Because $(X \cup Y) \times (A \cup B) \times (X \cup Y)$ is a finite set we deduce that there is a natural number n such that $Z_n = Z_{n+1}$. Let n_0 be the least number satisfying this property. It is easy to prove by induction on t that $Z_{n_0} = Z_{n_0+t}$ for every $t \geq 1$.
From (2.13) we have

$$pr_2 Z_{n_0} \subseteq A \cup B \tag{2.14}$$

Let us prove that

$$R \subseteq Z_{n_0} \tag{2.15}$$

If we take into consideration (2.6) then $A = \bigcup_{i \geq 0} (A \cap A_i)$. For each $i \geq 0$ we denote $R_i = R \cap (X \times (A \cap A_i) \times X)$. Obviously

$$\bigcup_{i\geq 0} R_i = \bigcup_{i\geq 0}(R \cap (X \times (A \cap A_i) \times X)) =$$

$$R \cap (X \times \bigcup_{i\geq 0}(A \cap A_i) \times X) = R$$

In order to prove (2.15) we observe that it is enough to prove

$$R_i \subseteq Z_{n_0} \tag{2.16}$$

for every $i \geq 0$. For $i = 0$ the inclusion is true because $R_0 \subseteq Z_0 \subseteq Z_{n_0}$. Suppose (2.16) is true for $i \in \{0, \ldots, k\}$. Take an element $(x, \theta(u, v), y) \in R_{k+1}$. But $R_{k+1} \subseteq R$ and therefore applying (2.2) we deduce that there is z such that $(x, u, z) \in R$ and $(z, v, y) \in R$. Moreover, we can say that $(x, u, z) \in R_n$ and $(z, v, y) \in R_m$ for some $n \leq k$ and $m \leq k$. Applying the inductive assumption we have $(x, u, z) \in Z_{n_0}$ and $(z, v, y) \in Z_{n_0}$, therefore $(x, \theta(u, v), y) \in Z_{n_0+1} = Z_{n_0}$. Thus, (2.16) is true for every $i \geq 0$. Now, from (2.16) we deduce (2.15) and similarly we have

$$Q \subseteq Z_{n_0} \tag{2.17}$$

The condition (2.4) can be written also as $A = pr_2R$. Then, from (2.15) we deduce $A = pr_2R \subseteq pr_2Z_{n_0}$ and similarly, $B = pr_2Q \subseteq pr_2Z_{n_0}$. It follows that $A \cup B \subseteq pr_2Z_{n_0}$ and taking into consideration (2.14) we obtain

$$A \cup B = pr_2Z_{n_0} \tag{2.18}$$

We observe now that $A_0 \cup B_0 \subseteq A \cup B \subseteq \overline{A} \cup \overline{B}$ and by Proposition 1 we obtain $A_0 \cup B_0 \subseteq A \cup B \subseteq \overline{A \cup B}$. The relation (2.4) for (2.12) becomes (2.18), which is fulfilled.

Let us verify (2.2) and (2.3) for (2.12). Suppose $(x, \theta(u, v), y) \in Z_{n_0}$. There is $i \in \{0, \ldots, n_0 - 1\}$ such that $(x, \theta(u, v), y) \in Z_{i+1} \setminus Z_i$. From (2.11) we deduce that there is $z \in X \cup Y$ such that $(x, u, z) \in Z_i$ and $(z, v, y) \in Z_i$. But $Z_i \subseteq Z_{n_0}$, therefore $(x, u, z) \in Z_{n_0}$ and $(z, v, y) \in Z_{n_0}$. It follows that (2.2) is satisfied by (2.12).

Suppose $\theta(u, v) \in A \cup B$, $(x, u, z) \in Z_{n_0}$ and $(z, v, y) \in Z_{n_0}$. Applying (2.11) we obtain $(x, \theta(u, v), y) \in Z_{n_0+1} = Z_{n_0}$. Thus, (2.4) is satisfied. It follows that $\mathcal{S}_1 \vee \mathcal{S}_2$ is a θ-schema.

We have $\mathcal{S}_i \sqsubseteq \mathcal{S}_1 \vee \mathcal{S}_2$ for $i \in \{1, 2\}$. Let $\mathcal{S} = (Z, C_0, C, P)$ such that $\mathcal{S}_1 \sqsubseteq \mathcal{S}$ and $\mathcal{S}_2 \sqsubseteq \mathcal{S}$. Consequently we have $X \subseteq Z$, $Y \subseteq Z$, $R \subseteq P$ and $Q \subseteq P$. We have also $X \cup Y \subseteq Z$, $A = pr_2R \subseteq pr_2P$, $B = pr_2Q \subseteq pr_2P$, therefore $A \cup B \subseteq pr_2P = C$. We observe that if $\{Z_n\}_{n\geq 0}$ is given by (2.11) then for each natural number n,

$$Z_n \subseteq P \tag{2.19}$$

Really, for $n = 0$ the relation (2.19) is verified because $Z_0 = R_0 \cup Q_0$ and $R_0 \subseteq R \subseteq P$, $Q_0 \subseteq Q \subseteq P$. Suppose (2.19) is satisfied by Z_n. Take $(x, \theta(u, v), y) \in Z_{n+1}$. If $(x, \theta(u, v), y) \in Z_n$ then by (2.19) we have

$(x, \theta(u, v), y) \in P$. Otherwise, there is $z \in X \cup Y$ such that $(x, u, z) \in Z_n \subseteq P$ and $(z, v, y) \in Z_n \subseteq P$. By the property (2.3) of P we have $(x, \theta(u, v), y) \in P$. Thus, (2.19) is verified for every $n \geq 0$. Particularly we have $Z_{n_0} \subseteq P$. In conclusion, $\mathcal{S}_1 \vee \mathcal{S}_2 \subseteq \mathcal{S}$ and therefore $\mathcal{S}_1 \vee \mathcal{S}_2$ is the least upper bound for $\{\mathcal{S}_1, \mathcal{S}_2\}$. ■

Corollary 2.3.1. *The collection of all θ-schemas is a join semilattice.*

Proof. Really, a partially ordered set $(S, \subseteq)$ is a join semilattice if for every $x, y \in S$ there exists $sup\{x, y\}$.

2.4 Syntactical aspects

2.4.1 Overview

In this chapter we present the concept of derivation in a semantic schema $\mathcal{S}$, we define the mapping generated by $\mathcal{S}$, we define the set $\mathcal{F}_{comp}(\mathcal{S})$ containing the final result of the derivation process, we establish several algebraic properties for the set $\mathcal{F}_{comp}(\mathcal{S})$, we prepare the notions requested by the interpretation concept (the sort of an element from $\mathcal{F}_{comp}(\mathcal{S})$ and the class of an object).

2.4.2 The mapping generated by $\mathcal{S}$

Let $\mathcal{S} = (X, A_0, A, R)$ be a semantic schema. We consider a symbol h of arity 1, a symbol σ of arity 2 and take the set:

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

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

We denote by Z the alphabet which includes 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. As in the case of a rewriting system we define two rewriting rules in the next definition.

Definition 2.4.1. *Let be $w_1, w_2 \in Z^*$. We define the binary relation $\Rightarrow$ as follows:*

- *If $(x, a, y) \in R_0$ then $w_1(x, a, y)w_2 \Rightarrow w_1 h(x, a, y)w_2$*
- *Let be $(x, \theta(u, v), y) \in R$. If $(x, u, z) \in R$ and $(z, v, y) \in R$ then*

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

The relation $\Rightarrow$ is named **direct derivation** *over Z^*. We denote by $\Rightarrow^*$ and $\Rightarrow^+$ the reflexive and transitive closure of the relation $\Rightarrow$, respectively the transitive closure. The relation $\Rightarrow^*$ will be called simply* **derivation** *over Z^*.*

Definition 2.4.2. *For each* $w \in Z^*$ *where* $w = w_1 \dots w_n$ *with* $w_i \in Z, i \in \{1, ..., n\}$, $n \geq 1$, *we denote* $first(w) = w_1$ *and* $last(w) = w_n$.

Definition 2.4.3. *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* $a \in A_0$
- $\mathcal{G}_{\mathcal{S}}(x, \theta(u, v), y) = \{w \in \mathcal{H} \mid (x, \theta(u, v), y) \Rightarrow^* w\}$

The set $\mathcal{H}$ is an infinite one. We extract from $\mathcal{H}$ those elements which can be derived from R and we denote this set by $\mathcal{F}_{comp}(S)$. In other words,

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

Obviously we have

$$\mathcal{F}_{comp}(\mathcal{S}) = \bigcup_{(x,u,y) \in R} \mathcal{G}_{\mathcal{S}}(x, u, y) \tag{2.20}$$

Proposition 2.4.1. *Suppose* $(x, \theta(u, v), y) \in R$. *If* $(x, \theta(u, v), y) \Rightarrow^+ w$ *then:*

i) There is $z \in X$ *such that* $(x, \theta(u, v), y) \Rightarrow \sigma((x, u, z), (z, v, y)) \Rightarrow^* w$
ii) There are α *and* β *such that:*
1) $w = \sigma(\alpha, \beta)$
2) $(x, u, z) \Rightarrow^* \alpha$, $(z, v, y) \Rightarrow^* \beta$

Proof. The assertion *i)* is obviously true. We verify by induction on $n \geq 1$ that if $\sigma((x, u, z), (z, v, y)) \Rightarrow^n w$ then *ii)* is true and moreover, $last(\alpha) \in \{)\}$ and $first(\beta) \in \{(, \sigma, h\}$.
For $n = 1$ the following cases can be encountered:

1) $u \in A_0$ and $w = \sigma(h(x, u, z), (z, v, y))$. In that case $\alpha = h(x, u, z)$, $\beta = (z, v, y)$ and $(x, u, z) \Rightarrow h(x, u, z)$.
2) $v \in A_0$ and $w = \sigma((x, u, z), h(z, v, y))$. We have $\alpha = (x, u, z)$, $\beta = h(z, v, y)$ and $(z, v, y) \Rightarrow h(z, v, y)$.
3) $u = \theta(u_1, v_1)$, $w = \sigma(\sigma((x, u_1, z_1), (z_1, v_1, z)), (z, v, y))$,
$\alpha = \sigma((x, u_1, z_1), (z_1, v_1, z))$, $\beta = (z, v, y)$ for some $z_1 \in X$.
4) $v = \theta(u_2, v_2)$, $w = \sigma((x, u, z), \sigma((z, u_2, z_2), (z_2, v_2, y)))$, $\alpha = (x, u, z)$, $\beta = \sigma((z, u_2, z_2), (z_2, v_2, y))$ for some $z_2 \in X$.

We observe that the assertion is true for these cases. Suppose the assertion is true for n and consider a derivation:

$$\sigma((x, u, z), (z, v, y)) \Rightarrow^n w_1 \Rightarrow w$$

By the inductive assumption, there are α_1 and β_1 such that

$w_1 = \sigma(\alpha_1, \beta_1)$,

$(x,u,z) \Rightarrow^* \alpha_1$, $(z,v,y) \Rightarrow^* \beta_1$,
$last(\alpha_1) \in \{)\}$ and $first(\beta_1) \in \{(,\sigma,h\}$

We have $w_1 \Rightarrow w$, therefore the following cases can be encountered:

$i_1)$ $\sigma(\alpha_1,\beta_1) = \omega_1(x_1,a,y_1)\omega_2 \Rightarrow \omega_1 h(x_1,a,y_1)\omega_2 = w$, $a \in A_0$
$i_2)$ $\sigma(\alpha_1,\beta_1) = \omega_1(x_1,\theta(u_1,v_1),y_1)\omega_2 \Rightarrow \omega_1\sigma((x_1,u_1,z_1),(z_1,v_1,y_1))\omega_2$ $= w$ for some $z_1 \in X$

Let us take into consideration the assumption $last(\alpha_1) \in \{)\}$ and $first(\beta_1) \in \{(,\sigma,\, h\}$. It follows that the word

$$last(\alpha_1), first(\beta_1)$$

can be only one of the following words:

$$),($$

$$),\sigma$$

$$),h$$

therefore either α_1 is a subword of ω_1 or β_1 is a subword of ω_2.

The following cases are taken into consideration:

$a)$ Suppose α_1 is a subword of ω_1.
From $i_1)$ and $i_2)$ we deduce that (x_1,a,y_1) or $(x_1,\theta(u_1,v_1),y_1)$ is a subword of β_1.

- If (x_1,a,y_1) is a subword of β_1 then $\beta_1 = \mu_1(x_1,a,y_1)\mu_2$ for some words μ_1 and μ_2. In that case, from $i_1)$ we deduce that

$$\sigma(\alpha_1,\beta_1) = \sigma(\alpha_1,\mu_1(x_1,a,y_1)\mu_2) \Rightarrow \sigma(\alpha_1,\mu_1 h(x_1,a,y_1)\mu_2) = w$$

  therefore $w = \sigma(\alpha,\beta)$ for $\alpha = \alpha_1$ and $\beta = \mu_1 h(x_1,a,y_1)\mu_2$. But

$$(x,u,z) \Rightarrow^* \alpha_1 \Rightarrow^* \alpha$$

$$(z,v,y) \Rightarrow^* \beta_1 \Rightarrow \beta$$

  $last(\alpha) = last(\alpha_1) \in \{)\}$, $first(\beta) = first(\mu_1) = first(\beta_1)$ if μ_1 is a non-empty word and $first(\beta) = h$ if μ_1 is the empty word.
- Let us suppose that $(x_1,\theta(u_1,v_1),y_1)$ is a subword of β_1. In that case we obtain $\beta_1 = \mu_1(x_1,\theta(u_1,v_1),y_1)\mu_2$ and from $i_2)$ we deduce that
  $\sigma(\alpha_1,\beta_1) = \sigma(\alpha_1,\mu_1(x_1,\theta(u_1,v_1),y_1)\mu_2)$
  $\sigma(\alpha_1,\mu_1(x_1,\theta(u_1,v_1),y_1)\mu_2) \Rightarrow \sigma(\alpha_1,\mu_1\sigma((x_1,u_1,z_1),(z_1,v_1,y_1))\mu_2)$
  $\sigma(\alpha_1,\mu_1\sigma((x_1,u_1,z_1),(z_1,v_1,y_1))\mu_2) = w$
  therefore we have $w = \sigma(\alpha,\beta)$ for $\alpha = \alpha_1$ and $\beta = \mu_1\sigma((x_1,u_1,z_1),$ $(z_1,v_1,y_1))\mu_2$.

b) Suppose now that β_1 is a subword of ω_2. From i_1) and i_2) we deduce that (x_1, a, y_1) or $(x_1, \theta(u_1, v_1), y_1)$ is a subword of α_1. Suppose that (x_1, a, y_1) is a subword of α_1, therefore $\alpha_1 = \mu_1(x_1, a, y_1)\mu_2$. From i_1) we deduce that $\sigma(\alpha_1, \beta_1) = \sigma(\mu_1(x_1, a, y_1)\mu_2, \beta_1) \Rightarrow \sigma(\mu_1 h(x_1, a, y_1)\mu_2, \beta_1) = w$, therefore $w = \sigma(\alpha, \beta)$ for $\alpha = \mu_1 h(x_1, a, y_1)\mu_2$ and $\beta = \beta_1$. But $(x, u, z) \Rightarrow^* \alpha_1$ and $\alpha_1 \Rightarrow \alpha$, therefore $(x, u, z) \Rightarrow^* \alpha$. We have also $(z, v, y) \Rightarrow^* \beta_1$ and $\beta_1 = \beta$, therefore $(z, v, y) \Rightarrow^* \beta$. In addition, $first(\beta) = first(\beta_1)$ and $last(\alpha) \in \{)\}$ if μ_2 is the empty word. If μ_2 is a non-empty word, then $last(\alpha) = last(\mu_2) = last(\alpha_1)$.

Thus the proposition is proved. ■

Proposition 2.4.2. *If* $(x, u, y) \Rightarrow^+ \alpha$ *and* $\alpha \in (\{\sigma\} \cup M)^*$ *then* $\alpha \in \mathcal{H}$.

Proof. We prove by induction on n that if $(x, u, y) \Rightarrow^n \alpha$ and $\alpha \in (\{\sigma\} \cup M)^*$ then $\alpha \in \mathcal{H}$. We verify this property for n=1. If $(x, u, y) \Rightarrow \alpha$ then two cases are possible:

1) $u \in A_0$ and $\alpha = h(x, u, y)$. In this case we have $\alpha \in \mathcal{H}$.
2) $u \in A \setminus A_0$, therefore $u = \theta(u_1, v_1)$. In thhis case

$$\alpha = \sigma((x, v_1, z_1), (z_1, v_2, y))$$

for some $z_1 \in X$. This case is not possible because $\alpha \notin (\{\sigma\} \cup M)^*$.

Suppose the assertion is true for $n \in \{1, ..., k\}$ and take a derivation $(x, u, y) \Rightarrow^{k+1} \alpha$ such that $\alpha \in (\{\sigma\} \cup M)^*$. Because $k + 1 \geq 2$ and $\alpha \in (\{\sigma\} \cup M)^*$ we have $u = \theta(v_1, v_2)$ for some $v_1, v_2 \in A$. Really, if by contrary we suppose that $u \in A_0$ then we have:

$$(x, u, y) \Rightarrow h(x, u, y) \Rightarrow^k h^k(x, u, y) = \alpha$$

therefore $\alpha \notin (\{\sigma\} \cup M)^*$.

The derivation $(x, u, y) \Rightarrow^{k+1} \alpha$ can be written as follows:

$$(x, \theta(v_1, v_2), y) \Rightarrow \sigma((x, v_1, z), (z, v_2, y)) \Rightarrow^k \alpha$$

for some $z \in X$. Applying Proposition 2.4.1 we deduce that there are β_1, β_2 such that $(x, v_1, z) \Rightarrow^* \beta_1$, $(z, v_2, y) \Rightarrow^* \beta_2$ and $\alpha = \sigma(\beta_1, \beta_2)$. Because $\alpha \in (\{\sigma\} \cup M)^*$ we have $\beta_1, \beta_2 \in (\{\sigma\} \cup M)^*$. Applying the inductive assumption we have $\beta_1, \beta_2 \in \mathcal{H}$, therefore $\alpha = \sigma(\beta_1, \beta_2) \in \mathcal{H}$. ■

Proposition 2.4.3. *Suppose that* $w \in \mathcal{G}_{\mathcal{S}}(x, \theta(u, v), y)$ *and denote by* α *and* β *those elements of* $\mathcal{H}$*, uniquely determined, such that* $w = \sigma(\alpha, \beta)$*. There is* $z \in X$*, such that*

$(x, \theta(u, v), y) \Rightarrow \sigma((x, u, z), (z, v, y)) \Rightarrow^* w$
$\alpha \in \mathcal{G}_{\mathcal{S}}(x, u, z)$ *and* $\beta \in \mathcal{G}_{\mathcal{S}}(z, v, y)$

Proof. We have $(x, \theta(u, v), y) \Rightarrow^{+} w$ and $w \in \mathcal{H}$ because $w \in \mathcal{G}_{\mathcal{S}}(x, \theta(u, v), y)$. But $\mathcal{H}$ is a Peano σ-algebra, therefore w is written as $w = \sigma(\alpha, \beta)$ for $\alpha, \beta \in \mathcal{H}$ uniquely determined. By Proposition 2.4.1 there is $z \in X$ such that:

$$(x, \theta(u, v), y) \Rightarrow \sigma((x, u, z), (z, v, y)) \Rightarrow^{*} w$$

and there are α_1, β_1 such that $w = \sigma(\alpha_1, \beta_1)$, $(x, u, z) \Rightarrow^{*} \alpha_1$, $(z, v, y) \Rightarrow^{*} \beta_1$. By Proposition 2.4.2 we obtain $\alpha_1 \in \mathcal{H}$ and $\beta_1 \in \mathcal{H}$. But $w = \sigma(\alpha, \beta) = \sigma(\alpha_1, \beta_1)$, where $\alpha, \beta, \alpha_1, \beta_1 \in \mathcal{H}$. By the property of the Peano σ-algebra $\mathcal{H}$, we have $\alpha = \alpha_1$ and $\beta = \beta_1$. In conclusion, the proposition is proved. ■

Remark 2.4.1. Finally we shall prove that just one element z satisfies the conditions of the previous proposition.

Proposition 2.4.4. *If* $(x, u, z) \Rightarrow^{*} \alpha$ *and* $(z, v, y) \Rightarrow^{*} \beta$ *then*

$$\sigma((x, u, z), (z, v, y)) \Rightarrow^{*} \sigma(\alpha, \beta)$$

Proof. There are the following derivations:

$$(x, u, z) \Rightarrow \omega_1 \Rightarrow \omega_2 \Rightarrow ... \Rightarrow \omega_k \Rightarrow \alpha$$
$$(z, v, y) \Rightarrow w_1 \Rightarrow w_2 \Rightarrow ... \Rightarrow w_r \Rightarrow \beta$$

We know that if $\mu \Rightarrow \nu$ is a direct derivation and $w \in Z^{*}$ then $w\mu \Rightarrow w\nu$ and $\mu w \Rightarrow \nu w$. Based on this property we obtain the following derivations:

$$\sigma((x, u, z), (z, v, y)) \Rightarrow \sigma(\omega_1, (z, v, y)) \Rightarrow ... \Rightarrow \sigma(\alpha, (z, v, y))$$
$$\sigma(\alpha, (z, v, y)) \Rightarrow \sigma(\alpha, w_1) \Rightarrow ... \Rightarrow \sigma(\alpha, \beta)$$

and the proposition is proved. ■

Corollary 2.4.1.

$$\mathcal{G}_{\mathcal{S}}(x, \theta(u, v), y) = \bigcup_{z \in X} \mathcal{G}_{\mathcal{S}}(x, u, z) \otimes \mathcal{G}_{\mathcal{S}}(z, v, y)$$

where $P \otimes Q = \{\sigma(u, v) \mid u \in P, v \in Q\}$.

Proof. By Proposition 2.4.3, if $w \in \mathcal{G}_{\mathcal{S}}(x, \theta(u, v), y)$ then $w \in \mathcal{G}_{\mathcal{S}}(x, u, z) \otimes_{\sigma} \mathcal{G}_{\mathcal{S}}(z, v, y)$. Conversely, consider $w = \sigma(\alpha, \beta)$, where $\alpha \in \mathcal{G}_{\mathcal{S}}(x, u, z)$ and $\beta \in \mathcal{G}_{\mathcal{S}}(z, v, y)$.

It follows that $(x, u, z) \Rightarrow^{*} \alpha$, $(z, v, y) \Rightarrow^{*} \beta$ and $\alpha \in \mathcal{H}$, $\beta \in \mathcal{H}$. On the other hand, if $(x, u, z) \Rightarrow^{*} \alpha$ and $(z, v, y) \Rightarrow^{*} \beta$ then

$$\sigma((x, u, z), (z, v, y)) \Rightarrow^{*} \sigma(\alpha, \beta) = w \tag{2.21}$$

as is stated in Proposition 2.4.4. But $\theta(u, v) \in A$, $(x, \theta(u, v), y) \in R$, $(x, u, z) \in R$ and $(z, v, y) \in R$. It follows that:

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

therefore using (2.4.1) we deduce $(x, \theta(u, v), y) \Rightarrow^{*} w$. We recall that $\alpha, \beta \in \mathcal{H}$ and $w = \sigma(\alpha, \beta)$, therefore $w \in \mathcal{H}$. In this way we have $w \in \mathcal{G}_{\mathcal{S}}(x, \theta(u, v), y)$ and the proposition is proved. ■

Definition 2.4.4. *We define:*

$H(h(x,a,y)) =< h(x,a,y) >$ *for* $h(x,a,y) \in M$
$H(\sigma(\alpha,\beta)) =< p,q >$, *where* $H(\alpha) =< p >$ *and* $H(\beta) =< q >$, $\sigma(\alpha,\beta) \in \mathcal{H}$, $\alpha \in \mathcal{H}$, $\beta \in \mathcal{H}$.

Proposition 2.4.5. *Let be* $u \in A$ *such that* $trace(u) =< a_1,\ldots,a_n >$. *For every* $\alpha \in \mathcal{G}_{\mathcal{S}}(x_1,u,z_1)$ *there are* $y_1,\ldots,y_{n-1} \in X$ *such that* $H(\alpha) =< h(x_1,a_1,y_1), h(y_1,a_2,y_2), \ldots, h(y_{n-1},a_n,z_1) >$ *for* $n \geq 2$ *and* $H(\alpha) =< h(x_1,u,z_1) >$ *for* $n = 1$.

Proof. We proceed by induction on n. For n=1 we have $trace(u) =< a_1 >$, therefore $u = a_1 \in A_0$. If α is an arbitrary element of $\mathcal{G}_{\mathcal{S}}(x_1,u,z_1)$ then $(x_1,u,z_1) \Rightarrow^* \alpha$ and $\alpha \in \mathcal{H}$. This derivation is a direct one, that is $(x_1,u,z_1) \Rightarrow h(x_1,u,z_1) = \alpha$. It follows that $H(\alpha) =< h(x_1,u,z_1) >$ and the property is verified for n=1.

Consider $k \geq 1$ and suppose the proposition is true for $n \in \{1,\ldots,k\}$. Take an element $u \in A$ such that $trace(u) =< a_1,\ldots,a_{k+1} >$. There is $u_1, v_1 \in A$ such that $u = \theta(u_1,v_1)$. Take an element $\alpha \in \mathcal{G}_{\mathcal{S}}(x_1,u,z_1) = \mathcal{G}_{\mathcal{S}}(x_1,\theta(u_1,v_1),z_1)$. By Corollary 2.4.1 we deduce that there is $z \in X$ such that $\alpha = \sigma(\alpha_1,\beta_1)$, where $\alpha_1 \in \mathcal{G}_{\mathcal{S}}(x_1,u_1,z)$ and $\beta_1 \in \mathcal{G}_{\mathcal{S}}(z,v_1,z_1)$. We use the inductive assumption. Because $u = \theta(u_1,v_1)$ and $trace(u) =< a_1,\ldots,a_{k+1} >$, it follows that there is $i \in \{1,\ldots,k\}$ such that $trace(u_1) =< a_1,\ldots,a_i >$ and $trace(v_1) =< a_{i+1},\ldots,a_{k+1} >$.

By the inductive assumption we have the following properties:

1) there are $y_1,\ldots,y_{i-1} \in X$ such that

$$H(\alpha_1) =< h(x_1,a_1,y_1), h(y_1,a_2,y_2), \ldots, h(y_{i-1},a_i,z) >$$

2) there are $t_1,\ldots,t_{k-i} \in X$ such that

$$H(\beta_1) =< h(z,a_{i+1},t_1), h(t_1,a_{i+2},t_2), \ldots, h(t_{k-i},a_{k+1},z_1) >$$

But $\alpha = \sigma(\alpha_1,\beta_1)$, therefore $H(\alpha)$ is the following system:

$$< h(x_1,a_1,y_1), h(y_1,a_2,y_2), \ldots, h(y_{i-1},a_i,z),$$
$$h(z,a_{i+1},t_1), \ldots, h(t_{k-i},a_{k+1},z_1) >$$

and the proposition is proved. ■

Corollary 2.4.2. *If* $\mathcal{G}_{\mathcal{S}}(x_1,u,z_1) \cap \mathcal{G}_{\mathcal{S}}(x_2,v,z_2) \neq \emptyset$ *then* $x_1 = x_2$, $trace(u) = trace(v)$ *and* $z_1 = z_2$.

Proof. If $\alpha \in \mathcal{G}_{\mathcal{S}}(x_1,u,z_1) \cap \mathcal{G}_{\mathcal{S}}(x_2,v,z_2)$ and $trace(u) =< a_1,\ldots,a_n >$, $trace(v) =< b_1,\ldots,b_k >$ then by Proposition 2.4.5 there are $y_1,\ldots,y_{n-1}$, $t_1,\ldots,t_{k-1} \in X$ such that:

$H(\alpha) =< h(x_1,a_1,y_1), h(y_1,a_2,y_2), \ldots, h(y_{n-1},a_n,z_1) >$

$$H(\alpha) =< h(x_2, b_1, t_1), h(t_1, b_2, t_2), \ldots, h(t_{k-1}, b_k, z_2) >$$

therefore $n = k$, $a_1 = b_1$, ..., $a_n = b_k$, $x_1 = x_2$, $y_1 = t_1$, ..., $y_{n-1} = t_{k-1}$ and $z_1 = z_2$. Thus, $x_1 = x_2$, $trace(u) = trace(v)$ and $z_1 = z_2$. ■

Corollary 2.4.3. *The element $z \in X$ from Proposition 2.4.3 is uniquely determined.*

Proof. If $\alpha \in \mathcal{G}_{\mathcal{S}}(x, u, z_1) \cap \mathcal{G}_{\mathcal{S}}(x, u, z_2)$ then $z_1 = z_2$ by Corollary 2.4.2. ■

2.4.3 Algebraic properties of $\mathcal{F}_{comp}(\mathcal{S})$

In this section several new properties of the set $\mathcal{F}_{comp}(\mathcal{S})$ are established. Because we work with two semantic schemas $\mathcal{S}$ and $\mathcal{P}$, we denote by $\mathcal{H}_{\mathcal{S}}$ and $\mathcal{H}_{\mathcal{P}}$ the Peano σ-algebras generated by $\mathcal{S}$, respective $\mathcal{P}$.

Proposition 2.4.6. $\mathcal{F}_{comp}(\mathcal{S}) \in Initial(\mathcal{H}_{\mathcal{S}})$

Proof. Let be $w \in \mathcal{F}_{comp}(\mathcal{S})$ and $\alpha, \beta \in \mathcal{H}_{\mathcal{S}}$ such that $w = \sigma(\alpha, \beta)$. Applying Proposition 2.4.3 we deduce that there are $(x, u, z) \in R$, $(z, v, y) \in R$ such that $(x, u, z) \Rightarrow^* \alpha$, $(z, v, y) \Rightarrow^* \beta$, therefore $\alpha, \beta \in \mathcal{F}_{comp}(\mathcal{S})$. ■

An useful property of the Peano algebras is specified in the following proposition:

Proposition 2.4.7. $\overline{A_0 \cap B_0} = \overline{A_0} \cap \overline{B_0}$

Proof. Using (2.1) we can write

$$\overline{A_0} = \bigcup_{i \geq 0} M_i, \quad \overline{B_0} = \bigcup_{i \geq 0} P_i$$

where $M_0 = A_0$, $M_{i+1} = A_{i+1} \setminus A_i$, $P_0 = B_0$, $P_{i+1} = B_{i+1} \setminus B_i$ $(i \geq 0)$. We observe that for every $i \geq 0$ and $j \geq 0$ we have

$$M_{i+1} = \bigcup_{k=0}^{i} (M_k \otimes M_i) \cup \bigcup_{k=0}^{i-1} (M_i \otimes M_k) \tag{2.22}$$

$$P_{j+1} = \bigcup_{k=0}^{j} (P_k \otimes P_j) \cup \bigcup_{k=0}^{j-1} (P_j \otimes P_k) \tag{2.23}$$

We verify by induction on i $(i \geq 1)$ that

$$M_k \cap P_s = \emptyset, \quad k \neq s, \quad k, s \in \{0, \ldots, i\} \tag{2.24}$$

Obviously $M_0 \cap P_1 = M_1 \cap P_0 = \emptyset$ and therefore (2.24) is true for $i = 1$. Suppose (2.24) is true for i and due to the symmetry of the reasoning we observe that to prove (2.24) for $i + 1$ is enough to verify that

$$M_{i+1} \cap P_s = \emptyset \tag{2.25}$$

for every $s \in \{0, \ldots, i\}$. For $s = 0$ the property (2.25) is obviously true. From (2.23) we have

$$P_s = \bigcup_{r=0}^{s-1} (P_r \otimes P_{s-1}) \cup \bigcup_{r=0}^{s-2} (P_{s-1} \otimes P_r) \tag{2.26}$$

for $s \geq 1$. Using the distributivity of the set operations, the inductive assumption and the following properties:

- $(M_k \otimes M_i) \cap (P_r \otimes P_{s-1}) = \emptyset$ because $s - 1 < i$
- $(M_k \otimes M_i) \cap (P_{s-1} \otimes P_r) = \emptyset$ because $r \leq s - 2 < i$
- $(M_i \otimes M_k) \cap (P_r \otimes P_{s-1}) = \emptyset$ because $r \leq s - 1 < i$
- $(M_i \otimes M_k) \cap (P_{s-1} \otimes P_r) = \emptyset$ because $s - 1 < i$

we obtain (2.25) from (2.22) and (2.26). ■

Proposition 2.4.8. *If $\mathcal{F}_{comp}(\mathcal{S}) \subseteq \mathcal{H}_{\mathcal{S}}$ and $\mathcal{F}_{comp}(\mathcal{P}) \subseteq \mathcal{H}_{\mathcal{P}}$ then*

$$\mathcal{F}_{comp}(\mathcal{S}) \cap \mathcal{F}_{comp}(\mathcal{P}) \in Initial(\mathcal{H}_{\mathcal{S}} \cap \mathcal{H}_{\mathcal{P}})$$

Proof. Take $w \in \mathcal{F}_{comp}(\mathcal{S}) \cap \mathcal{F}_{comp}(\mathcal{P})$. Denote $\mathcal{H}_{\mathcal{S}} = \overline{C_0}$ and $\mathcal{H}_{\mathcal{P}} = \overline{D_0}$. It follows that $w \in \mathcal{F}_{comp}(\mathcal{S}) \subseteq \mathcal{H}_{\mathcal{S}} = \overline{C}_0$ and $w \in \mathcal{F}_{comp}(\mathcal{P}) \subseteq \mathcal{H}_{\mathcal{P}} = \overline{D}_0$ therefore $w \in \mathcal{H}_{\mathcal{S}} \cap \mathcal{H}_{\mathcal{P}}$. By Proposition 2.4.7 we have $\mathcal{H}_{\mathcal{S}} \cap \mathcal{H}_{\mathcal{P}} = \overline{C_0 \cap D_0}$ therefore $w \in \overline{C_0 \cap D_0}$. Suppose $w = \sigma(\alpha, \beta)$, where $\alpha, \beta \in \overline{C_0 \cap D_0}$. From the fact that $w \in \mathcal{F}_{comp}(\mathcal{S}) \in Initial(\mathcal{H}_{\mathcal{S}})$ we deduce that there are $\alpha_S, \beta_S \in \mathcal{F}_{comp}(\mathcal{S})$, uniquely determined, such that $w = \sigma(\alpha_S, \beta_S)$. Similarly, from $w \in \mathcal{F}_{comp}(\mathcal{P}) \in Initial(\mathcal{H}_{\mathcal{P}})$ we deduce that there are $\alpha_P, \beta_P \in \mathcal{F}_{comp}(\mathcal{P})$, uniquely determined, such that $w = \sigma(\alpha_P, \beta_P)$. We conclude that the following relations were obtained:

$w = \sigma(\alpha, \beta) \in \mathcal{F}_{comp}(\mathcal{S}) \subseteq \mathcal{H}_{\mathcal{S}}$
$\alpha, \beta \in \mathcal{H}_{\mathcal{S}} \cap \mathcal{H}_{\mathcal{P}} \subseteq \mathcal{H}_{\mathcal{S}}$
$w = \sigma(\alpha_S, \beta_S)$
$\alpha_S, \beta_S \in \mathcal{F}_{comp}(\mathcal{S}) \subseteq \mathcal{H}_{\mathcal{S}}$

From these relations we deduce that $\alpha = \alpha_s$ and $\beta = \beta_S$. Similar we obtain $\alpha = \alpha_P$ and $\beta = \beta_P$. But $\alpha_S \in \mathcal{F}_{comp}(\mathcal{S})$, $\alpha_P \in \mathcal{F}_{comp}(\mathcal{P})$ and $\alpha = \alpha_S = \alpha_P$. It follows that $\alpha \in \mathcal{F}_{comp}(\mathcal{S}) \cap \mathcal{F}_{comp}(\mathcal{P})$. Similar we have $\beta \in \mathcal{F}_{comp}(\mathcal{S}) \cap \mathcal{F}_{comp}(\mathcal{P})$. ■

In order to simplify the proof of some properties we introduce the following definition.

Definition 2.4.5. *A* **left derivation** *is a derivation $w_1 \Rightarrow w_2 \Rightarrow \ldots$such that for every $i \geq 1$ the direct derivation $w_i \Rightarrow w_{i+1}$ has the property that the leftmost triple from w_i is replaced to obtain w_{i+1}.*

Obviously we have the following property:

Proposition 2.4.9. *If w is derived from (x, u, y) then there is a left derivation of w from (x, u, y).*

Proposition 2.4.10. *We consider two semantic schemas $\mathcal{S}$ and $\mathcal{P}$. If $w \in F_{comp}(\mathcal{S}) \cap \mathcal{F}_{comp}(\mathcal{P})$ then $(x,u,y) \Rightarrow^* w$ is a left derivation in $\mathcal{S}$ if and only if it is a left derivation in $\mathcal{P}$.*

Proof. Consider the schemas $\mathcal{S} = (X, A_0, A, R)$ and $\mathcal{P} = (Y, B_0, B, Q)$ and denote $R_0 = R \cap (X \times A_0 \times X)$, $Q_0 = Q \cap (Y \times B_0 \times Y)$. Obviously it is enough to prove the following property: *if $(x,u,y) \Rightarrow^* w$ is a left derivation in $\mathcal{S}$ then it is a left derivation in $\mathcal{P}$*. We consider the increasing sequence of natural numbers $1 = l_1 < l_2 < \ldots$ such that l_i is the length of a derivation in $\mathcal{S}$. In order to verify our sentence we proceed by induction on i.

• If $i = 1$ then $w = h(x,u,y)$ and $(x,u,y) \in R_0$. But $w \in \mathcal{F}_{comp}(\mathcal{P})$, therefore $(x,u,y) \in Q_0$ and thus the property is true for $i = 1$.

• Suppose the property is true for $l_1, \ldots, l_i$ and take a left derivation $(x, \theta(u,v), y) \Rightarrow^* w$ of length l_{i+1} in $\mathcal{S}$. By Proposition 2.4.6 and Proposition 2.4.8 there are $\alpha, \beta \in \mathcal{F}_{comp}(\mathcal{S}) \cap \mathcal{F}_{comp}(\mathcal{S})$ and there is z such that $w = \sigma(\alpha, \beta)$ and the following computations give left derivations in $\mathcal{S}$:

$(x, \theta(u,v), y) \Rightarrow \sigma((x,u,z),(z,v,y)) \Rightarrow^* w$

$(x,u,z) \Rightarrow^* \alpha$, $(z,v,y) \Rightarrow^* \beta$

By the inductive assumption $(x,u,z) \Rightarrow^* \alpha$ and $(z,v,y) \Rightarrow^* \beta$ are derivations in $\mathcal{P}$. We have also $(x,u,z) \in R \cap Q$ and $(z,v,y) \in R \cap Q$. Thus $(x, \theta(u,v), y) \Rightarrow \sigma((x,u,z),(z,v,y))$ is a left derivation in $\mathcal{P}$ and the proposition is proved. ■

Proposition 2.4.11. *If $\mathcal{S} \sqsubseteq \mathcal{P}$ then $\mathcal{F}_{comp}(\mathcal{S}) \subseteq \mathcal{F}_{comp}(\mathcal{P})$.*

Proof. We observe that $R = \bigcup_{n \geq 0} R_n$, where $R_n = \{(x,u,y) \in R \mid u \in A_n\}$. We verify by induction on n the following property:

$$\mathcal{G}_{\mathcal{S}}(x,u,y) \subseteq \mathcal{G}_{\mathcal{P}}(x,u,y) \tag{2.27}$$

for every $(x,u,y) \in R_n$.

• If $(x,u,y) \in R_0$ then $u \in A_0$. In this case, if $w \in \mathcal{G}_{\mathcal{S}}(x,u,y)$ then $w = h(x,a,y)$. But $x,y \in A_0$, $u \in A_0$, $X^1 \subseteq X^2$ and $A_0 \subseteq B_0$, therefore $h(x,a,y) \in \mathcal{G}_{\mathcal{P}}(x,u,y)$. Thus (2.27) is verified for $n = 1$.

• Suppose (2.27) is true for $n = k$ and take an element $w \in \mathcal{G}_{\mathcal{S}}(x,u,y)$ such that $(x,u,y) \in R_{k+1}$. It follows that $u = \theta(u_1,v_1) \in A_{k+1}$. Either $u \in A_k$ or $u \in A_{k+1} \setminus A_k$. If $u \in A_k$ then $(x,u,y) \in R_k$ and the property is true for $n = k$. If $u \in A_{k+1} \setminus A_k$ then $u_1, v_1 \in A_k$. But from Corollary 2.4.1 we have

$$\mathcal{G}_{\mathcal{S}}(x, \theta(u_1,v_1), y) = \bigcup_{z \in X^1} \mathcal{G}_{\mathcal{S}}(x,u_1,z) \otimes \mathcal{G}_{\mathcal{S}}(z,v_1,y)$$

It follows that we find $z \in X^1$, $\alpha \in \mathcal{G}_{\mathcal{S}}(x,u_1,z)$, $\beta \in \mathcal{G}_{\mathcal{S}}(z,v_1,y)$ such that $w = \sigma(\alpha,\beta)$. We observe that $(x,u_1,z) \in R_k$ and $(z,v_1,y) \in R_k$ therefore we can apply the inductive assumption. Based on this assumption we have $\mathcal{G}_{\mathcal{S}}(x,u_1,z) \subseteq \mathcal{G}_{\mathcal{P}}(x,u_1,z)$ and $\mathcal{G}_{\mathcal{S}}(x,v_1,z) \subseteq \mathcal{G}_{\mathcal{P}}(x,v_1,z)$. therefore $\alpha \in \mathcal{G}_{\mathcal{P}}(x,u_1,z)$, $\beta \in \mathcal{G}_{\mathcal{P}}(x,v_1,z)$. But

$$\mathcal{G}_{\mathcal{P}}(x, \theta(u_1, v_1), y) = \bigcup_{t \in X^2} \mathcal{G}_{\mathcal{P}}(x, u_1, t) \otimes \mathcal{G}_{\mathcal{P}}(t, v_1, y)$$

On the other hand $z \in X^1$ and $X^1 \subseteq X^2$, therefore $z \in X^2$. Choosing $t = z$ we obtain $\sigma(\alpha, \beta) \in \mathcal{G}_{\mathcal{P}}(x, u, y)$ and (2.27) is proved. Using (2.27) we obtain

$$\bigcup_{(x,u,y) \in R} \mathcal{G}_{\mathcal{S}}(x, u, y) \subseteq \bigcup_{(x,u,y) \in R} \mathcal{G}_{\mathcal{P}}(x, u, y)$$

But $R \subseteq Q$ therefore

$$\bigcup_{(x,u,y) \in R} \mathcal{G}_{\mathcal{P}}(x, u, y) \subseteq \bigcup_{(x,v,y) \in Q} \mathcal{G}_{\mathcal{P}}(x, u, y)$$

Now the proposition is proved if we use (2.20). ∎

2.4.4 Sorted elements

In order to introduce the concept of sorted element some preliminary results are needed. An useful property is proved in the following proposition.

Proposition 2.4.12. *If* $w \in \mathcal{G}_{\mathcal{S}}(x, u_1, z) \cap \mathcal{G}_{\mathcal{S}}(x, u_2, z)$ *then* $u_1 = u_2$.

Proof. First we observe that by Corollary 2.4.2 we have $trace(u_1) = trace(u_2)$. If $trace(u) =< a_1, \ldots, a_s >$ then we denote $length(u) = s$. Using this notation we consider the increasing sequence $l_1 = 1 < l_2 < \ldots$ of the elements from A. We verify the property by induction on i, where $length(u_1) = length(u_2) = l_i$.

• For $i = 1$ we have $u_1, u_2 \in A_0$, $w = h(x, u_1, y) = h(x, u_2, y)$, therefore $u_1 = u_2$.

• Suppose the property is true for every $u_1, u_2 \in A$ such that $length(u_1) = length(u_2) = l_i$, $i \in \{1, \ldots, k\}$. We verify the property for $i = k + 1$. Thus we suppose that $w \in \mathcal{G}_{\mathcal{S}}(x, u_1, z) \cap \mathcal{G}_{\mathcal{S}}(x, u_2, z)$ and $length(u_1) = length(u_2) = l_{k+1}$. By a basic property of a Peano algebra we deduce that there are $p_1, p_2, q_1, q_2 \in A$, uniquely determined, such that $u_1 = \theta(p_1, p_2)$ and $u_2 = \theta(q_1, q_2)$. Let us consider the elements $\alpha \in \mathcal{H}$ and $\beta \in \mathcal{H}$, uniquely determined, such that $w = \sigma(\alpha, \beta)$. We have

$(x, u_1, y) = (x, \theta(p_1, p_2), y) \Longrightarrow^* w = \sigma(\alpha, \beta)$

Applying Proposition 2.4.3 we deduce that there is z_1, uniquely determined, such that

$(x, \theta(p_1, p_2), y) \Rightarrow \sigma((x, p_1, z_1), (z_1, p_2, y)) \Rightarrow^* w$

$\alpha \in \mathcal{G}_{\mathcal{S}}(x, p_1, z_1)$ and $\beta \in \mathcal{G}_{\mathcal{S}}(z_1, p_2, y)$

Similarly we have

$(x, u_2, y) = (x, \theta(q_1, q_2), y) \Longrightarrow^* w = \sigma(\alpha, \beta)$ therefore

$(x, \theta(q_1, q_2), y) \Rightarrow \sigma((x, q_1, z_2), (z_2, q_2, y)) \Rightarrow^* w$

$\alpha \in \mathcal{G}_{\mathcal{S}}(x, q_1, z_2)$ and $\beta \in \mathcal{G}_{\mathcal{S}}(z_2, q_2, y)$

for an element $z_2 \in X$, uniquely determined. We observe that

$\alpha \in \mathcal{G}_{\mathcal{S}}(x, p_1, z_1) \cap \mathcal{G}_{\mathcal{S}}(x, q_1, z_2)$ $\beta \in \mathcal{G}_{\mathcal{S}}(z_1, p_2, y) \cap \mathcal{G}_{\mathcal{S}}(z_2, q_2, y)$
Applying Corollary 2.4.2 we obtain $z_1 = z_2$, $trace(p_1) = trace(q_1)$ and $trace(p_2) = trace(q_2)$. Denoting $z = z_1 = z_2$ we observe that
$\alpha \in \mathcal{G}_{\mathcal{S}}(x, p_1, z) \cap \mathcal{G}_{\mathcal{S}}(x, q_1, z)$
and $length(p_1) = length(q_1) < l_{k+1}$. Applying the inductive assumption we have $p_1 = q_1$. Similarly we obtain $p_2 = q_2$, therefore $u_1 = u_2$. ∎

Now we can prove the basic property which allows us to introduce the concept of sorted element.

Proposition 2.4.13. *For every $w \in \mathcal{F}_{comp}(\mathcal{S})$ just one element $(x, u, y) \in R$ satisfies the property $(x, u, y) \Longrightarrow^* w$.*

Proof. Based on Corollary 2.4.2 we deduce that if $w \in \mathcal{G}_{\mathcal{S}}(x_1, u_1, z_1) \cap \mathcal{G}_{\mathcal{S}}(x_2, u_2, z_2)$ then $x_1 = x_2$, $z_1 = z_2$ and $trace(u_1) = trace(u_2)$. Now we apply Proposition 2.4.12 and we deduce $u_1 = u_2$. ∎

This result allows us to give the following definition.

Definition 2.4.6. *If $w \in \mathcal{F}_{comp}(\mathcal{S})$ then the element $u \in A$ such that $(x, u, y) \in R$ and $(x, u, y) \Longrightarrow^* w$ is named* **the sort** *of w and we denote $sort(w) = u$.*

Proposition 2.4.14. *Every element $w \in \mathcal{F}_{comp}(\mathcal{S})$ has a sort, which is uniquely determined.*

Proof. By definition of $\mathcal{F}_{comp}(\mathcal{S})$ every element has a sort. This is uniquely determined by Proposition 2.4.13. ∎

Proposition 2.4.15. *Suppose $\sigma(\alpha, \beta) \in \mathcal{F}_{comp}(\mathcal{S})$, $\alpha \in \mathcal{F}_{comp}(\mathcal{S})$ and $\beta \in \mathcal{F}_{comp}(\mathcal{S})$. If $sort(\sigma(\alpha, \beta)) = \theta(u, v)$ then $sort(\alpha) = u$ and $sort(\beta) = v$.*

Proof. Immediate from Proposition 2.4.3. ∎

We introduce now the following binary relation on $\mathcal{F}_{comp}(\mathcal{S})$:

Definition 2.4.7. *Let us consider $w_1, w_2 \in \mathcal{F}_{comp}(\mathcal{S})$. We write $w_1 \sim w_2$ if $sort(w_1) = sort(w_2)$ and for every $u \in A$ we denote*

$$[u]_{\mathcal{F}} = \{w \in \mathcal{F}_{comp}(\mathcal{S}) \mid sort(w) = u\}$$

An useful property in studying these sets is the following:

Proposition 2.4.16. *For every $(x, u, y) \in R$ we have $\mathcal{G}_S(x, u, y) \neq \emptyset$.*

Proof. Taking into consideration the relation (2.1) we observe that $A = \bigcup_{n \geq 0} B_n$, where $B_n = A \cap A_n$. We verify by induction on n the following property T(n):

For every $(x, u, y) \in R$ such that $u \in B_n$ we have $\mathcal{G}_S(x, u, y) \neq \emptyset$.

From Definition 2.4.3 and Definition 2.4.1 we observe that T(0) is true. Suppose T(0),..., T(m) are true and take an arbitrary element $(x, \theta(u, v), y) \in R$ such that $\theta(u, v) \in B_{m+1}$. There are $i \le m$ and $j \le m$ such that $u \in B_i$ and $v \in B_j$. By equation (2.2) there is an element $z \in X$ such that $(x, u, z) \in R$ and $(z, v, y) \in R$. By the inductive assumption we have $\mathcal{G}_\mathcal{S}(x, u, z) \neq \emptyset$ and $\mathcal{G}_\mathcal{S}(z, u, y) \neq \emptyset$. Take $w_1 \in \mathcal{G}_\mathcal{S}(x, u, z)$ and $w_2 \in \mathcal{G}_\mathcal{S}(z, u, y)$. By Corollary 2.4.1 we know that

$$\mathcal{G}_\mathcal{S}(x, \theta(u, v), y) = \bigcup_{z \in X} \mathcal{G}_\mathcal{S}(x, u, z) \otimes \mathcal{G}_\mathcal{S}(z, v, y)$$

Based on this property we obtain $\sigma(w_1, w_2) \in \mathcal{G}_\mathcal{S}(x, \theta(u, v), y)$, therefore $\mathcal{G}_\mathcal{S}(x, \theta(u, v), y) \neq \emptyset$. In other words, T(m+1) is true and the proposition is proved. ■

Proposition 2.4.17. *For every $u \in A$ we have $[u]_\mathcal{F} \neq \emptyset$.*

Proof. From (2.4) we deduce that for every $u \in A$ there is $x, y \in R$ such that $(x, u, y) \in R$. By Proposition 2.4.16 we have $\mathcal{G}_S(x, u, y) \neq \emptyset$. But $[u]_\mathcal{F} \supseteq \mathcal{G}_S(x, u, y)$, therefore $[u]_\mathcal{F} \neq \emptyset$. ■

Now we observe that the relation defined in Definition 2.4.7 is reflexive, symmetric and transitive, therefore it is an equivalence relation. Thus, the set $\mathcal{F}_{comp}(\mathcal{S})$ is divided into equivalence classes and all elements of an equivalence class have the same sort. In other words,

$$\mathcal{F}_{comp}(\mathcal{S}) = \bigcup_{u \in A} [u]_\mathcal{F}$$

The set $\mathcal{F}_{comp}(\mathcal{S})$ is the result of the formal computations defined by the schema $\mathcal{S}$.

In order to relieve the computations we consider the labeled graph represented in Figure 2.5 and $A = A_0 \cup \{\theta(a, b), \theta(b, a)\}$, where we observe that $A_0 = \{a, b\}$. It follows that

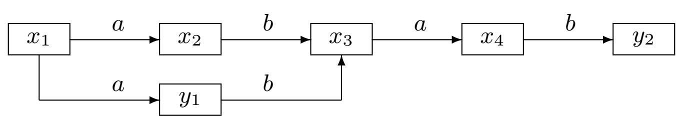

Fig. 2.5. The graph used to exemplify the computations

$$R = R_0 \cup \{(x_1, \theta(a, b), x_3), (x_3, \theta(a, b), y_2), \\ (x_2, \theta(b, a), x_4), (y_1, \theta(b, a), x_4)\}$$

where

$$R_0 = \{(x_1, a, x_2), (x_1, a, y_1), (x_3, a, x_4), (x_2, b, x_3),$$

$(y_1, b, x_3), (x_4, b, y_2)\}$

We obtain

$\mathcal{G}_S(x_1, \theta(a,b), x_3) = \{\sigma(h(x_1, a, x_2), h(x_2, b, x_3)),$
$\sigma(h(x_1, a, y_1), h(y_1, b, x_3))\}$

because

- $(x_1, \theta(a,b), x_3) \Longrightarrow \sigma((x_1, a, x_2), (x_2, b, x_3)) \Longrightarrow$
$\sigma(h(x_1, a, x_2), h(x_2, b, x_3))$
- $(x_1, \theta(a,b), x_3) \Longrightarrow \sigma((x_1, a, y_1), (y_1, b, x_3)) \Longrightarrow$
$\sigma(h(x_1, a, y_1), h(y_1, b, x_3))$

We have also

$\mathcal{G}_S(x_3, \theta(a,b), y_2) = \{\sigma(h(x_3, a, x_4), h(x_4, b, y_2))\}$

therefore

$$[\theta(a,b)]_{\mathcal{F}} = \{\alpha_1, \alpha_2, \beta_1\}$$

where $\alpha_1 = \sigma(h(x_1, a, x_2), h(x_2, b, x_3))$, $\alpha_2 = \sigma(h(x_1, a, y_1), h(y_1, b, x_3))$ and $\beta_1 = \sigma(h(x_3, a, x_4), h(x_4, b, y_2))$.

Similarly we obtain

$$[\theta(b,a)]_{\mathcal{F}} = \{\gamma_1, \gamma_2\}$$

where

$\gamma_1 = \sigma(h(x_2, a, x_3), h(x_3, a, x_4))$
$\gamma_2 = \sigma(h(y_1, a, x_3), h(x_3, a, x_4))$

and obviously

$[a]_{\mathcal{F}} = \{h(x_1, a, x_2), h(x_3, a, x_4), h(x_1, a, y_1)\}$
$[b]_{\mathcal{F}} = \{h(x_2, b, x_3), h(y_1, b, x_3), h(x_4, a, y_2)\}$

It follows that

$\mathcal{F}_{comp}(\mathcal{S}) = [a]_{\mathcal{F}} \cup [b]_{\mathcal{F}} \cup [\theta(a,b)]_{\mathcal{F}} \cup [\theta(b,a)]_{\mathcal{F}}$

2.4.5 Classes of objects

We consider:

- A semantic schema $\mathcal{S} = (X, A_0, A, R)$
- A bijective mapping $ob : X \rightarrow Ob$, where Ob is a set of objects (named **simple objects**).
- For each $u \in A$ we consider an algorithm Alg_u such that from two objects o_1 and o_2 another object $Alg_u(o_1, o_2)$ is obtained. In addition, we suppose that for $a \in A_0$, the output element $Alg_a(o_1, o_2)$ is defined only for $o_1, o_2 \in Ob$ (simple objects). The elements $Alg_u(o_1, o_2)$ are **complex objects**.

Definition 2.4.8. *We define recursively:*

- *The object $o = Alg_a(ob(x), ob(y))$ for $a \in A_0$ and $x, y \in X$ is a* **complex object** *of class a and we denote this property by $cls(o) = a$.*
- *If $cls(o_1) = u$, $cls(o_2) = v$ and $\theta(u,v) \in A$ then $o = Alg_{\theta(u,v)}(o_1, o_2)$ is a* **complex object** *and $cls(o) = \theta(u,v)$.*

We observe that an object of class $\theta(u, v)$ is the output object of the algorithm $Alg_{\theta(u,v)}$ for two input objects of class u, respectively v.

We remark that the elements of A are viewed as **sorts** for elements of $\mathcal{F}_{comp}(\mathcal{S})$ and **classes** for objects.

2.5 Semantical aspects

2.5.1 Overview

In this chapter we treat the concepts and results in conjunction with the semantics of a semantic schema. The semantics or meaning is obtained by means of the concept of interpretation. We introduce this concept and each interpretation defines a valuation mapping, which gets values from some space called semantic (or output) space.

2.5.2 Interpretation of a schema

At the beginning of this section we introduce the concept of interpretation. In essence, an interpretation attaches objects to the nodes of a semantic schema and is able to compute various objects of the semantic space. This structure is endowed with a set of algorithms, which organize the output space as a set of layers, each layer containing objects of the same class.

Definition 2.5.1. *Let be $\mathcal{S} = (X, A_0, A, R)$ a semantic schema. An* **interpretation** *$\mathcal{I}$ of $\mathcal{S}$ is a system $\mathcal{I} = (Ob, ob, \{Alg_u\}_{u\in A})$ where*

- *Ob is a finite set of elements which are called the* **objects** *of the interpretation*
- *$ob : X \to Ob$ is a bijective function.*
- *$\{Alg_u\}_{u\in A}$ is a set of algorithms such that each algorithm has two input parameters and an output parameter.*

Definition 2.5.2. *Consider an interpretation $\mathcal{I} = (Ob, ob, \{Alg_u\}_{u\in A})$ for $\mathcal{S}$. The* **output space** *Y of $\mathcal{I}$ is defined as follows:*

$$Y = \bigcup_{u\in A} Y_u \tag{2.28}$$

where

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

if $a \in A_0$ *and*

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

As we can view in (2.28) the output space Y is broken into layers. A layer is a set Y_u for some $u \in A$. We observe that each element of Y_u has the class u.

The mapping defined by an interpretation in Figure 2.1 is given in the next definition.

Definition 2.5.3. *We define recursively the valuation mapping*

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

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* $sort(\sigma(\alpha,\beta)) = \theta(u,v)$.

Remark 2.5.1.

$$sort(\sigma(\alpha,\beta)) = cls(Val_{\mathcal{I}}(\sigma(\alpha,\beta)))$$

Remark 2.5.2. $Val_{\mathcal{I}}$ is a partial mapping. More precisely, $Val_{\mathcal{I}}(\sigma(\alpha,\beta))$ is defined if and only if the following conditions are fulfilled:

$Val_{\mathcal{I}}(\alpha)$ and $Val_{\mathcal{I}}(\beta)$ are defined.
$Alg_{\theta(u,v)}$ returns a value if its input arguments are $Val_{\mathcal{I}}(\alpha)$ and $Val_{\mathcal{I}}(\beta)$.

Now we can define the output mapping of a semantic schema generated by an interpretation. This mapping computes for each pair of nodes all the meanings assigned in the output space Y. To do this, all the paths connecting the first node of the pair with the second node are considered. Each such path is characterized by some element of $\mathcal{F}_{comp}(\mathcal{S})$. We take the value of the mapping $Val_{\mathcal{I}}$ at this element and obtain the corresponding element from Y.

Definition 2.5.4. *If $\mathcal{I}$ is an interpretation of the semantic schema $\mathcal{S}$ then we can define the output mapping*

$$Out_{\mathcal{I}} : X \times X \rightarrow 2^Y$$

as follows:

$$Out_{\mathcal{I}}(x,y) = \bigcup_{(x,u,y)\in R} \; \bigcup_{w \in \mathcal{G}_{\mathcal{S}}(x,u,y)} \{Val_{\mathcal{I}}(w)\}$$

Let us consider the following example of computations. We consider the semantic schema

$$\mathcal{S} = (\{x_1,x_2,x_3,x_4\}, \{a,b,c\}, \{a,b,\theta(a,b),\theta(\theta(a,b),c)\}, \{(x_1,a,x_2),$$
$$(x_2,b,x_3),(x_3,c,x_4),(x_1,\theta(a,b),x_3),(x_1,\theta(\theta(a,b),c),x_4)\})$$

which is represented in Figure 2.6.

If C is a geometric figure such as a circle, a square, a rectangle etc then we denote by $int(C)$ the inner side of C and its frontier.

We take as interpretation the system $\mathcal{I} = (Ob, ob, \{Alg_u\}_{u\in A})$ containing the following components:

- $Ob = \{1, (2,2), (2,1/2), 1.2\}$
- $ob(x_1) = 1$; $ob(x_2) = (2,2)$; $ob(x_3) = (2,1/2)$; $ob(x_4) = 1.2$
- **Algorithm** $Alg_a(r : real, (x,y) : (real, real))$
  Take $o = int(C)$, where C is the circle of radius r and center (x,y); return o;
  End of algorithm
- **Algorithm** $Alg_b((x,y) : (real, real), (r_1, r_2) : (real, real))$
  Take $o = int(E)$, where E is the ellipse with horizontal radius r_1, vertical radius r_2 and center (x,y); return o;
  End of algorithm
- **Algorithm** $Alg_c((x,y) : (real, real), r : real)$
  Take $o = int(E)$, where E is the hexagon with center $(x, y+3/2)$, radius r and two sides parallel with Ox ; return o;
  End of algorithm
- **Algorithm** $Alg_{\theta(a,b)}(o_1 : a, o_2 : b)$
  If $E = (int(o_1) \cup int(o_2)) \setminus (int(o_2) \cap int(o_1)) \neq \emptyset$ then return E;
  End of algorithm
- **Algorithm** $Alg_{\theta(\theta(a,b),c)}(o_1 : \theta(a,b), o_2 : c)$
  If $E = (int(o_1) \cup int(o_2)) \setminus (int(o_2) \cap int(o_1)) \neq \emptyset$ then return E;
  End of algorithm

We obtain the following computation for $\mathcal{F}_{comp}(\mathcal{S})$:

$(x_1, a, x_2) \Rightarrow h(x_1, a, x_2)$
$(x_2, b, x_3) \Rightarrow h(x_2, b, x_3)$
$(x_3, c, x_4) \Rightarrow h(x_3, c, x_4)$
$(x_1, \theta(a,b), x_3) \Rightarrow^* \sigma(h(x_1, a, x_2), h(x_2, b, x_3))$
$(x_1, \theta(\theta(a,b),c), x_4) \Rightarrow^* \sigma(\sigma(h(x_1, a, x_2), h(x_2, b, x_3)), h(x_3, c, x_4))$

therefore

$$\mathcal{F}_{comp}(\mathcal{S}) = \{h(x_1, a, x_2), h(x_2, b, x_3), h(x_3, c, x_4), \sigma(h(x_1, a, x_2), h(x_2, b, x_3)) \\ \sigma(\sigma(h(x_1, a, x_2), h(x_2, b, x_3)), h(x_3, c, x_4))\}$$

and

$sort(h(x_1, a, x_2)) = a$, $sort(h(x_2, b, x_3)) = b$, $sort(h(x_3, c, x_4)) = c$
$sort(\sigma(h(x_1, a, x_2), h(x_2, b, x_3))) = \theta(a,b)$
$sort(\sigma(\sigma(h(x_1, a, x_2), h(x_2, b, x_3)), h(x_3, c, x_4))) = \theta(\theta(a,b),c)$

We obtain the following computations:

- $Val_{\mathcal{I}}(h(x_1, a, x_2)) = Alg_a(ob(x_1), ob(x_2)) = o_1$, $Y_a = \{o_1\}$, where o_1 is the circle represented in Figure 2.7(a);
- $Val_{\mathcal{I}}(h(x_2, b, x_3)) = Alg_b(ob(x_2), ob(x_3)) = o_2$, $Y_b = \{o_2\}$, where o_2 is the ellipse represented in Figure 2.7(b);
- $Val_{\mathcal{I}}(h(x_3, c, x_4)) = Alg_c(ob(x_3), ob(x_4)) = o_1$, $Y_c = \{o_3\}$, where o_3 is represented in Figure 2.8(b);

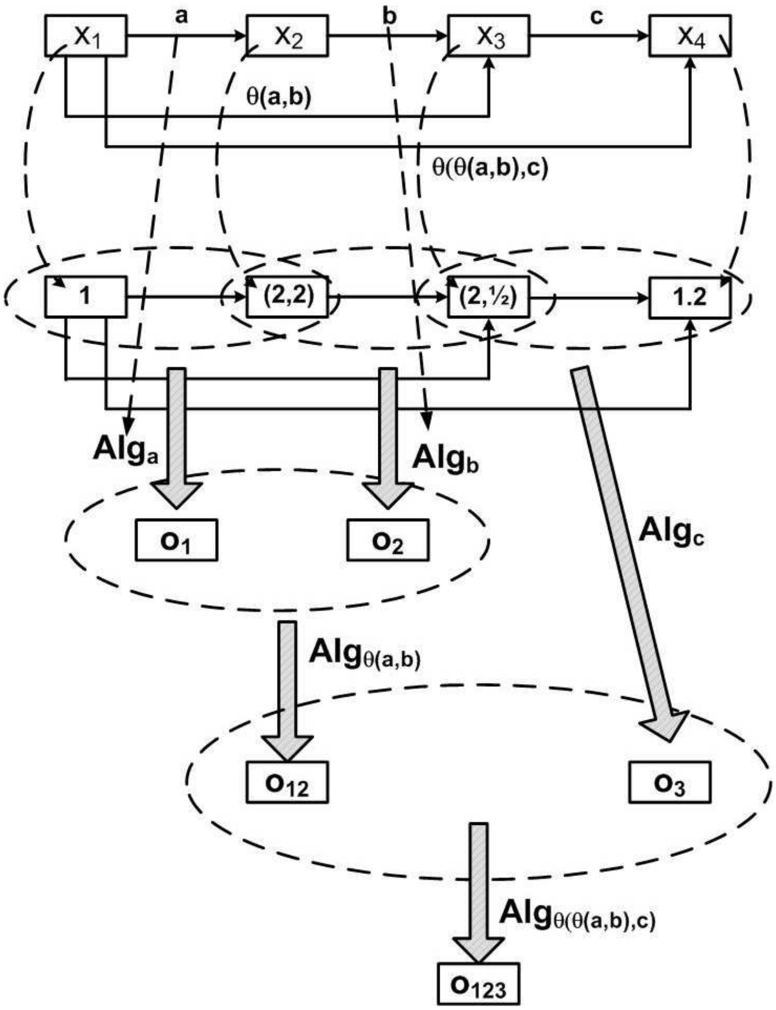

Fig. 2.6. An interpretation

$Val_{\mathcal{I}}(\sigma(h(x_1, a, x_2), h(x_2, b, x_3))) =$

$$Alg_{\theta(a,b)}(Val_{\mathcal{I}}(h(x_1, a, x_2)), Val_{\mathcal{I}}(h(x_2, b, x_3))) =$$

$$Alg_{\theta(a,b)}(o_1, o_2) = o_{12}$$

$Y_{\theta(a,b)} = \{o_{12}\}$, where o_{12} is the object represented in Figure 2.8(a);
$Val_{\mathcal{I}}(\sigma(\sigma(h(x_1, a, x_2), h(x_2, b, x_3)), h(x_3, c, x_4))) =$

$$Alg_{\theta(\theta(a,b),c)}(Val_{\mathcal{I}}(\sigma(h(x_1, a, x_2), h(x_2, b, x_3)), Val_{\mathcal{I}}(h(x_3, c, x_4)))) =$$

$$Alg_{\theta(\theta(a,b),c)}(o_{12}, o_3) = o_{123}$$

$Y_{\theta(\theta(a,b),c)} = \{o_{123}\}$, where o_{123} is represented in Figure 2.9.

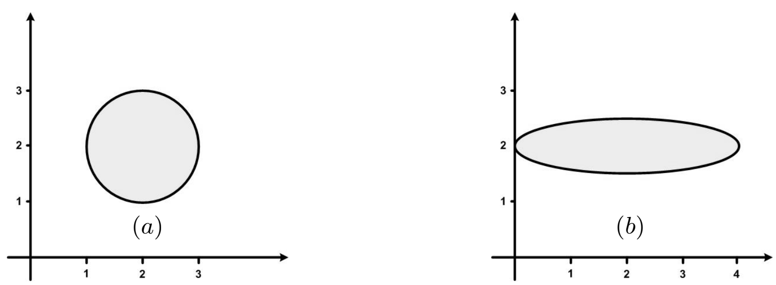

Fig. 2.7. (a)The object o_1; (b) The object o_2

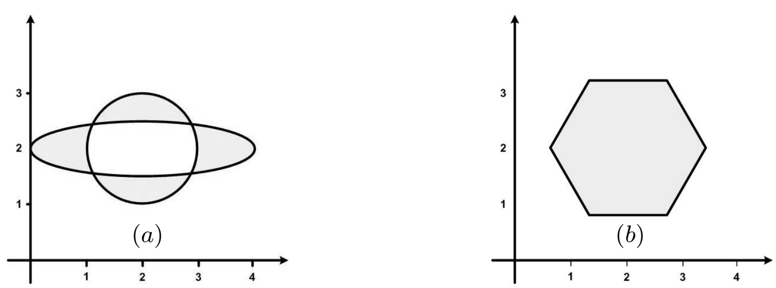

Fig. 2.8. (a)The object o_{12}; (b)The object o_3

In conclusion we obtain

$Out(x_1, x_2) = \{o_1\}$, $Out(x_2, x_3) = \{o_2\}$, $Out(x_3, x_4) = \{o_3\}$
$Out(x_1, x_3) = \{o_{12}\}$, $Out(x_1, x_4) = \{o_{123}\}$, $Out(x_2, x_4) = \emptyset$

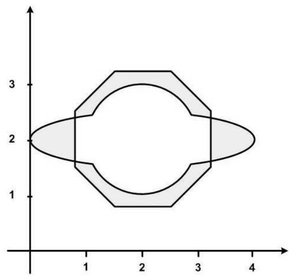

Fig. 2.9. The final object o_{123}

2.5.3 Compatible interpretations

In this section we give some details concerning the union of two algorithms and then we introduce the concept of compatible interpretations.

Definition 2.5.5. *The algorithms* Alg_1 *and* Alg_2 *are called* **compatible algorithms** *if*

$$Alg_1(v_1, v_2) = Alg_2(v_1, v_2)$$

for every $(v_1, v_2) \in dom(Alg_1) \cap dom(Alg_2)$*. We denote by* $Alg_1 \sim_C Alg_2$ *the property "*Alg_1 *and* Alg_2 *are compatible".*

The relation $\sim_C$ is reflexive and symmetric, but is not transitive as we can view in the following example.

Algorithm $Alg_1^1(r : real, (x, y) : (real, real))$
If $r > 10$ then **return** $s = x + y$;
End of algorithm

It is not difficult to observe that for this case

$$dom(Alg_1^1) = \{r \in \mathcal{R} | r > 10\} \times (\mathcal{R} \times \mathcal{R})$$

where $\mathcal{R}$ is the set of the real numbers.

Algorithm $Alg_2^1(r : real, (x, y) : (real, real))$
If $r \leq 10$ then **return** $s = x - y$; if $r > 11$ then return $s = x + y$;
End of algorithm

For Alg_2^1 we have

$$dom(Alg_2^1) = \{r \in \mathcal{R} | r \leq 10\} \times (\mathcal{R} \times \mathcal{R})$$

Now we consider the following algorithm Alg_3^1:
Algorithm $Alg_3^1(r : real, (x, y) : (real, real))$
If $r \leq 10$ then **return** $s = x - y$; if $r > 10$ and $r < 11$ then return $s = 5$;
End of algorithm

We observe that $Alg_1^1 \sim_C Alg_2^1$ and $Alg_2^1 \sim_C Alg_3^1$. But $Alg_1^1 \not\sim_C Alg_3^1$ because for $10 < r < 11$, $x = 4$ and $y = 3$ we have $Alg_1^1(r, (x, y)) = 7$ and $Alg_3^1(r, (x, y)) = 5$.

We can introduce a partial binary relation between two algorithms as we show in the following definition:

Definition 2.5.6. *We write $Alg_a \prec Alg_b$ if $dom(Alg_a) \subseteq dom(Alg_b)$ and $Alg_a(v_1, v_2) = Alg_b(v_1, v_2)$ for every $(v_1, v_2) \in dom(Alg_a)$.*

Obviously the relation $\prec$ is reflexive and transitive. But this is not a partial order because it is not antisymmetric. This property can be relieved as follows. We give an example of two algorithms Alg_4 and Alg_5 such that $dom(Alg_4) = dom(Alg_5)$, $Alg_4(v_1, v_2) = Alg_5(v_1, v_2)$ for all (v_1, v_2) and each algorithm is based on a distinct method. Such an example can be given to compute the value $z = a + \lfloor\sqrt{k}\rfloor$, where a and k are natural numbers and $\lfloor\sqrt{k}\rfloor$ is the largest integer p such that $p \leq \sqrt{k}$. A non trivial algorithm to compute z is based on the identity $1+3+5+\ldots+2n+1 = (n+1)^2$ and can be described as follows (Manna (1974), where the correctness of this algorithm is studied):

Algorithm $Alg_4(a : integer, k : integer)$
  if $k > 0$ then { $j := 1$; $sum := 1$; $i := 0$;
  while $sum \leq k$ do
    $i := i + 1$; $j := j + 2$; $sum := sum + j$;
  endwhile }
  endif
  return a+i;
End of algorithm

For example, $Alg_4(-2, 3) = -1$, $Alg_4(1, 6) = 3$ and so on.

The same problem can be solved by using the well known approximate method $x_{n+1} = 1/2(x_n + k/x_n)$. Using this method we obtain the algorithm Alg_5:

Algorithm $Alg_5(a : integer, k : integer)$
  if $k > 0$ then { $x := k$;
  while $|x^2 - k| > 0.001$ do
    $x := 1/2(x + k/x)$;
  endwhile }
  endif
  find the natural number n such that $n \leq x < n + 1$;
  return a+n;
End of algorithm

Definition 2.5.7. *We consider two compatible algorithms Alg_n and Alg_m. We denote by $Alg_n \sqcup Alg_m$ the following algorithm Alg:*

Algorithm $Alg(v_1, v_2)$
  If $(v_1, v_2) \in dom(Alg_n)$ then apply Alg_n;
  $o := Alg(v_1, v_2)$; endif
  If $(v_1, v_2) \in dom(Alg_m) \setminus dom(Alg_n)$ then apply
  Alg_m; $o := Alg_m(v_1, v_2)$; endif
  return o;
End of algorithm

We have the following properties of the operation $\sqcup$:

Proposition 2.5.1. *If* $Alg_n \sim_C Alg_m$ *then*
- $Alg_n \sqcup Alg_m \prec Alg_m \sqcup Alg_n \prec Alg_n \sqcup Alg_m$
- $Alg_n \prec Alg_n \sqcup Alg_m$
- *If* $Alg_n \prec Alg$ *and* $Alg_m \prec Alg$ *then* $Alg_n \sqcup Alg_m \prec Alg$

Proof. Immediate by the fact that $dom(Alg_n \sqcup Alg_m) = dom(Alg_n) \cup dom(Alg_n)$ and Definition 2.5.7. ■

In order to relieve some useful aspects of the computations in a semantic schema we consider the following example. We consider the labeled graph from Figure 2.10, which defines the schema $\mathcal{S} = (X, A_0, A, R)$ if we take $X = \{x_1, x_2, x_3\}$, $A_0 = \{a, b\}$, $A = A_0 \cup \{\theta(a, b)\}$, $R_0 = \{(x_1, a, x_2), (x_2, b, x_3)\}$ and $R = R_0 \cup \{(x_1, \theta(a, b), x_3)\}$.

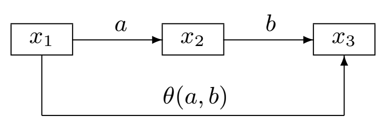

Fig. 2.10. A graph representing a semantic schema

We consider the interpretation $\mathcal{I} = (Ob, ob, \{Alg^1_u\}_{u \in A})$, where $Ob = \{1, 8, 2\}$, $ob(x_1) = 1$, $ob(x_2) = 8$, $ob(x_3) = 2$ and the following algorithms:

Algorithm $Alg^1_a(r_1 : integer, r_2 : integer)$
if $r_1 + r_2 \geq 10$ then return $o = (r_1, r_2)$;
End of algorithm

Algorithm $Alg^1_b(r_1 : integer, r_2 : integer)$
if $r_1 - r_2 > 5$ then return $o = (-r_1, -r_2)$;
End of algorithm

Algorithm $Alg^1_{\theta(a,b)}(o_1, o_2)$
if $o_1 = (x, y)$ and $o_2 = (p, q)$ then return
the integer value $x \times p + y \times q$;
End of algorithm

We consider also the interpretation $\mathcal{J} = (Ob, ob, \{Alg^2_u\}_{u \in A})$, where

Algorithm $Alg^2_a(r_1 : integer, r_2 : integer)$
if $r_1 + r_2 \geq 10$ then return $o = (r_1, r_2)$; otherwise
return $o = (2r_1, 3r_2)$;
End of algorithm

and take $Alg^2_b = Alg^1_b$, $Alg^2_{\theta(a,b)} = Alg^1_{\theta(a,b)}$.

We observe that $Val_{\mathcal{I}}(h(x_1, a, x_2))$ is not defined because $(ob(x_1), ob(x_2)) = (1, 8) \notin dom(Alg^1_a)$. We have also the following derivation in $\mathcal{S}$:

$$(x_1, \theta(a,b), x_3) \Rightarrow^* \sigma(h(x_1,a,x_2), h(x_2,b,x_3))$$

In other words we have $\sigma(h(x_1,a,x_2), h(x_2,b,x_3)) \in \mathcal{F}_{comp}(\mathcal{S})$. But we observe that $Val_{\mathcal{I}}(\sigma(h(x_1,a,x_2), h(x_2,b,x_3)))$ is not defined because the element $Val_{\mathcal{I}}(h(x,a,y))$ is not defined.

If we perform the same computations for $\mathcal{J}$ then we obtain:
$Val_{\mathcal{J}}(h(x_1,a,x_2)) = Alg_a^2(1,8) = (2,24)$
$Val_{\mathcal{J}}(h(x_2,b,x_3)) = Alg_b^2(8,2) = (-8,-2)$
$Val_{\mathcal{J}}(\sigma(h(x_1,a,x_2), h(x_2,b,x_3))) =$
$Alg^2_{\theta(a,b)}((2,24),(-8,-2)) = -64$

For the example presented before we observe that $Alg_a^1 \sim_C Alg_a^2$, $Alg_b^1 \sim_C Alg_b^2$ and $Alg^1_{\theta(a,b)} \sim_C Alg^2_{\theta(a,b)}$. In conclusion we relieved the following aspect: we defined two distinct interpretations $\mathcal{I} \in Int(\mathcal{S})$ and $\mathcal{J} \in Int(\mathcal{S})$ for the same semantic schema $\mathcal{S}$ such that the corresponding algorithms are compatible and there is $w \in \mathcal{F}_{comp}(\mathcal{S})$ such that $Val_{\mathcal{I}}(w)$ is not defined, but $Val_{\mathcal{J}}(w)$ has some value.

Definition 2.5.8. *The interpretations* $\mathcal{I} = (Ob_1, ob_1, \{Alg_u^1\}_{u\in A}) \in Int(\mathcal{S})$ *and* $\mathcal{J} = (Ob_2, ob_2, \{Alg_v^2\}_{v\in B}) \in Int(\mathcal{P})$ *are* **compatible** *if*

- *If* $x \in X^1 \cap X^2$ *then* $ob_1(x) = ob_2(x)$
- $Alg_u^1 \sim_C Alg_u^2$ *for every* $u \in A \cap B$

We denote this property by $\mathcal{I} \approx_C \mathcal{J}$.

Particularly this definition can be used for the same semantic schema, that is $\mathcal{S} = \mathcal{P}$. This is the case of the example presented before. Thus for our example we have $\mathcal{I} \approx_C \mathcal{J}$, but we observe that we have also $Alg_a^1 \prec Alg_a^2$, $Alg_b^1 \prec Alg_b^2$ and $Alg^1_{\theta(a,b)} \prec Alg^2_{\theta(a,b)}$. Thus the following definition is consistent.

Definition 2.5.9. *Let us consider the interpretations*

$$\mathcal{I} = (Ob_1, ob_1, \{Alg_u^1\}_{u\in A}) \in Int(\mathcal{S})$$

$$\mathcal{J} = (Ob_2, ob_2, \{Alg_v^2\}_{v\in B}) \in Int(\mathcal{P})$$

We write $\mathcal{I} \prec \mathcal{J}$ *if the the following conditions are satisfied:*

- $\mathcal{S} \sqsubseteq \mathcal{P}$
- ob_2 *is an extension of* ob_1
- $Alg_u^1 \prec Alg_u^2$ *for* $u \in A$

Particularly we can consider two interpretations for the same semantic schema. This is the case encountered in our example, where $\mathcal{I} \prec \mathcal{J}$ and $\mathcal{I} \approx_C \mathcal{J}$. The connection between these two concepts is stated in the following proposition:

Proposition 2.5.2. *If* $\mathcal{I} \prec \mathcal{J}$ *then* $Ob_1 \subseteq Ob_2$ *and* $\mathcal{I} \approx_C \mathcal{J}$.

Proof. The first part is obtained from the fact that $X^1 \subseteq X^2$, ob_2 is an extension of ob_1 therefore $ob_2(x) = ob_1(x)$ for $x \in X^1$. For the second part we observe that if $Alg^1_u \prec_C Alg^2_u$ then $Alg^1_u \sim_C Alg^2_u$. Now the proof is immediate by the fact that $X^1 \cap X^2 = X^1$, ob_2 is an extension of ob_1 and $A \cap B = A$. ■

Proposition 2.5.3. *If $\mathcal{I} \in Int(\mathcal{S})$, $\mathcal{J} \in Int(\mathcal{P})$ and $\mathcal{I} \prec \mathcal{J}$ then for every $w \in \mathcal{F}_{comp}(\mathcal{S})$ we have the following property: if $Val_{\mathcal{I}}(w)$ is defined then $Val_{\mathcal{J}}(w)$ is defined and $Val_{\mathcal{I}}(w) = Val_{\mathcal{J}}(w)$.*

Proof. We prove this property by induction on the length of $sort(w)$.

• If $length(sort(w)) = 1$ then $w = h(x, a, y)$ for some $(x, a, y) \in R_0$. But $\mathcal{S} \sqsubseteq \mathcal{P}$ therefore $R_0 \subseteq Q_0$. By definition of the mapping Val we have

$Val_{\mathcal{I}}(h(x, a, y)) = Alg^1_a(ob_1(x), ob_1(y))$ if $(ob_1(x), ob_1(y)) \in dom(Alg^1_a)$

$Val_{\mathcal{J}}(h(x, a, y)) = Alg^2_a(ob_2(x), ob_2(y))$ if $(ob_2(x), ob_2(y)) \in dom(Alg^2_a)$

But $x \in X^1$, $y \in X^1$, $X^1 \subseteq X^2$ and ob_2 extends ob_1. This means that $ob_1(x) = ob_2(x)$, $ob_1(y) = ob_2(y)$. We have also $Alg^1_a \prec Alg^2_a$ because $a \in A_0 = A_0 \cap B_0$. If $Val_{\mathcal{I}}(w)$ is defined then $(ob_1(x), ob_1(y)) \in dom(Alg^1_a)$ therefore $(ob_2(x), ob_2(y)) \in dom(Alg^2_a)$ because $(ob_1(x), ob_1(y)) = (ob_2(x), ob_2(y))$. It follows that in this case $Val_{\mathcal{J}}(w)$ is defined. Moreover, from $Alg^1_a \prec Alg^2_a$ we obtain $Alg^1_a(ob_1(x), ob_1(y) = Alg^2_a(ob_2(x), ob_2(y)))$. In other words, $Val_{\mathcal{I}}(w) = Val_{\mathcal{J}}(w)$ in this case.

• Suppose the property is true for every w such that $length(sort(w)) < n$ and take $w \in \mathcal{F}_{comp}(\mathcal{S})$ such that $length(sort(w)) = n$. Because $\mathcal{F}_{comp}(\mathcal{S}) \in Initial(\mathcal{H}_{\mathcal{S}})$ and $\mathcal{H}_{\mathcal{S}}$ is a Peano algebra, we have $w = \sigma(\alpha, \beta)$ for some $\alpha, \beta \in \mathcal{F}_{comp}(\mathcal{S})$, uniquely determined. If $sort(w) = \theta(u, v)$ then $sort(\alpha) = u$, $sort(\beta) = v$ and

$Val_{\mathcal{I}}(w) = Alg^1_{\theta(u,v)}(Val_{\mathcal{I}}(\alpha), Val_{\mathcal{I}}(\beta))$

if $(Val_{\mathcal{I}}(\alpha), Val_{\mathcal{I}}(\beta)) \in dom(Alg^1_{\theta(u,v)})$

$Val_{\mathcal{J}}(w) = Alg^2_{\theta(u,v)}(Val_{\mathcal{J}}(\alpha), Val_{\mathcal{J}}(\beta))$

if $(Val_{\mathcal{J}}(\alpha), Val_{\mathcal{J}}(\beta)) \in dom(Alg^2_{\theta(u,v)})$.

If $Val_{\mathcal{I}}(w)$ is defined then $(Val_{\mathcal{I}}(\alpha), Val_{\mathcal{I}}(\beta)) \in dom(Alg^1_{\theta(u,v)})$, therefore $Val_{\mathcal{I}}(\alpha)$ and $Val_{\mathcal{I}}(\beta))$ are defined. But $length(sort(\alpha)) < n$ and $length(sort(\beta)) < n$. Applying the inductive assumption we obtain that $Val_{\mathcal{J}}(\alpha)$ and $Val_{\mathcal{J}}(\beta))$ are defined and

$Val_{\mathcal{I}}(\alpha) = Val_{\mathcal{J}}(\alpha)$, $Val_{\mathcal{I}}(\beta) = Val_{\mathcal{J}}(\beta)$

Moreover, $Alg^1_{\theta(u,v)} \prec Alg^2_{\theta(u,v)}$ therefore $Val_{\mathcal{I}}(w) = Val_{\mathcal{J}}(w)$ and the proposition is proved. ■

2.5.4 An interpretation for $\mathcal{S} \vee \mathcal{P}$

In this section we treat the following problem:

Suppose that $\mathcal{I}$ is an interpretation for $\mathcal{S}$ and $\mathcal{J}$ is an interpretation for $\mathcal{P}$ such that they are compatible interpretations; we define and give a method *to obtain an interpretation* $\mathcal{I} \vee \mathcal{J}$ *of* $sup\{\mathcal{S}, \mathcal{P}\}$*, which is an upper bound for* $\{\mathcal{I}, \mathcal{J}\}$ *and for every interpretation* $\mathcal{K}$ *of* $\mathcal{S} \vee \mathcal{P}$ *such that* $\mathcal{I} \prec \mathcal{K}$ *and* $\mathcal{J} \prec \mathcal{K}$ *we have* $\mathcal{I} \vee \mathcal{J} \prec \mathcal{K}$.

Proposition 2.5.4. *We consider* $\mathcal{I} = (Ob_1, ob_1\{Alg^1_u\}_{u\in A}) \in Int(\mathcal{S})$ *and* $\mathcal{J} = (Ob_2, ob_2\{Alg^2_v\}_{v\in B}) \in Int(\mathcal{P})$ *such that* $\mathcal{I} \approx_C \mathcal{J}$.

The system $\mathcal{I} \vee \mathcal{J} = (Ob, ob, \{Alg_u\}_{u\in A\cup B})$*, where*

- $Ob = Ob_1 \cup Ob_2$
- $ob(x) = \begin{cases} ob_1(x) \text{ if } x \in X^1 \\ ob_2(x) \text{ if } x \in X^2 \end{cases}$
- $Alg_u = \begin{cases} Alg^1_u \sqcup Alg^2_u \text{ if } u \in A \cap B \\ Alg^1_u \text{ if } u \in A \setminus B \\ Alg^2_u \text{ if } u \in B \setminus A \end{cases}$

has the following properties:

- $\mathcal{I} \vee \mathcal{J} \in Int(\mathcal{S} \vee \mathcal{P})$
- $\mathcal{I} \prec \mathcal{I} \vee \mathcal{J}$; $\mathcal{J} \prec \mathcal{I} \vee \mathcal{J}$

Proof.

The mapping $ob : X^1 \cup X^2 \longrightarrow Ob$ is well defined because $ob_1(x) = ob_2(x)$ for $x \in X^1 \cap X^2$. Obviously ob is a bijective mapping. We have $dom(Alg^1_u \sqcup Alg^2_u) = dom(Alg^1_u) \cup dom(Alg^2_u)$, therefore

• If $u = a \in A_0 \cap B_0$ then $dom(Alg_u) = dom(Alg^1_u) \cup dom(Alg^2_u) \subseteq Ob \times Ob$. If $u \in A_0 \setminus B_0$ then $dom(Alg_u) = dom(Alg^1_u) \subseteq Ob_1 \times Ob_1 \subseteq Ob \times Ob$. Similar for $u \in B_0 \setminus A_0$ then $dom(Alg_u) = dom(Alg^2_u) \subseteq Ob_2 \times Ob_2 \subseteq Ob \times Ob$.

• If $u = \theta(u_1, v_1) \in A \cap B$ then $dom(Alg_{\theta(u_1,v_1)}) = dom(Alg^1_{\theta(u_1,v_1)}) \cup dom(Alg^2_{\theta(u_1,v_1)}) \subseteq (Y_{u_1}(\mathcal{I}) \times Y_{v_1}(\mathcal{I})) \cup (Y_{u_1}(\mathcal{J}) \times Y_{v_1}(\mathcal{J})) \subseteq Y_{u_1}(\mathcal{I} \vee J) \times Y_{v_1}(\mathcal{I} \vee J)$

• If $u = \theta(u_1, v_1) \in A \setminus B$ then $dom(Alg_{\theta(u_1,v_1)}) = dom(Alg^1_{\theta(u_1,v_1)}) \subseteq Y_{u_1}(\mathcal{I}) \times Y_{v_1}(\mathcal{I}) \subseteq Y_{u_1}(\mathcal{I} \vee J) \times Y_{v_1}(\mathcal{I} \vee J)$

Let us verify the condition $\mathcal{I} \prec \mathcal{I} \vee \mathcal{J}$. Because ob is an extension of ob_1, it remains to verify that $Alg^1_u \prec Alg_u$ for every $u \in A$. If $u \in A \setminus B$ then $Alg_u = Alg^1_u$ therefore the relation is true. If $u \in A \cap B$ then $Alg^1_u \prec Alg^1_u \sqcup Alg^2_u = Alg_u$.

Proposition 2.5.5. $\mathcal{I} \approx_C \mathcal{I} \vee \mathcal{J}$ *and* $\mathcal{J} \approx_C \mathcal{I} \vee \mathcal{J}$.

Proof. Immediate by Proposition 2.5.4 and Proposition 2.5.2. ■

Proposition 2.5.6. *Suppose* $\mathcal{I} \in Int(\mathcal{S})$, $\mathcal{J} \in Int(\mathcal{P})$ *and* $\mathcal{I} \approx_C \mathcal{J}$. *For every* $\mathcal{K} \in Int(\mathcal{S} \vee \mathcal{P})$ *such that* $\mathcal{I} \prec \mathcal{K}$ *and* $\mathcal{J} \prec \mathcal{K}$ *we have* $\mathcal{I} \vee \mathcal{J} \prec \mathcal{K}$.

Proof. We denote $\mathcal{K} = (Ob_3, ob_3, \{Alg^3_u\}_{u\in A\cup B})$. If we preserve the notations from Proposition 2.5.4 then we have $Alg_u \prec Alg^3_u$ for every $u \in A \cup B$. Really, we have:

• If $u \in A \cap B$ then $Alg_u = Alg^1_u \sqcup Alg^2_u$. But $\mathcal{I} \prec \mathcal{K}$ and $\mathcal{J} \prec \mathcal{K}$ therefore $Alg^1_u \prec Alg^3_u$ and $Alg^2_u \prec Alg^3_u$. By Proposition 2.5.1 we have $Alg^1_u \sqcup Alg^2_u \prec Alg^3_u$.

- If $u \in A \setminus B$ then $Alg_u = Alg_u^1$. But $Alg_u^1 \prec Alg_u^3$ because $\mathcal{I} \prec \mathcal{K}$.
- If $u \in B \setminus A$ then $Alg_u = Alg_u^2 \prec Alg_u^3$ because $\mathcal{J} \prec \mathcal{K}$.

The mapping ob_3 is an extension of ob. Take an arbitrary element $x \in X^1 \cup X^2$.If $x \in X^1$ then $ob(x) = ob_1(x)$. But $\mathcal{I} \prec \mathcal{K}$ therefore ob_3 extends ob_1. Thus if $x \in X^1$ then $ob(x) = ob_3(x)$. Similar for the case $x \in X^2$ we have $ob(x) = ob_3$ because $ob_3(x) = ob_2(x) = ob(x)$. ■

2.6 Applications

2.6.1 Application in logic programming with constraints

In this section we apply the concept of semantic schema to model a possible implementation in a client-server technology of the logic programming with constraints. Suppose we dispose of a computer network such that each work station WS_i, $(i = 1, \dots, n)$, contains some logic program P_i. The network server NS is endowed with a knowledge manager KM which is able to combine the information of the programs P_i. Suppose some query from a client is received by NS. The component KM analyses the query, consults the programs P_i and gives an answer. Several combinations of semantics can be used by NS to prepare this answer. In our application we shall consider the least fixed point semantics for Horn programs and the semantics of the distributed knowledge in the sense of Halpern et al. (1995).

In order to exemplify this case we consider the following problem:

- Consider the logic program P_1 of below:

$$\begin{cases} p(a) \leftarrow \\ p(f^2(x)) \leftarrow p(x) \\ q(g(x)) \leftarrow p(x) \end{cases}$$

where p, q are unary predicate symbols, f, g are unary function symbols, a is a constant and $f^k(x)$ denotes the term $f(f(\dots(f(x))\dots)$ that contains k symbols f. Denoting by $LFS(P_1)$ the least fixed point semantics of P_1, we have obviously

$$LFS(P_1) = \{p(f^{2n}(a)), q(g(f^{2n}(a)))\}_{n \geq 0}$$

- Extract from $LFS(P_1)$ the infinite set

$$C = \{p(f^{4k}(a))\}_{k \geq 0} \cup \{q(g(f^{4k}(a))), q(g(f^{3k}(a)))\}_{k \geq 0}$$

by introducing a cascade of filters such that each filter is a logic program.

- Specify the network architecture and the corresponding collaborations between the nodes of the network such that KM benefits of the semantics $LFS(P_1) \setminus C$.

In order to solve this problem we consider the following components of a semantic schema $\mathcal{S}$, which is represented in Figure 2.11:

- $X = \{x_1, x_2, x_3, x_4\}$
- $A_0 = \{m, c\}$
- $R = \{(x_1, m, x_2), (x_2, c, x_3), (x_3, c, x_4),$
  $(x_1, \theta(m, c), x_3), (x_1, \theta(\theta(m, c), c), x_4)\}$

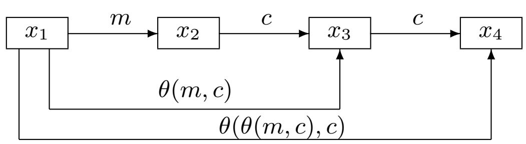

Fig. 2.11. A semantic schema

We introduce three programs which are used as filters. In the first step the set $F_1 = \{p(f^{4k}(a))\}_{k \geq 0}$ is eliminated from $LSF(P_1)$ and in the next steps the following sets are eliminated consecutively :

$$F_2 = \{q(g(f^{4k}(a)))\}_{k \geq 0}$$

$$F_3 = \{q(g(f^{3k}))\}_{k \geq 0}$$

The filters are defined as follows:

a) Consider the logic program P_2:

$$\begin{cases} p(a) \leftarrow \\ p(f^4(x)) \leftarrow p(x) \end{cases}$$

The fixed point semantics is $LFS(P_2) = F_1$.

b) Consider the logic program P_3:

$$\begin{cases} q(g(x)) \leftarrow p(x) \\ q(a) \leftarrow \end{cases}$$

If P and Q are two Horn programs we define a simplified form of the distributed knowledge for P and Q, denoted by $D(P, Q)$. More precisely, $p(c) \in D(P, Q)$ iff $p(c)$ is a ground atom that can be deduced from Q by using at least one element of $LFS(P)$. For example, if P is the program

$$\begin{cases} p(f(x)) \leftarrow p(x) \\ p(a) \leftarrow \end{cases}$$

and Q is the following program

$$\begin{cases} q(g(y)) \leftarrow p(y) \\ q(b) \leftarrow \end{cases}$$

then $LFS(P) = \{p(f^k(a))\}_{k\geq 0}$ and $D(P,Q) = \{q(g(f^k(a)))\}_{k\geq 0}$.
We observe that if $w \in D(P,Q)$ then neither P nor Q can prove w, but w can be proved by $P \cup Q$. For our application we have $D(P_2, P_3) = F_2$.

c) Consider the following logic program P_4:

$$\begin{cases} q(g(f^3(a)) \leftarrow q(a) \\ q(g(f^3(x)) \leftarrow q(g(x)) \end{cases}$$

Because $LFS(P_3) = \{q(a)\}$, we shall have $D(P_3, P_4) = F_3$.
We consider the following interpretation:

- $Ob = \{P_1, P_2, P_3, P_4\}$
- $ob(x_i) = P_i$; $i = 1, 2, 3, 4$
- $Y \subseteq 2^{HB}$, where HB is the Herbrand base (the set of all ground atoms over $\mathcal{S}_C = \{a\}, \mathcal{S}_F = \{f, g\}$ and $\mathcal{S}_P = \{p, q\}$)
- $J_\sigma(A, B) = A \setminus B$, the difference operation from the set theory
- The mapping J_h is defined by

Algorithm $Alg_m(o_1, o_2)$
  compute $X = LFS(o_1)$ and $Y = LFS(o_2)$; return $X \setminus Y$;
End of algorithm
Algorithm $Alg_c(o_1, o_2)$
  compute $X = D(o_1, o_2)$; return X;
End of algorithm
Algorithm $Alg_{\theta(m,c)}(o_1, o_2)$
  compute $X = o_1 \setminus o_2$; return X;
End of algorithm
Algorithm $Alg_{\theta(\theta(m,c),c)}(o_1, o_2)$
  compute $X = o_1 \setminus o_2$; return X;
End of algorithm

We obtain:

- $Val_{\mathcal{I}}(h(x_1, m, x_2)) = Alg_m(ob(x_1), ob(x_2)) = Alg_m(P_1, P_2) = LFS(P_1) \setminus LFS(P_2) = LFS(P_1) \setminus F_1$;
- $Val_{\mathcal{I}}(h(x_2, c, x_3)) = Alg_c(ob(x_2), ob(x_3)) = Alg_c(P_2, P_3) = D(P_2, P_3) = F_2$
- $Val_{\mathcal{I}}(h(x_3, c, x_4)) = Alg_c(ob(x_3), ob(x_4)) = Alg_c(P_3, P_4) = D(P_3, P_4) = F_3$

Obviously we have:

$$(x_1, \theta(\theta(m, c), c), x_4) \Rightarrow^* \sigma(\sigma(h(x_1, m, x_2), h(x_2, c, x_3)), h(x_3, c, x_4))$$

and therefore

$$sort(\sigma(\sigma(h(x_1, m, x_2), h(x_2, c, x_3)), h(x_3, c, x_4))) = \theta(\theta(m, c), c), x_4)$$

Similarly we have

$$(x_1, \theta(m, c), x_3) \Rightarrow^* \sigma(h(x_1, m, x_2), h(x_2, c, x_3))$$

therefore

$$sort(\sigma(h(x_1, m, x_2), h(x_2, c, x_3))) = \theta(m, c)$$

In order to use a shorter notation we denote $\alpha = h(x_1, m, x_2)$, $\beta = h(x_2, c, x_3)$ and $\gamma = h(x_3, c, x_4)$.

We obtain the following sequence of computations:

- $sort(\sigma(\sigma(\alpha, \beta), \gamma)) = \theta(\theta(m, c), c)$ therefore $Val_{\mathcal{I}}(\sigma(\sigma(\alpha, \beta), \gamma)) = Alg_{\theta(\theta(m,c),c)}(Val_{\mathcal{I}}(\sigma(\alpha, \beta)), Val_{\mathcal{I}}(\gamma))$
- $sort(\sigma(\alpha, \beta)) = \theta(m, c)$, therefore $Val_{\mathcal{I}}(\sigma(\alpha, \beta)) = Alg_{\theta(m,c)}(Val_{\mathcal{I}}(\alpha), Val_{\mathcal{I}}(\beta))$
- $Val_{\mathcal{I}}(\alpha) = Alg_m(ob(x_1, ob(x_2))) = D(P_1, P_2) = LFS(P_1) \setminus F_1$, $Val_{\mathcal{I}}(\beta)) = Alg_c(ob(x_2), ob(x_3))) = D(P_2, P_3) = F_2$ therefore

$$Val_{\mathcal{I}}(\sigma(\alpha, \beta)) = Alg_{\theta(m,c)}(LFS(P_1) \setminus F_1, F_2) = (LFS(P_1) \setminus F_1) \setminus F_2$$

- $Val_{\mathcal{I}}(\sigma(\sigma(\alpha, \beta), \gamma)) = Alg_{\theta(\theta(m,c),c)}((LFS(P_1) \setminus F_1) \setminus F_2, F_3) = ((LFS(P_1) \setminus F_1) \setminus F_2) \setminus F_3$

Applying Definition 2.5.4 we have

$$Out_{\mathcal{I}}(x_1, x_4) = Val_{\mathcal{I}}(x_1, \theta(\theta(m, c), c), x_4)$$

therefore based on the relation $(A \setminus B) \setminus C = A \setminus (B \cup C)$ from the set theory we have $Out_{\mathcal{I}}(x_1, x_4) = LFS(P_1) \setminus (F_1 \cup F_2 \cup F_3)$.

2.6.2 Distributed knowledge representation and processing

In this section we apply the existence of $sup\{\mathcal{S}_1, \mathcal{S}_2\}$ to model the representation of distributed knowledge in semantics of communication.

Consider the θ-schemas $\mathcal{S}_1 = (X_1, A_0, A, R)$, $\mathcal{S}_2 = (X_2, B_0, B, Q)$ such that R_0 and Q_0 are represented in Figure 2.12 and Figure 1.2 respectively.

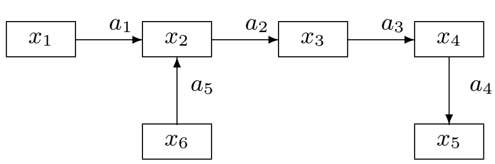

Fig. 2.12. The set R_0

We take the following sets R and Q:

$$R = R_0 \cup \{(x_2, \theta(a_2, a_3), x_4), (x_1, \theta(a_1, \theta(a_2, a_3)), x_4)\}$$
$$Q = Q_0 \cup \{(x_7, \theta(a_1, a_6), x_8), (x_7, \theta(\theta(a_1, a_6), a_3), x_9)\}$$

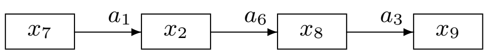

Fig. 2.13. The set Q_0

It follows that

$$A_0 = \{a_1, a_2, a_3, a_4, a_5\}$$
$$B_0 = \{a_1, a_6, a_3\}$$
$$A = A_0 \cup \{\theta(a_2, a_3), \theta(a_1, \theta(a_2, a_3))\}$$
$$B = B_0 \cup \{\theta(a_1, a_6), \theta(\theta(a_1, a_6), a_3)\}$$

Applying Proposition 2.3.4 we obtain:

$$Z_0 = R_0 \cup Q_0$$
$$Z_1 = Z_0 \cup \{(x_2, \theta(a_2, a_3), x_4), ((x_7, \theta(a_1, a_6), x_8))\}$$
$$Z_2 = Z_1 \cup \{(x_1, \theta(a_1, \theta(a_2, a_3)), x_4), (x_7, \theta(\theta(a_1, a_6), a_3), x_9), (x_7, \theta(a_1, \theta(a_2, a_3)), x_4)\}$$
$$Z_3 = Z_2$$

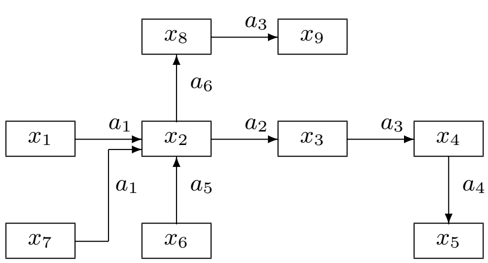

Fig. 2.14. The set Z_0

In what follows we shall consider several sentential forms. Such a structure is a sentence containing two variables. If we substitute each variable by an object then a sentential form becomes a sentence in a natural language. We shall consider the following sentential forms:

$p_1(x, y)$=”x is the mother of y”
$p_2(x, y)$=”x is a y”
$p_3(x, y)$=”x is the mother of a z”
$q_1(x, y)$=”x is the brother of y”
$q_2(x, y)$=”x is the uncle of a y”
$q_3(x, y)$=”every x is a good y”
$q_4(x, y)$=”x is a friend of y”

For $\mathcal{S}_1$ we consider the interpretation $\mathcal{I}_1 = (Ob_1, ob_1, \{Alg^1_u\}_{u\in A})$ where

- $Ob_1 = \{Peter, Mary, Helen, teacher, driver, John\}$
- $ob_1(x_1) = Peter, ob_1(x_2) = Mary, ob_1(x_3) = Helen, ob_1(x_4) = teacher, ob_1(x_5) = driver, ob_1(x_6) = John$
- The set of algorithms are the following:

Algorithm $Alg^1_{a_1}(o_1, o_2)$
  return $q_1(o_1, o_2)$;
End of algorithm
Algorithm $Alg^1_{a_2}(o_1, o_2)$
  return $p_1(o_1, o_2)$;
End of algorithm
Algorithm $Alg^1_{a_3}(o_1, o_2)$
  if $o_1 \in Ob_1$ then return $p_2(o_1, o_2)$;
End of algorithm
Algorithm $Alg^1_{a_4}(o_1, o_2)$
  return $q_3(o_1, o_2)$;
End of algorithm
Algorithm $Alg^1_{a_5}(o_1, o_2)$
  return $q_4(o_1, o_2)$;
End of algorithm
Algorithm $Alg^1_{\theta(a_2,a_3)}(o_1, o_2)$
  take t_1, t_2, t_3, t_4 such that $o_1 = p_1(t_1, t_2)$, $o_2 = p_2(t_3, t_4)$; if $t_2 = t_3$ then return $p_3(t_1, t_4)$;
End of algorithm
Algorithm $Alg^1_{\theta(a_1,\theta(a_2,a_3))}(o_1, o_2)$
  take t_1, t_2, t_3, t_4 such that $o_1 = q_1(t_1, t_2)$, $o_2 = p_3(t_3, t_4)$; if $t_2 = t_3$ then return $q_2(t_1, t_4)$;
End of algorithm

In order to obtain some interpretation for $\mathcal{S}_2$ we take the following sentential forms:

$r_2(x, y)$=”x learns y”
$r_3(x, y)$=” A sister of x learns y”
$s_2(x, y)$=”x is a kind of y”

Now we can define the components of the following interpretation

$$\mathcal{I}_2 = (Ob_2, ob_2, \{Alg^2_u\}_{u\in B})$$

for $\mathcal{S}_2$:

- $OB_2 = \{George, Mary, Java\_Programming, Object\_Oriented\_Programming\}$
- $ob_2(x_7) = George, ob_2(x_2) = Mary, ob_2(x_8) = Java\_Programming, ob_2(x_9) = Object\_Oriented\_Programming$

- The following algorithm are used:
  $Alg^2_{a_1} = Alg^1_{a_1}$
  Algorithm $Alg^2_{a_6}(o_1, o_2)$
  return $r_2(o_1, o_2)$;
  End of algorithm
  Algorithm $Alg^2_{a_3}(o_1, o_2)$
  if $o_1 \in Ob_2 \setminus \{Mary\}$ then return $s_2(o_1, o_2)$;
  End of algorithm
  Algorithm $Alg^2_{\theta(a_1,a_6)}(o_1, o_2)$
  take t_1, t_2, t_3, t_4 such that $o_1 = q_1(t_1, t_2)$, $o_2 = r_2(t_3, t_4)$; if $t_2 = t_3$ then return $r_3(t_1, t_4)$;
  End of algorithm
  Algorithm $Alg^2_{\theta(\theta(a_1,a_6),a_3)}(o_1, o_2)$
  take t_1, t_2, t_3, t_4 such that $o_1 = r_3(t_1, t_2)$, $o_2 = s_2(t_3, t_4)$; if $t_2 = t_3$ then return $r_3(t_1, t_4)$;
  End of algorithm

We observe that $\mathcal{I}_1 \approx_C \mathcal{I}_2$ and therefore we can use the interpretation $\mathcal{I} = \mathcal{I}_1 \vee \mathcal{I}_2 = (Ob, ob, \{Alg_u\}_{u \in A \cup B})$.

A simple computation shows that

$$(x_7, \theta(a_1, \theta(a_2, a_3)), x_4) \Longrightarrow^* \sigma(h(x_7, a_1, x_2), \sigma(h(x_2, a_2, x_3), h(x_3, a_3, x_4)))$$

If we denote $\alpha = h(x_7, a_1, x_2)$ and $\beta = \sigma(h(x_2, a_2, x_3), h(x_3, a_3, x_4))$ then we obtain:

- $Val_{\mathcal{I}}(\alpha) = Alg_{a_1}(ob(x_7), ob(x_2)) = q_1(George, Mary)$
- We have $(x_2, \theta(a_2, a_3), x_4) \Rightarrow^* \beta$ therefore $sort(\beta) = \theta(a_2, a_3)$. It follows that

  $$Val_{\mathcal{I}}(\beta) = Alg_{\theta(a_2,a_3)}(Val_{\mathcal{I}}(h(x_2, a_2, x_3)), Val_{\mathcal{I}}(h(x_3, a_3, x_4)))$$

  But we observe that

  $$Val_{\mathcal{I}}(h(x_2, a_2, x_3)) = Alg_{a_2}(ob(x_2), ob(x_3)) = Alg_{a_2}(Mary, Helen) = p_1(Mary, Helen)$$

  and

  $$Val_{\mathcal{I}}(h(x_3, a_3, x_4)) = Alg_{a_3}(ob(x_3), ob(x_4)) = Alg_{a_3}(Helen, teacher) = p_2(Helen, teacher)$$

  It follows that

  $$Val_{\mathcal{I}}(\beta) = Alg_{\theta(a_2,a_3)}(p_1(Mary, Helen), p_2(Helen, teacher)) = p_3(Mary, teacher)$$

- In $\mathcal{S}_1 \vee \mathcal{S}_2$ we have

$$(x_7, \theta(a_1, \theta(a_2, a_3)), x_4) \Rightarrow^* \sigma(\alpha, \beta)$$

therefore $sort(\sigma(\alpha,\beta)) = \theta(a_1, \theta(a_2, a_3))$. Let us compute the value $Val_{\mathcal{I}}(\sigma(\alpha,\beta))$. We obtain

$$Val_{\mathcal{I}}(\sigma(\alpha,\beta)) = Alg_{\theta(a_1,\theta(a_2,a_3))}(Val_{\mathcal{I}}(\alpha), Val_{\mathcal{I}}(\beta)) =$$
$$Alg_{\theta(a_1,\theta(a_2,a_3))}(q_1(George, Mary), p_3(Mary, teacher)) =$$
$$q_2(George, teacher)$$

Finally we obtained

$$Val_{\mathcal{I}}(\sigma(\alpha,\beta)) = \text{"}\textit{George is the uncle of a teacher}\text{"}$$

We observe that neither $\mathcal{S}_1$ nor $\mathcal{S}_2$ can deduce this element, but this knowledge is deduced by $\mathcal{S}_1 \vee \mathcal{S}_2$. Thus $Val_{\mathcal{I}}(\sigma(\alpha,\beta))$ is a distributed knowledge.

2.6.3 Reasoning by analogy

In this section we deal with the case of the reasoning by analogy. In order to give an example we consider the schemas $\mathcal{S}_1 = (X, A_0, A, R)$ and $\mathcal{S}_2 = (Y, A_0, B, Q)$, where

- R_0 and Q_0 are defined in Figure 2.15 and Figure 2.16 respectively
- $A_0 = \{a_1, a_2, a_3\}$, $A = A_0 \cup \{\theta(a_2, a_3), \theta(a_1, \theta(a_2, a_3))\}$
- $R = R_0 \cup \{(x_2, \theta(a_2, a_3), x_4), (x_1, \theta(a_2, \theta(a_2, a_3)), x_4)\}$
- $B = A_0$ and $Q = Q_0$

Obviously the schema $\mathcal{S}_2$ can be used only to retrieve the initial information of a knowledge piece. This is due to the fact that $Q \setminus Q_0 = \emptyset$.

The components of $\mathcal{S}_1 \vee \mathcal{S}_2$ are represented in Figure 2.17. From Proposition 2.3.4 for our case we have

$$\mathcal{S}_1 \vee \mathcal{S}_2 = (X, A_0, R_0 \cup Q_0, Z)$$

where

$$Z = R \cup Q \cup \{(y_2, \theta(a_2, a_3), y_4), (y_1, \theta(a_2, \theta(a_2, a_3)), y_4)\}$$

We observe a transfer of meta-knowledge from $\mathcal{S}_1$ to $\mathcal{S}_2$ and the reasoning given by the element $(y_1, \theta(a_2, \theta(a_2, a_3)), y_4)$ is obtained by analogy from that of $(x_1, \theta(a_2, \theta(a_2, a_3)), x_4)$.

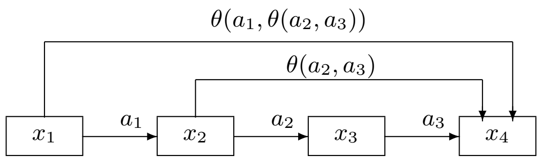

Fig. 2.15. Schema $\mathcal{S}_1$

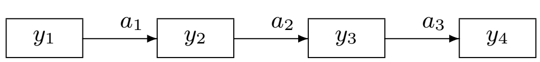

Fig. 2.16. Schema $\mathcal{S}_2$

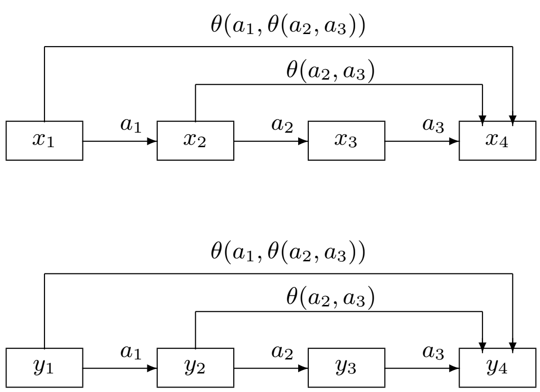

Fig. 2.17. Schema $\mathcal{S}_1 \vee \mathcal{S}_2$

2.6.4 A distributed system with a three-level structure

Another application of semantic schemas is given in this section, where we define a distributed knowledge system organized on three levels. The structure is the following:

• On the first level we find the *observers* or the *agents* of the system which send phrases in a natural language to the second level of the system.

• The second level includes several *primary knowledge managers (PKM)* of the system. Every *PKM* receives the phrases from some observers, has an own semantic schema and identifies a useful part of it.

• On the third level we find the *general knowledge manager (GKM)* of the system. It processes the structures of the second level of the system using the supremum of these structures and obtains its corresponding interpretation. Based on the theoretical results presented in the previous sections, the GKM is able to perform a distributed computing.

Definition 2.6.1. *A* **distributed system based on semantic schemas** *is a tuple* $DS = (L_1, L_2, L_3, SDB, G)$ *where:*

• $L_1 = \{Obs_1, \ldots, Obs_k\}$*, where* $Obs_i = \{O^i_1, \ldots, O^i_{s_i}\}$ *and* $O^i_1, \ldots, O^i_{s_i}$ *are named* **agents***;* L_1 *defines the first level of* DS *and we suppose that* $Obs_i \cap Obs_j = \emptyset$ *for* $i \neq j$*.*

• $L_2 = \{PKM_1, \ldots, PKM_k\}$ *and its elements are named* **primary knowledge managers***; the components of* L_2 *define the second level of* DS*;*

• $L_3 = \{GKM\}$*,* GKM *is named the* **general knowledge manager** *of* DS*; this manager defines the third level of* DS*.*

• *The component* SDB *is a data base, which contains a correspondence between the name of a binary relation and its element from* $\bigcup_{i=1}^{k} A^i_0$ *(specified in the next definition). This is shared by the knowledge managers of the system.*

• G *is a grammar for natural languages.*

The structure is represented in Figure 2.18.

In what follows we consider that the universe of an observer consists of some objects and the relations between them. We suppose these relations are binary ones. The communications between agents and *PKM*s are guided by some grammar G for natural language. We consider that all *PKM*s share the same **grammar**.

Definition 2.6.2. *A* **primary knowledge manager** *is a system* $PKM_i = (\mathcal{S}_i, KB_i)$ *where:*

• $\mathcal{S}_i = (X^i, A^i_0, A^i, R^i)$ *is the general semantic schema of* PKM_i

• $KB_i = (\{Alg_u\}_{u \in A^i}, \mathcal{USS}_i, \mathcal{I}_i)$*, where* $\mathcal{USS}_i \sqsubseteq \mathcal{S}_i$ *is a subschema of* $\mathcal{S}_i$ *and* $\mathcal{I}_i$ *is an interpretation for* $\mathcal{USS}_i$*, which uses* $\{Alg_u\}_{u \in A^i}$*.*

The tasks of PKM_i are the following:

• Receives the phrases from its observers, analyses them and obtains the *useful semantic schema* USS_i.

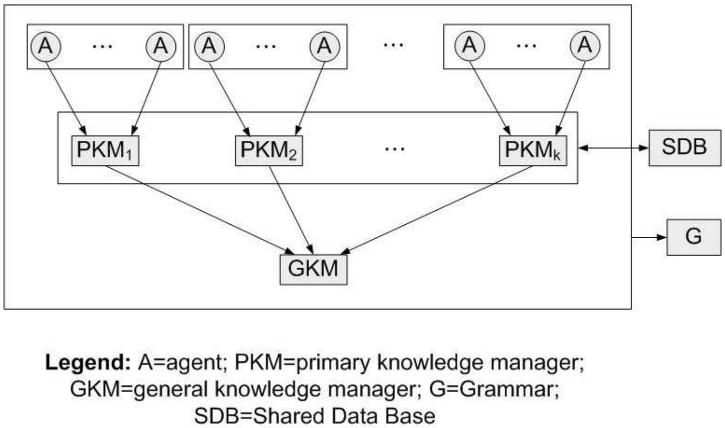

Fig. 2.18. The structure of DS on three levels

- Obtains an interpretation $\mathcal{I}_i$ for USS_i.

Definition 2.6.3. *The* **general knowledge manager (GKM)** *is a tuple* $GKM = (SupSi, \mathcal{I}, Displ)$*, where* $SupSi = sup\{\mathcal{USS}_1, \ldots, \mathcal{USS}_k\} = (X, A_0, A, R)$ *is the supremum of the useful semantic schemas from the second level,* $\mathcal{I}$ *is an interpretation of* $SupSi$ *and for every pair* $(x, y) \in X \times X$ *such that there is a path in* $SupSi$ *from* x *to* y*,* $Displ(x, y)$ *is the action performed to display the elements of* $Out_{\mathcal{I}}(x, y)$*.*

In order to exemplify this situation we consider the case $k = 2$, $s_1 = s_2 = 1$, $\mathcal{S}_1$ and $\mathcal{S}_2$ represented in Figure 2.19. We have:

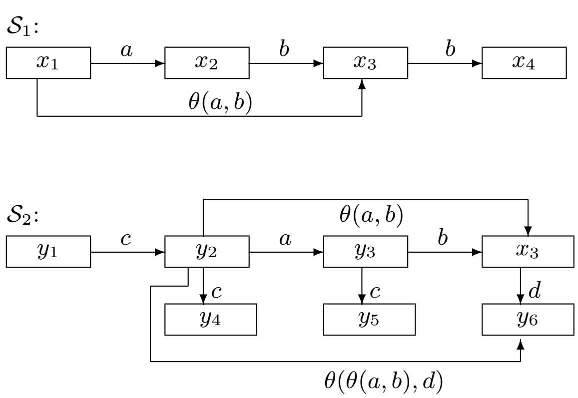

Fig. 2.19. The schemas $\mathcal{S}_1$ and $\mathcal{S}_2$

• $\mathcal{S}_1 = (X^1, A_0^1, A^1, R^1)$, where:
$X^1 = \{x_1, x_2, x_3, x_4\}$, $A_0^1 = \{a, b\}$, $A^1 = A_0^1 \cup \{\theta(a, b)\}$,
$R^1 = \{(x_1, a, x_2), (x_2, b, x_3), (x_3, b, x_4), (x_1, \theta(a, b), x_3)\}$

• $\mathcal{S}_2 = (X^2, A_0^2, A^2, R^2)$, where:
$X^2 = \{x_3, y_1, \ldots, y_6\}$, $A_0^2 = \{a, b, c, d\}$, $A^2 = A_0^2 \cup$ $\{\theta(a, b), \theta(\theta(a, b), d)\}$, $R^2 = \{(y_1, c, y_2), (y_2, a, y_3),$ $(y_3, b, x_3), (y_2, c, y_4), (y_3, c, y_5), (x_3, d, y_6),$ $(y_2, \theta(a, b), x_3), (y_2, \theta(\theta(a, b), d), y_6)\}$

We suppose that PKM_1 receives from O_1^1 some phrases from which it deduces that Ob_1 and ob_1 are the following:

- $Ob_1 = \{1, (2, 2), (2, 1/2)\}$
- $ob_1(x_1) = 1$, $ob_1(x_2) = (2, 2)$, $ob_1(x_3) = (2, 1/2)$

For PKM_2 we suppose that

- $Ob_2 = \{(2, 1/2), (2, 1), (2, 2), 4, (5, 3)\}$
- $ob_2(x_3) = (2, 1/2)$, $ob_2(y_6) = (2, 2)$, $ob_2(y_2) = 4$, $ob_2(y_3) = (5, 3)$

We suppose that $Alg_a^1, Alg_a^2, Alg_b^1, Alg_b^2$ are the algorithms defined as follows:

Algorithm $Alg_a^1(r : real, (x, y) : (real, real))$
$o = in(C)$, where C is the circle of radius r and center (x, y); **return** o;
End of algorithm

Algorithm $Alg_b^1((r_1, r_2) : (real, real), (x, y) : (real, real))$
$o = in(E)$, where E is the ellipse with horizontal radius r_1, vertical radius r_2 and center (x, y); **return** o;
End of algorithm

$Alg_a^2 = Alg_a^1$
$Alg_a^2 = Alg_a^1$

We take also the following algorithms:

Algorithm $Alg_d^2((r_1, r_2) : (real, real), (x, y) : (real, real))$
$o = in(E)$, where E is the ellipse with horizontal radius r_1, vertical radius $r_2 + 1/2$ and center (x,y); **return** o;
End of algorithm

Algorithm $Alg_{\theta(\theta(a,b),d)}^2(o_1, o_2)$
$F = int(o_2) \setminus int(o_1)$; **return** F;
End of algorithm

We suppose that PKM_1 and PKM_2 obtain the useful semantic schemas USS_1 and respectively USS_2, drawn in Figure 2.20. The schema $SupSi = sup\{USS_1, USS_2\}$ is represented in Figure 2.21.

For our purpose we suppose that $Displ(x, y)$ draws the objects from $Out_{\mathcal{I}}(x, y)$. Applying Definition 2.4.1 and the relation (2.5.4) we obtain for example:

$$Out_{\mathcal{I}}(x_1, y_6) = \{Val_{\mathcal{I}}(w)\}$$

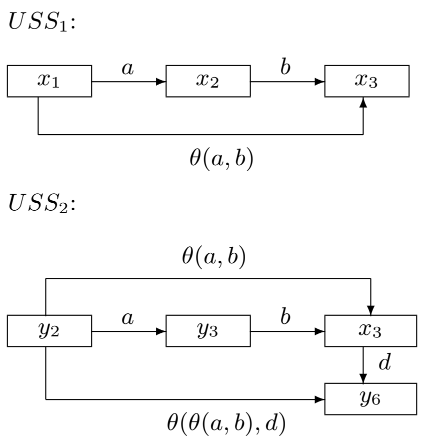

Fig. 2.20. The schemas USS_1 and USS_2

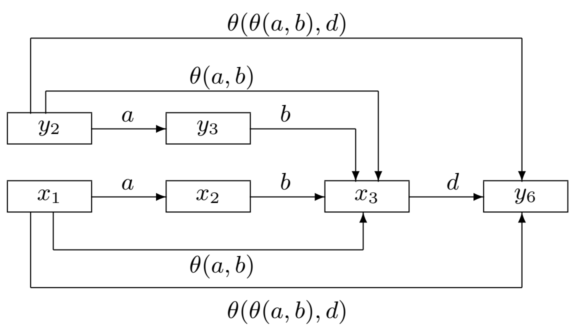

Fig. 2.21. $SupSi = sup\{USS_1, USS_2\}$

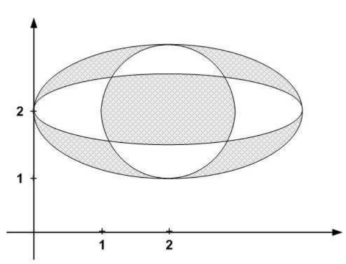

Fig. 2.22. The effect of $Displ(x_1, y_6)$

where $w = \sigma(\sigma(h(x_1, a, x_2), h(x_2, b, x_3)), h(x_3, d, y_6))$. As a consequence, $Displ(x_1, y_6)$ will draw the object represented in Figure 2.22.

References

[1] Boicescu V., Filipoiu A., Georgescu G. and Rudeanu S. (1991): "Lukasiewicz-Moisil Algebra". *Annals of Discrete Mathematics* **49** North-Holland

Chein M. (1996) (Ed): *Revue d'Intelligence artificielle*, Special Issue on Conceptual Graphs, vol. 10, no. 1

Chein M. and Mugnier M.L. (2000): Graph Homomorphism: a tool for managing labeled graphs bases. *Int. Conf. on Ordinal and Symbolic Data Analysis (OSDA 2000)* (Brussels, Belgium, July 2000)

Dastani, M., Hulstijn, J., Dignum, F. Meyer, J-J. Ch. (2004a): Issues in Multiagent System Development. N. R. Jennings, C. Sierra, L. Sonenberg, M. Tambe (eds.) Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS'04), ACM, p. 922-929

Dastani M., L. van der Torre (2004b): Games for Cognitive Agents. Alferes J.J., Leite J. (eds.) Proceedings of the 9th European Conference on Logics in Artificial Intelligence (JELIA'04), Springer LNAI 3229, p. 5-17

Eklund P.W., Ellis D. and Mann G. (1996) (Eds.): Conceptual Structures: Knowledge Representation as Interlingua. *Lecture Notes in AI* **1115** Springer-Verlag, Berlin

Ellis G., Levinson R.A., Rich W. and Sowa J.F. (Eds.) (1995): Conceptual Structures: Applications, Implementation, and Theory, *Lecture Notes in AI* **954** Springer-Verlag, Berlin

Ganter B., Mineau G.W. (Eds.) (2000): Conceptual Structures: Logical, Linguistic, and Computational Issues. *Lecture Notes in AI* **1867** Springer-Verlag, Berlin

R.Fagin, J.Y.Halpern, Y.Moses, M.Y.Vardi (1995): Reasoning about Knowledge, MIT Press, 1995

Hindriks K.V., F.S. de Boer, van der Hoek, Meyer J.J.Ch. (1999): Semantics of Communicating Agents Based on Deduction and Abduction. Proceedings of the IJCAI'99 Workshop on ACL, 1999; also in F.Dignum and M.Greaves (Eds), Issue in Agent Communication, Springer Verlag, p.63-79, 2000

Hulstijn, J., Dignum, F. Dastani, M. (2004): Coherence in Multi-agent Interaction. Proceedings of the Agent Communication Languages (AC), New York

Kaufmann, M., Wagner, D (Eds): Drawing Graphs : Methods and Models. Lecture Notes in Computer Science 2025, Springer

L. van der Torre, Hulstijn J., Dastani M., Broersen J. (2004) Specifying Multiagent Organizations. Proceedings of the Seventh Workshop on Deontic Logic in Computer Science (Deon'2004), Madeira, Springer, LNAI 3065, 243-257

Lukose, D., Delugach H., Keeler, M., Searle L., Sowa J.F. (Eds.) (1997): Conceptual Structures: Fulfilling Peirce's Dream. *Lecture Notes in AI* **1257** Springer-Verlag, Berlin

Z. Manna (1974): Mathematical Theory of Computation, McGraw-Hill Book Company, 1974

Mineau, G. W., Moulin, B., Sowa J.F. (Eds.) (1993): Conceptual Graphs for Knowledge Representation. *Lecture Notes in AI* **699** Springer-Verlag, Berlin
Mugnier, M.L., Chein M. (Eds.) (1998): Conceptual Structures: Theory, Tools, and Applications *Lecture Notes in AI* **1453** Springer-Verlag, Berlin
Mugnier, M.L (2000): Knowledge Representation and Reasonings Based on Graph Homomorphism. Research Report LIRMM 00098, August 2000; a preliminary version in Ganter and Mineau (2000), p.172-192
Nishizeki,T., Rahman, S.(2004): Planar Graph Drawing. Lecture Notes Series on Computing, World Scientific Publishing Company
Priss, U., Corbett, D., Angelova, G. (Eds.) (2002): Conceptual Structures: Integration and Interfaces, 10th Int. Conf. on Conceptual Structures, ICCS 2002 Borovets, Bulgaria, July 15–19
Rudeanu S. (2001): Lattice Functions and Equations, Discrete Mathematics and Computer Science, Springer
Sowa, J.F. (1984): Conceptual structures- Information Processing in Mind and Machine, Addison-Wesley
Sowa, J.F. (1992): *Knowledge-Based Systems*, "Special Issue: Conceptual Graphs", 5, (3), September
Sugiyama, K. (2002): Graph Drawing and Applications for Software and Knowledge Engineers. World Scientific Publishing Company
Tepfenhart, W.M., Dick, J.P., Sowa, J.F. (Eds.) (1994): Conceptual Structures: Current Practice, *Lecture Notes in AI* **835** Springer-Verlag, Berlin
Tepfenhart, W., Cyre, W. (Eds.) (1999): Conceptual Structures: Standards and Practices, *Lecture Notes in AI* **1640** Springer-Verlag, Berlin
Tollis, I. G., Giuseppe Di Battista, Eades P. Tamassia, R. (1998): Graph Drawing: Algorithms for the Visualization of Graphs, Prentice Hall
Ţăndăreanu, N. (2000a): Proving the Existence of labeled Stratified Graphs, *Annals of the University of Craiova*, Vol. XXVII, 81–92; on line version at http://inf.ucv.ro/~ntand/en/publications.html
Ţăndăreanu, N., (2000b): Knowledge Bases with Output, *Knowledge and Information Systems* 2(2000) 4, 438-460.
Ţăndăreanu, N. (2001): Intuitive Aspects of the Semantic Computations in KBO, Research Notes in Artificial Intelligence and Digital Communications 101, 1^{th} National Conference on Artificial Intelligence and Digital Communications, Craiova, June 2001, 1-8.
Ţăndăreanu, N. (2002): Using Java-Prolog Connection to implement automated reasoning, Research Notes in Artificial Intelligence and Digital Communications, Vol.102, 2^{nd} National Conference on Artificial Intelligence and Digital Communications, Craiova, June, p.9-18
Ţăndăreanu, N., Ghindeanu, M. (2003a): Towards a Mathematical Modelling of Conditional Knowledge, Research Notes in Artificial Intelligence and Digital Communications, Vol.103, 3^{nd} Romanian Conference on Artificial Intelligence and Digital Communications, Craiova, June, p.5-15
Ţăndăreanu, N., Ghindeanu, M. (2003b): Image Synthesis from Natural Language Description, Research Notes in Artificial Intelligence and Digital Communications, Vol.103, 3^{nd} Romanian Conference on Artificial Intelligence and Digital Communications, Craiova, June, p.82-96
Ţăndăreanu, N. (2003c): Collaborations between distinguished representatives for labeled stratified graphs, Annals of the University of Craiova, Mathematics and Computer Science Series, Vol. 30(2), p.184-192
Ţăndăreanu, N. (2004a): JaSSNet: a Java Shell for Semantic Networks via Semantic Schemas, Research Notes in Artificial Intelligence and Digital Communications,

Vol.104, 4^{th} International Conference on Artificial Intelligence and Digital Communications, Craiova, June, p.16-21

Ţăndăreanu, N. (2004b): Distinguished Representatives for Equivalent labeled Stratified Graphs and Applications, Discrete Applied Mathematics, Vol. 144/1-2, p.183-208

Ţăndăreanu, N. (2004c): Knowledge Representation by Labeled Stratified Graphs, The 8th World Multi-Conference on Systemics, Cybernetics and Informatics, July 18-21, Orlando, Florida, Vol. V: Computer Science and Engineering, p.345-350

Ţăndăreanu, N. (2004d): Semantic Schemas and Applications in Logical representation of Knowledge, Proceedings of the 10th International Conference on Cybernetics and Information Technologies, Systems and Applications (CITSA2004), July 21-25, Orlando, Florida, Vol. III p.82-87

Ţăndăreanu, N., Ghindeanu, M. (2004e) - Hierarchical Reasoning Based on Stratified Graphs. Application in Image Synthesis., 15th International Workshop on Database and Expert Systems Applications, Proceedings of DEXA2004, Zaragoza, IEEE Computer Society, Los Alamitos California, p.498-502

Ţăndăreanu, N. (2004d): Semantic Schemas extend Semantic Networks, Research Notes in Artificial Intelligence and Digital Communications, June, 104, 4^{th} International Conference on Artificial Intelligence and Digital Communications, Craiova, Romania, p.8-15, 2004.

Ţăndăreanu, N. (2005a): Transfer of knowledge via semantic schemas, July 10-13, Vol. IV, p.70-75, 2005

Ţăndăreanu, N., Ghindeanu, M. (2005b): 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

Ţăndăreanu N., Ghindeanu M. (2006)- Properties of derivations in a Semantic Schema, Annals of University of Craiova, Math. Comp. Sci. Ser. (to appear)