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

Labelled Stratified Graphs and their Applications in Knowledge Representation

Technical report

2005

Contents

Chapter 1

Introduction to labelled stratified graphs

1.1 Overview

A great number of research works and practical implementations have confirmed the interest of mathematicians and computer scientists in developing and applying the methods of graph theory. The subject presented in this chapter can be placed in a common area of the applied mathematics and computer science. The motivation is based on the following facts. First, we incorporate the concept of *labelled graph* into an algebraic environment given by a tuple of components, which are obtained applying several concepts of *universal algebra*. The result of this approach is materialized in a proper point of view concerning the use of the universal algebra in graph theory leading to a simple and concise theory with applications in several domains. This explains the connection with applied mathematics. In what concerns the computer science, there are two fields by which our research is connected. One of them is the *automatic graph drawing*, which is relatively a new field in computer science. A short description of the connection with the graph drawing field is given in the final of the next chapter, where several intentions concerning the continuation of this work are presented. Another field is that of *knowledge representation by graph methods*. The concepts of our approach, incorporated in that of *labelled stratified graph* (*LSG*), can be used successfully in knowledge representation.

The first section of this chapter establishes all prerequisites necessary to the study of labelled stratified graphs. We recall some elementary concepts and results of universal algebra, we introduce the concept of labelled graph in a distinctive manner and we establish some connections which allow us to introduce the concept of labelled stratified graph in the next section. Finally we prove the existence of this structure.

1.2 Labelled graphs and universal algebras

We consider a non empty set A. The notation $B \subseteq A$ specifies that B is a subset of A. If $B \subseteq A$ and $B \neq A$ then we write $B \subset A$. The empty set is denoted by $\emptyset$. We denote by 2^A the power set of A, that is the set of all subsets of A. We denote by A^n the Cartesian product $A \times \ldots \times A$.

By a *partial operation* f on A we understand a partial mapping f from A^n to A. This means that f is defined for the elements of some set $dom(f)$, where $dom(f) \subset A^n$. We shall use the notation $f : dom(f) \longrightarrow A$. The number n is called the *arity* of f. In the case when $dom(f) = A \times A$ we say that f is a *binary operation* on A.

We write $f \prec g$ if $f : dom(f) \longrightarrow A$ and $g : dom(g) \longrightarrow A$ are two functions such that $dom(f) \subseteq dom(g)$ and $f(x) = g(x)$ for all $x \in dom(f)$. If this is the case, we say that f is a *restriction* of g.

We consider a symbol σ of arity 2. By a *partial σ-algebra* we understand a pair $\mathcal{A}=(A, \sigma_A)$, where A is the support set of $\mathcal{A}$ and σ_A is *a partial binary operation* on A. If $dom(\sigma_A) = A \times A$ then we say that $\mathcal{A}$ is a *σ-algebra*.

Let $\mathcal{A}=(A, \sigma_A)$ be a partial σ-algebra. A subset $B \subseteq A$ is a *closed set* in $\mathcal{A}$ if the following condition is fulfilled:

$$(x_1, x_2) \in dom(\sigma_A) \cap (B \times B) \Longrightarrow \sigma_A(x_1, x_2) \in B$$

If $B \subseteq A$ then *the closure of B in $\mathcal{A}$* is the least closed set containing B. The closure of B in (A, σ_A) is denoted by $Cl_{\sigma_A}(B)$ and obviously if B is a closed set then $Cl_{\sigma_A}(B) = B$. It can be shown that if B is not a closed set then $Cl_{\sigma_A}(B) = \bigcup_{n \geq 0} B_n$ where

$$\begin{cases} B_0 = B \\ B_{n+1} = B_n \cup \{\sigma_A(x_1, x_2) \mid (x_1, x_2) \in dom(\sigma_A) \cap (B_n \times B_n)\}, n \geq 0 \end{cases} \tag{1.1}$$

We consider now the partial σ-algebras $\mathcal{A}=(A, \sigma_A)$ and $\mathcal{B}=(B, \sigma_B)$. The mapping $h : A \longrightarrow B$ is a *morphism of partial algebras* from $\mathcal{A}$ to $\mathcal{B}$ if for every $(x_1, x_2) \in dom(\sigma_A)$ the following conditions are fulfilled:

- $(h(x_1), h(x_2)) \in dom(\sigma_B)$
- $\sigma_B(h(x_1), h(x_2)) = h(\sigma_A(x_1, x_2))$

We can define the mapping $h \times h : A \times A \longrightarrow B \times B$ taking by definition, $h \times h(x, y) = (h(x), h(y))$. Using this notation and taking into consideration the representation given in Figure 1.1, we can express intuitively the condition *$h : A \longrightarrow B$ is a morphism* as follows: if we are able to go along the path $(A \times A, A, B)$ then we are able also to go along the path $(A \times A, B \times B, B)$ and we obtain the same result.

We shall use the notation $h : \mathcal{A} \longrightarrow \mathcal{B}$ to specify that h is a morphism from $\mathcal{A}$ to $\mathcal{B}$. A bijective morphism is an *isomorphism.* Two partial σ-algebras $\mathcal{A}$ and $\mathcal{B}$ are *isomorphic algebras* if there is an isomorphism from $\mathcal{A}$ to $\mathcal{B}$.

Let $\mathcal{A}=(A, \sigma_A)$ be a σ-algebra and $M \subseteq A$. By definition, $\mathcal{A}$ is a *Peano σ-algebra over M* if the following conditions are fulfilled ([1]):

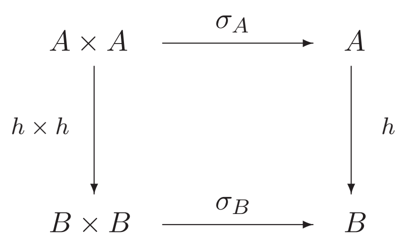


Figure 1.1: The morphism condition

- $Cl_{\sigma_A}(M) = A$
- $\sigma_A(x_1, x_2) \notin M$ for every $x_1, x_2 \in A$
- for every $x_1, x_2, y_1, y_2 \in A$, from $\sigma_A(x_1, x_2) = \sigma_A(y_1, y_2)$ we deduce $x_1 = y_1$ and $x_2 = y_2$

By definition, the σ-algebra $\mathcal{A}$=(A, σ_A) is *free-generated* by $M \subseteq A$ if for every σ-algebra $\mathcal{B}$=(B, σ_B) and every mapping $f : M \longrightarrow B$ there exists a morphism and only one, $h : A \longrightarrow B$, such that $f \prec h$.
For every set M there is a Peano σ-algebra over M. In order to obtain such an algebra we proceed as follows. We may assume $\sigma \notin M$ because otherwise we can rename this element. We take the σ-algebra $\mathcal{H}$=(H, σ_H), where

- H is the set of all nonempty words over $\{\sigma\} \cup M$
- $\sigma_H : H \longrightarrow H$ is defined by $\sigma_H(x_1, x_2) = \sigma x_1 x_2$

Taking $A = Cl_{\sigma_H}(M)$ we obtain the σ-algebra $\mathcal{A}$=(A, σ_A), where σ_A is the restriction of σ_H on A. $\mathcal{A}$ is a Peano σ-algebra over M and we denote this algebra by $PA(M)$. Two Peano σ-algebras over the same set M are isomorphic algebras (particularly they are isomorphic with $\mathcal{A}$) because a Peano σ-algebra over M is a σ-algebra free generated by M and two σ-algebras free generated by M are isomorphic algebras ([1]). Thus, if $M = \{a, b\}$ then the set

$$A = \{a, b, \sigma(a, a), \sigma(b, b), \sigma(a, b), \sigma(a, \sigma(a, a)) \ldots\}$$

gives the support set of the Peano σ-algebra $PA(M)$, $\sigma_A(a, a) = \sigma aa$, $\sigma_A(a, \sigma(a, b))$ $= \sigma a \sigma ab$ and so on. The elements of A are called *terms* by some authors. We observe that the elements of A are nonempty *strings* or *words*.

We consider a nonempty set S. A *binary relation* over S is a subset $\rho \subseteq S \times S$. Equivalently we can write $\rho \in 2^{S \times S}$. If $\rho_1 \in 2^{S \times S}$ and $\rho_2 \in 2^{S \times S}$ then we define:

$$\rho_1 \circ \rho_2 = \{(x, y) \in S \times S \mid \exists z \in S : (x, z) \in \rho_1, (z, y) \in \rho_2\}$$

We remark that the following case can be encountered: $\rho_1 \neq \emptyset$, $\rho_2 \neq \emptyset$ and nevertheless $\rho_1 \circ \rho_2 = \emptyset$. But the empty relation is not a useful one in knowledge representation. In order to avoid this situation we introduce the mapping $prod_S : dom(prod_S) \longrightarrow 2^{S \times S}$ as follows:

$$dom(prod_S) = \{(\rho_1, \rho_2) \in 2^{S\times S} \times 2^{S\times S} \mid \rho_1 \circ \rho_2 \neq \emptyset\}$$

$$prod_S(\rho_1, \rho_2) = \rho_1 \circ \rho_2$$

We denote by $R(prod_S)$ the set of all restrictions of the mapping $prod_S$:

$$R(prod_S) = \{u \mid u \prec prod_S\}$$

We observe that if u is an element of $R(prod_S)$ then the pair $(2^{S\times S}, u)$ is a partial algebra, which is used in the next section.

In the remainder of this section we present in a distinctive manner the concept of labelled graph such that several useful connections with the concepts of universal algebra can be established. These connections allow us to introduce the concept of labelled stratified graph in the next section of this chapter.

A *labelled graph* is a tuple $G = (S, L_0, T_0, f_0)$, where

- S is a finite set, an element of S is a *node* of G
- L_0 is a set of elements named *labels*
- T_0 is a set of binary relations on S
- $f_0 : L_0 \longrightarrow T_0$ is a surjective mapping

Such a structure admits a graphical representation. Each element of S is represented by a rectangle specifying the corresponding node. We draw an arc from $n_1 \in S$ to $n_2 \in S$ and this arc is labelled by $e \in L_0$ if $(n_1, n_2) \in f_0(e)$. This case is shown in Figure 1.2. If we proceed in this manner for each element of $\bigcup_{e\in L_0} f_0(e)$ then we obtain a graphical representation of the whole structure.

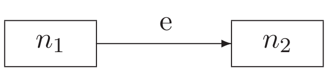


Figure 1.2: A labelled arc

Conversely, from a graphical representation of a labelled graph we can recompose the components of the corresponding structure. In this case a binary relation from T_0 consists of all the pairs of nodes that are linked by an arc and the corresponding arc is labelled by the same element of L_0. For example, Figure 1.3 specifies a graphical representation of some labelled graph. For this graph we have $S = \{x_1, x_2, x_3, x_4\}$, $L_0 = \{a, b, c\}$, $T_0 = \{\rho_1, \rho_2\}$, $\rho_1 = \{(x_1, x_2), (x_2, x_4)\}$, $\rho_2 = \{(x_2, x_3)\}$ and $f_0(a) = f_0(b) = \rho_1$, $f_0(c) = \rho_2$.

The concepts of *labelled graph* and *directed graph* are differentiated by the following two aspects:

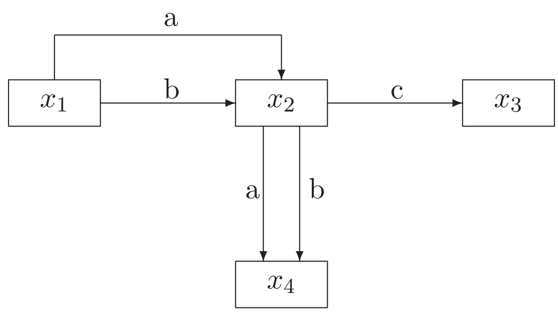


Figure 1.3: Labelled graph

1) It is known that a directed graph is defined by a pair (X, Γ_X), where X is the set of its nodes and Γ_X is a binary relation on X. Only one binary relation is relieved by a directed graph and therefore its graphical representation specifies at most one arc from x to y for each pair (x, y) of nodes. In comparison with such a structure, a labelled graph relieves *a set* of binary relations. Moreover, a given binary relation ρ may be labelled by several distinct elements of L_0. In other words, the set $\{y \in L_0 \mid f_0(y) = \rho\}$ may contain two or more elements. Such a situation is frequently encountered in knowledge representation, where each label of L_0 is assigned to some meaning. For instance, we consider the following knowledge piece: *John is Peter's friend. Peter is George's friend. John likes Peter's sister and Peter likes George's sister. Peter likes cakes.* If we denote $x_1 = John$, $x_2 = Peter$, $x_3 = cakes$ and $x_4 = George$ then the knowledge piece can be transposed in a labelled graph (as in Figure 1.3). The pairs $(John, Peter)$ and $(Peter, George)$ have two common properties: friendship and affection. For this reason the binary relation $\{(John, Peter), (Peter, George)\}$ will be labelled by two distinct symbols.

2) If all labels appearing on a labelled graph are erased then we do not obtain necessarily a directed graph. This can be easily viewed in Figure 1.3: there are two arcs for each pair (x_1, x_2) and (x_2, x_4), therefore we do not obtain a directed graph.

We consider a labelled graph $G = (S, L_0, T_0, f_0)$ and a symbol σ of arity 2. We consider the Peano σ-algebra $PA(L_0)$ over L_0. The support set of this algebra is

$$B = \bigcup_{n \geq 0} B_n \tag{1.2}$$

where

$$\begin{cases} B_0 = L_0 \\ B_{n+1} = B_n \cup \{\sigma(x_1, x_2) \mid (x_1, x_2) \in B_n \times B_n\}, n \geq 0 \end{cases} \tag{1.3}$$

For simplicity we denote $PA(L_0) = (B, \sigma)$. We consider some collection of subsets of B, denoted by $Initial(L_0)$. Namely, we say that $X \in Initial(L_0)$ if the following conditions are fulfilled:

- $L_0 \subseteq X \subseteq B$
- if $\sigma(u, v) \in X$ then $u \in X$ and $v \in X$

For example, if $L_0 = \{a, b\}$ then the set $X = \{a, b, \sigma(a, b), \sigma(a, \sigma(a, b))\}$ is in $Initial(L_0)$.

Remark 1.2.1 *Generally speaking, if $L \in Initial(L_0)$ then the pair $(L, \{\sigma_L\})$, where*

- $dom(\sigma_L) = \{(x, y) \in L \times L \mid \sigma(x, y) \in L\}$
- $\sigma_L(x, y) = \sigma(x, y)$ *for every* $(x, y) \in dom(\sigma_L)$

is a partial algebra.

For instance, if we consider the Peano σ-algebra $PA(\{a, b\})$ then

$$L = \{a, b, \sigma(a, a), \sigma(a, b), \sigma(b, a), \sigma(a, \sigma(a, b))\} \in Initial(\{a, b\})$$

and

$$dom(\sigma_L) = \{(a, a), (a, b), (b, a), (a, \sigma(a, b))\}$$

Remark 1.2.2 *Consider the closure $Cl_u(T_0)$ of T_0 in the partial algebra $(2^{S \times S}, u)$, where $u \in R(prod_S)$. We have $Cl_u(T_0) = \bigcup_{n \geq 0} X_n$, where*

$$\begin{cases} X_0 = T_0 \\ X_{n+1} = X_n \cup \{u(\rho_1, \rho_2) \mid (\rho_1, \rho_2) \in dom(u) \cap (X_n \times X_n)\}, n \geq 0 \end{cases} \tag{1.4}$$

But S is a finite set, therefore there is n such that $X_n = X_{n+1}$ and thus $Cl_u(T_0) = \bigcup_{k=0}^{n} X_k = X_n$ ([17]).

1.3 Labelled stratified graphs

In this section we define the concept of *labelled stratified graph* (shortly, LSG) and we show that for each labelled graph we can build a LSG, therefore we prove the existence of this structure. Some examples of LSGs are also presented.

Definition 1.3.1 *A* **labelled stratified graph** $\mathcal{G}$ *over G is a tuple (G, L, T, u, f) where*

- $G = (S, L_0, T_0, f_0)$ *is a labelled graph*
- $L \in Initial(L_0)$

- *$u \in R(prod_S)$ and $T = Cl_u(T_0)$*
- *$f : (L, \{\sigma_L\}) \longrightarrow (2^{S\times S}, \{u\})$ is a morphism of partial algebras such that $f_0 \prec f$, $f(L) = T$ and if $(f(x), f(y)) \in dom(u)$ then $(x, y) \in dom(\sigma_L)$*

We give now some details concerning the last condition imposed in the definition of LSG. Because f is a morphism of partial algebras, in the diagram of Figure 1.4 we have the following property: if we are able to go along the path $(L \times L, L, 2^{S\times S})$ then we are able also to go along the path $(L \times L, 2^{S\times S} \times 2^{S\times S}, 2^{S\times S})$ and we obtain the same result.

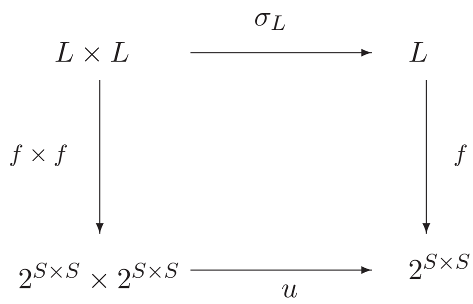


Figure 1.4: Commutative diagram

Thus from the morphism property we have

$$dom(\sigma_L) \subseteq \{(x, y) \in L \times L \mid (f(x), f(y)) \in dom(u)\} \tag{1.5}$$

The last condition specified in Definition 1.3.1 gives the converse condition: if we can go along the path $(L \times L, 2^{S\times S} \times 2^{S\times S}, 2^{S\times S})$ then we can go along the path $(L \times L, L, 2^{S\times S})$ and we obtain the same final result. Indeed, if $(f(x), f(y)) \in dom(u)$ then $(x, y) \in dom(\sigma_L)$ therefore $\sigma_L(x, y) \in L$. Now by the morphism property we have $f(\sigma_L(x, y)) = u(f(x), f(y))$. It follows that

$$dom(\sigma_L) \supseteq \{(x, y) \in L \times L \mid (f(x), f(y)) \in dom(u)\} \tag{1.6}$$

From (1.5) and (1.6) we obtain

$$dom(\sigma_L) = \{(x, y) \in L \times L \mid (f(x), f(y)) \in dom(u)\} \tag{1.7}$$

In order to relieve some aspects of this concept we consider the labelled graph drawn in Figure 3.4. This labelled graph is defined by the following components:

$S = \{x_1, x_2, x_3, x_4, x_5\}$

$L_0 = \{a, b, c, d\}$

$T_0 = \{\rho_1, \rho_2, \rho_3\}$, where $\rho_1 = \{(x_1, x_2), (x_2, x_4)\}$, $\rho_2 = \{(x_2, x_3)\}$, $\rho_3 = \{(x_5, x_1)\}$

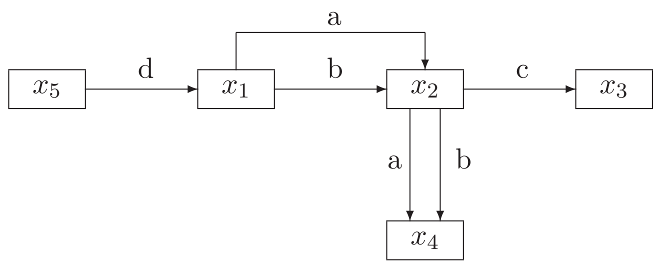


Figure 1.5: Labelled graph

$f_0(a) = f_0(b) = \rho_1,\ f_0(c) = \rho_2,\ f_0(d) = \rho_3$

Taking $dom(u) = \{(\rho_1, \rho_1)\}$ we obtain $u(\rho_1, \rho_1) = \rho_4$, where $\rho_4 = \{(x_1, x_4)\}$. In this case various sets $L \in Initial(L_0)$ and corresponding morphisms f of partial algebras as in Figure 1.4 can be taken. For example,

$$L = L_0 \cup \{\sigma(a, a)\}$$

$$f_0 \prec f,\ f(\sigma(a, a)) = u(f(a), f(a)) = u(\rho_1, \rho_1) = \rho_4$$

We obtain $T = Cl_u(T_0) = T_0 \cup \{\rho_4\}$. Moreover, we have $f(L) = T$, $f_0 \prec f$, $f : L \longrightarrow 2^{S\times S}$ is a morphism of partial algebras but (G, L, T, u, f) is not a LSG. Really, $(f(a), f(b)) = (\rho_1, \rho_1) \in dom(u)$ but $\sigma(a, b) \notin L$.
A similar situation is encountered if we take $L = L_0 \cup \{\sigma(a, a), \sigma(b, b)\}$ or $L = L_0 \cup \{\sigma(a, a), \sigma(a, b)\}$.
In order to obtain a LSG we take

$$L = L_0 \cup \{\sigma(a, a), \sigma(b, b), \sigma(a, b), \sigma(b, a)\}$$

$$f_0 \prec f;\ f(\sigma(a, a)) = f(\sigma(a, b)) = f(\sigma(b, b)) = f(\sigma(b, a)) = \rho_4$$

and in this case the last condition of the definition of a LSG is fulfilled. We shall refer to this LSG in the next chapter.

Obviously, if we choose another mapping u then we obtain also a labelled stratified graph over the same labelled graph. In some cases it is not easy to build directly a labelled stratified graph if we do not apply an algorithm. Such a case is presented in [17], where an example of a LSG with an infinite set L is given.

Let $G = (S, L_0, T_0, f_0)$ be a labelled graph. Take $u \in R(prod_S)$ and consider the closure $T = Cl_u(T_0)$ of T_0 in the algebra $(2^{S\times S}, u)$. We consider the Peano σ-algebra $PA(L_0) = (B, \sigma)$ over L_0, where B is given by (1.2) and (1.3).
We observe that $dom(f_0) = B_0$ and we can define recursively for every natural number $n \geq 0$:

- $D_{n+1} = \{\sigma(p, q) \in B_{n+1} \setminus B_n \mid p, q \in dom(f_n),\ (f_n(p), f_n(q)) \in dom(u)\}$

- $dom(f_{n+1}) = dom(f_n) \cup D_{n+1}$
- $f_{n+1}(x) = \begin{cases} f_n(x) & \text{if} \quad x \in dom(f_n) \\ u(f_n(p), f_n(q)) & \text{if} \quad x = \sigma(p,q) \in D_{n+1} \end{cases}$

It is not difficult to observe that the following properties are satisfied by these entities:

(a) $D_i \cap D_j = \emptyset$ for $i \neq j$; $L_0 \cap D_i = \emptyset$ for every $i \geq 1$

(b) $dom(f_n) = B_0 \cup \bigcup_{k=1}^{n} D_k \subseteq B_n$, for every $n \geq 0$

(c) $dom(f_n) \cap D_{n+1} = \emptyset$ for every $n \geq 0$

Taking into account these properties we obtain the following proposition:

Proposition 1.3.1 *For every $n \geq 0$ the following properties are satisfied:*

1) *the mapping f_{n+1} is well defined*

2) $f_n \prec f_{n+1}$

3) $f_n : dom(f_n) \longrightarrow T$

Proof. Because B is given by (1.2) and (1.3), it follows that if $x \in B_{n+1} \setminus B_n$ then x can be uniquely written as $x = \sigma(p,q)$ for some $p, q \in B_n$. Taking into account the definition of f_{n+1} and the fact that $dom(f_n) \cap D_{n+1} = \emptyset$, we deduce that the mapping f_{n+1} is well defined for every $n \geq 0$.
The property $f_n \prec f_{n+1}$ is obtained directly from the definition of f_{n+1}. The last sentence can be verified by induction on n. Because $G = (S, L_0, T_0, f_0)$ is a labelled graph, we have $f_0 : dom(f_0) \longrightarrow T_0$. But $T_0 \subseteq T$, therefore the property is verified for $n = 0$. Suppose the property is true for n. Take an element $x \in dom(f_{n+1})$. If $x \in dom(f_n)$ then $f_{n+1}(x) = f_n(x) \in T$. If $x \in D_{n+1}$ then $x = \sigma(p,q)$ for some $p, q \in dom(f_n)$ and $(f_n(p), f_n(q)) \in dom(u)$. By the inductive assumption we have $f_n(p) \in T$ and $f_n(q) \in T$. But $f_{n+1}(x) = u(f_n(p), f_n(q))$ and T is closed under u, therefore $f_{n+1}(x) \in T$. ■

Definition 1.3.2 *We define the mapping $f^* : dom(f^*) \longrightarrow T$ as follows:*

$$dom(f^*) = \bigcup_{n \geq 0} dom(f_n) = L_0 \cup \bigcup_{k \geq 1} D_k$$

$$f^*(x) = \begin{cases} f_0(x) & \text{if} \quad x \in L_0 \\ f_k(x) & \text{if} \quad x \in D_k \end{cases}$$

Proposition 1.3.2 *For every $n \geq 0$ we have $dom(f^*) \cap B_n = dom(f_n)$.*

Proof. We observe that for each $n \geq 0$ we have

$$dom(f^*) = dom(f_n) \cup \bigcup_{k \geq n+1} D_k$$

therefore

$$dom(f^*) \cap B_n = (dom(f_n) \cap B_n) \cup \bigcup_{k \geq n+1} (D_k \cap B_n) = dom(f_n)$$

because $dom(f_n) \subseteq B_n$ and for $k \geq n+1$ we have $D_k \cap B_n = \emptyset$.

Proposition 1.3.3 *Let be $L^* = dom(f^*)$. The set L^* satisfies the following properties:*

$$L^* \supseteq L_0 \tag{1.8}$$

$$\sigma(p,q) \in L^* \quad \text{iff} \quad \{p, q \in L^* \quad \text{and} \quad (f^*(p), f^*(q)) \in dom(u)\} \tag{1.9}$$

$$L^* \in Initial(L_0) \tag{1.10}$$

Proof. The relation (1.8) is obtained directly from Definition 1.3.2. Let us prove (1.9). We suppose $\sigma(p,q) \in L^* = dom(f^*)$. Let n be the smallest natural number such that $\sigma(p,q) \in dom(f_n)$. We have $n \geq 1$ since $\sigma(p,q) \notin L_0 = dom(f_0)$. It follows that $\sigma(p,q) \in dom(f_n) \setminus dom(f_{n-1})= D_n$. By the definition of D_n it follows that $p, q \in dom(f_{n-1}) \subseteq L^*$ and $(f^*(p), f^*(q)) = (f_{n-1}(p), f_{n-1}(q)) \in dom(u)$.
Conversely, let $p, q \in L^*$ such that $(f^*(p), f^*(q)) \in dom(u)$. To use an uniform notation we consider $D_0 = L_0$. Because $L^* = \bigcup_{n \geq 0} D_n$ we deduce that there are the natural numbers i_p, i_q such that $p \in D_{i_p}$ and $q \in D_{i_q}$. Taking $k = max\{i_p, i_q\}$ we obtain $p \in dom(f_k)$, $q \in dom(f_k)$ and $\sigma(p,q) \in B_{k+1} \setminus B_k$. It follows that $(f^*(p), f^*(q)) = (f_{i_p}(p), f_{i_q}(q))= (f_k(p), f_k(q)) \in dom(u)$, therefore $\sigma(p,q) \in D_{k+1} \subseteq L^*$. Thus (1.9) is true. Now, (1.10) is obtained from (1.8) and (1.9). ■

Based on Remark 1.2.1 and Proposition 1.3.3 we can consider the partial σ-algebra $\mathcal{A}_{L^*} = (L^*, \sigma_{L^*})$. On the other hand the same propositions allow us to obtain the following property for σ_{L^*}, which is used later:

Corollary 1.3.1

$$dom(\sigma_{L^*}) = \{(x,y) \in L^* \times L^* \mid (f^*(x), f^*(y)) \in dom(u)\} \tag{1.11}$$

Proof. If $(x,y) \in L^* \times L^*$ and $(f^*(x), f^*(y)) \in dom(u)$ then by (1.9) we have $\sigma(x,y) \in L^*$. By Remark 1.2.1 it follows that $(x,y) \in dom(\sigma_{L^*})$. Conversely, if $(x,y) \in dom(\sigma_{L^*})$ then $(x,y) \in L^* \times L^*$ and $\sigma(x,y) \in L^*$. By (1.9) we have $(f^*(x), f^*(y)) \in dom(u)$. ■

Because $T = Cl_u(T_0)$, applying Remark 1.2.2 we obtain a natural number n_0 and the increasing sequence $T_0 = X_0 \subset X_1 \subset \ldots \subset X_{n_0} = X_{n_0+1} = \ldots = T$, whose elements are given by (1.4). We consider the following sets:

$$\begin{cases} C_0 = X_0 \\ C_i = X_i \setminus X_{i-1}, \qquad i \in \{1, \ldots, n_0\} \end{cases}$$

We have $C_i \cap C_j = \emptyset$ for $i \neq j$, $C_i \neq \emptyset$ and $X_i = \bigcup_{j=0}^{i} C_j$ for $i \in \{0, \ldots, n_0\}$.

Proposition 1.3.4 *For every $i \in \{1, \ldots, n_0\}$, $d \in C_i$ if and only if*

i) there exist $p \in C_{i-1}$, $q \in X_{i-1}$ such that $d = u(p,q)$ or $d = u(q,p)$

ii) $d \neq u(d_1, d_2)$ for every $d_1, d_2 \in X_{i-2}$

Proof We suppose that $d \in C_i = X_i \setminus X_{i-1}$ for some $i \in \{1, \ldots, n_0\}$. There exist $p, q \in X_{i-1}$ such that $(p,q) \in dom(u)$ and $d = u(p,q)$. Two cases are possible:

a) $p, q \in X_{i-2}$; in this case we have $d \in X_{i-1}$, which is not true

b) $p \in C_{i-1}$ or $q \in C_{i-1}$, therefore *i)* is true

In order to prove *ii)* we suppose by contrary that $d = u(d_1, d_2)$ for some $d_1, d_2 \in X_{i-2}$. Thus we have $d \in X_{i-1}$, which is not true.
Conversely, if $p \in C_{i-1}$ and $q \in X_{i-1}$ then $d = u(p,q) \in T_i$. If $d \neq u(d_1, d_2)$ for every $d_1, d_2 \in X_{i-2}$ then $d \notin X_{i-1}$. Therefore $d \in X_i \setminus X_{i-1} = C_i$. ■

Proposition 1.3.5 *For every $i \in \{1, \ldots, n_0\}$ we have $C_i \subseteq f_i(D_i) \subseteq X_i$*

Proof. We prove first by induction on i that $C_i \subseteq f_i(D_i)$ for every $i \in \{1, \ldots, n_0\}$. We verify this property for $i = 1$. We have $D_1 = \{\sigma(p,q) \in B_1 \setminus B_0 \mid p, q \in dom(f_0), (f_0(p), f_0(q)) \in dom(u)\}$. Let be $z \in C_1$. By Proposition 1.3.4 it follows that $z = u(d_1, d_2)$ for some $d_1, d_2 \in X_0$. Because $f_0 : L_0 \longrightarrow X_0$ is a surjective mapping and $L_0 = B_0$, we deduce that there are $a, b \in B_0$ such that $d_1 = f_0(a)$, $d_2 = f_0(b)$. Therefore we have $z = u(f_0(a), f_0(b))$ for some $a, b \in B_0$. But $\sigma(a,b) \in B_1 \setminus B_0$, $a, b \in dom(f_0) = B_0$, $(f_0(a), f_0(b)) \in dom(u)$, therefore $\sigma(a,b) \in D_1 = dom(f_1)$. We obtain $f_1(\sigma(a,b)) = u(f_0(a), f_0(b)) = z$, therefore $C_1 \subseteq f_1(D_1)$.
We suppose now that $C_j \subseteq f_j(D_j)$ for $j \in \{1, \ldots, i\}$ and we shall prove that $C_{i+1} \subseteq f_{i+1}(D_{i+1})$. We consider an arbitrary element $z \in C_{i+1}$. By Proposition 1.3.4 it follows that there are $d_1 \in C_i$ and $d_2 \in C_j$ for some $j \in \{0, \ldots, i\}$ such that $z = u(d_1, d_2)$ or $z = u(d_2, d_1)$. Obviously it is enough to consider the situation $z = u(d_1, d_2)$. By the inductive assumption we have $C_i \subseteq f_i(D_i)$ and $C_j \subseteq f_j(D_j)$, therefore there are $a \in D_i$, $b \in D_j$ such that $d_1 = f_i(a)$, $d_2 = f_j(b)$. It follows that $\sigma(a,b) \in B_{i+1} \setminus B_i$, $a \in dom(f_i)$, $b \in dom(f_j) \subseteq dom(f_i)$ and $(f_i(a), f_i(b)) = (f_i(a), f_j(b)) \in dom(u)$. Thus $\sigma(a,b) \in D_{i+1}$ and $f_{i+1}(\sigma(a,b)) = u(f_i(a), f_i(b)) = u(f_i(a), f_j(b)) = u(d_1, d_2) = z$, therefore $z \in f_{i+1}(D_{i+1})$.
We prove now that $f_i(D_i) \subseteq X_i$. First we verify this property for $i = 1$. If $z \in f_1(D_1)$ then $z = f_1(\sigma(p,q))= u(f_0(p), f_0(q))$ for some $p, q \in L_0$. Because $(f_0(p), f_0(q)) \in (X_0 \times X_0) \cap dom(u)$, using the definition of X_1 we deduce that $z \in X_1$. We suppose that $f_j(D_j) \subseteq X_j$ for every $j < i$ and let be $z \in f_i(D_i)$. It follows that $z = f_i(\sigma(p,q)) = u(f_{i-1}(p), f_{i-1}(q))$ for some $p, q \in dom(f_{i-1})$ such that $(f_{i-1}(p), f_{i-1}(q)) \in dom(u)$. There are $i_1, i_2 \in \{0, \ldots, i-1\}$ such that $p \in D_{i_1}$ and $q \in D_{i_2}$ because $p, q \in dom(f_{i-1}) = \bigcup_{j<i} D_j$. We have $f_{i-1}(p) = f_{i_1}(p)$ and $f_{i-1}(q) = f_{i_2}(q)$. By the inductive assumption we have $f_{i_1}(p) \in X_{i_1} \subseteq X_{i-1}$ and $f_{i_2}(q) \in X_{i_2} \subseteq X_{i-1}$. Thus we have $z = u(f_{i_1}(p), f_{i_2}(q))$, $f_{i_1}(p) \in X_{i-1}$ and $f_{i_2}(q) \in X_{i-1}$. From the definition of X_i we deduce $z \in X_i$. ■

Remark 1.3.1 *If we denote $D_0 = L_0$ then we observe that Proposition 1.3.5 is true also for $n = 0$.*

Corollary 1.3.2 *For every $i \in \{1, \ldots, n_0\}$ we have $D_i \neq \emptyset$.*

Proof. We apply Proposition 1.3.5. If by contrary, we suppose that for some $i \in \{1, \ldots, n_0\}$ we have $D_i = \emptyset$ then $f_i(D_i) = \emptyset \supseteq C_i$. Thus $C_i = \emptyset$, which is not true. ∎

Proposition 1.3.6 *The mapping $f^* : (L^*, \sigma_{L^*}) \longrightarrow (T, u)$ is a surjective morphism of σ-algebras.*

Proof. We consider an arbitrary pair $(x_1, x_2) \in dom(\sigma_{L^*})$. By (1.11) we have $(x_1, x_2) \in L^* \times L^* = dom(f^*) \times dom(f^*)$ and $(f^*(x_1), f^*(x_2)) \in dom(u)$. Based on Remark 1.2.1 and Proposition 1.3.3 we have $\sigma_{L^*}(x_1, x_2) = \sigma(x_1, x_2)$. Using the definition of the mapping f^* we obtain $f^*(\sigma_{L^*}(x_1, x_2)) = f^*(\sigma(x_1, x_2)) = u(f^*(x_1), f^*(x_2))$, therefore f^* is a morphism. Taking into account the properties

$dom(f^*) = L_0 \cup \bigcup_{k \geq 1} D_k$,

$T = \bigcup_{k=0}^{n_0} C_k$

and using Proposition 1.3.5, we deduce that f^* is a surjective mapping. ∎

We can obtain now the following theorem:

Theorem 1.3.1 *Let $G = (S, L_0, T_0, f_0)$ be a labelled graph. For every $u \in R(prod_S)$ the tuple $\mathcal{G}^* = (G, L^*, T, u, f^*)$ is a labelled stratified over G, where $T = Cl_u(T_0)$, $L^* = dom(f^*)$ and f^* is given in Definition 1.3.2.*

Proof. Taking into consideration Definition 1.3.1 and the Proposition 1.3.6 it remains to verify the following property: if $(f^*(x), f^*(y)) \in dom(u)$ then $(x, y) \in dom(\sigma_{L^*})$. But using (1.9) we obtain $\sigma(x, y) \in L^*$, therefore by Remark 1.2.1 we have $(x, y) \in dom(L^*)$. ∎

Based on the next definition we can explain why the structure introduced in Definition 1.3.1 is named labelled stratified graph.

Definition 1.3.3 *Let $\mathcal{G} = (G, L, T, u, f)$ be a labelled stratified graph. We define*

$$\begin{cases} Layer(L, 0) = L_0 \\ Layer(L, n+1) = L \cap (B_{n+1} \setminus B_n), \ n \geq 0 \end{cases} \tag{1.12}$$

where $PA(L_0) = (B, \sigma)$ is the Peano σ-algebra over L_0 and B is given by (1.2) and (1.3).

The set $Layer(L, n)$ is called the n^{th} layer of L.

We can explain now why the structure given in Definition 1.3.1 is named *labelled stratified graph.* Let us denote by $\mathcal{G} = (G, L, T, u, f)$ such a structure. The explanation is based on the following remarks:

(1) By the surjective morphism f, the set T is covered by L; in other words every binary relation of T has at least one label and the labels are assigned by f; thus, the elements of T are *labelled* binary relations.

(2) Based on Definition 1.3.3 we observe that the set L of all the labels is divided into several layers; the first layer is given by L_0; each element of the layer i is obtained by means of two elements, one of them belonging to the layer $i-1$ and the other being in the set union of the layers $0, 1, \ldots, i-1$. We obtain a *stratified* structure of the set L of labels.

The next proposition shows that the layers of $\mathcal{G}^*$ are exactly the sets D_n.

Proposition 1.3.7 *Let $\mathcal{G}^* = (G, L^*, T, u, f^*)$ be the labelled stratified graph obtained in Theorem 1.3.1. Then $Layer(L^*, n) = D_n$ for every $n \geq 0$.*

Proof. By Definition 1.3.2 we have $L^* = \bigcup_{j\geq 0} D_j$, where $D_0 = L_0$. For every $j \geq 0$ we have $D_j \subseteq B_j \setminus B_{j-1}$, where for $j = 0$ we consider $B_{j-1} = B_{-1} = \emptyset$. It follows that

$$D_j \cap (B_{k+1} \setminus B_k) = \begin{cases} \emptyset & \text{for } j \neq k+1 \\ D_{k+1} & \text{for } j = k+1 \end{cases} \tag{1.13}$$

From (1.12) and (1.13) we obtain:

$$Layer(L^*, k+1) = L^* \cap (B_{k+1} \setminus B_k) = \bigcup_{j\geq 0} D_j \cap (B_{k+1} \setminus B_k) = D_{k+1}$$

For $n = 0$ the property is obviously true and thus the proposition is proved. ■

The set L^* may be divided into an infinite number of layers. In order to emphasize this fact we take the following example. We consider the labelled graph from Figure 1.6.

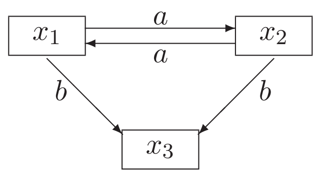


Figure 1.6: A labelled graph giving an infinite hierarchy of layers

We take $S = \{x_1, x_2, x_3\}$ and $L_0 = \{a, b\}$. We consider the binary relations

$$\rho_1 = \{(x_1, x_2), (x_2, x_1)\},\ \rho_2 = \{(x_1, x_3), (x_2, x_3)\}$$

Take $T_0 = \{\rho_1, \rho_2\}$, $T = Cl_u(T_0)$, $u = prod_S$ and $\rho_3 = \{(x_1, x_1), (x_2, x_2)\}$. We obtain:

$u(\rho_1, \rho_1) = \rho_3$, $u(\rho_1, \rho_2) = \rho_2$, $u(\rho_1, \rho_3) = \rho_1$

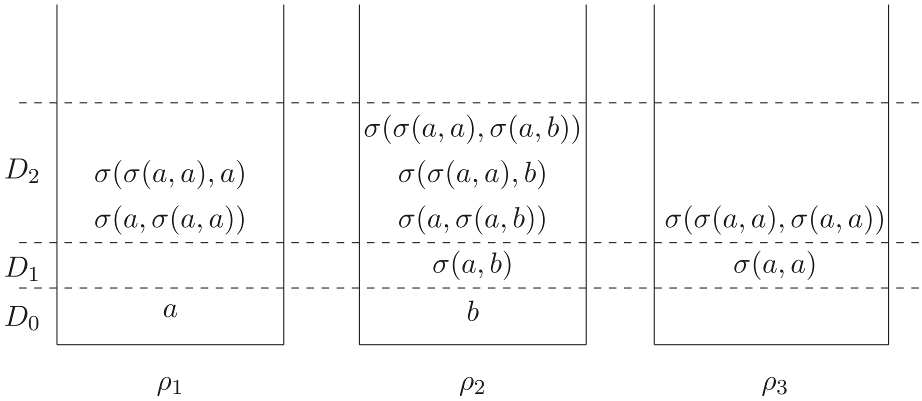


Figure 1.7: An infinite hierarchy of layers

$u(\rho_3,\rho_1)=\rho_1$, $u(\rho_3,\rho_2)=\rho_2$, $u(\rho_3,\rho_3)=\rho_3$

Applying (1.4) we have $X_1 = T_0 \cup \{\rho_3\}$ and $X_2 = X_1$ therefore $T = \{\rho_1, \rho_2, \rho_3\}$. The computation of the elements D_n will conduce us to the elements specified in Figure 1.7. Taking into consideration the values of the mapping f^* we obtain three containers of labels, each of them containing all the labels for ρ_1, ρ_2, ρ_3 respectively. Each container contains an infinite set of labels. In order to verify this fact we denote

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

and for each natural number n we take

$$\sigma_n(A,B) = \bigcup_{j \le n} [\sigma(A_n, B_j) \cup \sigma(A_j, B_n)]$$

where A_j, B_j are subsets of L, A is the sequence $A_0, A_1, \ldots$ and B is the sequence $B_0, B_1, \ldots$. For every $j \geq 0$ and $i \in \{1,2,3\}$ we denote $D_j(\rho_i) = \{u \in D_j \mid f(u) = \rho_i\}$ and let $D(\rho_i)$ be the sequence $D_0(\rho_i), D_1(\rho_i), \ldots$.

Taking into account the manner in which σ_T is defined we obtain the following equations:

$$\begin{cases} D_{n+1}(\rho_3) = \sigma_n(D(\rho_1), D(\rho_1)) \cup \sigma_n(D(\rho_3), D(\rho_3)) \\ D_{n+1}(\rho_2) = \sigma_n(D(\rho_1), D(\rho_2)) \cup \sigma_n(D(\rho_3), D(\rho_2)) \\ D_{n+1}(\rho_1) = \sigma_n(D(\rho_1), D(\rho_3)) \cup \sigma_n(D(\rho_3), D(\rho_1)) \end{cases} \tag{1.14}$$

We observe that $D_2(\rho_1), D_2(\rho_2)$ and $D_2(\rho_3)$ are nonempty sets. Based on (1.14) we can verify by induction that $D_n(\rho_1)$, $D_n(\rho_2)$ and $D_n(\rho_3)$ are also nonempty sets for every $n \geq 3$. Thus we obtain an infinite hierarchy of layers for L^*.

Remark 1.3.2 *The construction given in Theorem 1.3.1 is possible even if T_0 is a closed set, that is, $Cl_u(T_0) = T_0$. In order to observe this fact we consider the labelled graph from Figure 1.8.*

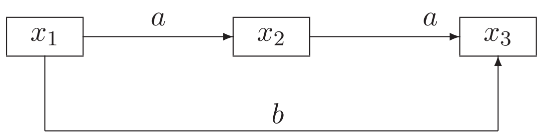


Figure 1.8: A labelled graph such that $Cl_u(T_0) = T_0$

Taking $u = prod_S$, $\rho_1 = \{(x_1, x_2), (x_2, x_3)\}$, $\rho_2 = \{(x_1, x_3)\}$ *and* $L_0 = \{a, b\}$ *we obtain* $Cl_u(T_0) = T_0$, $L^* = \{a, b, \sigma(a, a)\}$, $f^*(a) = \rho_1$, $f^*(b) = \rho_2$, $f^*(\sigma(a, a)) = \rho_2$.

Remark 1.3.3 *As we proved in this chapter, for every labelled graph* G *there is a labelled stratified graph over* G. *We denote by* $Strat(G)$ *the set of all labelled graphs over* G. *By Proposition 1.3.1 we have* $Strat(G) \neq \emptyset$.

Chapter 2

Algebraic properties of LSGs

2.1 Overview

2.2 Environments for labelled graphs.

We consider a labelled graph $G = (S, L_0, T_0, f_0)$. For each $u \in R(prod_S)$ we denote by $env_G(u)$ the set of all tuples (L, T, f) such that $(G, L, T, u, f) \in Strat(G)$. As we shown in the previous chapter, for every $u \in R(prod_S)$ there is $(G, L, T, u, f) \in Strat(G)$, therefore $env_G(u) \neq \emptyset$. We denote

$$Env(G) = \bigcup_{u \in R(prod_S)} env_G(u)$$

An element of $Env(G)$ is named *environment for G* and the elements of $env_G(u)$ are called *environments generated by u*. Because $env_G(u) \subseteq Env(G)$, we obtained a mapping

$$env_G : R(prod_S) \longrightarrow 2^{Env(G)} \tag{2.1}$$

The terms used above can be explained as follows. First, if we transpose some problem in a labelled graph G and we build an appropriate $(G, L, T, u, f) \in Strat(G)$ then the pair (L, T) offers the space by means of which we can identify all the solutions of the problem. The connection between L and T is given by f. In section 3.3 of the next chapter this process can be observed in detail. On the other hand, in order to build a LSG corresponding to $u \in R(prod_S)$, only the features of the mapping u are used. This fact is proved in this section and it constitutes a basic result to prove other properties for $LSGs$. This can explain the context "environment generated by u ".

The following mapping will be used also in this chapter:

$$H_G : R(prod_S) \longrightarrow 2^{2^{S\times S}},\ H_G(u) = Cl_u(T_0) \tag{2.2}$$

The pairs $(R(prod_S), \prec)$ and $(2^{2^{S\times S}}, \subseteq)$ are partially ordered sets, where $\subseteq$ is the set inclusion.

Proposition 2.2.1 *The mapping H_G is monotone, that is, if $u \prec v$ then $H_G(u) \subseteq H_G(v)$.*

Proof. $H_G(v)$ is a closed set under v and includes T_0. Because $u \prec v$, $H_G(v)$ is closed under u. But $H_G(u)$ is the smallest set which is closed under u such that $T_0 \subseteq H_G(u)$, therefore $H_G(u) \subseteq H_G(v)$. ∎

A basic result concerning *LSGs* refers to the number of the environments generated by a mapping. In order to prove this result the next proposition is used.

Proposition 2.2.2 *We consider a labelled graph $G = (S, L_0, T_0, f_0)$. Let be $u, v \in R(prod_S)$ such that $u \prec v$. If $(L_1, T_1, f) \in env_G(u)$ and $(L_2, T_2, g) \in env_G(v)$ then $L_1 \subseteq L_2$, $T_1 \subseteq T_2$ and $f \prec g$.*

Proof. The support set of the Peano-algebra over L_0 is defined by (1.3) and it is the set $B = \bigcup_{n\geq 0} B_n$, where $B_0 = L_0$. Because $L_0 \subseteq L_1 \subseteq B$ and $L_0 \subseteq L_2 \subseteq B$ it follows that

$$L_1 = L_0 \cup \bigcup_{n\geq 0} (L_1 \cap (B_{n+1} \setminus B_n)) \tag{2.3}$$

$$L_2 = L_0 \cup \bigcup_{n\geq 0} (L_2 \cap (B_{n+1} \setminus B_n)) \tag{2.4}$$

There exist $\mathcal{G}_1 = (G, L_1, T_1, u, f)$ and $\mathcal{G}_2 = (G, L_2, T_2, v, g)$ in $Strat(G)$, therefore by (1.7) we have

$$dom(\sigma_{L_1}) = \{(x, y) \in L_1 \times L_1 \mid (f(x), f(y)) \in dom(u)\} \tag{2.5}$$

$$dom(\sigma_{L_2}) = \{(x, y) \in L_2 \times L_2 \mid (g(x), g(y)) \in dom(v)\} \tag{2.6}$$

We shall verify by induction on n that for every $n \geq 0$

$$L_1 \cap (B_{n+1} \setminus B_n) \subseteq L_2 \cap (B_{n+1} \setminus B_n) \tag{2.7}$$

$$f(x) = g(x) \text{ if } x \in L_1 \cap (B_{n+1} \setminus B_n) \tag{2.8}$$

1) *Initial step*:
Suppose $n = 0$ and take $x \in L_1 \cap (B_1 \setminus B_0)$. There are $a, b \in B_0 = L_0$, uniquely determined, such that $x = \sigma(a, b) = \sigma_{L_1}(a, b)$. The following properties are satisfied by a and b:

$$(a, b) \in dom(\sigma_{L_1}) \tag{2.9}$$

$$f_0(a) = f(a) = g(a),\ f_0(b) = f(b) = g(b) \tag{2.10}$$

Really, (2.9) is true because $x = \sigma_{L_1}(a, b)$ and (2.10) is obtained from $a, b \in L_0$, $f_0 \prec f$, $f_0 \prec g$. From (2.9) and (2.5) we obtain $(f(a), f(b)) \in dom(u)$. Consequently, from (2.10) we obtain $(g(a), g(b)) \in dom(u)$. But $dom(u) \subseteq dom(v)$, therefore $(g(a), g(b)) \in dom(v)$. Because $(a, b) \in L_0 \times L_0$ and $L_0 \subseteq L_2$, we have $(a, b) \in L_2 \times L_2$. Using (2.6) we obtain $(a, b) \in dom(\sigma_{L_2})$ and therefore $\sigma_{L_2}(a, b) = \sigma(a, b) = x \in L_2$.

Thus, the property (2.7) is verified for $n = 0$. Moreover, f and g are morphisms therefore

$$f(x) = f(\sigma(a,b)) = f(\sigma_{L_1}(a,b)) = u(f(a), f(b)) = v(g(a), g(b))$$

$$g(x) = g(\sigma(a,b)) = g(\sigma_{L_2}(a,b)) = v(g(a), g(b))$$

Thus, $f(x) = g(x)$ and (2.8) is true for $n = 0$.
2) *Inductive step*:
We suppose the properties (2.7) and (2.8) are true for $n \in \{0, \ldots, k\}$. Let us take an arbitrary element x such that

$$x \in L_1 \cap (B_{k+2} \setminus B_{k+1}) \tag{2.11}$$

Using (1.3) we deduce that there are $a, b \in B_{k+1}$, uniquely determined, such that $x = \sigma_{L_1}(a, b)$. We prove that

$$a, b \in L_0 \cup \bigcup_{n \in \{0,\ldots,k\}} L_1 \cap (B_{n+1} \setminus B_n) \tag{2.12}$$

$$(a, b) \in dom(\sigma_{L_2}) \tag{2.13}$$

We observe that $a, b \in L_1$ because $x \in L_1$ and $L_1 \in Initial(L_0)$. It is not possible to have simultaneously $a \in B_k$ and $b \in B_k$. Really, if by contrary we suppose $a \in B_k$ and $b \in B_k$ then $x = \sigma(a, b) \in B_{k+1}$, which is not true. Therefore $a \in B_{k+1} \setminus B_k$ or $b \in B_{k+1} \setminus B_k$. In order to make a choice we suppose $a \in B_{k+1} \setminus B_k$. Two cases are possible:

Case 1: $b \in B_{k+1} \setminus B_k$

Case 2: $b \in B_k$

We observe that either in case 1 or in case 2 we have (2.12). It remains to prove (2.13). In the first case we have

$a \in L_1 \cap (B_{k+1} \setminus B_k) \subseteq L_2 \cap (B_{k+1} \setminus B_k)$

$b \in L_1 \cap (B_{k+1} \setminus B_k) \subseteq L_2 \cap (B_{k+1} \setminus B_k)$

therefore by the inductive assumption we have $f(a) = g(a)$ and $f(b) = g(b)$. We have $(a, b) \in dom(\sigma_{L_1})$ and from (2.5) we obtain $(f(a), f(b)) \in dom(u)$. But $dom(u) \subseteq dom(v)$ and $a \in L_2$, $b \in L_2$. Thus, $(g(a), g(b)) = (f(a), f(b)) \in dom(u) \subseteq dom(v)$ and by (2.6) we have (2.13).
In the second case we shall take into consideration the cases $b \in B_0 = L_0$ and $b \in B_q \setminus B_{q-1}$ for some $q \in \{1, \ldots, k\}$.

- If $b \in B_0$ then $f(b) = f_0(b) = g(b)$. We have $a \in L_1 \cap (B_{k+1} \setminus B_k)$. By the inductive assumption we have $L_1 \cap (B_{k+1} \setminus B_k) \subseteq L_2 \cap (B_{k+1} \setminus B_k)$ and

$f(a) = g(a)$. On the other hand, the mapping f is a morphism and $u \prec v$, therefore

$$f(x) = f(\sigma_{L_1}(a,b)) = u(f(a), f(b)) = u(g(a), g(b)) = v(g(a), g(b)) \quad (2.14)$$

We remark that $a \in L_2$, $b \in L_0 \subseteq L_2$ and $(g(a), g(b)) \in dom(v)$. Applying (2.6) we deduce (2.13).

- If $b \in B_q \setminus B_{q-1}$ for some $q \in \{1, \dots, k\}$ then

$$b \in L_1 \cap (B_q \setminus B_{q-1}) \subseteq L_2 \cap (B_q \setminus B_{q-1})$$
$$a \in L_1 \cap (B_{k+1} \setminus B_k) \subseteq L_2 \cap (B_{k+1} \setminus B_k)$$

and $f(a) = g(a)$, $f(b) = g(b)$ by the inductive assumption. We obtain (2.14), therefore $(g(a), g(b)) \in dom(v)$. From (2.6) we deduce (2.13).

Now, from (2.13) we deduce $\sigma_{L_2}(a,b) \in L_2$. But $\sigma_{L_2}(a,b) = \sigma(a,b) = x$, therefore $x \in L_2$ and the inclusion $L_1 \cap (B_{k+2} \setminus B_{k+1}) \subseteq L_2 \cap (B_{k+2} \setminus B_{k+1})$ is proved.
It remains to verify the equality $f(x) = g(x)$ for every $x \in L_1 \cap (B_{k+2} \setminus B_{k+1})$. We recall that as we proved above, if $x \in L_1 \cap (B_{k+2} \setminus B_{k+1})$ then there are $a, b \in L_0 \cup \bigcup_{n \in \{0,\dots,k\}} L_1 \cap (B_{n+1} \setminus B_n)$ such that $x = \sigma_{L_1}(a,b) = \sigma_{L_2}(a,b)$. Based on the inductive assumption we have $f(a) = g(a)$, $f(b) = g(b)$. Thus,

$$f(x) = f(\sigma_{L_1}(a,b)) = u(f(a), f(b)) = v(g(a), g(b)) = g(x)$$

and the relations (2.7) and (2.8) are proved by induction.
Using the relations (2.3), (3.1.1), (2.7) and (2.8) we obtain $L_1 \subseteq L_2$ and $f \prec g$. Applying the monotony of the mapping H_G and taking into account the equalities $T_1 = H_G(u)$ and $T_2 = H_G(v)$ we obtain $T_1 \subseteq T_2$. ∎

Corollary 2.2.1 *For each $u \in R(prod_S)$, the set $env_G(u)$ is a singleton. In other words, just one element belongs to $env_G(u)$.*

Proof. Take $v = u$ in the previous proposition. It follows that if $(L_1, T_1, f) \in env_G(u)$ and $(L_2, T_2, g) \in env_G(u)$ then $L_1 \subseteq L_2$, $T_1 \subseteq T_2$ and $f \prec g$. For the same reason we have also $L_2 \subseteq L_1$, $T_2 \subseteq T_1$ and $g \prec f$. Thus we have $L_2 = L_1$, $T_2 = T_1$ and $f = g$. ∎

Remark 2.2.1 *Corollary 2.2.1 allows to denote*

$$env_G(u) = (L, T, f)$$

instead of $env_G(u) = \{(L, T, f)\}$.

Thus the mapping defined in (2.1) becomes

$$env_G : R(prod_S) \longrightarrow Env(G)$$

where

$$Env(G) = \{env_G(u) \mid u \in R(prod_S)\} \quad (2.15)$$

Definition 2.2.1 *We define the following relation on* $Env(G)$*:*

$$(L_1, T_1, f) \ll (L_2, T_2, g) \text{ iff } \mathrm{L}_1 \subseteq \mathrm{L}_2, \mathrm{T}_1 \subseteq \mathrm{T}_2 \text{ and } \mathrm{f} \prec \mathrm{g}$$

We obtain a partially ordered set $(Env(G), \ll)$ because $\ll$ is reflexive, antisymmetric and transitive. In these terms, Proposition 2.2.2 can be restated as follows:

Proposition 2.2.3 *The mapping* $env_G : (R(prod_S), \prec) \longrightarrow (Env(G), \ll)$ *is monotone.*

Remark 2.2.2 *The mapping* $env_G : R(prod_S) \longrightarrow Env(G)$ *is surjective, but it is not injective.*

Really, by (2.15) env_G is surjective. It is not injective and this property can be proved as follows. We consider again the labelled graph $G = (S, L_0, T_0, f_0)$ which is represented in Figure 3.4. In Section 1.3 the following environment $env_G(u) = (L, T, f)$ was obtained:

$L = L_0 \cup \{\sigma(a, a), \sigma(b, b), \sigma(a, b), \sigma(b, a)\}$

$T = Cl_u(T_0) = T_0 \cup \{\rho_4\}$

$f_0 \prec f;\ f(\sigma(a, a)) = f(\sigma(a, b)) = f(\sigma(b, b)) = f(\sigma(b, a)) = \rho_4$

Let us choose now the mapping $v \in R(prod_S)$ defined by

$dom(v) = \{(\rho_1, \rho_1), (\rho_5, \rho_2)\}$, where $\rho_5 = \{(x_5, x_2)\}$

$v(\rho_1, \rho_1) = \rho_4$, $v(\rho_5, \rho_2) = \rho_6$, where $\rho_6 = \{(x_5, x_3)\}$

By computation we obtain $env_G(v) = env_G(u)$ and thus the mapping env_G is not injective. This is due to the fact that the domain of v includes the *useless symbol* ρ_5. Really, $\rho_5 \notin Cl_v(T_0)$ and thus this element is not used in the computation of $env_G(v)$. The useless symbols can be rejected by a method which is described in what follows.

Definition 2.2.2 *Let* $G = (S, L_0, T_0, f_0)$ *be a labelled graph. We define the operator*

$$\theta_G : R(prod_S) \longrightarrow R(prod_S)$$

taking $\theta_G(u) \prec u$ *such that* $dom(\theta_G(u)) = (Cl_u(T_0) \times Cl_u(T_0)) \cap dom(u)$*. We denote by* $MGE(G)$ *the image of the set* $R(prod_S)$ *by* θ_G*, that is,* $MGE(G) = \theta_G(R(prod_S))$*.*

The notation $MGE(G)$ is derived from the fact that its elements will be considered as *mappings generating environments.*

Directly from Definition 2.2.2 we express the mapping θ_G by the following two conditions:

$$\begin{cases} \theta_G(u) \prec u \\ dom(\theta_G(u)) = (H_G(u) \times H_G(u)) \cap dom(u) \end{cases} \tag{2.16}$$

Let us consider an example. We start with the sets S and T_0 of below:

$S = \{x_1, x_2, x_3, x_4\}$

$T_0 = \{\rho_1, \rho_2\}$, where $\rho_1 = \{(x_1, x_2), (x_2, x_1)\}$, $\rho_2 = \{(x_1, x_3), (x_2, x_3)\}$

We consider the mapping $u \in R(prod_S)$ defined in Table 2.1, where

$\rho_3 = \{(x_3, x_4)\}$, $\rho_4 = \{(x_1, x_1), (x_2, x_2)\}$, $\rho_5 = \{(x_1, x_4), (x_2, x_4)\}$

$dom(u) = \{(\rho_1, \rho_1), (\rho_1, \rho_2), (\rho_2, \rho_3), (\rho_1, \rho_4), (\rho_4, \rho_2)\}$

u	ρ_1	ρ_2	ρ_3	ρ_4
ρ_1	ρ_4	ρ_2		ρ_1
ρ_2			ρ_5	
ρ_4		ρ_2		

Table 2.1: The mapping u

Applying (1.4) we obtain

$X_0 = T_0$

$X_1 = T_0 \cup \{\rho_4\}$, $X_2 = X_1$

therefore $H_G(u) = Cl_u(T_0) = \{\rho_1, \rho_2, \rho_4\}$. Consequently, we have

$$(Cl_u(T_0) \times Cl_u(T_0)) \cap dom(u) = \{(\rho_1, \rho_1), (\rho_1, \rho_2), (\rho_1, \rho_4), (\rho_4, \rho_2)\}$$

and thus we obtain the mapping $\theta_G(u) \prec u$ represented in Table 2.2.

$\theta_G(u)$	ρ_1	ρ_2	ρ_4
ρ_1	ρ_4	ρ_2	ρ_1
ρ_2			
ρ_4		ρ_2	

Table 2.2: The mapping $\theta_G(u)$

Finally we shall remark that it suffices to consider the set $MGE(G)$ to obtain all *LSGs* over G. This will be shown in a next section (see Corollary 2.4.2), where we shall prove the following equality:

$$Env(G) = \{env_G(u) \mid u \in MGE(G)\}$$

If we corroborate this property with the bijectivity of the mapping env_G, that is, distinct mappings produce distinct environments (by the same corollary), then we can anticipate that in order to investigate the general properties of a *LSG*, $MGE(G)$ is a useful set of mappings. This explain why the next section is dedicated to the study of several algebraic properties of this set.

2.3 Algebraic properties for $MGE(G)$

Two main results concerning the set $MGE(G)$ will be proved in this section. The first result states that $MGE(G)$ is the set of all fixed points of the operator θ_G. This is not only an interesting result by itself, but it is also used to prove the final result of this section, which states that $MGE(G)$ is a join semilattice with greatest element (Proposition 2.3.6). This property is used in the next section.

In order to prove the first result, the following three properties of the operator θ_G are used.

Proposition 2.3.1 *$H_G(u) = H_G(\theta_G(u))$ for every $u \in R(prod_S)$.*

Proof. Because $\theta_G(u) \prec u$, by Proposition 2.2.1 we obtain $H_G(\theta_G(u)) \subseteq H_G(u)$. To prove the converse implication we observe first that from (1.4) and (2.2) we have $H_G(u) = \bigcup_{n\geq 0} X_n$. We verify by induction on n that

$$X_n \subseteq H_G(\theta_G(u)) \tag{2.17}$$

For $n = 0$ we have $X_0 = T_0 \subseteq H_G(\theta_G(u))$. Assume $X_n \subseteq H_G(\theta_G(u))$. Let $d \in X_{n+1}$. If $d \in X_n$ then $d \in H_G(\theta_G(u))$ by the inductive assumption. Otherwise there are $d_1, d_2 \in X_n$ such that $d = u(d_1, d_2)$. By the inductive assumption we have (2.17), therefore $d_1, d_2 \in H_G(\theta_G(u))$. We have also $(d_1, d_2) \in (X_n \times X_n) \cap dom(u)$ $\subseteq (H_G(u) \times H_G(u)) \cap dom(u) = dom(\theta_G(u))$. But $H_G(\theta_G(u))$ is closed under $\theta_G(u)$, therefore $\theta_G(u)(d_1, d_2) \in H_G(\theta_G(u))$. On the other hand $\theta_G(u) \prec u$, therefore $\theta_G(u)(d_1, d_2) = u(d_1, d_2)$. But $u(d_1, d_2) = d$ and $\theta_G(u)(d_1, d_2) \in H_G(\theta_G(u))$. Thus $d \in H_G(\theta_G(u))$ and (2.17) is proved by induction. From (2.17) we obtain $\bigcup_{n\geq 0} X_n \subseteq H_G(\theta_G(u))$, that is, $H_G(u) \subseteq H_G(\theta_G(u))$. ∎

Applying this proposition we deduce directly from (2.16) that θ_G satisfies also the following property

$$dom(\theta_G(u)) = (H_G(\theta_G(u)) \times H_G(\theta_G(u))) \cap dom(u)$$

therefore

$$dom(\theta_G(u)) \subseteq H_G(\theta_G(u)) \times H_G(\theta_G(u)) \tag{2.18}$$

Proposition 2.3.2 *The operator θ_G is idempotent. In other words, $\theta_G(\theta_G(u)) = \theta_G(u)$ for every $u \in R(prod_S)$.*

Proof. Using Definition 2.2.2 and Proposition 2.3.1 we obtain

$$dom(\theta_G(\theta_G(u))) = (H_G(\theta_G(u)) \times H_G(\theta_G(u))) \cap dom(\theta_G(u)) =$$

$$(H_G(\theta_G(u)) \times H_G(\theta_G(u))) \cap (H_G(u) \times H_G(u)) \cap dom(u) =$$

$$(H_G(u) \times H_G(u)) \cap dom(u) = dom(\theta_G(u))$$

But $\theta_G(\theta_G(u)) \prec \theta_G(u)$, therefore $\theta_G(\theta_G(u)) = \theta_G(u)$. ∎

Proposition 2.3.3

The operator $\theta_G : (R(prod_S), \prec) \longrightarrow (R(prod_S), \prec)$ *is monotone.*

Proof. Let be $u \prec v$. We have $dom(\theta_G(u)) = (H_G(u) \times H_G(u)) \cap dom(u)$, therefore by Proposition 2.2.1 we obtain

$$dom(\theta_G(u)) \subseteq (H_G(v) \times H_G(v)) \cap dom(u) \subseteq (H_G(v) \times H_G(v)) \cap dom(v)$$

because $dom(u) \subseteq dom(v)$. But $(H_G(v) \times H_G(v)) \cap dom(v) = dom(\theta_G(v))$. The relation $dom(\theta_G(u)) \subseteq dom(\theta_G(v))$ can be written equivalently $\theta_G(u) \prec \theta_G(v)$, that is, θ_G is monotone. ∎

We can now prove that $MGE(G)$ is the set of all fixed points of θ_G:

Proposition 2.3.4

$$MGE(G) = \{u \in R(prod_S) \mid \theta_G(u) = u\}$$

Proof. We observe that θ_G is an interior operator because it is the dual of a closure operator ($\theta_G(u) \prec u$, θ_G is monotone and idempotent). Now, the proof is immediate ([1], p.7). ∎

We define now a binary operation for two extensions of the same mapping, which allows to introduce a binary operation on $MGE(G)$ such that $MGE(G)$ becomes a join semilattice.

Definition 2.3.1 *Let* f *and* g *be two mappings such that* $f(x) = g(x)$ *for every* $x \in dom(f) \cap dom(g)$. *We define the mapping* $f \vee g$ *as follows:*

$$\begin{aligned} & dom(f \vee g) = dom(f) \cup dom(g) \\ & (f \vee g)(x) = \begin{cases} f(x) \text{ if } x \in dom(f) \\ g(x) \text{ if } x \in dom(g) \end{cases} \end{aligned} \tag{2.19}$$

We observe that the mapping $f \vee g$ is well defined because f and g have the same values on $dom(f) \cap dom(g)$.

Remark 2.3.1

1) *For every* $u \in R(prod_S)$ *and* $v \in R(prod_S)$ *the condition specified in Definition 2.3.1 is fulfilled and thus we may consider the mapping* $u \vee v$. *Indeed, for every* $(d_1, d_2) \in dom(u) \cap dom(v)$ *we have* $u(d_1, d_2) = prod_S(d_1, d_2) = v(d_1, d_2)$ *because* $u \prec prod_S$ *and* $v \prec prod_S$.

2) *Particularly we may consider the mapping* $\theta_G(u) \vee \theta_G(v)$.

It is not difficult to observe that for every $u, v \in R(prod_S)$ the following properties are equivalent:

$u \prec v$

$u \vee v = v$

We intend to prove now that $\vee$ is a binary operation on $MGE(G)$, that is, if $u, v \in MGE(G)$ then $u \vee v \in MGE(G)$. This is not an obvious property and in order to prove it we use the following property specified in the next proposition:

Proposition 2.3.5 *For every $u, v \in R(prod_S)$ we have*

$$\theta_G(u) \vee \theta_G(v) \in MGE(G)$$

Proof. By Proposition 2.3.4 we see that it is enough to prove that $\theta_G(u) \vee \theta_G(v)$ is a fixed point of θ_G, that is, $\theta_G(\theta_G(u) \vee \theta_G(v)) = \theta_G(u) \vee \theta_G(v)$. We have $\theta_G(\theta_G(u) \vee \theta_G(v)) \prec \theta_G(u) \vee \theta_G(v)$ and thus in order to prove the proposition it is enough to verify that $dom(\theta_G(\theta_G(u) \vee \theta_G(v))) = dom(\theta_G(u) \vee \theta_G(v))$. We have

$$dom(\theta_G(\theta_G(u) \vee \theta_G(v))) = (H_G(\theta_G(u) \vee \theta_G(v)) \times H_G(\theta_G(u) \vee \theta_G(v))) \cap dom(\theta_G(u) \vee \theta_G(v)) \quad (2.20)$$

From (2.18) and by monotony of H_G (Proposition 2.2.1) we have

$dom(\theta_G(u)) \subseteq H_G(\theta_G(u)) \times H_G(\theta_G(u)) \subseteq H_G(\theta_G(u) \vee \theta_G(v)) \times H_G(\theta_G(u) \vee \theta_G(v))$

$dom(\theta_G(v)) \subseteq H_G(\theta_G(v)) \times H_G(\theta_G(v)) \subseteq H_G(\theta_G(u) \vee \theta_G(v)) \times H_G(\theta_G(u) \vee \theta_G(v))$

therefore $dom(\theta_G(u) \vee \theta_G(v)) = dom(\theta_G(u)) \cup dom(\theta_G(v)) \subseteq H_G(\theta_G(u) \vee \theta_G(v)) \times H_G(\theta_G(u) \vee \theta_G(v))$. Using this inclusion in (2.20) we obtain

$$dom(\theta_G(\theta_G(u) \vee \theta_G(v))) = dom(\theta_G(u) \vee \theta_G(v))$$

■

Corollary 2.3.1 *The pair $(MGE(G), \vee)$ is an algebra, that is, for every $u, v \in MGE(G)$ the element $u \vee v$ is in $MGE(G)$.*

Proof. Immediate by Proposition 2.3.4 and Proposition 2.3.5. ■

Proposition 2.3.6 *$(MGE(G), \prec)$ is a join semilattice with greatest element. More precisely, for every $u, v \in MGE(G)$ there exists $sup\{u, v\}$, $sup\{u, v\} = u \vee v$ and $\theta_G(prod_S)$ is the greatest element of $MGE(G)$.*

Proof. We consider an arbitrary subset $\{u, v\} \subseteq MGE(G)$. We have $u \prec u \vee v$ and $v \prec u \vee v$. So the element $u \vee v$ is an upper bound for $\{u, v\}$ and by Corollary 2.3.1 the element $u \vee v$ belongs to $MGE(G)$. Let us show that $u \vee v$ is the least upper bound for $\{u, v\}$. To do this we consider an arbitrary upper bound $w \in MGE(G)$ for $\{u, v\}$. Thus, $u \prec w$ and $v \prec w$. By the definition of $\prec$ we have $dom(u) \subseteq dom(w)$ and $dom(v) \subseteq dom(w)$, therefore $dom(u \vee v) \subseteq dom(w)$. Consequently, $u \vee v \prec w$

and therefore $u \vee v$ is the least upper bound of $\{u, v\}$. Let us consider now an arbitrary element $u \in MGE(G)$. We have $u \prec prod_S$ and by Proposition 2.3.3 the mapping θ_G is monotone, therefore $\theta_G(u) \prec \theta_G(prod_S)$. But $\theta_G(u) = u$ by Proposition 2.3.4, therefore $u \prec \theta_G(prod_S)$. Because u is an arbitrary element in $MGE(G)$ and $\theta_G(prod_S) \in MGE(G)$, it results that $\theta_G(prod_S)$ is the greatest element of $MGE(G)$. ■

2.4 Algebraic properties for $Env(G)$

From Proposition 2.2.3 and Remark 2.2.2 we know that the mapping

$$env_G : (R(prod_S), \prec) \longrightarrow (Env(G), \ll)$$

is monotone and surjective, but it is not injective. At the beginning of this section we show that if we take

$$env_G : (MGE(G), \prec) \longrightarrow (Env(G), \ll)$$

then we obtain a bijective morphism of partially ordered sets, therefore we obtain an isomorphism.

Proposition 2.4.1 *Let be $u, v \in MGE(G)$. The following conditions are equivalent:*

i) $u \prec v$

ii) $env_G(u) \ll env_G(v)$

Proof. Implication $i) \Longrightarrow ii)$ is obtained by Proposition 2.2.3 because we have $MGE(G) \subseteq R(prod_S)$. Thus it remains to prove the implication $ii) \Longrightarrow i)$. Suppose $(L_1, T_1, f) = env_G(u)$ and $(L_2, T_2, g) = env_G(v)$. We have $H_G(u) = T_1$, $H_G(v) = T_2$, $f(L_1) = T_1$, $g(L_2) = T_2$. By Proposition 2.3.4 we have $\theta_G(u) = u$ and $\theta_G(v) = v$, therefore using (2.16) we have $dom(u) \subseteq T_1 \times T_1$ and $dom(v) \subseteq T_2 \times T_2$. Let be $(\rho_1, \rho_2) \in dom(u)$. There are $a, b \in L_1$ such that $f(a) = \rho_1$ and $f(b) = \rho_2$. Using (1.7) we deduce that $\sigma(a, b) \in L_1$. But $L_1 \subseteq L_2$ and $f \prec g$, therefore $\sigma(a, b) \in L_2$, $a \in L_2$, $b \in L_2$, $f(a) = g(a)$, $f(b) = g(b)$. Applying (1.7) for v, g and L_2, we deduce $(g(a), g(b)) \in dom(v)$, therefore $(\rho_1, \rho_2) \in dom(v)$. Thus we proved that $dom(u) \subseteq dom(v)$, that is, $u \prec v$. ■

Corollary 2.4.1 *The mapping $env_G : MGE(G) \longrightarrow Env(G)$ is injective.*

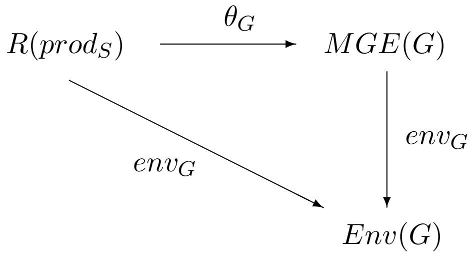
Proof. Suppose $env_G(u) = env_G(v)$. We have $env_G(u) \ll env_G(v)$ and $env_G(v) \ll env_G(u)$. Applying Proposition 2.4.1 we deduce $u \prec v$ and $v \prec u$. Thus $u = v$. ■

The next result can explain why the mapping

$$env_G : (R(prod_S), \prec) \longrightarrow (Env(G), \ll)$$

is not injective. The argument is that $u \in R(prod_S)$ and $\theta_G(u)$ generate the same environment. This property is stated in the next proposition. On the other hand, this result is used to prove the surjectivity of the mapping specified in Corollary 2.4.1.

Proposition 2.4.2 *The following diagram is commutative:*



that is, $env_G(u) = env_G(\theta_G(u))$ *for each* $u \in R(prod_S)$.

Proof. Take an arbitrary element $u \in R(prod_S)$. By Proposition 2.2.3 we have $env_G(\theta_G(u)) \ll env_G(u)$ because $\theta_G(u) \prec u$. It remains to verify that $env_G(u) \ll env_G(\theta_G(u))$. Let us denote $env_G(u) = (L_1, T_1, f)$ and $env_G(\theta_G(u)) = (L_2, T_2, g)$. Because $T_1 = H_G(u)$ and $T_2 = H_G(\theta_G(u))$, by Proposition 2.3.1 we have $T_1 = T_2$. We consider again the relations (2.3) and (3.1.1). Let us verify (2.7) and (2.8) by induction on n. If $\sigma(a,b) \in L_1 \cap (B_1 \setminus B_0)$ then $a, b \in L_0$ and $(f(a), f(b)) \in dom(u)$ by (1.7). Because $f(L_1) = T_1$, we have $(f(a), f(b)) \in dom(u) \cap (T_1 \times T_1) = dom(\theta_G(u))$. On the other hand, $f(a) = f_0(a) = g(a)$ and $f(b) = f_0(b) = g(b)$. Thus we have $(g(a), g(b)) \in dom(\theta_G(u))$, therefore by (1.7) we obtain $\sigma(a,b) \in L_2$. Moreover,

$$f(\sigma(a,b)) = u(f(a), f(b)) = prod_S(f_0(a), f_0(b))$$

$$g(\sigma(a,b)) = \theta_G(u)(g(a), g(b)) = prod_S(f_0(a), f_0(b))$$

that is, $f(\sigma(a,b)) = g(\sigma(a,b))$. Thus (2.7) and (2.8) are true for $n = 0$.
Suppose (2.7) and (2.8) are true for $n \leq k$. If $\sigma(x,y) \in L_1 \cap (B_{k+2} \setminus B_{k+1})$ then $x \in L_1 \cap (B_{s+1} \setminus B_s)$ and $y \in L_1 \cap (B_{p+1} \setminus B_p)$ for some $s \leq k$ and $p \leq k$. Applying the inductive assumption we have $x \in L_2 \cap (B_{s+1} \setminus B_s)$, $y \in L_2 \cap (B_{p+1} \setminus B_p)$ and $f(x) = g(x)$, $f(y) = g(y)$. Because $f(L_1) = T_1$, $x \in L_1$, $y \in L_1$ and $\sigma(x,y) \in L_1$, by (1.7) we have $(f(x), f(y)) \in dom(u) \cap (T_1 \times T_1) = dom(u) \cap (H_G(u) \times H_G(u)) = dom(\theta_G(u))$. In other words, we have $(g(x), g(y)) \in dom(\theta_G(u))$, where $x, y \in L_2$. Applying (1.7) we deduce $\sigma(x,y) \in L_2$. Obviously $f(\sigma(x,y)) = u(f(x), f(y)) = prod_S(f(x), f(y)) = prod_S(g(x), g(y)) = \theta_G(u)(g(x), g(y)) = g(\sigma(x,y))$. Thus (2.7) and (2.8) are true for every $n \geq 0$. Using (2.3) and (3.1.1) we deduce $L_1 \subseteq L_2$ and $f \prec g$. We recall that $T_1 = T_2$, therefore $env_G(u) \ll env_G(\theta_G(u))$. ■

Corollary 2.4.2 *The mapping* $env_G : (MGE(G), \prec) \longrightarrow (Env(G), \ll)$ *is bijective and monotone, therefore it is an isomorphism.*

Proof. The mapping env_G is injective by Corollary 2.4.1. To prove the surjectivity we observe that for each $(L, T, f) \in Env(G)$ there is $u \in R(prod_S)$ such that $env_G(u) = (L, T, f)$. But $\theta_G(u) \in MGE(G)$ and $env_G(u) = env_G(\theta_G(u))$ by Proposition 2.4.2. The monotony is obtained from Proposition 2.4.1. ∎

Taking into consideration Proposition 2.4.2 we deduce that

$$\{env_G(u) \mid u \in R(prod_S)\} = \{env_G(u) \mid u \in MGE(G)\}$$

therefore

$$Env(G) = \{env_G(u) \mid u \in MGE(G)\}$$

By Corollary 2.4.2 it follows that there are as many environments over G as the cardinal number of $MGE(G)$.

We can now prove the main result of this section, which is presented in the next proposition.

Proposition 2.4.3 *The pair $(Env(G), \ll)$ is a join semilattice with greatest element. More precisely,*

- $sup\{env_G(u), env_G(v)\} = env_G(u \vee v)$
- $env_G(\theta_G(prod_S))$ *is the greatest element of* $Env(G)$.

Proof. The mapping $env_G : MGE(G) \longrightarrow Env(G)$ is monotone, therefore env_G $(u \vee v)$ is an upper bound for $\{env_G(u), env_G(v)\}$. Moreover, $env_G(u \vee v)$ is the least upper bound for $\{env_G(u), env_G(v)\}$. Indeed, if $env_G(w)$ is also an upper bound for $\{env_G(u), env_G(v)\}$ then by Proposition 2.4.1 we have $u \prec w$ and $v \prec w$. By Proposition 2.3.6 it follows that $u \vee v \prec w$, therefore $env_G(u \vee v) \ll env(w)$. Now let us consider an arbitrary element $env_G(u) \in Env(G)$, where $u \in MGE(G)$. By Proposition 2.3.6 the greatest element of $MGE(G)$ is $\theta_G(prod_S)$. Thus, $u \prec \theta_G(prod_S)$ and therefore env_G $(u) \ll env_G(\theta_G(prod_S))$. This shows that $env_G(\theta_G(prod_S))$ is the greatest element in $Env(G)$. ∎

2.5 An equivalence relation on $Strat(G)$

Using the notations introduced in the previous sections we can write

$$Strat(G) = \{(G, L, T, u, f) \mid u \in R(prod_S), env_G(u) = (L, T, f)\}$$

We can define

$$lsg : R(prod_S) \longrightarrow Strat(G)$$

by $lsg(u) = (G, L, T, u, f)$, where $(L, T, f) = env_G(u)$.

Obviously, the mapping lsg is surjective and thus

$$Strat(G) = \{lsg(u) \mid u \in R(prod_S)\} \tag{2.21}$$

Definition 2.5.1 *For every* $u, v \in R(prod_S)$ *we define*

$$lsg(u) \simeq lsg(v) \text{ iff } env_G(u) = env_G(v)$$

Obviously, $\simeq$ is reflexive, symmetric and transitive, therefore it is an equivalence relation. The factor set is denoted by $Strat(G)/_{\simeq}$. We denote by $[lsg(u)]$ the equivalence class defined by $lsg(u)$.

We introduce on $Strat(G)/_{\simeq}$ the following relation:

Definition 2.5.2

$$[lsg(u)] \sqsubseteq [lsg(v)] \text{ iff } \theta_G(u) \prec \theta_G(v) \tag{2.22}$$

Because the relation (2.22) is defined in terms of representatives, the next property gives the answer to a natural question:

Proposition 2.5.1 *If* $lsg(u) \simeq lsg(u_1)$ *then* $\theta_G(u) = \theta_G(u_1)$. *Consequently, the relation* $\sqsubseteq$ *defined in (2.22) does not depend on representatives.*

Proof. If $lsg(u) \simeq lsg(u_1)$ then $env_G(u) = env_G(u_1)$, therefore by Proposition 2.4.2 we have $env_G(\theta_G(u)) = env_G(\theta_G(u_1))$. But $\theta_G(u)$ and $\theta_G(u_1)$ belong to $MGE(G)$ and by Corollary 2.4.1 the mapping $env_G : MGE(G) \longrightarrow Env(G)$ is injective. Thus, $\theta_G(u) = \theta_G(u_1)$. ■

Proposition 2.5.2 $(Strat(G)/_{\simeq}, \sqsubseteq)$ *is a partially ordered set.*

Proof. The relation $\sqsubseteq$ is reflexive, antisymmetric and transitive. The reflexivity and transitivity are verified immediately and the antisymmetry can be proved as follows. Suppose $[lsg(u)] \sqsubseteq [lsg(v)]$ and $[lsg(v)] \sqsubseteq [lsg(u)]$. We have $\theta_G(u) \prec \theta_G(v)$ and $\theta_G(v) \prec \theta_G(u)$, therefore $\theta_G(u) = \theta_G(v)$. It follows that $env_G(\theta_G(u)) = env(\theta_G(v))$ and thus by Proposition 2.4.2 we have $env_G(u) = env_G(v)$, that is, $[lsg(u)] = [lsg(v)]$. ■

2.6 $Strat(G)/_{\simeq}$ is a join semilattice

We observe that if $u, v \in R(prod_S)$ then

$$[lsg(u)] \sqsubseteq [lsg(v)] \text{ iff } env_G(u) \ll env_G(v) \tag{2.23}$$

Really, this property is obtained immediately because the following relations are equivalent by Proposition 2.4.1 and Proposition 2.4.2:

$$env_G(u) \ll env_G(v)$$

$$env_G(\theta_G(u)) \ll env_G(\theta_G(v))$$

$$\theta_G(u) \prec \theta_G(v)$$

We define now the mapping

$$\mathcal{E} : Env(G) \longrightarrow Strat(G)/\simeq$$

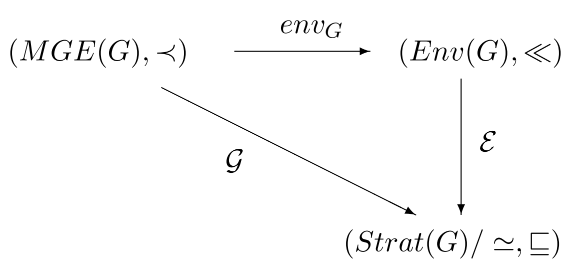
by $\mathcal{E}(env_G(u)) = [lsg(u)]$ for each $u \in R(prod_S)$.

Proposition 2.6.1 *The mapping* $\mathcal{E} : (Env(G), \ll) \longrightarrow (Strat(G)/\simeq, \sqsubseteq)$ *is an isomorphism.*

Proof. By (2.23) the mapping $\mathcal{E}$ is a morphism of partially ordered sets. Obviously $\mathcal{E}$ is surjective. If $\mathcal{E}(env_G(u)) = \mathcal{E}(env_G(v))$ then $lsg(u) \simeq lsg(v)$, therefore $env_G(u) = env_G(v)$. Thus $\mathcal{E}$ is injective. ∎

Proposition 2.6.2 *The mapping* $\mathcal{G} : (MGE(G), \prec) \longrightarrow (Strat(G)/\simeq, \sqsubseteq)$*, defined by* $\mathcal{G}(u) = [lsg(u)]$*, is an isomorphism.*

Proof. Immediate, if we use the diagram



This diagram is commutative and env_G and $\mathcal{E}$ are isomorphisms. ∎

In the next proposition we characterize $Strat(G)/_{\simeq}$ as a join semilattice.

Proposition 2.6.3 *The pair* $(Strat(G)/_{\simeq}, \sqsubseteq)$ *is a join semilattice with greatest element. More precisely,*

1) *If* $\mathcal{G}(u) \in Strat(G)/_{\simeq}$ *and* $\mathcal{G}(v) \in Strat(G)/_{\simeq}$ *are two arbitrary elements then there exists* $sup\{\mathcal{G}(u), \mathcal{G}(v)\} \in Strat(G)/_{\simeq}$ *and*

$$sup\{\mathcal{G}(u), \mathcal{G}(v)\} = \mathcal{G}(u \vee v)$$

2) $\mathcal{G}(\theta_G(prod_S))$ *is the greatest element in* $(Strat(G)/_{\simeq}, \sqsubseteq)$.

Proof. Immediate by Proposition 2.3.6 and Proposition 2.6.2. ∎

2.7 Distinguished representatives

In the previous sections we introduced some partial order between equivalence classes such that there exists the greatest equivalence class. In this section we specify some property for the elements of an equivalence class. This is stated in the next definition and afterwards we show that for each equivalence class just one element satisfies this property. In this way each equivalence class contains a "special" representative, which is uniquely determined.

Definition 2.7.1 *Let be $u \in R(prod_S)$. We consider the set*

$$S_u = \{v \in R(prod_S) \mid lsg(v) \simeq lsg(u)\}$$

The element $lsg(u_0)$ is called **the distinguished representative** *of $[lsg(u)]$ if u_0 is the least element of $(S_u, \prec)$.*

The distinguished representative of an equivalence class is uniquely determined by the uniqueness of the least element (if any) of the set $(S_u, \prec)$. We shall develop this idea in what follows but before this, we remark the following fact. Each equivalence class of $Strat(G)/_{\simeq}$ and $Strat(G)$ itself, is partially ordered by the relation defined in the following definition:

Definition 2.7.2 *For every $u, v \in R(prod_S)$ we define*

$$lsg(u) \leq lsg(v) \text{ iff } u \prec v$$

In view of this definition one might say that the distinguished representative of an equivalence class is the least element of the corresponding class.

The next proposition is devoted to the existence of the distinguished representative.

Proposition 2.7.1 *For each $u \in R(prod_S)$ there exists the distinguished representative of the class $[lsg(u)]$ and moreover, this is the element $lsg(\theta_G(u))$.*

Proof. The element $\theta_G(u)$ belongs to S_u and it is the least element of S_u. Really, if $v \in S_u$ is an arbitrary element then

- $env_G(u) = env_G(v)$, therefore $env_G(\theta_G(u)) = env_G(\theta_G(v))$
- $\theta_G(u), \theta_G(v) \in MGE(G)$

Applying Corollary 2.4.1 we obtain $\theta_G(u) = \theta_G(v)$. But $\theta_G(v) \prec v$, therefore $\theta_G(u) \prec v$. Thus $\theta_G(u)$ is the least element of S_u. ■

Corollary 2.7.1 *For each $u \in MGE(G)$ the element $lsg(u)$ is the distinguished representative of $\mathcal{G}(u)$. Particularly, $lsg(\theta_G(prod_S))$ is the distinguished representative of $\mathcal{G}(\theta_G(prod_S))$.*

Proof. In view of Proposition 2.7.1, if $u \in MGE(G)$ then $lsg(\theta_G(u))$ is the distinguished representative of $\mathcal{G}(u)$. But for $u \in MGE(G)$ we have shown that $\theta_G(u) = u$, therefore $lsg(\theta_G(u)) = lsg(u)$. ■

This result allows us to denote

$$DR(u) = lsg(\theta_G(u))$$

for each $u \in R(prod_S)$. In other words, $DR(u)$ is the distinguished representative of the equivalence class defined by $u \in R(prod_S)$.

Proposition 2.7.2 *For each $u, v \in R(prod_S)$ the following conditions are equivalent:*

$[lsg(u)] \sqsubseteq [lsg(v)]$

$DR(u) \leq DR(v)$

Proof. The following conditions are equivalent:

$[lsg(u)] \sqsubseteq [lsg(v)]$

$\theta_G(u) \prec \theta_G(v)$

$lsg(\theta_G(u)) \leq lsg(\theta_G(v))$

$DR(u) \leq DR(v)$

■

Because $\mathcal{G}(\theta_G(prod_S))$ is the greatest element of $(Strat(G)/_{\simeq}, \sqsubseteq)$, it follows that for every $u \in R(prod_S)$ we have

$$[lsg(u)] \sqsubseteq \mathcal{G}(\theta_G(prod_S)) = [lsg(\theta_G(prod_S))]$$

In virtue of Proposition 2.7.2 we deduce that

$$DR(u) \leq DR(\theta_G(prod_S))$$

for every $u \in R(prod_S)$. In other words, $DR(\theta_G(prod_S))$ is the greatest element of the set $\{DR(u) \mid u \in R(prod_S)\}$. For this reason, the element $DR(\theta_G(prod_S))$ is called *the greatest distinguished LSG over G* and it is denoted by GD.

2.8 Conclusions and open problems

In this chapter we presented several algebraic properties of labelled stratified graphs. Thus, we introduced a set $MGE(G)$ of mappings which generate in a bijective manner all the labelled stratified graphs over some labelled graph G. We introduced an equivalence relation $\simeq$ on $Strat(G)$ and we organized the factor set $Strat(G)/\simeq$ as a join semilattice with greatest element.

We defined a partial order $\leq$ on the set $Strat(G)$. For each equivalence class $C \in Strat(G)/\simeq$ there is a distinguished representative and this is the least element of C with respect to $\leq$. Particularly, this property is true for the greatest element of $Strat(G)/\simeq$ and the corresponding representative is named the greatest distinguished LSG over G.

In conclusion, the set $Strat(G)$ is first divided into equivalence classes as in Figure 2.1, where by EC we denoted an equivalence class. The elements of a class are grouped by the fact that they have the same environment. The partial order $\sqsubseteq$ is drawn by a double arrow and it is given by the order between environments.

On the highest level it is drawn the greatest equivalence class GEC. The inner side of an equivalence class is shown in Figure 2.2. The elements of an equivalence class EC are partially ordered by $\leq$, which is represented by a single arrow. On the lowest level of each class EC we find its distinguished element DE, which is the least element of EC.

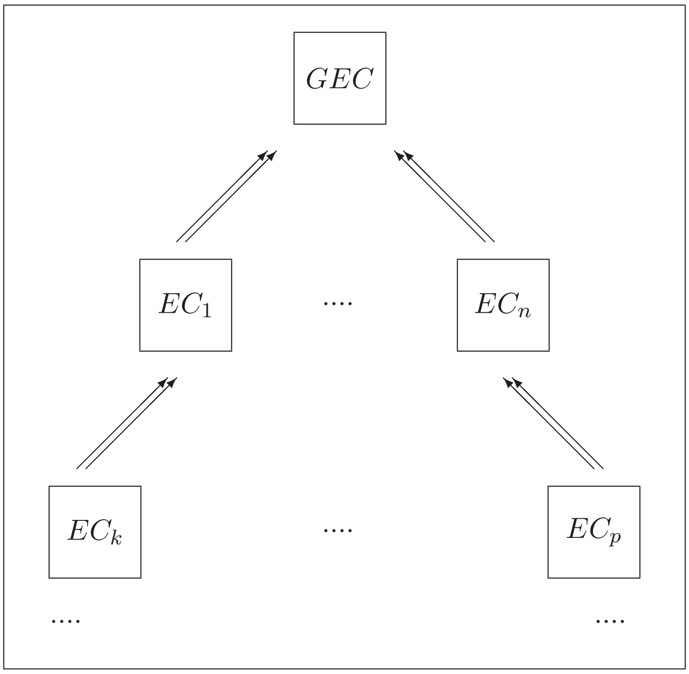


Figure 2.1: The factor set $Strat(G)/\simeq$

Several questions with respect to the subject treated in this chapter are left as open problems. We are interested in developing the following aspects:

- Find a method to extract automatically the objects and the relations specified in a knowledge piece.
- Apply some methods of graph drawing to draw the attached labelled graph in such a way that its objects and relations are easy to read and understand. As a model of such work can be taken the project of the German Science Foundation ([7]).
- Define $inf\{u, v\}$, where $u, v \in MGE(G)$; use this definition to find the greatest lower bound of two labelled stratified graphs, if any; identify an application of this concept.
- Consider $env_G(u)$ and $env_G(v)$ for $u, v \in MGE(G)$. Evaluate the components of $env_G(u \vee v)$ by means of the components of $env_G(u)$ and $env_G(v)$. For

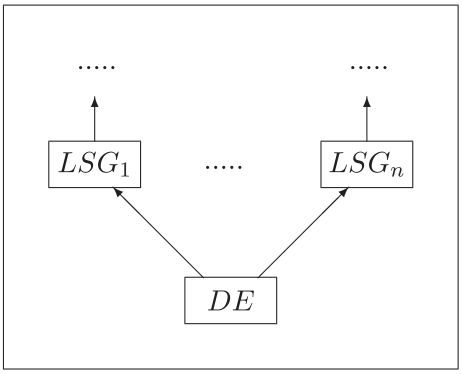


Figure 2.2: An equivalence class

example, try to prove $H_G(u \vee v) = H_G(u) \cup H_G(v)$ and/or other appropriate relations.

- Apply the concept of LSG to model an inference process.

Chapter 3

Applications of LSG

3.1 Knowledge bases with output

3.1.1 Introduction

This section is devoted to *knowledge base with output* (KBO), which are based on labelled stratified graphs. For such a knowledge base we distinguish *a structure* and *two kinds of computations.* The structure can be described by the equation $KBO = LSG{+}OS$, that is a knowledge base with output is a *labelled stratified graph* K and an *output space* Y. The connection of these two components is accomplished by means of some mapping, which is named *the output function.* The output space Y is endowed with an algebraic operation and K is obtained taking into consideration a labelled graph G. The structural aspect of a KBO is obtained by specifying the elements K, Y, the output function and the algebraic operation of Y. The structure of the labelled stratified graph K allows us to obtain the partial algebra $Tree_{us}(K)$. Using the output function an *algebraic morphism* from $Tree_{us}(K)$ to Y is obtained.

The computations performed in a KBO are represented in Figure 3.1 by two levels: *the syntactic level* and *the semantic level.*

An *interrogation* or a *query* is obtained by specifying an ordered pair of objects. This pair is taken into consideration by the syntactic level, the information included in K is used and consequently some subset of $Tree_{us}(K)$ is obtained. The *answer*, i.e. the result of the semantic computation, is the image of this set by the algebraic morphism.

All the properties specified in this section are proved in a separate section. Several examples are taken in order to specify the main aspects of the method.

This section is organized as follows: in Section 3.1.2 we introduce the concept of knowledge base with output; Section 3.1.3 and Section 3.1.4 contain the descriptions of the syntactic, respectively semantic computations in a KBO; in Section 3.1.6 we present an application in travel scheduling; finally, Section 3.1.5 contains the proofs of all the properties specified in Section 3.1.3.

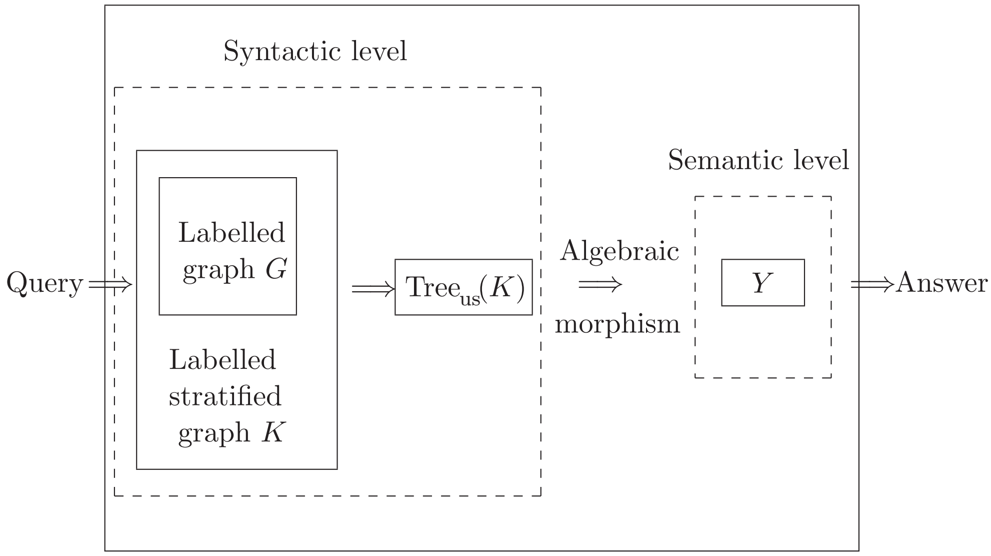


Figure 3.1: Computational levels in a KBO

3.1.2 The structure of a KBO

In what follows we introduce the concept of *knowledge base with output*. We consider a labelled graph $G = (S, L_0, T_0, f_0)$. We denote

$$Elem(G) = \{(x, r, y) \in S \times L_0 \times S \mid (x, y) \in f(r)\}$$

and their elements are called **the basic elements** of G.

Definition 3.1.1 *Let* $K = (G, L, T, u, f)$ *be a labelled stratified graph over* G. *Let* Y *be an arbitrary nonempty set and* $* : Y \times Y \longrightarrow Y$ *an algebraic operation. Let* $g : Elem(G) \longrightarrow Y$ *be an arbitrary function. The system* $(K, Y, g, *)$ *is called a* **knowledge base with output (KBO)** *and* g *is called the* **output function**. *For every* $u \in Elem(G)$ *the element* $g(u)$ *is called the* **output element** *corresponding to the basic element* u.

We observe that *the structure* of a KBO is given by the following three entities:

1) a labelled stratified graph $\mathcal{G}$

2) an algebra $(Y, *)$

3) a link between the above entities, that is the output function

Two kinds of computations can be accomplished in a KBO. They are named *syntactic computations* and *semantic computations*. They must be performed in this order.

The final result of a syntactic computation is taken into consideration to obtain a semantic computation and the result of such computation is an element of Y. Let us consider a pair $(x, y) \in S \times S$. If there is $a \in L_0$ such that $(x, a, y) \in Elem(G)$ then $g(x, a, y)$ is a *meaning* assigned to (x, y). This is the simplest computation. More complex computations are accomplished if there is a *path* from x to y in G. For this reason we shall specify in the next definition the concept of path and several notations.

Definition 3.1.2 *Let G be a labelled graph. Let be $x, y \in S$ and $n \geq 1$. A* **path** *(of length n) from x to y in G is a pair $([x_1, x_2, \ldots, x_{n+1}], [a_1, a_2, \ldots, a_n])$, where $x_1 = x$, $x_{n+1} = y$ and $(x_i, a_i, x_{i+1}) \in Elem(G)$ for every $i \in \{1, \ldots, n\}$. For a given $n \geq 1$ we consider the set $Path_n(x, y)$ of all the paths of length n from x to y and we denote*

$$P_G(x, y) = \bigcup_{n \geq 1} Path_n(x, y); \; PATH(G) = \bigcup_{(x,y) \in S \times S} P_G(x, y)$$

3.1.3 Syntactic computations in KBOs

At the beginning of this section we shall give the intuitive meaning for the concept of *syntactic computation in K*. The result of this computation is represented in Figure 3.1 by an element of $Tree_{us}(K)$.

Let be $(x, y) \in S \times S$. A syntactic computation for (x, y) is accomplished as follows. We consider a path $p = ([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$ in G such that $x_1 = x$ and $x_{n+1} = y$. We denote by $d_1(p)$ the following sequence of basic elements

$$d_1(p) : z_1, \ldots, z_n \tag{3.1}$$

where $z_j = (x_j, a_j, x_{j+1})$ for $j = 1, \ldots, n$. We choose $i \in \{1, \ldots, n-1\}$ such that $(a_i, a_{i+1}) \in dom(\sigma_L)$. The subsequence z_i, z_{i+1} of the sequence (3.1) is replaced by the element $(x_i, \sigma_L(a_i, a_{i+1}), x_{i+2})$ and we obtain the following sequence $d_2(p)$ containing $n-1$ elements:

$$d_2(p) : z_1, \ldots, z_{i-1}, (x_i, \sigma_L(a_i, a_{i+1}), x_{i+2}), z_{i+2}, \ldots, z_n \tag{3.2}$$

We repeat step by step the replacement process and finally we obtain a sequence $d_n(p)$ of the form

$$d_n(p) : (x_1, u, x_{n+1}) \tag{3.3}$$

for some $u \in L$. The element $d_n(p)$ is a result of the syntactic computation. Taking all the elements $p \in P_G(x, y)$ and all the possible replacements of two consecutive elements in $d_r(p)$ for $r \geq 1$ we obtain all the syntactic computations for the pair (x, y).

Some elements $p \in P_G(x, y)$ can be *useless* paths. They are those paths that can not give any syntactic computation. The problem is to identify these paths.

In order to give a formal description of this computation we shall represent the steps (3.1),(3.2), ..., by means of some labelled tree. The following notations will be used in what follows. If t is a labelled tree then **root(t)** denotes the node which is the root of t and *label(s)* denotes the label of the node s. For every node s, $label(s)$ will be an element of $S \times L \times S$. For $i < j$ we denote by $CON_{l=i}^{j}(w_l)$ the concatenation of the symbols $w_i, w_{i+1}, \ldots, w_j$. If $s_1, \ldots, s_n$ are all terminal nodes of the labelled tree t from left to right in this order and $label(s_1) = w_1$, …, $label(s_n) = w_n$ then $CON_{l=1}^{n}(w_l)$ is *the frontier* of t and it is denoted by $front(t)$.

The next definition will show the manner in which a syntactic computation can be represented by means of a labelled tree.

Definition 3.1.3 *Let* $([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$ *be a path in* G. *We define the set* $T([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$ *as follows:*

1. *for* $n = 1$, $T([x_1, x_2], [a_1])$ *contains only one tree, consisting in a single node. This is the root of the tree and it is labelled by* (x_1, a_1, x_2).

2. *for* $n \geq 2$, *the labelled tree* t *is an element of* $T([x_1, \ldots x_{n+1}], [a_1, \ldots, a_n])$ *if and only if the following conditions are satisfied:*
   - $front(t) = CON_{i=1}^{n}((x_i, a_i, x_{i+1}))$.
   - *every node* s *of* t, *which is not a terminal node, has two direct descendants: the left descendant* s_l *and the right descendant* s_r. *If* $label(s) = (x, r, y)$ *then the following conditions are fulfilled:*

     (a) there are $u, v \in L$, $z \in S$ *such that* $label(s_l) = (x, u, z)$, $label(s_r) = (z, v, y)$

     (b) $(u, v) \in dom(\sigma_L)$ *and* $r = \sigma_L(u, v)$

The process by means of which we obtain an element of $T(p)$ *is a* **syntactic computation** *assigned to* $p \in PATH(G)$. *If* $T(p) = \emptyset$ *then* p *is named* **useless path**.

We are interested to solve the following two problems:

(P1) identify the useless paths in G; consequently, we find the pairs from $S \times S$ for which we can accomplish the corresponding syntactic computations

(P2) establish the properties connecting the components of $label(root(t))$ for each $t \in T(p)$, where p is not an useless path in G

In order to solve these problems we introduce the following notation. The arity of a mapping will be denoted as a superscript of the mapping symbol. If $h^{(2)} \prec prod_S$ then for every $n \geq 3$ we can define $h^{(n)}$ recursively as follows:

$$(\rho_1, \rho_2, \ldots, \rho_n) \in dom(h^{(n)}) \text{ and } h^{(n)}(\rho_1, \rho_2, \ldots, \rho_n) = \rho$$

if and only if at least one of the following conditions is fulfilled:

1) $(\rho_1, h^{(n-1)}(\rho_2, \ldots, \rho_n)) \in dom(h^{(2)})$, $\rho = h^{(2)}(\rho_1, h^{(n-1)}(\rho_2, \ldots, \rho_n))$

2) $(h^{(n-1)}(\rho_1, \ldots, \rho_{n-1}), \rho_n) \in dom(h^{(2)})$, $\rho = h^{(2)}(h^{(n-1)}(\rho_1, \ldots, \rho_{n-1}), \rho_n)$

3) there is $k \in \{2, \ldots, n-2\}$ such that

3.1) $(\rho_1, \ldots, \rho_k) \in dom(h^{(k)})$, $(\rho_{k+1}, \ldots, \rho_n) \in dom(h^{(n-k)})$

3.2) $(h^{(k)}(\rho_1, \ldots, \rho_k), h^{(n-k)}(\rho_{k+1}, \ldots, \rho_n)) \in dom(h^{(2)})$

3.3) $\rho = h^{(2)}(h^{(k)}(\rho_1, \ldots, \rho_k), h^{(n-k)}(\rho_{k+1}, \ldots, \rho_n))$

The above definition seems to be trivial because it is known that $prod_S$ is an associative binary operation. But sometimes h is not an associative operation as we can see in the following example.

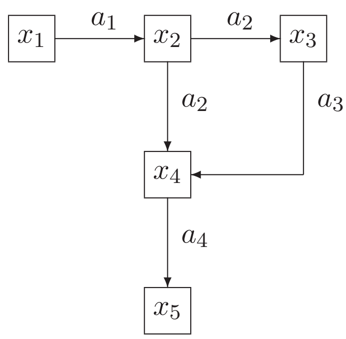


Figure 3.2: The labelled graph for Example 3.1.1

Example 3.1.1 *Let be* $S = \{x_1, x_2, x_3, x_4, x_5\}$. *We take* $\rho_1 = \{(x_1, x_2)\}$; $\rho_2 = \{(x_2, x_3), (x_2, x_4)\}$; $\rho_3 = \{(x_3, x_4)\}$; $\rho_4 = \{(x_4, x_5)\}$. *We suppose* $T_0 = \{\rho_1, \rho_2, \rho_3, \rho_4\}$. *Let be* $T = \{\rho_i\}_{i \in \{1,\ldots,7\}}$, *where* $\rho_5 = \{(x_2, x_4)\}$; $\rho_6 = \{(x_2, x_5)\}$; $\rho_7 = \{(x_1, x_5)\}$. *Let us take* $L_0 = \{a_1, a_2, a_3, a_4\}$ *and* $f_0 : L_0 \longrightarrow T_0$, $f_0(a_i) = \rho_i$ *for* $i \in \{1, \ldots, 4\}$. *The tuple* $G = (S, L_0, T_0, f_0)$ *is a labelled graph and it is drawn in Figure 3.2. Let us consider the mapping* $u \prec prod_S$, *which is defined as follows:*

$$u(\rho_2, \rho_3) = \rho_5; u(\rho_5, \rho_4) = \rho_6; u(\rho_1, \rho_6) = \rho_7$$

Obviously $Cl_u(T_0) = T$ *and* $\mathcal{A}_T = (T, u)$ *becomes a partial algebra. Building a labelled stratified graph over* G *we obtain successively:*

$D_0 = L_0 = \{a_1, a_2, a_3, a_4\}$

$D_1 = \{\sigma(a_2, a_3)\}$, $f_1(\sigma(a_2, a_3)) = \rho_5$

$D_2 = \{\sigma(\sigma(a_2, a_3), a_4)\}$, $f_2(\sigma(\sigma(a_2, a_3), a_4)) = \rho_6$

$D_3 = \{\sigma(a_1, \sigma(\sigma(a_2, a_3), a_4))\}$, $f_3(\sigma(a_1, \sigma(\sigma(a_2, a_3), a_4))) = \rho_7$

Now we obtain the labelled stratified graph $\mathcal{G} = (G, L, T, u, f)$*, where* $L = \{a_1, a_2, a_3, a_4, m_1, m_2, m\}$*,* $m_1 = \sigma(a_2, a_3)$*,* $m_2 = \sigma(m_1, a_4)$*,* $m = \sigma(a_1, m_2)$*,* $f(a_i) = \rho_i$ *for* $i \in \{1, 2, 3, 4\}$*,* $f(m_1) = \rho_5$*,* $f(m_2) = \rho_6$ *and* $f(m) = \rho_7$*.*
We observe that the operation u *is not an associative one although* $u \prec prod_S$ *and* $prod_S$ *is associative. Really,* $u(\rho_2, u(\rho_3, \rho_4))$ *is not defined, whereas* $u(u(\rho_2, \rho_3), \rho_4)) = \rho_6$*. It follows that* $u^{(3)}$ *is defined in* (ρ_2, ρ_3, ρ_4) *and* $u^{(3)}(\rho_2, \rho_3, \rho_4) = \rho_6$*.*

Let $K = (G, L, T, u, f)$ be a labelled stratified graph over $G = (S, L_0, T_0, f_0)$. We consider an arbitrary element $p = ([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n]) \in PATH(G)$. The answers for the problems (P1) and (P2) are the following and their proofs are included in Section 3.1.5:

(1) p is useless iff $(f(a_1), \ldots, f(a_n)) \notin dom(u^{(n)})$ (Corollary 3.1.3)

(2) if $t \in T(p)$ then there is $r \in L$, uniquely determined, such that

$label(root(t)) = (x_1, r, x_{n+1})$,
$(x_1, x_{n+1}) \in f(r)$,
$f(r) = u^{(n)}(f(a_1), \ldots, f(a_n))$ (Proposition 3.1.3 and Proposition 3.1.4)

(3) if $a_1, \ldots, a_n \in L_0$ are arbitrary elements such that $(f(a_1), \ldots, f(a_n)) \in dom(u^{(n)})$ then for every $(x, y) \in u^{(n)}(f(a_1), \ldots, f(a_n))$ there is a labelled tree t such that $label(root(t)) = (x, r, y)$ for some $r \in L$ (Proposition 3.1.5)

In other words, each syntactic computation assigned to a path whose arc sequence is $a_1, \ldots, a_n$ will produce an element of the form (x_1, r, x_{n+1}), where r is a label of the same binary relation, namely $u^{(n)}(f(a_1), \ldots, f(a_n))$, the first component is the initial node and the last component is the final node of the corresponding path. Moreover, all the elements of $u(f(a_1), \ldots, f(a_n))$ are used in the syntactic computations. Equivalently we can say that the set of all elements from $S \times S$ for which we obtain syntactic computations is exactly the set

$$\bigcup_{n \geq 2} \bigcup_{a_1, \ldots, a_n \in L_0} u^{(n)}(f(a_1), \ldots, f(a_n))$$

Example 3.1.2 *Let us consider the path* $([x_1, x_2, x_3, x_4, x_5], [a_1, a_2, a_3, a_4])$ *in* G*, where* G *is defined in Figure 3.2. We have*

$$u^{(4)}(f(a_1), \ldots, f(a_4)) = u^{(2)}(\rho_1, u^{(2)}(u^{(2)}(\rho_2, \rho_3), \rho_4)) = \rho_7$$

therefore by Proposition 3.1.6 we have $T([x_1, \ldots, x_5], [a_1, \ldots, a_4]) \neq \emptyset$*. It is easy to show that* $T([x_1, \ldots, x_5], [a_1, \ldots, a_4])$ *contains only one element, that is the tree which is represented in Figure 3.3.*

We observe that

$$T([x_1, x_2, x_4, x_5], [a_1, a_2, a_4]) = \emptyset$$

although the pair $([x_1, x_2, x_4, x_5], [a_1, a_2, a_4])$ *is a path in the corresponding labelled graph. This is explained by the fact that* $u^{(3)}(f(a_1), f(a_2), f(a_4)) = \emptyset$*.*

On the other hand we have

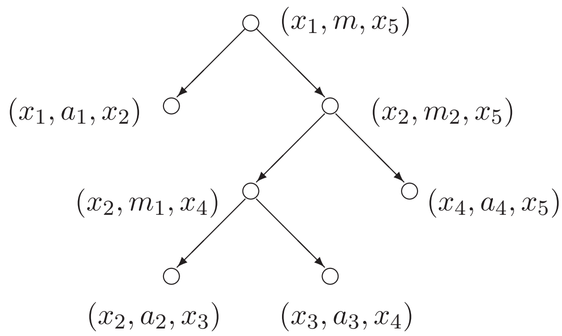


Figure 3.3: The unique element of $T([x_1, \ldots, x_5], [a_1, \ldots, a_4])$

$u^{(3)}(f(a_2), f(a_3), f(a_4)) = u^{(2)}(u^{(2)}(f(a_2), f(a_3)), f(a_4)) =$
$f(m_2) = \rho_6 = \{(x_2, x_5)\}$

Thus the syntactic structure of $m_2 = \sigma(\sigma(a_2, a_3), a_4)$ *gives us the computation described in Figure 3.3 by the subtree having* (x_2, m_2, x_5) *as root label.*

In conclusion, only for the paths $([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$ *for which*

$$(f(a_1), \ldots, f(a_n)) \in dom(u^{(n)})$$

we can accomplish syntactic computations.

We consider a knowledge base with output $KBO = (K, Y, g, *)$, where $K = (G, L, T, u, f)$. Frequently some restrictions concerning the connections between the objects are specified in a knowledge piece. These restrictions can appear in various manners and they are finally imposed on the set L. For this reason only a subset L_{us} of L will be used in the deduction process. Such restrictions can be seen in the application described in Section 3.1.6 of this chapter.

Let be L_{us} such that $L_0 \subseteq L_{us} \subseteq L$. The elements of L_{us} are called *useful labels.* We define

$$TREE(K) = \bigcup_{p \in PATH(G)} T(p)$$

Let us consider $t_1, t_2 \in TREE(K)$. We write $(t_1, t_2) \in dom(\sigma_K)$ iff the following conditions are fulfilled:

1) $label(root(t_1)) = (x, r_1, z)$ and $label(root(t_2)) = (z, r_2, y)$ for some nodes x, y, z and $r_1, r_2 \in L$

2) $(r_1, r_2) \in dom(\sigma_L)$ and $\sigma_L(r_1, r_2) \in L_{us}$

If this is the case we define $\sigma_K(t_1, t_2) = t$, where t is the tree such that t_1 is the left subtree, t_2 is the right subtree of t and $label(root(t)) = (x, \sigma_L(r_1, r_2), y)$. Thus $(TREE(K), \sigma_K)$ becomes a partial algebra.

Let K_0 be the set of all syntactic computations corresponding to the basic elements of G, that is

$$K_0 = \bigcup_{(x_1,a,x_2)\in Elem(G)} T([x_1, x_2], [a])$$

This means that every element $t \in K_0$ consists in a single node n_t and $front(t) = label(root(t)) \in Elem(G)$.

Definition 3.1.4 *We denote by $Tree_{us}(K)$ the closure of K_0 in $(TREE(K), \sigma_K)$, that is $Tree_{us}(K) = Cl_{\sigma_K}(K_0)$. The elements of $Tree_{us}(K)$ are called* **syntactic computations**.

Every node label of an element $t \in Tree_{us}(K)$ belongs to $S \times L_{us} \times S$. Equivalently, if $T_{us}(p)$ denotes the set of all trees $t \in T(p)$ such that $label(n) \in S \times L_{us} \times S$ for every node n of t then $Tree_{us}(K) = \bigcup_{p\in PATH(G)} T_{us}(p)$.

In conclusion, all the results of the syntactic computations performed for the pair $(x, y) \in S \times S$ are the elements of the set $\bigcup_{p\in P_G(x,y)} T_{us}(p)$.

3.1.4 Semantic computations in KBOs and inference

If S denotes the set of the objects belonging to some real world described by a knowledge piece and Y is *the output space* then the answer function will appear as a mapping $Ans : S \times S \longrightarrow 2^Y$. The elements of Y can be sentences in a natural language, pieces of images and so on. An interrogation will be obtained by specifying an *ordered pair* of objects, $(o_1, o_2) \in S \times S$. The order in which we specify the objects is taken into consideration. The simplest way to explain this property is based on the fact that the labelled graph is oriented one, but this can be viewed also in particular cases. If the elements of Y are sentences in a natural language then $Ans(o_1, o_2)$ will consist in all the properties of the object o_1 with respect to o_2. If all the properties of o_1 are needed then we take the set $\bigcup_{o\in S} Ans(o_1, o)$.

We consider the knowledge base with output $(K, Y, g, *)$, the partial algebra $(Tree_{us}(K), \sigma_K)$ and the algebra $(Y, *)$. Let $\widetilde{g} : K_0 \longrightarrow Y$ be the mapping $\widetilde{g}(t) = g(x, a, y)$, where $label(root(t)) = (x, a, y)$. There is a morphism $\widetilde{G} : Tree_{us}(K) \longrightarrow Y$, uniquely determined, such that $\widetilde{G}(t) = \widetilde{g}(t)$ for $t \in K_0$. Thus if $t_1, t_2 \in Tree_{us}(K)$ and $(t_1, t_2) \in dom(\sigma_K)$ then $\widetilde{G}(\sigma_K(t_1, t_2)) = \widetilde{G}(t_1) * \widetilde{G}(t_2)$.

We observe that in Definition 3.1.1 the mapping $*$ may be a *partial* algebraic operation. Really, the following condition must be fulfilled: if $t_1, t_2 \in Tree_{us}(K)$ and $(t_1, t_2) \in dom(\sigma_K)$ then $(\widetilde{G}(t_1), \widetilde{G}(t_2)) \in dom(*)$.

Definition 3.1.5 *Let $KBO = (K, Y, g, *)$ be a knowledge base with output and L_{us} such that $L_0 \subseteq L_{us} \subseteq L$. Let $\widetilde{G} : Tree_{us}(K) \longrightarrow Y$ be the morphism determined by $\widetilde{g} : K_0 \longrightarrow Y$. The computation performed to obtain the value $\widetilde{G}(t)$ for some*

$t \in Tree_{us}(K)$ is called **semantic computation**. *The element $\widetilde{G}(t) \in Y$ is* **the result** *of this computation. The function $Ans : S \times S \longrightarrow 2^Y$ defined by*

$$Ans(x,y) = \bigcup_{p \in P_G(x,y)} \bigcup_{t \in T_{us}(p)} \{\widetilde{G}(t)\}$$

is called **the answer function** *of KBO.*

We observe that the answer function is defined by means of the paths in the corresponding labelled graph. We shall remark that for some $(x,y) \in S \times S$ we can have $P_G(x,y) \neq \emptyset$, but $T_{us}(p) = \emptyset$ for every $p \in P_G(x,y)$ even if $\bigcup_{p \in P_G(x,y)} T(p) \neq \emptyset$. In the application presented in Section 3.1.6 we see that (x_6, x_4) is such a pair. In order to avoid this situation we give another description of the answer function, which is used in applications.

We denote
$Us(x,y) = \{r \in L_{us} \mid (x,y) \in f(r)\}$
$T_r(x,y) = \{t \in Tree_{us}(G) \mid label(root(t)) = (x,r,y)\}$

Proposition 3.1.1 *For every $(x,y) \in S \times S$ we have*

$$Ans(x,y) = \bigcup_{r \in Us(x,y)} \bigcup_{t \in T_r(x,y)} \{\widetilde{G}(t)\}$$

Proof. We verify the following relation

$$\bigcup_{p \in P_G(x,y)} T_{us}(p) = \bigcup_{r \in Us(x,y)} T_r(x,y)$$

If for some $p \in P_G(x,y)$ we have $t \in T_{us}(p)$ then by Proposition 3.1.3 and by the definition of $T_{us}(p)$ there is $r \in L_{us}$ such that $label(root(t)) = (x,r,y)$. By Proposition 3.1.4 we have also $(x,y) \in f(r)$, therefore $r \in Us(x,y)$. Obviously $t \in T_r(x,y)$. Conversely, if $t \in T_r(x,y)$ for some $r \in Us(x,y)$ then the frontier $front(t) = (x,a_1,x_1)(x_1,a_2,x_2)\ldots(x_n,a_{n+1},y)$ gives the path $p = ([x,x_1,\ldots,x_n,y],[a_1,\ldots,a_{n+1}]) \in P_G(x,y)$ such that $t \in T_{us}(p)$. ∎

The expression given in Proposition 3.1.1 is useful in some cases, for example when $L_{us} \in \mathcal{I}nitial(L_0)$. We observe that if this is the case then for every $r \in Us(x,y)$ we have $T_r(x,y) \neq \emptyset$. This property can be proved as follows. If $r \in Us(x,y)$ then $r \in L_{us}$ and $(x,y) \in f(r)$. Using the fact that $L_{us} \in \mathcal{I}nitial(L_0)$ it is not difficult to verify by induction on the length of $trace(r)$ that there is $t \in Tree_{us}(K)$ such that $label(root(t)) = (x,r,y)$. Thus $t \in T_r(x,y)$ and therefore $T_r(x,y) \neq \emptyset$.

Definition 3.1.6 *In a KBO the inference relation $\vdash \subseteq (S \times S) \times 2^Y$ is defined by $(x,y) \vdash M$ iff $Ans(x,y) = M$.*

3.1.5 Proofs of the theoretical results

We give in this section the formal proofs for the properties of the syntactic computations, which are presented in Section 3.1.3.

We introduce now the following notation, which is used also in the next sections.

Definition 3.1.7 *For every $\alpha \in L$ we define $trace(\alpha)$ as follows:*

(1) *if $\alpha \in L_0$ then $trace(\alpha) = (\alpha)$*

(2) *if $\alpha = \sigma(u, v)$ then $trace(\alpha) = (p, q)$, where $trace(u) = (p)$ and $trace(v) = (q)$*

The number n such that $trace(\alpha) \in L_0^n$ is called the length of $trace(\alpha)$.

The proof of Proposition 3.1.7 uses the following remark, which can be verified by induction on the length of $trace(\alpha)$:

Remark 3.1.1 *If $trace(\alpha) = (a_1, \ldots, a_n)$ and $(f(a_1), \ldots, f(a_n)) \in dom(u^{(n)})$ then*

$$f(\alpha) = \begin{cases} f_0(\alpha) \text{ if } n = 1 \\ \\ u^{(n)}(f(a_1), \ldots, f(a_n)) \text{ if } n \geq 2 \end{cases}$$

Proposition 3.1.2 *Let be $n \geq 2$. If $t \in T([x_1, x_2, \ldots, x_{n+1}], [a_1, a_2, \ldots, a_n])$ then there is $i \in \{2, \ldots, n\}$ such that $t_l \in T([x_1, x_2, \ldots, x_i], [a_1, \ldots, a_{i-1}])$ and $t_r \in T([x_i, \ldots, x_{n+1}], [a_i, \ldots, a_n])$, where t_l and t_r are the subtrees corresponding to the left descendant, respectively right descendant of $root(t)$.*

Proof. Obviously

$$front(t) = CON_{l=1}^{n}((x_l, a_l, x_{l+1})) = front(t_l).front(t_r)$$

therefore there is $i \in \{2, \ldots, n\}$ such that

$$front(t_l) = CON_{l=1}^{i-1}((x_l, a_l, x_{l+1})), front(t_r) = CON_{l=i}^{n}((x_l, a_l, x_{l+1}))$$

Thus $t_l \in T([x_1, \ldots, x_i], [a_1, \ldots, a_{i-1}])$ and $t_r \in T([x_i, \ldots, x_{n+1}], [a_i, \ldots, a_n])$. ∎

Proposition 3.1.3 *If $t \in T([x_1, x_2, \ldots, x_{n+1}], [a_1, a_2, \ldots, a_n])$ then $label(root(t)) = (x_1, r, x_{n+1})$ for some $r \in L$. Moreover, $trace(r) = (a_1, a_2, \ldots, a_n)$.*

Proof. Obviously the property is true for $n = 1$. If $T([x_1, x_2, x_3], [a_1, a_2]) \neq \emptyset$ then this set contains only one tree, namely the tree t which is defined by the conditions:

$$front(t) = (x_1, a_1, x_2)(x_2, a_2, x_3), label(root(t)) = (x_1, \sigma_L(a_1, a_2), x_3)$$

therefore the proposition is true for $n = 2$. We suppose that the property is true for $n \in \{1, \ldots, m-1\}$ and let be $t \in T([x_1, x_2, \ldots, x_{m+1}], [a_1, a_2, \ldots, a_m])$. By Proposition 3.1.2 it follows that there is $i \in \{2, \ldots, m\}$ such that

$$t_l \in T([x_1, x_2, \ldots, x_i], [a_1, \ldots, a_{i-1}]), t_r \in T([x_i, \ldots, x_{m+1}], [a_i, \ldots, a_m])$$

where t_l and t_r are the subtrees corresponding to the left descendant, respectively right descendant of $root(t)$. By inductive assumption we have $label(root(t_l)) = (x_1, r_1, x_i)$ for some $r_1 \in L$ and $label(root(t_r)) = (x_i, r_2, x_{m+1})$ for some $r_2 \in L$. Moreover, $trace(r_1) = (a_1, \ldots, a_{i-1})$ and $trace(r_2) = (a_i, \ldots, a_m)$. Since $t \in T([x_1, x_2, \ldots, x_{m+1}], [a_1, a_2, \ldots, a_m])$ we obtain

- $(r_1, r_2) \in dom(\sigma_L)$
- $label(root(t)) = (x_1, \sigma_L(r_1, r_2), x_{m+1})$

Obviously we have $trace(\sigma_L(r_1, r_2)) = (a_1, a_2, \ldots, a_m)$. ■

We denote by $Tree([a_1, \ldots, a_n])$ the union set of all $T([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n])$ such that $([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n]) \in Path_n(x_1, x_{n+1})$. This set contains precisely all the syntactic computations which can be realized in K by means of the sequence $a_1, \ldots, a_n$.

Proposition 3.1.4 *Let be $n \geq 2$ and $t \in Tree([a_1, \ldots, a_n])$. If $label(root(t)) = (x, r, y)$ then $(x, y) \in f(r) = \sigma_T^{(n)}(f(a_1), \ldots, f(a_n))$.*

Proof. First, we verify the property for $n = 2$. We suppose $t \in Tree([a_1, a_2]) = \bigcup_{x_1, x_2, x_3 \in S} T([x_1, x_2, x_3], [a_1, a_2])$. We denote $label(root(t)) = (x, r, y)$. There is $z \in S$ such that $t \in T([x, z, y], [a_1, a_2])$, therefore $(x, a_1, z) \in Elem(G)$, $(z, a_2, y) \in Elem(G)$. We have also $r = \sigma_L(a_1, a_2)$, therefore $(f(a_1), f(a_2)) \in dom(u)$. Since $u \prec prod$, $(x, z) \in f(a_1)$ and $(z, y) \in f(a_2)$ it follows that $(x, y) \in u(f(a_1), f(a_2)) = f(\sigma_L(a_1, a_2)) = f(r)$.

We suppose the proposition is true for $n \in \{2, \ldots, m\}$ and we prove it for $n = m+1$. Let us consider $t \in Tree([a_1, \ldots, a_{m+1}])$ and we denote $label(root(t)) = (x, r, y)$. From the definition of $Tree([a_1, \ldots, a_{m+1}])$ we deduce that there are $x_2, \ldots, x_{m+1} \in S$ such that $t \in T([x, x_2, \ldots, x_{m+1}, y], [a_1, \ldots, a_{m+1}])$. We denote by t_l, t_r the subtrees corresponding to the left descendant, respectively right descendant of $root(t)$. By Proposition 3.1.2 it follows that there is $k \in \{2, \ldots, m+1\}$ such that

$$t_l \in T([x, x_2, \ldots, x_k], [a_1, \ldots, a_{k-1}]), t_r \in T([x_k, \ldots, x_{m+1}, y], [a_k, \ldots, a_{m+1}])$$

First, we assume $k \neq 2$ and $k \neq m+1$. By Proposition 3.1.3 we have $label(root(t_l)) = (x, r_1, x_k)$, $label(root(t_r)) = (x_k, r_2, y)$ and $r = \sigma_L(r_1, r_2)$ for some $r_1, r_2 \in L$. By inductive assumption we have

$$(x, x_k) \in f(r_1) = u^{(k-1)}(f(a_1), \ldots, f(a_{k-1}))$$

$$(x_k, y) \in f(r_2) = u^{(m+2-k)}(f(a_k), \ldots, f(a_{m+1}))$$

Since $(r_1, r_2) \in dom(\sigma_L)$ it follows that $(f(r_1), f(r_2)) \in dom(u)$. Because $u \in R(prod_S)$, we have

$$(x, y) \in u^{(2)}(f(r_1), f(r_2)) = u^{(2)}(u^{(k-1)}(f(a_1), \ldots, f(a_{k-1})),$$
$$u^{(m+2-k)}(f(a_k), \ldots, f(a_{m+1}))) = u^{(m+1)}(f(a_1), \ldots, f(a_{m+1}))$$

We have also

$$f(r) = f(\sigma_L(r_1, r_2)) = u(f(r_1), f(r_2)) = u^{(m+1)}(f(a_1), \ldots, f(a_{m+1}))$$

If $k = 2$ then $label(root(t_l)) = (x, a_1, x_2)$, $label(root(t_r)) = (x_2, r_2, y)$ and $r = \sigma_L(a_1, r_2)$. From $(x, a_1, x_2) \in Elem(G)$ it follows that $(x, x_2) \in f(a_1)$. From $t_r \in Tree([a_2, \ldots, a_{m+1}])$ by inductive assumption it follows $(x_2, y) \in f(r_2) = u^{(m)}(f(a_2), \ldots, a_{m+1}))$. As in the previous case we have $(a_1, r_2) \in dom(\sigma_L)$, therefore $(f(a_1), f(r_2)) \in dom(u)$ and $(x, y) \in f(r) = u(f(a_1), f(r_2))=$
$u(f(a_1), u^{(m)}(f(a_2), \ldots, f(a_{m+1})))= u^{(m+1)}(f(a_1), \ldots, f(a_{m+1}))$.
The case $k = m + 1$ is similar to the case $k = 2$. ■

Corollary 3.1.1 *If* $T([x_1, x_2, \ldots, x_{n+1}], [a_1, a_2, \ldots, a_n]) \neq \emptyset$ *then*

$$(x_1, x_{n+1}) \in u^{(n)}(f(a_1), \ldots, f(a_n))$$

Proof. Let be $t \in T([x_1, x_2, \ldots, x_{n+1}], [a_1, a_2, \ldots, a_n])$. By Proposition 3.1.3 there is $r \in L$ euch that $label(root(t)) = (x_1, r, x_{n+1})$. By Proposition 3.1.4 we have $(x_1, x_{n+1}) \in f(r) = u^{(n)}(f(a_1), \ldots, f(a_n))$. ■

Proposition 3.1.5 *Let be* $n \geq 2$ *and* $a_1, \ldots, a_n \in L_0$ *such that* $(f(a_1), \ldots, f(a_n)) \in dom(u^{(n)})$. *For every* $(x, y) \in u^{(n)}(f(a_1), \ldots, f(a_n))$ *there is a labelled tree* $t \in Tree([a_1, \ldots, a_n])$ *such that* $label(root(t)) = (x, r, y)$ *for some* $r \in \bigcup_{k=1}^{n-1} D_k$.

Proof. We verify the property for $n = 2$. If $(f(a_1), f(a_2)) \in dom(u^{(2)})$ then $(a_1, a_2) \in dom(\sigma_L)$. Let be $(x, y) \in u^{(2)}(f(a_1), f(a_2))$. It follows that there is $z \in S$ such that $(x, z) \in f(a_1)$, $(z, y) \in f(a_2)$. Thus $(x, a_1, z) \in Elem(G)$, $(z, a_2, y) \in Elem(G)$. The set $T([x, z, y], [a_1, a_2])$ contains only one element, namely the tree t which is defined by $label(root(t)) = (x, \sigma_L(a_1, a_2), y)$ and $front(t) = (x, a_1, z)(z, a_2, y)$. Thus the property is true for $n = 2$.
We suppose the property is true for $n \in \{2, \ldots, m\}$ and $(f(a_1), \ldots, f(a_{m+1})) \in dom(u^{(m+1)})$. Let us consider an arbitrary element $(x, y) \in u^{(m+1)}(f(a_1), \ldots, f(a_{m+1}))$. We analyse the following three cases:

I) We denote $u = f(a_1)$, $v = (f(a_2), \ldots, f(a_{m+1}))$ and suppose $v \in dom(u^{(m)})$, $(u, u^{(m)}(v)) \in dom(u^{(2)})$.
It follows that $u^{(m+1)}(f(a_1), \ldots, f(a_{m+1})) = u^{(2)}(u, u^{(m)}(v))$. From $(x, y) \in u^{(2)}(u, u^{(m)}(v))$ it follows that there is $z \in S$ such that $(x, z) \in u$, $(z, y) \in u^{(m)}(v)$. Applying the inductive assumption it follows that there is a labelled tree $t_2 \in Tree([a_2, \ldots, a_{m+1}])$ such that $label(root(t_2)) = (z, r_2, y)$ for some $r_2 \in \bigcup_{k=1}^{m-1} D_k$. By Proposition 3.1.4 we have $f(r_2) = u^{(m)}(v)$. Taking into

account that $a_1 \in D_0$, $r_2 \in \bigcup_{k=1}^{m-1} D_k$ and $(u, f(r_2)) \in dom(u^{(2)})$ we deduce that $(a_1, r_2) \in dom(\sigma_L)$. We take $r = \sigma_L(a_1, r_2)$, which belongs to $\bigcup_{k=2}^{m} D_k$. There are $y_1, \ldots y_{m-1} \in S$ such that $t_2 \in T([z, y_1, \ldots y_{m-1}, y], [a_2, \ldots, a_{m+1}])$. We consider the tree t which is defined by the following conditions:

- $label(root(t)) = (x, \sigma_L(a_1, r_2), y)$
- the label of the left descendant of $root(t)$ is (x, a_1, z)
- the right descendant of $root(t)$ is labelled by (z, r_2, y) and its corresponding subtree is t_2

Obviously $t \in Tree([a_1, \ldots, a_{m+1}])$.

II) We denote $u = (f(a_1), \ldots, f(a_m))$, $v = f(a_{m+1})$. Suppose $u \in dom(u^{(m)})$, $(u^{(m)}(u), v) \in dom(u^{(2)})$. This is treated in a similar manner as in the previous case.

III) We denote $u = (f(a_1), \ldots, f(a_k))$ and $v = (f(a_{k+1}), \ldots, f(a_{m+1}))$, where $k \in \{2, \ldots, m-1\}$. Suppose $u \in dom(u^{(k)})$, $v \in dom(u^{(m-k+1)})$ and $(u^{(k)}(u)$, $u^{(m-k+1)}(v)) \in dom(u^{(2)})$. It follows that

$$u^{(m+1)}(f(a_1), \ldots, f(a_{m+1})) = u^{(2)}(u^{(k)}(u), u^{(m-k+1)}(v))$$

We denote $A = [a_1, \ldots, a_k]$ and $B = [a_{k+1}, \ldots, a_{m+1}]$. Let us consider an arbitrary element $(x, y) \in u^{(m+1)}(f(a_1), \ldots, f(a_{m+1}))$. There is $z \in S$ such that $(x, z) \in u^{(k)}(u), (z, y) \in u^{(m-k+1)}(v)$. By inductive assumption there are $t_l \in Tree(A)$, $t_r \in Tree(B)$ satisfying the conditions

$$label(root(t_l)) = (x, m_1, z); label(root(t_r)) = (z, m_2, y)$$

for some $m_1 \in \bigcup_{p=1}^{k-1} D_p$, $m_2 \in \bigcup_{q=1}^{m-k} D_q$. By Proposition 3.1.4 we obtain:

$$(x, z) \in f(m_1) = u^{(k)}(u); (z, y) \in f(m_2) = u^{(m-k+1)}(v)$$

We have $(m_1, m_2) \in L \times L$ and $(f(m_1), f(m_2)) \in dom(u^{(2)})$, therefore (m_1, m_2) $\in dom(\sigma_L)$. If we denote $r = \sigma_L(m_1, m_2)$ then $r \in \bigcup_{l=1}^{m} D_l$. We take the tree which is defined by the following conditions:

$label(root(t)) = (x, r, y)$

t_l is the subtree corresponding to the left descendant of $root(t)$

t_r is the subtree corresponding to the right descendant of $root(t)$

There are $y_1, \ldots, y_{k-1}, z_1, \ldots, z_{m-k} \in S$ such that

$$t_l \in T([x, y_1, \ldots, y_{k-1}, z], A), t_r \in T([z, z_1, \ldots, z_{m-k}, y], B)$$

It follows that $t \in T([x, y_1, \ldots, y_{k-1}, z, z_1, \ldots, z_{m-k}, y], [a_1, \ldots, a_{m+1}])$, therefore $t \in Tree([a_1, \ldots, a_{m+1}])$

■

Corollary 3.1.2 *$Tree([a_1, \ldots, a_n]) \neq \emptyset$ iff $(f(a_1), \ldots, f(a_n)) \in dom(u^{(n)})$*

Proof. We apply Proposition 3.1.4 and Proposition 3.1.5. ■

Proposition 3.1.6 *Let be $n \geq 2$ and $([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n]) \in PATH(G)$. If $(f(a_1), \ldots, f(a_n)) \in dom(u^{(n)})$ then $T([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n]) \neq \emptyset$.*

Proof. We proceed by induction on n. For $n = 2$ the property is true. Suppose the property is true for every $n < r$ and we prove it for $n = r$. Let be $p = ([x_1, \ldots, x_{r+1}], [a_1, \ldots, a_r]) \in PATH(G)$ such that $(f(a_1), \ldots, f(a_r)) \in dom(u^{(r)})$. Three cases are possible, but we shall examine here only the case when there is $k \in \{2, \ldots, r-2\}$ such that

$$(f(a_1), \ldots, f(a_k)) \in dom(u^{(k)})$$

$$(f(a_{k+1}), \ldots, f(a_r)) \in dom(u^{(r-k)})$$

$$(u^{(k)}(f(a_1), \ldots, f(a_k)), u^{(r-k)}(f(a_{k+1}), \ldots, f(a_r))) \in dom(u^{(2)})$$

$$u^{(r)}(f(a_1), \ldots, f(a_r)) =$$
$$u^{(2)}(u^{(k)}(f(a_1), \ldots, f(a_k)), u^{(r-k)}(f(a_{k+1}), \ldots, f(a_r)))$$

We denote

$$p_1 = ([x_1, \ldots, x_{k+1}], [a_1, \ldots, a_k]), p_2 = ([x_{k+1}, \ldots, x_{r+1}], [a_{k+1}, \ldots, a_r])$$

We have $p_1, p_2 \in PATH(G)$. By the inductive assumption we have $T(p_1) \neq \emptyset$ and $T(p_2) \neq \emptyset$. Let be $t_1 \in T(p_1)$ and $t_2 \in T(p_2)$. By Proposition 3.1.3 and Proposition 3.1.4 we have $label(root(t_1)) = (x_1, u_1, x_{k+1})$, $label(root(t_2)) = (x_{k+1}, u_2, x_{r+1})$ for some $u_1, u_2 \in L$, $(x_1, x_{k+1}) \in f(u_1) = u^{(k)}(f(a_1), \ldots, f(a_k))$, $(x_{k+1}, x_{r+1}) \in f(u_2) = u^{(r-k)}(f(a_{k+1}), \ldots, f(a_r))$. Thus we have $(x_1, x_r) \in u^{(2)}(f(u_1), f(u_2)) = u^{(r)}(f(a_1), \ldots, f(a_r))$.
Because

$$\{(u, v) \in L \times L \mid (f(u), f(v)) \in dom(u)\} \subseteq dom(\sigma_L)$$

and $(f(u_1), f(u_2)) \in dom(u^{(2)})$, we obtain $\sigma_L(u_1, u_2) \in L$. We consider the tree t such that $label(root(t)) = (x_1, \sigma_L(u_1, u_2), x_r)$ and obviously $t \in T(p)$. ■

Corollary 3.1.3 *Let be $p = ([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n]) \in PATH(G)$. We have $T(p) \neq \emptyset$ iff $(f(a_1), \ldots, f(a_n)) \in dom(u^{(n)})$.*

Proof. Suppose $T(p) \neq \emptyset$. If by contrary we have $(f(a_1), \ldots, f(a_n)) \notin dom(u^{(n)})$ then $Tree([a_1, \ldots, a_n]) = \emptyset$ therefore $T(p) = \emptyset$, which is not true. The converse implication is obtained by Proposition 3.1.6. ■

3.1.6 An application of $KBOs$ in travel scheduling.

The concept of KBO can be used in writing of interfaces for expert systems, problem solving, rewriting systems and other domains. In this section we consider a simple application of this concept. Let us consider the following knowledge piece KP_2:

> *We consider the airports $x_1, \ldots, x_m$. The companies $L_1, \ldots, L_k$ organize some non-stop flights between these airports. It is known a set R of restrictions concerning the continuation of a travel for a passenger. An element of R is a rule of the form $L_i \longrightarrow P_i$, where P_i is a nonempty subset of $\{L_1, \ldots, L_k\}$. Such rule specifies the following property: if a passenger arrives in some airport using the company L_i then he may continue his travel with the company L only if $L \in P_i$. Give the answer to the following interrogation: given a pair (x, y) of airports, is there a flight from x to y having at most n intermediary airports? In the affirmative case find all the solutions.*

In order to solve this problem we consider $S = \{x_1, \ldots, x_m\}$ and we denote by ρ_i the following binary relation on S, where $i \in \{1, \ldots, k\}$: $(x_p, x_q) \in \rho_i$ iff the company L_i organizes a non-stop flight from x_p to x_q. We take $L_0 = \{l_1, \ldots, l_k\}$, $T_0 = \{\rho_1, \ldots, \rho_k\}$ and $f_0 : L_0 \longrightarrow T_0$, where $f_0(l_i) = \rho_i$ for $i \in \{1, \ldots, k\}$. Thus we obtain the labelled graph $G = (S, L_0, T_0, f_0)$. We consider the greatest labelled stratified graph $\mathcal{G} = (G, L, T, prod_S, f)$ over G. We shall specify R as a set of pairs of the form (l_s, l_r) as follows: $(l_s, l_r) \in R$ iff there is a rule $L_s \longrightarrow P_s$ such that $L_r \in P_s$.

Intuitively, an *useful label* is an element $\alpha \in L$ such that if $trace(\alpha) = (l_{i_1}, \ldots, l_{i_s})$ then there is a path in G giving a solution of the problem, that is the path satisfies the restrictions and the corresponding non-stop flights are realized respectively by $L_{i_1}, \ldots, L_{i_s}$. We denote $last(\alpha) = l_{i_s}$.

We define inductively the set L_{us} of the *useful labels* as follows:

- $L_0 \subseteq L_{us}$
- $\alpha = \sigma(u, v) \in L_{us}$ iff $u \in L_{us}$, $v \in L_0$, $(last(u), v) \in R$ and $trace(u) \in \bigcup_{k=1}^{n+1} L_0^k$

Proposition 3.1.7 *There is a solution of the problem from x_i to x_j if and only if $Us(x_i, x_j) \neq \emptyset$. Moreover, if $\alpha \in Us(x_i, x_j)$ and $trace(\alpha) = (l_{i_1}, \ldots, l_{i_s})$ then the sequence $L_{i_1}, \ldots, L_{i_s}$ gives a solution.*

Proof. If there is a solution from x_i to x_j then there is a path p in G such that $p = ([x_i, x_{i_1}, \ldots, x_{i_s}, x_j], [l_{i_1}, \ldots, l_{i_{s+1}}])$, $(l_{i_u}, l_{i_{u+1}}) \in R$ for $u \in \{1, \ldots, s\}$ and $0 \leq s \leq n$. If $s = 0$ then $p = ([x_i, x_j], l_{i_1})$ is a path in G, therefore $Us(x_i, x_j) \neq \emptyset$ because $l_{i_1} \in Us(x_i, x_j)$. For the case $s \geq 1$ we define recursively

$$l_{i_1} = \alpha_0$$

$$\sigma(\alpha_u, l_{i_{u+2}}) = \alpha_{u+1} \text{ for } u \in \{0, \ldots, s-1\}$$

Applying the definition of D_k ($k \geq 0$) and using the fact that $u = prod_S$ we deduce that $\alpha_i \in D_i$ for $i \in \{0, \ldots, s\}$. The path p gives a solution for KP_2, therefore $\{(l_{i_1}, l_{i_2}), (l_{i_2}, l_{i_3}), \ldots, (l_{i_s}, l_{i_{s+1}})\} \subseteq R$. Equivalently we have $(last(\alpha_u), l_{i_{u+2}}) \in R$ for each $u \in \{0, \ldots, s-1\}$. But $0 \leq s \leq n$ and $trace(\alpha_s) = (l_{i_1}, \ldots, l_{i_{s+1}}) \in L_0^{s+1}$, therefore $\alpha_s \in L_{us}$. Moreover, $(x_i, x_{i_1}) \in f_0(l_{i_1}) = f(l_{i_1})$, ..., $(x_{i_s}, x_j) \in f_0(l_{i_{s+1}}) = f(l_{i_{s+1}})$ and thus $(x_i, x_j) \in prod_S(f(l_{i_1}), \ldots, f(l_{i_{s+1}}))$. Taking into account *Remark 3.1.1* and the choice $u = prod_S$, it follows that $(x_i, x_j) \in f(\alpha_s)$. Thus $\alpha_s \in Us(x_i, x_j)$ and therefore $Us(x_i, x_j) \neq \emptyset$.
Conversely, suppose $(x_i, x_j) \in f(\alpha)$ and $\alpha \in L_{us}$. Either $\alpha \in L_0$ or there are $s \geq 2$ and $l_{i_1}, \ldots, l_{i_s} \in L_0$ such that $\alpha = \alpha_{s-1}$, where

$$\sigma(l_{i_1}, l_{i_2}) = \alpha_1$$

$$\sigma(\alpha_u, l_{i_{u+2}}) = \alpha_{u+1} \text{ for } u \in \{1, \ldots, s-2\}$$

and $(l_{i_q}, l_{i_{q+1}}) \in R$ for $q \in \{1, \ldots, s-1\}$. Moreover, $s \leq n+1$.
If $\alpha \in L_0$ then there is a non-stop flight from x_i to x_j and thus there is a solution for KP_2.
If the second case holds, that is $\alpha = \alpha_{s-1}$, then by the morphism property of f we obtain

$$f(\alpha_{s-1}) = f(\sigma(\alpha_{s-2}, l_{i_s})) = \sigma_T^{(2)}(f(\alpha_{s-2}), f(l_{i_s}))$$

From $(x_i, x_j) \in f(\alpha_{s-1})$ we deduce that there is $y_{s-2} \in S$ such that $(x_i, y_{s-2}) \in f(\alpha_{s-2})$ and $(y_{s-2}, x_j) \in f(l_{i_s})$. We iterate this property and thus there are $y_0, \ldots, y_{s-2} \in S$ such that $(x_i, y_0) \in f(l_{i_1})$, $(y_0, y_1) \in f(l_{i_2})$, ...,$(y_{s-2}, x_j) \in f(l_{i_s})$. It follows that $([x_i, y_0, \ldots, y_{s-2}, x_j], [l_{i_1}, \ldots, l_{i_s}])$ is a path in G and it gives a solution for KP_2 because there are at most n intermediary nodes. ■

If $X \subseteq L$ and $Y \subseteq L_0$ then we denote $X \otimes Y = \{(u, v) \in X \times Y \mid (last(u), v) \in R\}$. We denote $D_k^{(us)} = D_k \cap L_{us}$ for any $k \geq 0$. It is not difficult to observe that $D_0^{(us)} = L_0$ and $D_{k+1}^{(us)} = \{\sigma(u, v) \mid (u, v) \in D_k^{(us)} \otimes D_0^{(us)}, f(u) \circ f(v) \neq \emptyset\}$ for $0 \leq k \leq n-1$. Thus in order to compute the set L_{us} we compute $D_0^{(us)}, \ldots, D_n^{(us)}$ and we take $L_{us} = \bigcup_{k=0}^{n} D_k^{(us)}$.

In order to illustrate the computation we consider $m = 9$, $n = 2$, $k = 3$ and the following binary relations on $S = \{x_1, \ldots, x_9\}$:

$$\rho_1 = \{(x_1, x_2), (x_2, x_3), (x_2, x_6)\},$$

$$\rho_2 = \{(x_3, x_4), (x_4, x_5), (x_6, x_7), (x_7, x_8), (x_8, x_9), (x_9, x_5)\}$$

$$\rho_3 = \{(x_1, x_3), (x_3, x_5), (x_7, x_4)\}$$

We obtain the labelled graph $G = (S, L_0, T_0, f_0)$ from Figure 3.4.

We suppose the restrictions of the problem are represented by the fact that the company L_3 does not take over the passengers of L_1 or L_2. It follows that $R = \{(l_1, l_1), (l_2, l_2), (l_3, l_3), (l_1, l_2), (l_2, l_1), (l_3, l_1), (l_3, l_2)\}$. We obtain the following computations:

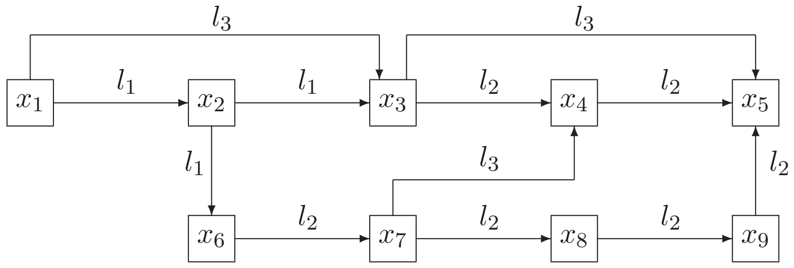


Figure 3.4: Labelled graph for KP_2, where $m = 9, n = 2, k = 3$

$1) D_0^{(us)} = \{l_1, l_2, l_3\};$

$$f_0(l_1) = \rho_1; f_0(l_2) = \rho_2; f_0(l_3) = \rho_3$$

$2) D_1^{(us)} = \{\sigma(l_1, l_1), \sigma(l_1, l_2), \sigma(l_2, l_2), \sigma(l_3, l_2), \sigma(l_3, l_3)\};$

$$f(\sigma(l_1, l_1)) = \rho_4; f(\sigma(l_1, l_2)) = \rho_5; f(\sigma(l_2, l_2)) = \rho_6;$$
$$f(\sigma(l_3, l_2)) = \rho_7; f(\sigma(l_3, l_3)) = \rho_8,$$

where

$$\rho_4 = \{(x_1, x_3), (x_1, x_6)\},\ \rho_5 = \{(x_2, x_4), (x_2, x_7)\},$$
$$\rho_6 = \{(x_3, x_5), (x_6, x_8), (x_7, x_9), (x_8, x_5)\},$$
$$\rho_7 = \{(x_1, x_4), (x_7, x_5)\},\ \rho_8 = \{(x_1, x_5)\}$$

$3) D_2^{(us)} = \{\sigma(\sigma(l_1, l_1), l_2), \sigma(\sigma(l_1, l_2), l_2), \sigma(\sigma(l_2, l_2), l_2), \sigma(\sigma(l_3, l_2), l_2)\};$

$$f(\sigma(\sigma(l_1, l_1), l_2)) = \rho_9; f(\sigma(\sigma(l_1, l_2), l_2)) = \rho_{10}; f(\sigma(\sigma(l_2, l_2), l_2)) = \rho_{11};$$
$$f(\sigma(\sigma(l_3, l_2), l_2)) = \rho_8,$$

where

$$\rho_9 = \{(x_1, x_4), (x_1, x_7)\},\ \rho_{10} = \{(x_2, x_5), (x_2, x_8)\},$$
$$\rho_{11} = \{(x_6, x_9), (x_7, x_5)\}$$

It follows that $L_{us} = D_0^{(us)} \cup D_1^{(us)} \cup D_2^{(us)}$.

In order to obtain the solutions of the given problem we proceed as follows. Let be $(x_i, x_j) \in S \times S$. We consider the sets $Us(x_i, x_j)$. If $Us(x_i, x_j) = \emptyset$ then it does not exist any solution. Otherwise, every element of $Us(x_i, x_j)$ will give a solution of the problem. For example, in order to learn if there is a flight from x_1 to x_4 we observe that $Us(x_1, x_4) = \{\sigma(l_3, l_2), \sigma(\sigma(l_1, l_1), l_2)\}$. It follows that there are two solutions. The same problem for (x_2, x_4) has only one solution, namely $\sigma(l_1, l_2)$.

We consider the labelled stratified graph $\mathcal{G} = (G, L, T, prod_S, f)$, where $G = (S, L_0, T_0, f_0)$. In order to obtain a KBO we denote by Y the space of all the sentences of the form *use the companies* $L_{i_1}, \ldots, L_{i_s}$ and *use the company* L, where $s \geq 2$ and $L_{i_1}, \ldots, L_{i_s}, L \in \{L_1, L_2, L_3\}$. We define the following partial algebraic operation $*$ on Y:

1) If p=*use the companies* $L_{i_1}, \ldots, L_{i_s}$ and q=*use the company* L then

$$p * q = \begin{cases} p & \text{if} \quad L \in \{L_{i_1}, \ldots, L_{i_s}\} \\ r & \text{otherwise} \end{cases}$$

where r=*use the companies* $L_{i_1}, \ldots, L_{i_s}, L$.

2) If p=*use the company* L_{i_1} and q=*use the company* L_{i_2} then

$$p * q = \begin{cases} p & \text{if} \quad L_{i_1} = L_{i_2} \\ r & \text{otherwise} \end{cases}$$

where r=*use the companies* L_{i_1}, L_{i_2}.

We consider the mapping $\widetilde{g} : K_0 \longrightarrow Y$ defined by $\widetilde{g}(x, l_i, y)$ = *use the company* L_i for $i \in \{1, 2, 3\}$.

In order to compute $Ans(x, y)$ we use Proposition 3.1.1. For example, if we are interested for the solutions from x_1 to x_4 then we find $Us(x_1, x_4) = \{\alpha, \beta\}$, where $\alpha = \sigma(l_3, l_2)$ and $\beta = \sigma(\sigma(l_1, l_1), l_2)$. If we denote $T_\alpha(x_1, x_4) = \{t_1\}$ and $T_\beta(x_1, x_4) = \{t_2\}$ then $\widetilde{G}(t_1)$ =*use the companies* L_3, L_2 and $\widetilde{G}(t_2)$ =*use the companies* L_1, L_2. Thus we have

$$Ans(x_1, x_4) = \{ \textit{ use the companies } L_3, L_2; \textit{ use the companies } L_1, L_2 \}$$

Similarly we have $Ans(x_6, x_4) = \emptyset$. It is not difficult to give an algorithm to solve the problem for an arbitrary natural number n.

Remark 3.1.2 *We observe that in this application the elements of L are not effectively enumerated. Only the elements of the set L_{us} are used. We observe also that $*$ is a partial algebraic operation.*

Remark 3.1.3 *The method presented in this section to obtain an inference in a KBO is based on the set $L_u s$, which is a part of the set L. Thus, the main problem is reduced to the extraction of the set L_{us}. In the next section we present another aspect concerning the inference process based on labelled stratified graphs. In this case the inference is obtained by means of an interpretation defined for a labelled stratified graph.*

3.2 Inference based on *LSGs* and applications

3.2.1 Overview

In this section we present another facet of the inference process based on labelled stratified graphs. We introduce the concept of structured path and we introduce in this way some order on a given path. In comparison with the inference based on KBO, we establish this order instead of the useful labels from the set L. Two applications of this method are presented: one for the case when the conclusion of the inference process is given in a natural language and other case when we process geometric images.

3.2.2 Structured paths in a *LSG*

We consider a path

$$d = ([x_1, \ldots, x_{n+1}], [a_1, \ldots, a_n]) \tag{3.4}$$

in a labeled graph $G = (S, L_0, T_0, f_0)$.

Definition 3.2.1 *Consider the least set $STR(d)$ satisfying the following conditions:*

- $([x_i, x_{i+1}], a_i) \in STR(d)$, $i \in \{1, \ldots, n\}$
- *if* $([x_i, \ldots, x_k], b_1) \in STR(d)$ *and* $([x_k, \ldots, x_r], b_2) \in STR(d)$, *where* $1 \leq i < k < r \leq n+1$, *then* $([x_i, \ldots, x_r], [b_1, b_2]) \in STR(d)$

The maximal length elements of $STR(d)$, namely, the elements of the form

$$([x_1, \ldots, x_{n+1}], c) \in STR(d)$$

are called **structured paths** *over d.*

We consider the projection of $STR(d)$ under the second axis:

$$STR_2(d) = \{\beta \mid \exists \alpha : (\alpha, \beta) \in STR(d)\}$$

Consider for example the path

$$d = ([x_1, x_2, x_3, x_4], [a, b, a])$$

in an arbitrary labeled graph.
We have

$$STR(d) = \{([x_1, x_2], a), ([x_2, x_3], b), ([x_3, x_4], a),$$

$$([x_1, x_2, x_3], [a, b]), ([x_2, x_3, x_4][b, a]), ([x_1, x_2, x_3, x_4],$$

$$[[a, b], a]), ([x_1, x_2, x_3, x_4], [a, [b, a]])\}$$

therefore

$$STR_2(d) = \{a, b, [a, b], [b, a], [[a, b], a], [a, [b, a]]\} \tag{3.5}$$

Thus, two structured paths are obtained:

$$([x_1, x_2, x_3, x_4], [[a, b], a]), ([x_1, x_2, x_3, x_4], [a, [b, a]])$$

Let d be a path as in (3.4). We define the mapping

$$h : STR_2(d) \longrightarrow B$$

where B is defined in (1.3), as follows:

- $h(x) = x$ for $x \in L_0$
- $h([u, v]) = \sigma(h(u), h(v))$

For the case given in (3.5) we obtain, for example, $h([a, b]) = \sigma(a, b)$, $h([a, [b, a]]) = \sigma(a, \sigma(b, a))$.

Definition 3.2.2 *The structured path $d_s \in STR(d)$ is named an* **accepted structured path** *over $\mathcal{G}$ if $d_s = ([x_1, \ldots, x_{n+1}], c)$ and $h(c) \in L$. We denote by $ASP(\mathcal{G})$ the set of all accepted structured paths over $\mathcal{G}$.*

In order to benefit by the properties of L as a subset of a Peano algebra, it is convenient to denote a structured path by $d_s = ([x_1, \ldots, x_{n+1}], h(c))$ instead of $d_s = ([x_1, \ldots, x_{n+1}], c)$. This notation simplifies also the decision concerning the acceptability of a structured path.

The following notations will be used in what follows. If t is a labeled tree then **root(t)** denotes the node which is the root of t and **label(s)** denotes the label of the node s. For every node s, $label(s)$ will be an element of $S \times L \times S$. For $i < j$ we denote by $CON_{l=i}^{j}(w_l)$ the concatenation of the symbols $w_i, w_{i+1}, \ldots, w_j$.

If $s_1, \ldots, s_n$ are the leaves of the labeled tree t from left to right in this order and $label(s_1) = w_1, \ldots, label(s_n) = w_n$ then $CON_{l=1}^{n}(w_l)$ is the *frontier* of t and it is denoted by $front(t)$.

Let us consider a structured path

$$d_s = ([x_1, \ldots, x_{n+1}], h(c)) \in STR(d)$$

for the path d considered in (3.4).

We denote by $t(d_s)$ some tree defined as follows. If $n = 1$ then $t(d_s)$ reduces to a single node, which is the root of the tree and its label is (x_1, a_1, x_2). If $n \geq 2$ then $t(d_s)$ satisfies the following rules:

- $front(t) = CON_{i=1}^{n}((x_i, a_i, x_{i+1}))$.
- every node s of t, which is not a leaf, has two direct descendants: the left descendant s_l and the right descendant s_r. If $label(s) = (x, r, y)$ then the following conditions are fulfilled: there are $u, v \in h(STR_2(d))$ and $z \in S$ such that $label(s_l) = (x, u, z)$, $label(s_r) = (z, v, y)$ and $r = \sigma(u, v)$.

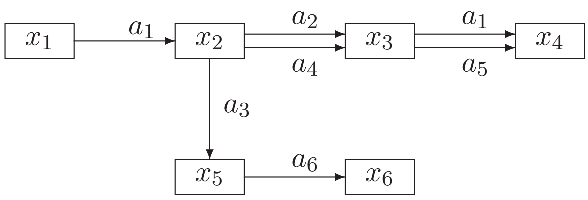


Figure 3.5: A labelled graph

We say that $t(d_s)$ is a **tree over** B. The element $h(c)$ is the label of $root(t(d_s))$ and obviously $h(c) \in B$.

Definition 3.2.3 *If t is a tree over B then we say that t is an accepted tree for $\mathcal{G}$ if $label(root(t)) \in L$. We denote by $AT(\mathcal{G})$ the set of all accepted trees for $\mathcal{G}$.*

Example 3.2.1 *We consider a labeled graph G_0 as in Figure 3.5, where we have a set $S = \{x_1, \ldots, x_6\}$ of nodes and several arcs labeled by $a_1, \ldots, a_6$. We denote by $L_0 = \{a_1, \ldots, a_6\}$ the set of all these labels.*

We consider the following binary relations:

$$\begin{cases} \rho_1 = \{(x_1, x_2), (x_3, x_4)\}; \\ \rho_2 = \{(x_2, x_3)\}; \rho_3 = \{(x_2, x_5)\}; \\ \rho_4 = \{(x_3, x_4)\}; \ \rho_5 = \{(x_5, x_6)\} \end{cases} \tag{3.6}$$

and take $T_0 = \{\rho_1, \rho_2, \rho_3, \rho_4, \rho_5\}$, the surjective mapping $f_0 : L_0 \longrightarrow T_0$ defined by $f_0(a_1) = \rho_1$, $f_0(a_2) = f_0(a_4) = \rho_2$, $f_0(a_3) = \rho_3$, $f_0(a_5) = \rho_4$, $f_0(a_6) = \rho_5$.

Take for the graph represented in Figure 3.5 the following values:

$$u(\rho_1, \rho_3) = \omega_1 = \{(x_1, x_5)\}; \ u(\omega_1, \rho_5) = \omega_2 = \{(x_1, x_6)\};$$

$$u(\rho_3, \rho_5) = \omega_3 = \{(x_2, x_6)\}; \ u(\rho_1, \omega_3) = \omega_2$$

We obtain the following components of the corresponding LSG:

$$T = T_0 \cup \{\omega_1, \omega_2, \omega_3\};$$

$$L = L_0 \cup \{\sigma(a_1, a_3), \sigma(a_3, a_6), \sigma(\sigma(a_1, a_3), a_6), \sigma(a_1, \sigma(a_3, a_6))\}$$

$$f(\sigma(a_1, a_3)) = \omega_1; f(\sigma(a_3, a_6)) = \omega_3$$

$$f(\sigma(\sigma(a_1, a_3), a_6)) = f(\sigma(a_1, \sigma(a_3, a_6))) = \omega_2$$

As we shall see in a separate section, in the inference process we use a path based mechanism. By the above choice of u only the path passing through the nodes x_1, x_2, x_5, x_6 can be used. If we intend to use also the node x_3 then we have to use also ρ_2 in the definition of u.

For the path $d = ([x_1, x_2, x_5, x_6], [a_1, a_3, a_6])$ there are only two structured paths:

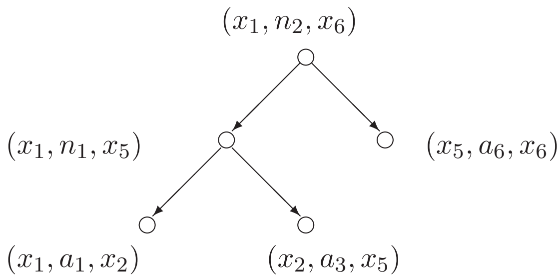


Figure 3.6: The element $t(d_s^1)$

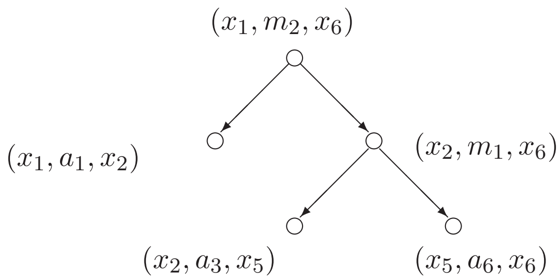


Figure 3.7: The element $t(d_s^2)$

$$d_s^1 = ([x_1, x_2, x_5, x_6], \sigma(\sigma(a_1, a_3), a_6)) \in STR(d)$$

$$d_s^2 = ([x_1, x_2, x_5, x_6], \sigma(a_1, \sigma(a_3, a_6))) \in STR(d)$$

The elements $t(d_s^1)$ and $t(d_s^2)$ are drawn in Figure 3.6 and Figure 3.7 respectively, where:

$$n_2 = \sigma(\sigma(a_1, a_3), a_6), n_1 = \sigma(a_1, a_3)$$

$$m_2 = \sigma(a_1, \sigma(a_3, a_6)), m_1 = \sigma(a_3, a_6)$$

Both $t(d_s^1)$ and $t(d_s^2)$ are trees over B. Moreover, $t(d_s^1) \in AT(\mathcal{G})$ and $t(d_s^2) \in AT(\mathcal{G})$ because $\sigma(\sigma(a_1, a_3), a_6) \in L$, $\sigma(a_1, \sigma(a_3, a_6)) \in L$. In fact, an accepted tree is obtained only from an accepted structured path.

Let be $d_s = ([x_1, \ldots, x_{n+1}], \sigma(v_1, v_2)) \in ASP(\mathcal{G})$, where $n \geq 2$. From $\sigma(v_1, v_2) \in L$ and $L \in Initial(L_0)$ we deduce that $v_1 \in L$ and $v_2 \in L$. We have also

$$label(root(t(d_s))) = (x_1, \sigma(v_1, v_2), x_{n+1})$$

Because $t(d_s)$ is a binary tree, there is two direct descendants of $root(t(d_s))$. We denote by t_l the subtree corresponding to the left descendant and by t_r the subtree defined by the right descendant of $root(t(d_s))$. Obviously, there is one and only one $i \in \{2, \ldots, n\}$ such that

$$front(t_l) = CON_{l=1}^{i-1}(x_l, a_l, x_{l+1})$$

$$front(t_r) = CON_{l=i}^{n}(x_l, a_l, x_{l+1})$$

$$label(root(t_l)) = v_1;\ label(root(t_r)) = v_2$$

Moreover, we have

$$([x_1, \ldots, x_i], v_1) \in ASP(\mathcal{G})$$

$$([x_i, \ldots, x_{n+1}], v_2) \in ASP(\mathcal{G})$$

Thus we proved the following proposition:

Proposition 3.2.1 *For every accepted structured path*

$$([x_1, \ldots, x_{n+1}], \sigma(v_1, v_2)) \in ASP(\mathcal{G})$$

where $n \geq 2$, there is one and only one $i \in \{2, \ldots, n\}$ such that the following two conditions are fulfilled:

$$([x_1, \ldots, x_i], v_1) \in ASP(\mathcal{G})$$

$$([x_i, \ldots, x_{n+1}], v_2) \in ASP(\mathcal{G})$$

In other words, Proposition 3.2.1 states that every accepted structured path over $\mathcal{G}$ can be broken into two accepted structured paths over $\mathcal{G}$. The number i stated in Proposition 1 is named the *break index* for the path d_s and is denoted by $ind(d_s)$. In general, if $v \in B = \bigcup_{k \geq 0} B_k$ then there is k, uniquely determined, such that $v \in B_k$. We denote $|v| = k + 1$ and this means that $|v|$ is the number of elements from L_0 that appear in v. Now, the value $ind(d_s)$ can be expressed as follows:

$$ind(d_s) = |v_1| + 1$$

For example, if $d_s = ([x_1, x_2, x_5, x_6], \sigma(\sigma(a_1, a_3), a_6))$ is an accepted structured path over $\mathcal{G}$ considered in Example 3.2.1, then $ind(d_s) = 3$.

3.2.3 Interpretations of labelled stratified graphs

Consider a stratified graph $\mathcal{G} = (G_0, L, T, u, f)$ over $G_0 = (S, L_0, T_0, f_0)$. In this section we deal with the inference process generated by an accepted structured path over $\mathcal{G}$. A basic concept is introduced in the next definition.

Definition 3.2.4 *An interpretation for $\mathcal{G}$ is a tuple*

$$\Sigma = (Ob, i, D, \mathcal{P})$$

where:

- *Ob is a finite set of objects such that*

$$Card(Ob) = Card(S)$$

- *$i : S \longrightarrow Ob$ is a bijective mapping*
- *$D = (Y, *)$ is a partial algebra; Y is called the domain of Σ and $*$ is a partial binary operation on Y*
- *$\mathcal{P} = \{p_a\}_{a \in L_0}$, where $p_a : Ob \times Ob \longrightarrow Y$*

In general, an interpretation is used to evaluate terms in mathematical logic. In our case, a term will be an element of $ASP(\mathcal{G})$. The evaluation process is described in the next definition.

Definition 3.2.5 *The* **valuation mapping** *generated by Σ is the mapping*

$$val_\Sigma : ASP(\mathcal{G}) \longrightarrow Y$$

defined inductively as follows:

$$\begin{cases} val_\Sigma([x, y], a) = p_a(i(x), i(y)) \\ \\ val_\Sigma(x(1; n+1), \sigma(v_1, v_2)) = val_\Sigma(x(1; i), v_1) * val_\Sigma(x(i; n+1), v_2) \end{cases} \tag{3.7}$$

where $i = ind([x_1, \ldots, x_{n+1}], \sigma(v_1, v_2))$ and $x(i; j) = [x_i, \ldots, x_j]$.

Applying this definition for the path $d_s = ([x_1, x_2, x_5, x_6], \sigma(\sigma(a_1, a_3), a_6))$ considered in the last part of the previous section we obtain:

- $v_1 = \sigma(a_1, a_3); v_2 = a_6; |v_1| = 2; |v_2| = 1$
- $ind(d_s) = 3$
- The value of the mapping val_Σ for d_s:
  $val_\Sigma(d_s) = val_\Sigma([x_1, x_2, x_5], \sigma(a_1, a_3)) * val_\Sigma([x_5, x_6], a_6) = (val_\Sigma([x_1, x_2], a_1) * val_\Sigma([x_2, x_5], a_3)) * val_\Sigma([x_5, x_6], a_6) = (p_{a_1}(i(x_1), i(x_2)) * p_{a_3}(i(x_2), i(x_5))) * p_{a_6}(i(x_5), i(x_6))$

Because a stratified graph is an abstract structure, an inference process and its conclusion are defined with respect to some interpretation. These concepts are stated in the following definition.

Definition 3.2.6 *Consider a stratified graph $\mathcal{G} = (G_0, L, T, u, f)$ over $G_0 = (S, L_0, T_0, f_0)$ and $\Sigma = (Ob, i, D, \mathcal{P})$ an interpretation for $\mathcal{G}$. A pair $(x, y) \in S \times S$ is called* **interrogation**. *For a given interrogation (x, y) we designate by $ASP(x, y)$ the set of all accepted structured paths from x to y in $\mathcal{G}$. The* **answer mapping** *is the mapping*

$$Ans : S \times S \longrightarrow Y \cup \{no\}$$

defined as follows:

$$\begin{cases} Ans(x,y) = no \quad \text{if } ASP(x,y) = \emptyset \\ \\ Ans(x,y) = \{val_\Sigma(d) \mid d \in ASP(x,y)\} \text{ if } ASP(x,y) \neq \emptyset \end{cases} \tag{3.8}$$

The *inference process* generated by $d_s \in ASP(\mathcal{G})$ is the computation performed to obtain $val_\Sigma(d_s)$ by (3.7). The element $val_\Sigma(d_s)$ is the *conclusion* of the corresponding process.

3.2.4 First application: conclusion is given in a natural language

In this section we deal with the case when the answer of an interrogation is given in a natural language. In general, for some simplified cases concerning the meaning of the binary relations, this problem can be modeled by a semantic network. In comparison with a semantic network, there is no restriction concerning the meaning of a binary relation in a LSG.

Consider the labeled graph G_0 represented in Figure 3.5 and take $u = prod_S$. Suppose the following knowledge piece KP_1 is given:
Peter is a friend of Emily. Emily is a teacher. Every teacher likes to drive a car. Emily learns Ann to drive a car. Usually, Emily helps Ann to learn mathematics. Ann likes to play tennis with Tom.
We observe KP_1 contains 6 objects and 6 binary relations. We can establish a connection between the labeled graph represented in Figure 3.5 and KP_1. More precisely, we can define the following interpretation $\Sigma_1 = \{Ob, i, D, \mathcal{P}\}$ for $\mathcal{G}$:

- $Ob = \{Peter, Emily, Ann, Tom, teacher, car\}$
- $i(x_1) = Peter$; $i(x_2) = Emily$; $i(x_3) = Ann$;
  $i(x_4) = Tom$; $i(x_5) = teacher$; $i(x_6) = car$;
- $\mathcal{P} = \{p_{a_i}\}_{i \in \{1,\ldots,6\}}$, where each mapping p_{a_i} is such that $p_{a_i}(x,y)$ is some sentence. For our case we choose the following sentences:

  $p_{a_1}(x,y)=$ *"x is a friend of y"*;

  $p_{a_2}(x,y)=$ *"x learns y to drive a car"*;

  $p_{a_3}(x,y)=$ *"x is a y"*;

  $p_{a_4}(x,y)=$ *"usually x helps y to learn mathematics"*;

  $p_{a_5}(x,y)=$ *"x likes to play tennis with y"*;

  $p_{a_6}(x,y)=$ *"every x likes to drive a y"*
- For every $x, y, z \in Ob$ we define:

  $p_{a_1}(x,y) * p_{a_3}(y,z) = r_1(x,z)$, where $r_1(x,z)=$ *"x is a friend of a z"*

$r_1(x,y) * p_{a_6}(y,z) = r_2(x,z)$, where $r_2(x,z)$= *"x is a friend of a person which likes to drive a z"*

$p_{a_3}(x,y) * p_{a_6}(y,z) = r_3(x,z)$, where $r_3(x,z)$= *"x is a person which likes to drive a z"*

$p_{a_1}(x,y) * r_3(y,z) = r_2(x,z)$

and so on.

Taking Y the set of all these sentences we obtain the partial algebra $(Y, *)$.

Computing the valuation mapping for some particular case we obtain:

$$val_{\Sigma_1}([x_1,x_2,x_5,x_6], \sigma(\sigma(a_1,a_3),a_6)) =$$

$$val_{\Sigma_1}([x_1,x_2,x_5], \sigma(a_1,a_3)) * val_{\Sigma_1}([x_5,x_6],a_6) =$$

$$(val_{\Sigma_1}([x_1,x_2],a_1) * val_{\Sigma_1}([x_2,x_3],a_3)) * val_{\Sigma_1}([x_5,x_6],a_6) =$$

$$(p_{a_1}(i(x_1),i(x_2)) * p_{a_3}(i(x_2),i(x_5))) * p_{a_6}(i(x_5),i(x_6)) =$$

$$r_1(i(x_1),i(x_5)) * p_{a_6}(i(x_5),i(x_6)) = r_2(i(x_1),i(x_6))$$

= *"Peter is a friend of a person which likes to drive a car"*

The same conclusion is obtained if we valuate

$$val_{\Sigma_1}([x_1,x_2,x_5,x_6], \sigma(a_1,\sigma(a_3,a_6)))$$

3.2.5 Second application: conclusion is a geometric image

This section is devoted to the case when the conclusion is a geometric image. Two accepted structured paths for the same path are used such that different conclusions are obtained. To highlight the fact that a stratified graph is an abstract structure we shall use the same stratified graph as in the previous section, but the context is given by other interpretation. So we change the context by choosing another interpretation.

Consider the stratified graph $\mathcal{G}$ from the previous section and the following interpretation $\Sigma_2 = \{Ob, i, D, \mathcal{P}\}$ for $\mathcal{G}$:

- $Ob = \{A_0(k,0), B_0(k{+}3,0), C_0(k{+}5,0), D_0(k{+}7,0), E_0(k{+}11), F_0(k{+}15,0)\}$
Thus Ob contains 6 points of the real plan, where k is some constant.

- $i(x_1) = A_0$; $i(x_2) = B_0$; $i(x_3) = E_0, i(x_4) = F_0, i(x_5) = C_0$; $i(x_6) = D_0$

- $\mathcal{P} = \{p_{a_i}\}_{i \in \{1,\ldots,6\}}$
where each $p_{a_i}(x,y)$ is a geometric figure in the plan $R \times R$, uniquely determined by the points x and y. For our purpose we choose the following figures:

  $p_{a_1}(x,y)$ is the square having the diagonal xy;

$p_{a_3}(x, y)$ is the circle of diameter xy;

$p_{a_6}(x, y)$ is the regular triangle such that xy is one of its altitude and x is a vertex of the triangle;

• The space Y is the set of all geometric figures of the plan $R \times R$ such that each figure has an axis of symmetry (a circle, a regular or isosceles triangle etc). Moreover, if we obtain $\alpha * \beta$ for some $\alpha, \beta \in Y$ then the following condition is satisfied: α, β and $\alpha * \beta$ have the same axis of symmetry.

In order to define the elements of Y we use the notation $pl_r(X_1, \dots, X_r)$ to designate the polygonal line passing successively by the points $X_1, \dots, X_r$. Based on this convention, p_{a_1} and p_{a_6} can be expressed by means of pl_5 and pl_4 respectively.

We relieve the following aspect of the operation $*$ from Y: if X and Y are two polygonal lines then $X * Y$ is, in general, the "least" polygon containing both X and Y. Now we give a detailed definition:

$p_{a_1}(M, Z) * p_{a_3}(Z, V) = pl_4(A, B, C, A)$, where A, B, C, Z, V, M are represented in Figure 3.8a)

$pl_4(A, B, C, A) * p_{a_6}(E, V) = pl_7(A, G, D, F, R, B, A)$, where all the points are represented in Figure 3.8b)

$p_{a_3}(M, A) * p_{a_6}(A, H) = pl_5(D, E, F, G, D)$ as we show in Figure 3.9a). The figure DEFG is an isosceles trapezoid and is obtained as follows: the regular triangle ABC is uniquely determined by the points A and H; the straight line MG, which is parallel with AB, cuts the circle in D; similarly is obtained the point E.

$p_{a_1}(A, C) * pl_4(E, F, G, H) = pl_8(A, B, E, F, G, H, D, A)$, where all the points are represented in Figure 3.9b). The figure EFGH is an isosceles trapezoid having a common axis of symmetry with the square ABCD: the straight line passing by the points A and C.

Computing the valuation mapping for the accepted structured path

$$([x_1, x_2, x_5, x_6], \sigma(\sigma(a_1, a_3), a_6))$$

we obtain:

$val_{\Sigma_2}([x_1, x_2, x_5, x_6], \sigma(\sigma(a_1, a_3), a_6))) =$

$val_{\Sigma_2}([x_1, x_2, x_5], \sigma(a_1, a_3)) * val_{\Sigma_2}([x_5, x_6], a_6) =$

$(val_{\Sigma_2}([x_1, x_2], a_1) * val_{\Sigma_2}([x_2, x_5], a_3)) * val_{\Sigma_2}([x_5, x_6], a_6) =$

$(p_{a_1}(i(x_1), i(x_2)) * p_{a_3}(i(x_2), i(x_5))) * p_{a_6}(i(x_5), i(x_6)) =$

$(p_{a_1}(A_0, B_0) * p_{a_3}(B_0, C_0)) * p_{a_6}(C_0, D_0) = pl_4(D, M, H, D) * p_{a_6}(C_0, D_0) =$

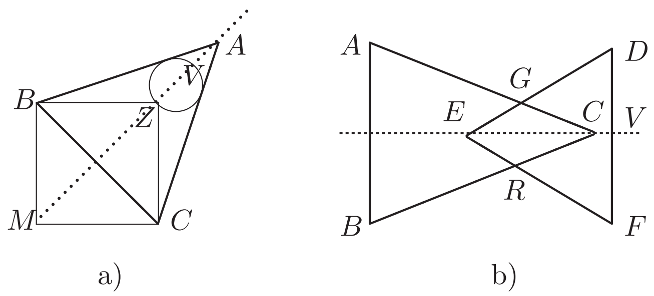


Figure 3.8: Operation ∗

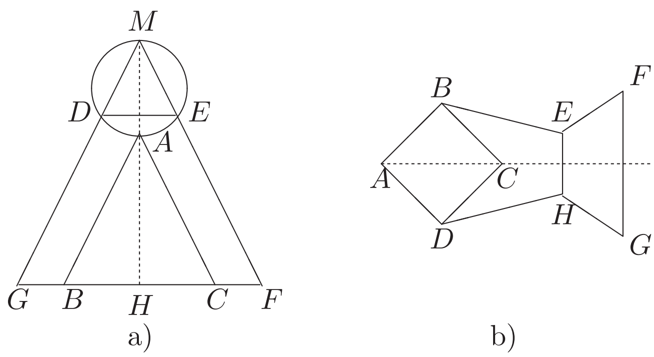


Figure 3.9:

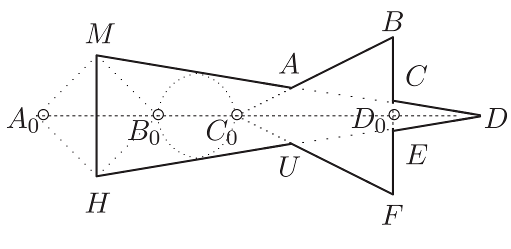


Figure 3.10:

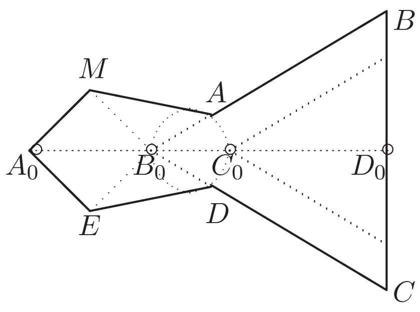


Figure 3.11:

$$pl_4(D, M, H, D) * pl_4(C_0, B, F, C_0) = pl_{10}(M, A, B, C, D, E, F, U, H, M)$$

where all the points are represented in Figure 3.10.

Now we compute the valuation mapping for the accepted structured path

$$([x_1, x_2, x_5, x_6], \sigma(a_1, \sigma(a_3, a_6)))$$

and obtain successively:

$$val_{\Sigma_2}([x_1, x_2, x_5, x_6], \sigma(a_1, \sigma(a_3, a_6)))) =$$

$$val_{\Sigma_2}([x_1, x_2], a_1) * val_{\Sigma_2}([x_2, x_5, x_6], \sigma(a_3, a_6)) =$$

$$val_{\Sigma_2}([x_1, x_2], a_1) * (val_{\Sigma_2}([x_2, x_5], a_3)) * val_{\Sigma_2}([x_5, x_6], a_6)) =$$

$$p_{a_1}(i(x_1), i(x_2)) * (p_{a_3}(i(x_2),\ i(x_5)) * p_{a_6}(i(x_5), i(x_6))) =$$

$$p_{a_1}(A_0, B_0) * (p_{a_3}(B_0, C_0) * p_{a_6}(C_0, D_0)) =$$

$$p_{a_1}(A_0, B_0) * pl_5(A, B, C, D, A) = pl_8(M, A, B, C, D, E, A_0, M)$$

where all the points are represented in Figure 3.11.

We shall remark that we considered the path

$$d = ([x_1, x_2, x_5, x_6], [a_1, a_3, a_6])$$

and the following accepted structured paths of d:

$$d_1 = ([x_1, x_2, x_5, x_6], \sigma(\sigma(a_1, a_3), a_6))$$

$$d_2 = ([x_1, x_2, x_5, x_6], \sigma(a_1, \sigma(a_3, a_6)))$$

Computing the valuation mapping for these structured paths with respect to the interpretation Σ_2 we obtain different conclusions. Applying (3.8) we conclude that $Ans(x_1, x_6)$ contains two geometric figures, but $Ans(x_6, x_1) = no$. The method presented in this section can be used in image synthesis.

3.3 An application for greatest distinguished *LSGs*

We consider a directed graph $G_0 = (S, \Gamma)$, where $S = \{x_1, \ldots, x_m\}$ is the set of its nodes. Let $A = \{a_1, \ldots, a_k\}$ be a set of properties or attributes for the elements of Γ. The attributes $a_1, \ldots, a_k$ may represent colours or other properties. For example, if $A = \{black, yellow, green\}$ then the arcs of G_0 may be coloured with *black*, *yellow* or *green*. Other attributes may represent properties such as *closed road, works in progress, rain* and so on.

An *attribute graph* is a tuple $(G_0, A, attr)$, where

- $G_0 = (S, \Gamma)$ is a directed graph
- $A = \{a_1, \ldots, a_k\}$ is the set of attribute names
- $attr : \Gamma \longrightarrow 2^A \setminus \{\emptyset\}$ is a mapping such that $attr(x_i, x_j)$ is the set of all attributes associated to $(x_i, x_j) \in \Gamma$
- for each $i \in \{1, \ldots, k\}$ there is $(x_p, x_q) \in \Gamma$ such that $a_i \in attr(x_p, x_q)$

A *path* in an attribute graph $(G_0, A, attr)$ is a pair $([x_1, \ldots, x_n], [a_1, \ldots, a_{n-1}])$, where $(x_i, x_{i+1}) \in \Gamma$ and $a_i \in attr(x_i, x_{i+1})$ for $i \in \{1, \ldots, n-1\}$.

We consider the mappings $R : A \longrightarrow 2^A$, $K : A \longrightarrow 2^A$ and a subset $N_0 \subseteq A$. By definition, an *accepted path* from y_1 to y_n is a path $([y_1, \ldots, y_n], [b_1, \ldots, b_{n-1}])$ in $(G_0, A, attr)$ such that $b_1 \in A \setminus N_0$ and $b_{i+1} \in R(b_i) \setminus K(b_i)$ for $i \in \{1, \ldots, n-2\}$. The tuple (R, K, N_0) defines *the restrictions* imposed on accepted paths. We shall denote by $Path_{acc}(y_1, y_n)$ the set of the accepted paths from y_1 to y_n.

Given two arbitrary nodes $y, z \in S$ and a natural number r the following problems arise:

P1) Decide whether or not the set $\bigcup_{s=0}^{r} Path_{acc}^{s}(y, z)$ is a non-empty set, where $Path_{acc}^{s}(y, z)$ denotes the set of all accepted paths from y to z that contain exactly s intermediary nodes. In other words, the problem is to decide whether or not there is an accepted path from y to z containing at most r intermediary nodes.

P2) In the affirmative case, find all these paths.

In order to transpose this problem in terms of labelled graphs we take $L_0 = A = \{a_1, \ldots, a_k\}$ and for each $i \in \{1, \ldots, k\}$ we consider the following binary relation on S:

$$f_0(a_i) = \{(x_p, x_q) \in \Gamma \mid a_i \in attr(x_p, x_q)\} \tag{3.9}$$

We consider $T_0 = \{\rho \mid \exists a_i \in A : f_0(a_i) = \rho\}$. We obtain the labelled graph $G = (S, L_0, T_0, f_0)$, where $f_0 : L_0 \longrightarrow T_0$ is the mapping defined in (3.9).
In order to exemplify this situation we consider the directed graph $G_0 = (S, \Gamma)$, where

$S = \{x_1, x_2, x_3, x_4\}$

$\Gamma = \{(x_1, x_2), (x_2, x_3), (x_3, x_4)\}$

Let us take:

$A = \{a_1, a_2, a_3\}$

$attr(x_1, x_2) = \{a_1, a_2\}$; $attr(x_2, x_3) = \{a_1, a_2\}$; $attr(x_3, x_4) = \{a_3\}$

We obtain the attribute graph $(G_0, A, attr)$. For this example we obtain:

$f_0(a_1) = f_0(a_2) = \{(x_1, x_2), (x_2, x_3)\} = \rho_1$

$f_0(a_3) = \{(x_3, x_4)\} = \rho_2$

In this case we have $L_0 = \{a_1, a_2, a_3\}$ and $T_0 = \{\rho_1, \rho_2\}$.

We consider the greatest distinguished LSG over G, that is $DR(\theta_G(prod_S))$:

$$\mathcal{G}(\theta_G(prod_S)) = (G, L, T, \theta_G(prod_S), f)$$

where $(L, T, f) = env_G(\theta_G(prod_S))$. This is the LSG used to solve the problems P1 and P2. An essential aspect is that only some elements of L and T are used and the selection of these elements is described in what follows.

For every $\alpha \in L$ we define $trace(\alpha)$ as follows:

(1) if $\alpha \in L_0$ then $trace(\alpha) = (\alpha)$

(2) if $\alpha = \sigma(u, v)$ then $trace(\alpha) = (p, q)$, where $trace(u) = (p)$ and $trace(v) = (q)$

For example, $trace(a) = (a)$ for $a \in L_0$, $trace(\sigma(a, b)) = (a, b)$, $trace(\sigma(\sigma(a, b), \sigma(a, c))) = (a, b, a, c)$ and so on. We observe that if $\alpha \in L$ then there are $s \geq 1$ and $a_{i_1}, \ldots, a_{i_s} \in L_0$ such that $trace(\alpha) = (a_{i_1}, \ldots, a_{i_s})$. If this is the case, then we denote $last(\alpha) = a_{i_s}$.

If $U \subseteq L$ then we denote

$$U^{\otimes} = \{(u, v) \mid u \in U, v \in R(last(u)) \setminus K(last(u))\} \tag{3.10}$$

As we stated above, only some of the elements of L will be used to solve the problems $P1$ and $P2$. We shall denote by L_{us} the set of these elements, which are called *useful labels*. In order to extract the set L_{us} we define recursively

$$\begin{cases} Q_0 = L_0 \setminus N_0 \\ Q_{n+1} = \{\sigma(u, v) \mid (u, v) \in Q_n^{\otimes}, (f(u), f(v)) \in dom(\theta_G(prod_S))\} \end{cases} \tag{3.11}$$

and take $L_{us} = \bigcup_{n=0}^{r} Q_n$.

Remark 3.3.1 *If* $\alpha \in Q_n$ *then* $trace(\alpha) \in L_0^{n+1}$.

We verify by induction on n that $Q_n \subseteq L$ and thus we shall have $L_{us} \subseteq L$. Obviously $Q_0 \subseteq L$. Suppose $Q_n \subseteq L$ for some n. If $\sigma(u,v) \in Q_{n+1}$ then $(u,v) \in Q_n^{\otimes}$ and $(f(u), f(v)) \in dom(\theta_G(prod_S))$. Applying (3.10) and the inductive assumption we obtain $u \in Q_n \subseteq L$, $v \in R(last(u)) \setminus K(last(u)) \subseteq A = L_0 \subseteq L$. Using (1.7) we obtain $(u,v) \in dom(\sigma_L)$, therefore $\sigma(u,v) \in L$. Thus $Q_n \subseteq L$ for each n.

Applying (2.16) for $u = prod_S$ we obtain

$$dom(\theta_G(prod_S)) = (H_G(prod_S) \times H_G(prod_S)) \cap dom(prod_S)$$

therefore the following relations are equivalent:

$$(\rho_i, \rho_j) \in dom(\theta_G(prod_S)) \tag{3.12}$$

$$\rho_i, \rho_j \in H_G(prod_S),\ \ \rho_i \circ \rho_j \neq \emptyset \tag{3.13}$$

Because $env_G(\theta_G(prod_S)) = (L, T, f)$, we have $f(L) = T = H_G(\theta_G(prod_S)) = H_G(prod_S)$. Based on (3.12) and (3.13) we deduce that the following relations are equivalent for $u, v \in L$:

$$(f(u), f(v)) \in dom(\theta_G(prod_S)) \tag{3.14}$$

$$f(u) \circ f(v) \neq \emptyset \tag{3.15}$$

Consequently, (3.11) can be written equivalently

$$\begin{cases} Q_0 = L_0 \setminus N_0 \\ Q_{n+1} = \{\sigma(u,v) \mid (u,v) \in Q_n^{\otimes}, f(u) \circ f(v) \neq \emptyset\} \end{cases} \tag{3.16}$$

and in applications we shall use (3.16) instead of (3.11).

For each $(x,y) \in S \times S$ we denote

$$Us(x,y) = \{\alpha \in L_{us} \mid (x,y) \in f(\alpha)\} \tag{3.17}$$

For a given sequence $b_1, b_2, \ldots$ of elements from L_0 we define

$$seq(b_1) = b_1$$

$$seq(b_1 \ldots b_{j+1}) = \sigma(seq(b_1 \ldots b_j), b_{j+1})$$

Thus we have $seq(b_1, b_2) = \sigma(b_1, b_2)$, $seq(b_1, b_2, b_3) = \sigma(\sigma(b_1, b_2), b_3)$ and so on.

We observe that if $\alpha \in Q_n$ then there are $b_1, \ldots, b_{n+1} \in L_0$ such that $\alpha = seq(b_1, \ldots, b_{n+1})$. Based on the fact that $L_{us} = \bigcup_{n=0}^{r} Q_n$ we deduce that for each $\alpha \in L_{us}$ there are $n \leq r$ and $b_1, \ldots, b_{n+1}$ such that $\alpha = seq(b_1, \ldots, b_{n+1})$.

Proposition 3.3.1 *The following properties are fulfilled:*

1) *If* $\alpha = seq(b_1, \ldots, b_{s+1}) \in Us(x_i, x_j)$ *then* $s \leq r$ *and there are* $z_1, \ldots, z_s \in S$ *such that* $([x_i, z_1, \ldots, z_s, x_j], [b_1, \ldots, b_{s+1}]) \in Path^s_{acc}(x_i, x_j)$.

2) Let be $s \leq r$ and $b_1, \ldots, b_{s+1} \in L_0$. If $([y_1, \ldots, y_{s+2}], [b_1, \ldots, b_{s+1}]) \in Path^s_{acc}(y_1, y_{s+2})$ then $seq(b_1 \ldots b_j) \in Us(y_1, y_{j+1})$ for $j = 1, \ldots, s+1$.

Proof. Let us prove the first property. We consider $\alpha = seq(b_1, \ldots, b_{s+1}) \in Us(x_i, x_j)$. As we observed before we have $s \leq r$. We prove that there are $z_1, \ldots, z_s \in S$ such that $([x_i, z_1, \ldots, z_s, x_j], [b_1, \ldots, b_{s+1}]) \in Path_{acc}\ (x_i, x_j)$. We proceed by induction on s.
1) *Initial step.* If $s = 0$ then $\alpha = b_1 \in Us(x_i, x_j) \cap L_0 \subseteq L_{us} \cap L_0 = Q_0 = L_0 \setminus N_0$. By the definition of $Us(x_i, x_j)$ we deduce $(x_i, x_j) \in f(b_1) = f_0(b_1)$, therefore $(x_i, x_j) \in \Gamma$ and $b_1 \in attr(x_i, x_j)$. Thus, $([x_i, x_j], [b_1]) \in Path_{acc}(x_i, x_j)$.
2) *Inductive step.* We suppose the property is true for some $s < r$ and take $\alpha = seq(b_1, \ldots, b_{s+2}) \in Us(x_i, x_j)$. Because $Us(x_i, x_j) \subseteq L_{us} = \bigcup_{n=0}^{r} Q_n$, it follows that $\alpha \in Q_{s+1}$. Using (3.10) and (3.16) we deduce that $\alpha = \sigma(\beta, b_{s+2})$, $\beta = seq(b_1, \ldots, b_{s+1}) \in Q_s$ and $b_{s+2} \in R(last(\beta)) \setminus K(last(\beta))$. Moreover, $f(\beta) \circ f_0(b_{s+2}) \neq \emptyset$. From $\alpha \in Us(x_i, x_j)$ we deduce $(x_i, x_j) \in f(\alpha)$. But f is a morphism in the structure of $\mathcal{G}(\theta_G(prod_S))$, therefore $f(\alpha) = f(\sigma(\beta, b_{s+2})) = \theta_G(prod_S)(f(\beta), f_0(b_{s+2})) = f(\beta) \circ f_0(b_{s+2})$. It follows that there is $z \in S$ such that $(x_i, z) \in f(\beta)$ and $(z, x_j) \in f_0(b_{s+2})$. From $\beta \in Q_s \subseteq L_{us}$ and $(x_i, z) \in f(\beta)$, by (3.17) we obtain $\beta \in Us(x_i, z)$. Applying the inductive assumption for β it follows that there are $z_1, \ldots, z_s \in S$ such that

$$([x_i, z_1, \ldots, z_s, z], [b_1, \ldots, b_{s+1}]) \in Path_{acc}(x_i, z)$$

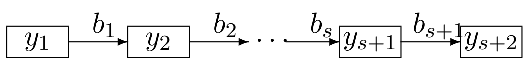
It follows that

$$([x_i, z_1, \ldots, z_s, z, x_j], [b_1, \ldots, b_{s+1}, b_{s+2}]) \in Path_{acc}(x_i, x_j)$$

because $(z, x_j) \in f_0(b_{s+2})$ and $b_{s+2} \in R(b_{s+1}) \setminus K(b_{s+1})$. Thus the proof of the first part is finished.
In order to prove the second part we suppose

$$([y_1, \ldots, y_{s+2}], [b_1, \ldots, b_{s+1}]) \in Path^s_{acc}(y_1, y_{s+2})$$

where $s \leq r$. Graphically, we have the following path in G:



We denote $l_m = seq(b_1 \ldots b_m)$ for $m = 1, \ldots, s+1$. We prove that $l_m \in Us(y_1, y_{m+1}) \cap Q_{m-1}$ for $m = 1, \ldots, s+1$. To do this we proceed by induction on m. We have $b_1 \in L_0 \setminus N_0 = Q_0 \subseteq L_{us}$ and $(y_1, y_2) \in f_0(b_1)$ by (3.9). Thus, $l_1 \in Us(y_1, y_2) \cap Q_0$ and the property is true for $m = 1$. We suppose $l_m \in Us(y_1, y_{m+1}) \cap Q_{m-1}$ for some $m \leq s$. We have $b_{m+1} \in R(b_m) \setminus K(b_m) = R(last(l_m)) \setminus K(last(l_m))$, therefore $(l_m, b_{m+1}) \in Q^{\otimes}_{m-1}$. Moreover, from $l_m \in Us(y_1, y_{m+1})$ we deduce

$$(y_1, y_{m+1}) \in f(l_m) \tag{3.18}$$

We have also

$$(y_{m+1}, y_{m+2}) \in f_0(b_{m+1}) = f(b_{m+1}) \tag{3.19}$$

by (3.9). From (3.18) and (3.19) we deduce

$$(f(l_m), f(b_{m+1})) \in dom(prod_S)$$

and

$$(y_1, y_{m+2}) \in prod_S(f(l_m), f(b_{m+1}))$$

The relation (3.15) is satisfied by the elements l_m and b_{m+1}, therefore by (3.14) we shall have $(f(l_m), f(b_{m+1})) \in dom(\theta_G(prod_S))$. Using again (3.18) and (3.19) we can now state that $(y_1, y_{m+2}) \in \theta_G(prod_S)(f(l_m), f(b_{m+1}))$. Taking into account the property $(l_m, b_{m+1}) \in Q_{m-1}^{\otimes}$ and using (3.11) we deduce $\sigma(l_m,\ b_{m+1}) \in Q_m$, that is $l_{m+1} \in Q_m \subseteq L_{us}$. We have $(y_1, y_{m+2}) \in \theta_G(prod_S)(f(l_m), f(b_{m+1})) = f(\sigma(l_m, b_{m+1})) = f(l_{m+1})$ and $l_{m+1} \in Q_m \subseteq L_{us}$, therefore $l_{m+1} \in Us(y_1,\ y_{m+2}) \cap Q_m$. The proof is complete. ∎

Corollary 3.3.1 *$\bigcup_{s=0}^{r} Path_{acc}^{s}(x_i, x_j) \neq \emptyset$ if and only if $Us(x_i, x_j) \neq \emptyset$.*

Thus the problem $P1$ is solved by Corollary 3.3.1. In order to solve the problem $P2$ we use the following property:

Proposition 3.3.2 *Let be $s \leq r$ and $b_1, \ldots, b_{s+1} \in L_0$. The following conditions are equivalent:*

$$([x_i, z_1, \ldots, z_s, x_j], [b_1, \ldots, b_{s+1}]) \in Path_{acc}^{s}(x_i, x_j) \tag{3.20}$$

$$\begin{cases} seq(b_1 \ldots b_{s+1}) \in Us(x_i, x_j) \\ (x_i, z_1) \in f_0(b_1), (z_1, z_2) \in f_0(b_2), \ldots, (z_s, x_j) \in f_0(b_{s+1}) \end{cases} \tag{3.21}$$

Proof. If (3.20) is true then $seq(b_1 \ldots b_{s+1}) \in Us(x_i, x_j)$ by Proposition 3.3.1 and

$$\begin{cases} (x_i, z_1) \in \Gamma, b_1 \in attr(x_i, z_1), \\ (z_1, z_2) \in \Gamma, b_2 \in attr(z_1, z_2), \\ \ldots\ldots\ldots \\ (z_s, x_j) \in \Gamma, b_{s+1} \in attr(z_s, x_j) \end{cases} \tag{3.22}$$

Equivalently we have (3.21). Conversely, if (3.21) is satisfied then (3.22) is satisfied from the definition of f_0. But $seq(b_1 \ldots b_{s+1}) \in Us(x_i, x_j)$ therefore $seq(b_1 \ldots b_{s+1}) \in Q_s$. It follows that $b_{s+1} \in R(b_s) \setminus K(b_s)$. Moreover, $seq(b_1 \ldots\ b_s) \in Q_{s-1}, \ldots,$ $seq(b_1, b_2) \in Q_1$. Thus, $b_j \in R(b_{j-1}) \setminus K(b_{j_1})$ for $j = s+1, \ldots, 2$, $b_1 \in Q_0 = L_0 \setminus N_0$ and therefore we have (3.20). ∎

Remark 3.3.2 *We observe that the second part of the condition (3.21) gives an ordinary path $([x_i,\ z_1, \ldots, z_s, x_j], [b_1, \ldots,\ b_{s+1}])$ in G_0. Thus, Proposition 3.3.2 states that the accepted paths from x_i to x_j are exactly the ordinary paths which are "guided" by the elements of $Us(x_i, x_j)$.*

In conclusion, in order to solve the problems $P1$ and $P2$ we can apply the following steps:

Step 1: compute $Q_0, Q_1, \ldots, Q_r$ and take $L_{us} = \bigcup_{n=0}^{r} Q_n$

Step 2: compute the values of the mapping f only for the elements of L_{us}

Step 3: let be x, y two nodes; take $Us(x,y) = \{\alpha \in L_{us} \mid (x,y) \in f(\alpha)\}$

Step 4: if $Us(x,y) = \emptyset$ then $\bigcup_{s=0}^{r} Path_{acc}^{s}(x_i, x_j) = \emptyset$

Step 5: otherwise, for each $\alpha \in Us(x,y)$ compute all the sequences $z_1, \ldots, z_s \in S$ such that $(x, z_1) \in f_0(b_1), (z_1, z_2) \in f_0(b_2), \ldots, (z_s, y) \in f_0(b_{s+1})$, where $\alpha = seq(b_1 \ldots b_{s+1})$

Now it is not difficult to append several steps to obtain other properties such as the length of the shortest path and so on.

3.4 Collaboration between distinguished representatives

In this section we present a simple application, which can be stated shortly as follows: given $\mathcal{G}(u)$ and $\mathcal{G}(v)$ use $\mathcal{G}(u \vee v)$. In this application we use a "graphs merging" operation for two labelled graphs. In order to realize this aim we consider the following problem: *the companies C_1 and C_2 accomplish transport services between some centers of the same county; the conveyance is performed by goods trains such that if a final center for a company is encountered then in order to continue the movement by means of other company, the goods are moved to the headquarter of the corresponding company; by some agreement which stipulates that C_1 and C_2 become members of a common association CA, the goods arrived to a final center for some company are taken over directly by the other company; the quality of the services are known for each company. The problem is the following: for a given pair (x, y) of centers is there a direct conveyance realized by CA?*

The services accomplished by C_i are described by a labelled graph G_i ($i = 1, 2$). In order to obtain a satisfactory description we have to consider the mappings $u = \theta_{G_1}(prod_S)$, $v = \theta_{G_2}(prod_S)$ and $u \vee v$. We are interested to integrate $DR(u)$ and $DR(v)$ in $DR(u \vee v)$. Some collaboration between $DR(u)$ and $DR(v)$ will be obtained in a natural manner but in order to impose additional collaboration we have to complete the graph $DR(u \vee v)$. The description of these operations is the aim of this section.

In order to fix the ideas we shall consider the labelled graphs G_1 from Figure 3.12 and G_2 from Figure 3.13.

We shall take:

1) $G_1 = (S, L_0^{(1)}, T_0^{(1)}, f_0^{(1)})$ where

- $S = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$
- $L_0^{(1)} = \{a_1, b_1, c_1\}$
- $T_0^{(1)} = \{\rho_1, \rho_2, \rho_3\}$, where
  $\rho_1 = \{(x_1, x_2), (x_3, x_4)\},\ \rho_2 = \{(x_2, x_3), (x_5, x_6)\},\ \rho_3 = \{(x_6, x_7)\}$
- $f_0(a_1) = \rho_1,\ f_0(b_1) = \rho_2,\ f_0(c_1) = \rho_3$

Taking $u = \theta_{G_1}(prod_S)$ we obtain Table 3.1, where x denotes the fact that the corresponding relations can not be composed by the product operation and

$$\mu_1 = \{(x_1, x_3)\},\ \mu_2 = \{(x_2, x_4)\},\ \mu_3 = \{(x_5, x_7)\},\ \mu_4 = \{(x_1, x_4)\}$$

Computing the components of $\mathcal{G}(G_1, u) = (G_1, L^{(1)}, T^{(1)}), f^{(1)}$ we obtain:

- $D_0^{(1)} = L_0^{(1)} = \{a_1/\rho_1, b_1/\rho_2, c_1/\rho_3\}$
- $D_1^{(1)} = \{\sigma(a_1, b_1)/\mu_1, \sigma(b_1, a_1)/\mu_2, \sigma(b_1, c_1)/\mu_3\}$
- $D_2^{(1)} = \{\sigma(a_1, \sigma(b_1, a_1))/\mu_4, \sigma(\sigma(a_1, b_1), a_1)/\mu_4\}$
- $L^{(1)} = D_0^{(1)} \cup D_1^{(1)} \cup D_2^{(1)}$
- $T^{(1)} = \{\rho_1, \rho_2, \rho_3, \mu_1, \mu_2, \mu_3, \mu_4\}$

where we denoted by α/ρ the property $f^{(1)}(\alpha) = \rho$.

2) $G_2 = (S, L_0^{(2)}, T_0^{(2)}, f_0^{(2)})$ where

- $S = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$
- $L_0^{(2)} = \{a_2, b_2, c_2\}$
- $T_0^{(2)} = \{\rho_1, \rho_2, \rho_4\}$, where $\rho_4 = \{(x_4, x_5)\}$
- $f_0(a_2) = \rho_1,\ f_0(b_2) = \rho_2,\ f_0(c_2) = \rho_4$

Taking $v = \theta_{G_2}(prod_S)$ we obtain Table 3.2, where

$$\mu_5 = \{(x_1, x_5)\};\ \mu_6 = \{(x_2, x_5)\};\ \mu_7 = \{(x_2, x_6)\}$$
$$\mu_8 = \{(x_3, x_6)\};\ \mu_{11} = \{(x_1, x_6)\}$$

If we compute the components of $\mathcal{G}(G_2, v) = (G_2, L^{(2)}, T^{(2)}, v, f^{(2)})$ we obtain:

- $D_0^{(2)} = L_0^{(2)} = \{a_2/\rho_1, b_2/\rho_2, c_2/\rho_4\}$
- $D_1^{(2)} = \{\sigma(a_2, b_2)/\mu_1, \sigma(a_2, c_2)/\nu_1, \sigma(b_2, a_2)/\mu_2, \sigma(c_2, b_2)/\nu_2\}$
- $D_2^{(2)} = \{\sigma(a_2, \sigma(b_2, a_2))/\mu_4, \sigma(a_2, \sigma(c_2, b_2))/\mu_8, \sigma(b_2, \sigma(a_2, c_2))/\mu_6,$
  $\sigma(\sigma(a_2, b_2), a_2)/\mu_4,\ \sigma(\sigma(a_2, b_2), \sigma(a_2, c_2))/\mu_5, \sigma(\sigma(b_2, a_2), c_2)/\mu_6,$
  $\sigma(\sigma(b_2, a_2), \sigma(c_2, b_2))/\mu_7, \sigma(\sigma(a_2, c_2), b_2)/\mu_8\}$
- $D_3^{(2)} = \{\sigma(a_2, \sigma(b_2, \sigma(a_2, c_2)))/\mu_5, \sigma(a_2, \sigma(\sigma(b_2, a_2), c_2))/\mu_5,$

u	ρ_1	ρ_2	ρ_3	μ_1	μ_2	μ_3	μ_4
ρ_1	x	μ_1	x	x	μ_4	x	x
ρ_2	μ_2	x	μ_3	x	x	x	x
ρ_3	x	x	x	x	x	x	x
μ_1	μ_4	x	x	x	x	x	x
μ_2	x	x	x	x	x	x	x
μ_3	x	x	x	x	x	x	x
μ_4	x	x	x	x	x	x	x

Table 3.1: The mapping u for G_1

$\sigma(a_2, \sigma(\sigma(b_2, a_2), \sigma(c_2, b_2)))/\mu_{11}, \sigma(b_2, \sigma(a_2, \sigma(c_2, b_2)))/\mu_7,$
$\sigma(b_2, \sigma(\sigma(a_2, c_2), b_2))/\mu_7,\ \sigma(\sigma(a_2, b_2), \sigma(\sigma(a_2, c_2), b_2))/\mu_{11},$
$\sigma(\sigma(a_2, b_2), \sigma(a_2, \sigma(c_2, b_2)))/\mu_{11}, \sigma(\sigma(a_2, \sigma(b_2, a_2)), c_2)/\mu_5,$
$\sigma(\sigma(\sigma(a_2, b_2), a_2), c_2)/\mu_5,\ \sigma(\sigma(a_2, \sigma(b_2, a_2)), \sigma(c_2, b_2))/\mu_{11},$
$\sigma(\sigma(\sigma(a_2, b_2), a_2), \sigma(c_2, b_2))/\mu_{11}, \sigma(\sigma(\sigma(a_2, b_2), \sigma(a_2, c_2)), b_2)/\mu_{11},$
$\sigma(\sigma(b_2, \sigma(a_2, c_2)), b_2)/\mu_7, \sigma(\sigma(\sigma(b_2, a_2), c_2), b_2)/\mu_7\}$

- $L^{(2)} = D_0^{(2)} \cup D_1^{(2)} \cup D_2^{(2)} \cup D_3^{(2)}$
- $T^{(2)} = T_0^{(2)} \cup \{\mu_1, \mu_2, \nu_1, \nu_2, \mu_4, \mu_5, \mu_6, \mu_7, \mu_8, \mu_{11}\}$

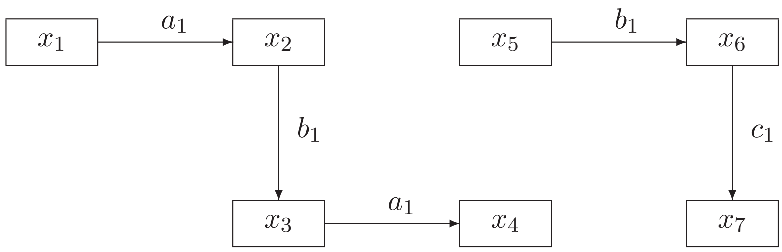


Figure 3.12: Labelled graph G_1

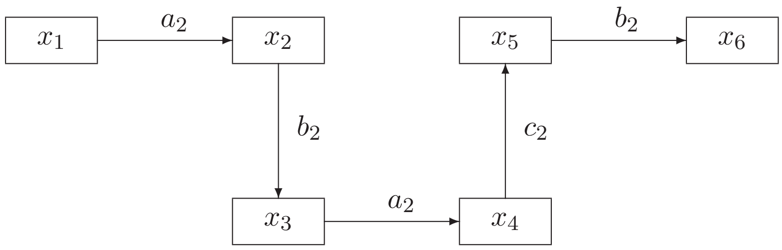


Figure 3.13: Labelled graph G_2

v	ρ_1	ρ_2	ρ_4	μ_1	μ_2	ν_1	ν_2	μ_4	μ_5	μ_6	μ_7	μ_8	μ_{11}
ρ_1	x	μ_1	ν_1	x	μ_4	x	μ_8	x	x	μ_5	μ_{11}	x	x
ρ_2	μ_2	x	x	x	x	μ_6	x	x	x	x	x	μ_7	x
ρ_4	x	ν_2	x	x	x	x	x	x	x	x	x	x	x
μ_1	μ_4	x	x	x	x	μ_5	x	x	x	x	x	μ_{11}	x
μ_2	x	x	μ_6	x	x	x	μ_7	x	x	x	x	x	x
ν_1	x	μ_8	x	x	x	x	x	x	x	x	x	x	x
ν_2	x	x	x	x	x	x	x	x	x	x	x	x	x
μ_4	x	x	μ_5	x	x	x	μ_{11}	x	x	x	x	x	x
μ_5	x	μ_{11}	x	x	x	x	x	x	x	x	x	x	x
μ_6	x	μ_7	x	x	x	x	x	x	x	x	x	x	x
μ_7	x	x	x	x	x	x	x	x	x	x	x	x	x
μ_8	x	x	x	x	x	x	x	x	x	x	x	x	x
μ_{11}	x	x	x	x	x	x	x	x	x	x	x	x	x

Table 3.2: The mapping v for G_2

$u \vee v$	ρ_1	ρ_2	ρ_3	ρ_4	μ_1	μ_2	μ_3	ν_1	ν_2	μ_4	μ_5	μ_6	μ_7	μ_8	μ_{11}
ρ_1	x	μ_1	x	ν_1	x	μ_4	x	x	μ_8	x	x	μ_5	μ_{11}	x	x
ρ_2	μ_2	x	μ_3	x	x	x	x	μ_6	x	x	x	x	x	μ_7	x
ρ_3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
ρ_4	x	ν_2	x	x	x	x		x	x	x	x	x	x	x	x
μ_1	μ_4	x	x	x	x	x	x	μ_5	x	x	x	x	x	μ_{11}	x
μ_2	x	x	x	μ_6	x	x	x	x	μ_7	x	x	x	x	x	x
μ_3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
ν_1	x	μ_8	x	x	x	x		x	x	x	x	x	x	x	x
ν_2	x	x		x	x	x	x	x	x	x	x	x	x	x	x
μ_4	x	x	x	μ_5	x	x	x	x	μ_{11}	x	x	x	x	x	x
μ_5	x	μ_{11}	x	x	x	x		x	x	x	x	x	x	x	x
μ_6	x	μ_7	x	x	x	x		x	x	x	x	x	x	x	x
μ_7	x	x		x	x	x	x	x	x	x	x	x	x	x	x
μ_8	x	x		x	x	x	x	x	x	x	x	x	x	x	x
μ_{11}	x	x		x	x	x	x	x	x	x	x	x	x	x	x

Table 3.3: The mapping $u \vee v$

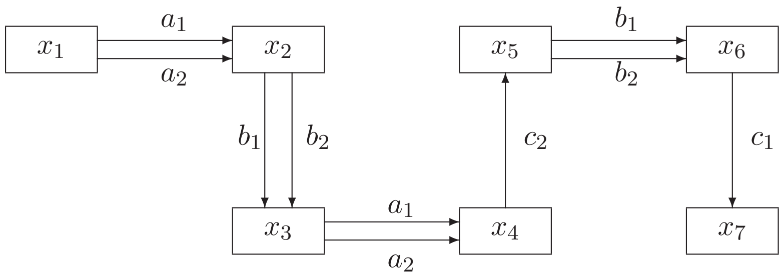


Figure 3.14: Labelled graph $G_{1\cup 2}$

Taking the union graph for G_1 and G_2 we obtain the labelled graph $G_{1\cup 2}$ represented in Figure 3.14.

The mapping $u \vee v$ is given in Table 3.3. If we compute the set $L_{u\vee v}$ of the labels for $\mathcal{G}(G_{1\cup 2}, u \vee v)$ we obtain

$$L_{u\vee v} = L^{(1)} \cup L^{(2)} \cup L_{(1,2)} \tag{3.23}$$

Each element $\alpha \in L_{(1,2)}$ has the property that $trace(\alpha)$ contains both elements from $L_0^{(1)}$ and elements from $L_0^{(2)}$. Thus the elements of $L_{(1,2)}$ have a structure that shows that $DR(u)$ and $DR(v)$ collaborate in $DR(u \vee v)$.

If we examine the set $L_{(1,2)}$ we find that there is a path from x_1 to x_6 because $(x_1, x_6) \in \mu_{11}$ and there is, for example, the "combined" label

$$\alpha = \sigma(\sigma(a_1, b_2), \sigma(a_1, \sigma(c_2, b_1)))$$

such that $f(\alpha) = \mu_{11}$.

In the same time, none of the binary relations ρ in Table 3.3 satisfies the condition $(x_1, x_7) \in \rho$. Equivalently, this means that $\mathcal{G}(G_{1\cup 2}, u \vee v)$ does not authorize the use of any path from x_1 to x_7 in $G_{1\cup 2}$. In order to benefit of such a path we have to fill in some position in Table 3.3. If we proceed in this manner then we obtain a completion of $\mathcal{G}(G_{1\cup 2}, u \vee v)$. For example, if in the place corresponding to the line μ_{11} and column ρ_3 we append in Table 3.3 the element

$$prod_S(\mu_{11}, \rho_3) = \mu_{12}$$

then the new LSG will authorize the use of the path

$$([x_1, x_1, x_3, x_4, x_5, x_6, x_7], [a_1, b_2, a_1, c_2, b_1, c_1])$$

in $G_{1\cup 2}$ because $\sigma(\alpha, c_1)$ becomes a label for this LSG.

Various completions for $\mathcal{G}(G_{1\cup 2}, u \vee v)$ can be obtained. If we fill in all the empty places in Table 3.3 then we take:

$prod_S(\nu_1, \mu_3) = prod_S(\mu_8, \rho_3) = \mu_9$

$prod_S(\rho_4, \mu_3) = prod_S(\nu_2, \rho_3) = \mu_{10}$

$prod_S(\mu_{11}, \rho_3) = prod_S(\mu_5, \mu_3) = \mu_{12}$

$prod_S(\mu_7, \rho_3) = prod_S(\mu_6, \mu_3) = \mu_{13}$

Let us denote by L_c the label set for this case. We can imagine the following situation appeared in an application: the nodes of a labelled graph represent localities of a county; the arcs represent variants for motor-ways; a label represents the weather state for a variant. We may be interested to convey some goods from x_1 to x_7. Suppose the conditions imposed by the quality of the goods require the use of a path containing a minimum number of symbols a_2, b_2, c_2. This problem reduces to the finding of the set

$$L_{\mu_{12}} = \{\alpha \in L_c \mid f(\alpha) = \mu_{12}, Pl(\alpha) = min\}$$

where $Pl(\alpha)$ represents the number of places from α such that each place contains a symbol a_2, b_2 or c_2.

If we compute the elements of L_c we find that for each $\alpha \in L_{\mu_{12}}$ we have

$$trace(\alpha) = (a_1, b_1, a_1, c_2, b_1, c_1)$$

We can conclude by this particular computation that there is only one authorized path satisfying the conditions imposed above and this path has the length 6.

3.4.1 Conclusions and open problems

In this section we inaugurated a possible research line concerning the combination of two such structures which are built over distinct labelled graphs. Various completions of $\mathcal{G}(G, u \vee v)$ can be obtained in order to obtain a better collaboration between $\mathcal{G}(G_1, u)$ and $\mathcal{G}(G_2, v)$. We are interested to develop the following questions:

- Consider the label set $L_{(1,2)}$ from (3.23) and give an expression by means of which we can compute this set only by the components of $\mathcal{G}(G_1, u)$ and $\mathcal{G}(G_2, v)$. This expression will give an analytical characterization for the collaboration between $DR(u)$ and $DR(v)$ in $DR(u \vee v)$.

- Consider the concept of distinguished representative and study the impact of the ideas presented in this section concerning various combination of such representatives which are built over different labelled graphs.

Bibliography

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

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

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

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

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

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

[7] Thomas Lengauer et al.: Efficient Algorithms for Discrete Problems and Their Applications, http://www.gmd.de/SCAI/dfg/

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

[9] G. W. Mineau, B. Moulin and J.F. Sowa (Eds.), Conceptual Graphs for Knowledge Representation, *Lecture Notes in AI* **699** (Springer-Verlag, Berlin, 1993)

[10] M.L. Mugnier and M. Chein (Eds.), Conceptual Structures: Theory, Tools, and Applications, *Lecture Notes in AI* **1453** (Springer-Verlag, Berlin, 1998)

[11] M.L. Mugnier, Knowledge Representation and Reasonings Based on Graph Homomorphism, Research Report LIRMM 00098, August 2000; a preliminary version in [6], p.172-192

[12] U. Priss, D. Corbett and G. Angelova (Eds.), Conceptual Structures: Integration and Interfaces, **10th Int. Conf. on Conceptual Structures, ICCS 2002** (Borovets, Bulgaria, July 15-19, 2002)

[13] J.F. Sowa, Conceptual structures- Information Processing in Mind and Machine (Addison-Wesley, 1984)

[14] J.F. Sowa, *Knowledge-Based Systems*, "Special Issue: Conceptual Graphs", 5, (3) (September 1992)

[15] W.M. Tepfenhart, J.P. Dick and J.F. Sowa (Eds.), Conceptual Structures: Current Practice, *Lecture Notes in AI* **835** (Springer-Verlag, Berlin, 1994)

[16] W. Tepfenhart and W. Cyre (Eds.): Conceptual Structures: Standards and Practices, *Lecture Notes in AI* **1640** (Springer-Verlag, Berlin, 1999)

[17] N. Ţăndăreanu, Proving the Existence of Labelled Stratified Graphs, *Annals of the University of Craiova*, Vol. XXVII (2000), 81-92; on line version at http://inf.ucv.ro/~ntand/en/publications.html

[18] N. Ţăndăreanu, Knowledge Bases with Output, *Knowledge and Information Systems* 2(2000) 4, 438-460.

[19] E. Way, *Journal of Experimental & Theoretical Artificial Intelligence (JETAI)*, "Special Issue: Conceptual Graphs", 4, (2), (September 1992)