# Treewidth and logical definability of graph products

Selma Djelloul
LRI, UMR 8623, Bât 490 Université de Paris-Sud
91405 Orsay Cedex, France
e-mail: djelloul@lri.fr

November 23, 2005

**Abstract**

In this paper we describe an algorithm that, given a tree-decomposition of a graph $G$ and a path-decomposition of a graph $H$, provides a tree-decomposition of the cartesian product of $G$ and $H$. Using this algorithm, we derive upper bounds on the treewidth and on the pathwidth of the cartesian product of two graphs, expressed in terms of the treewidth and of the pathwidth of the two involved graphs. In the context of graph grammars and graph logic, we prove that the cartesian product of a class of graphs by a finite set of graphs preserves the property of being a context-free set. Moreover, if the graphs in the finite set are all connected, then we prove that the property of being an $MS$-definable set is also preserved.
**keywords:** Tree-decomposition, path-decomposition, graph grammars, graph logic.

## 1 Introduction

In this paper, we consider cartesian products of graphs. (All graphs considered here are finite and undirected). Given two graphs $G$ and $H$, their cartesian product, $G \otimes H$, is the graph whose vertex set is the cartesian product $V_G \times V_H$. Every pair of vertices of the form $\{(u,v),(u',v)\}$ with $[u,u'] \in E_G$ and every pair of vertices of the form $\{(u,v),(u,v')\}$ with $[v,v'] \in E_H$ are connected by an edge in $G \otimes H$. Grids and hypercubes are among the most popular families of cartesian product of graphs. The interested reader is referred to the book [IK00] that is exclusively dedicated to results on graph products.

Our research is motivated by the following:

- questions about parameters of graphs related to structural decompositions: treewidth, pathwidth, etc;
- questions about grammatical descriptions of graphs;
- questions related to *monadic second-order logic (MSOL)* and decidability of $MS$*-theories* of certain classes of graphs;

with application to classes of graphs obtained by natural operations such as product of graphs.
We now present in more detail the theoretical background of our contributions with its two main motivations: (1) polynomial algorithms, and (2) decidability of logical theories.

In the early eighties, Robertson and Seymour published their first results of their *Graph Minor* series of papers. They defined the concept of *tree-decomposition* and *treewidth*. This notion has come

to play an important role in recent investigations in algorithmic graph theory. Tree-decomposition is equivalent to the notion of *partial $k$-tree* which has already been used, directly or undirectly, in many algorithmic constructions designed independently. Courcelle [Cou90] established a unified framework for many graph properties that were considered separately by several authors. He proved that every graph property that can be expressed in *monadic second-order logic (MSOL)* is decidable in linear time (on the number of vertices) for simple graphs of bounded treewidth. The reference book [DF97] gives a framework to study the links between on one hand results by Robertson and Seymour, Courcelle, and other authors, and on the other hand algorithmic *parameterized* complexity. Determining whether a graph has a treewidth at most $k$ is *NP*-complete if $k$ is part of the input, but decidable in linear time for fixed $k$ (Bodlaender [Bod96]). In case of a positive answer, a tree-decomposition is even produced by the algorithm in [Bod96]. However, this linear time algorithm is not practical because of the large size of constants depending on $k$.
Courcelle and Bauderon [BC87] defined operations that enable viewing the set of all finite (hyper)-graphs as a *many sorted algebra* over an infinite signature. Such algebras have been extensively studied, particularly those defined by HR-*(hyperedge replacement)* and VR-*(vertex replacement)* signatures. In this paper we are focussing on HR grammars. The key interest of HR grammars is that, by such algebraic operations, one can generate all finite graphs of treewidth at most $k$, for fixed $k$, from elementary graphs [Cou92]. This can be done in a way that can be compared to context-free grammars defining *words*. HR-*equational* sets of graphs have been defined by means of systems of recursive equations (see, e.g., [Cou97]). An equational set is defined as the least solution (for set-inclusion) of a component (i.e., an equation) of such a system. For the reader who is not familiar with these notions, we recall the notions of graph algebras and equational sets of graphs in the appendix. Graphs that are members of such sets have a derivation tree according to the grammar. This often enables to design proofs by induction on the size of the derivation tree. HR-equational sets of graphs have another interest related to MSOL. This logic is first-order logic on power-sets. Hence, in case of a graph considered as a relational structure with the set of vertices as the domain, and using the binary relation of adjacency, quantified variables can denote sets of vertices. We use the same notations as the ones introduced by Courcelle, and denote by $MS_1$ this variant of MSOL. In the $MS_2$ variant, quantified variables can denote sets of edges because the domain also includes the set of edges. A set of graphs $L$ has a decidable *$MS_i$-theory* iff there exists an algorithm that decides whether a given $MS_i$-formula is satisfied by every graph of $L$. Since this language is closed under negation, asking whether a formula is satisfied by all the graphs of the class is the same as asking whether a formula is satisfied by some graph in the class. Hence the decidability of $MS_i$-theories of a graph class is equivalent to the decidability of the emptiness problem for this class. The $MS_2$-theory of an HR-equational set of graphs is decidable [Cou90]. A similar result relates decidability of $MS_1$-theory and VR-equational sets. Sets of graphs of decidable $MS_2$-theory are of bounded treewidth.
A set of graphs is *$MS_i$-definable* iff it is the set of graphs satisfying an $MS_i$-formula. The intersection of an HR-equational set and an $MS_2$-definable set is HR-equational [Cou90]. As a consequence, if a class of graphs with bounded treewidth is $MS_2$-definable then its $MS_2$-theory is decidable.

## Our results

We prove a new upper bound on the treewidth of the cartesian product expressed in terms of the treewidth and of the *pathwidth* of the two involved graphs. This bound is obtained by means of two algorithmic procedures. We obtain refinements of upper bounds on the treewidth of certain classes of cartesian products. We use Chlebíková's result [Chl92] stating that a grid $G_{p\times q}$ has treewidth

$\mathsf{min}\{p,q\}$ to derive a graph class of cartesian products containing the class of grids, and with the property that each of its elements has treewidth $\mathsf{min}\{p,q\}$ where $p$ and $q$ are the number of vertices of the two involved graphs. We also derive from our general construction an upper bound on the pathwidth of the cartesian product expressed in terms of the pathwidths of the two involved graphs.

The first-order logic (FO) theory of graphs is undecidable even if we limit ourselves to finite graphs. The FO-theory of the class of grids is decidable [FG01], whereas the MSOL-theory of grids is undecidable [See91]. One of the major incidence of Courcelle's developments on graph grammars and monadic second order logic for graphs is that, if $k$ is fixed, the class of all graphs with treewidth at most $k$ has decidable $MS_2$-theory. A class of graphs of decidable $MS_2$-theory is of bounded *treewidth* [See91]; the converse does not hold. It is natural to ask what kinds of compositions of structures preserve the decidability of theories. The Feferman-Vaught theorem [FV59] concerns decidability of the FO-theory for (infinite) disjoint union and (infinite) product of structures. Our contribution in this context, is that the composition of an equational set of graphs by a finite set of graphs using the cartesian product yields a class of graphs whose $MS_2$-theory is decidable.

Regular grammars on words generate languages that are recognized by finite state automata. There is no convenient notion of finite automaton for graphs. Nevertheless, it follows from results by Courcelle that a monadic second order formula is finite state for graphs of bounded treewidth. Hence an $MS$-definable class of graphs can be viewed in a way comparable to a regular language on words. Furthermore, from the aforementioned results, we deduce that an MS-formula acts, for a fixed $k$, as a filter that extracts a subclass that has a decidable $MS_2$-theory from the class of all graphs with *treewidth* at most $k$. Our contribution in this context, is that graph classes obtained using the cartesian product of an $MS$-definable class of graphs by a finite set of connected graphs are $MS$-definable.

## 2 Definitions and notations

**Definition 2.1.** *Tree-decomposition, path-decomposition* -. Let $G = (V, E)$ be a graph, a tree-decomposition of $G$ is a pair $(T, (X_t)_{t\in V_T})$ where $T$ is a tree and $(X_t)_{t\in V_T}$ a family of subsets of $V_G$, with the following properties:

**(P1)** $\bigcup_{t\in V_T} X_t = V_G$.

**(P2)** For every edge $e$ of $G$ there exists $t \in V_T$ such that $e$ has both ends in $X_t$.

**(P3)** For $t$, $t'$, $t''$, if $t'$ is on the path between $t$ and $t''$ then $X_t \cap X_{t''} \subseteq X_{t'}$.

The *width* of the tree-decomposition is $\max_{t\in V_T}(|X_t| - 1)$.
The graph $G$ has *treewidth* $w$ if $w$ is the smallest integer such that $G$ has a tree-decomposition of width $w$. We write $\mathsf{twd}(G) = w$.
If $T$ is a path, the decomposition is called a path-decomposition. The graph $G$ has *pathwidth* $w$ if $w$ is the smallest integer such that $G$ has a path-decomposition of width $w$. We write $\mathsf{pwd}(G) = w$.

A decomposition is said fundamental if it is of the minimum width, and with the smallest number of nodes of $T$.

We use the following notation :

$\mathsf{TWD}(= k)$ denotes the set of all graphs of treewidth $k$.
$\mathsf{TWD}(\leq k)$ denotes the set of all graphs of treewidth at most $k$.

**Definition 2.2.** *Parent and descendent bags mappings -.* Let $(\mathcal{B}, (Y_r))$ be a tree-decomposition of a graph $G$ rooted at $Y_{r_0}$.

i. We define the mapping *parent-bag* $p : V_{\mathcal{B}} \to \mathcal{P}(V_G)$ such that, for $r \neq r_0$, $p(r) = Y_{r'}$, where $r'$ is the parent of $r$ in the rooted tree $\mathcal{B}$, and $p(r_0) = \emptyset$.

   Note that, if the decomposition is fundamental, there always exists $y \in Y_r - p(r)$.

ii. We define the mapping *descendent-bags* $d : V_{\mathcal{B}} \to \mathcal{P}(V_G)$ such that for every internal node $r$, $d(r) = \bigcup_{\alpha,\, r_\alpha \text{ child of } r} Y_{r_\alpha}$, and $d(r) = \emptyset$ if $r$ is a leaf.

**Definition 2.3.** *Bag types -.* Let $(\mathcal{B}, (Y_r))$ be a rooted fundamental tree-decomposition of a graph $G$.

*Relax-introduce* - We say that a bag $Y_r$ is *relax-introduce* if $|Y_r| = \mathsf{twd}(G)+1$, and there exist two distinct vertices $y \in Y_r - p(r)$ and $z \in Y_r - d(r)$. Then, a bag $Y_r$ is relax-introduce if $|Y_r|$ is tight but there exists a vertex $z$ that comes into $Y_r$ just when another vertex will leave $Y_r$. Such a couple $(y, z)$ is said to be *synchronized.*

*Furtive-introduce* - We say that a bag $Y_r$ is *furtive-introduce* if $|Y_r \cap p(r)| = \mathsf{twd}(G)$ and $Y_r - p(r) \not\subset d(r)$. Note that, $|Y_r \cap p(r)| = \mathsf{twd}(G)$ implies that $|Y_r - p(r)| = 1$. Let $\{y\} = Y_r - p(r)$. This vertex is called the *furtive* vertex. Then, a bag $Y_r$ is furtive-introduce if $|Y_r \cap p(r)|$ is tight and the leaving vertex does not come from $d(r)$.

*Child-hand-shaking* - Let $r$ be a node with at least two children. We say that $Y_r$ is $r_\beta$*-child-hand-shaking* if $r_\beta$ is a child of $r$, $|Y_r| = \mathsf{twd}(G) + 1$, $|Y_r| \subseteq d(r)$, and there exists at least one vertex $y \in Y_r - p(r)$ for which $r_\beta$ is the unique child such that $y \in Y_{r_\beta}$. In this case, such a vertex $y$ is called the *single-branch-leaving* vertex. Then, a bag $Y_r$ is child-hand-shaking if it has at least two children, $|Y_r|$ is tight and at least one leaving vertex comes from a single branch of the descendence of $Y_r$. Since the decomposition is fundamental, there exists $x \in Y_r - Y_{r_\beta}$. The couple $(y, x)$ is said to be *synchronized.*

*Blocking* - A bag $Y_r$ is *blocking* if $r$ has at least two children, $|Y_r| = \mathsf{twd}(G) + 1$, $|Y_r| \subseteq d(r)$ and, for every vertex $y \in Y_r - p(r)$ there are at least two distinct children $r_\beta$, $r_\gamma$ of $r$ such that both $Y_{r_\beta}$ and $Y_{r_\gamma}$ contain $y$.

**Remark 2.4.** Let $(\mathcal{B}, (Y_r))$ be a rooted fundamental tree-decomposition.

i. A bag can be both relax-introduce and furtive-introduce.

ii. Types relax-introduce (resp. furtive-introduce), child-hand-shaking and blocking are mutually excluding.

iii. A leaf bag cannot be blocking and cannot be child-hand-shaking.

iv. The root bag cannot be furtive-introduce.

v. Every bag of size $\mathsf{twd}(G) + 1$ is of (at least) one of the types described above.

vi. If $\mathcal{B}$ is a path then no bag is blocking.

vii. A clique has a unique rooted fundamental tree-decomposition reduced to a single bag that is relax-introduce.

**Definition 2.5.** *Compounds of a tree by a path -.* In the following, $\mathcal{B}$ denotes a rooted tree, $\mathcal{A}$ denotes a rooted path (disjoint from $\mathcal{B}$), and $r$ denotes a node of $\mathcal{B}$.

- The *simple compound* of $\mathcal{B}$ by $\mathcal{A}$ at $r$ is the rooted tree obtained by replacing $r$ by a copy of $\mathcal{A}$, and, if $r$ is not the root, connecting the root of $\mathcal{A}$ to the parent of $r$ and, if $r$ is not a leaf, connecting the leaf of $\mathcal{A}$ to each of the children of $r$. We denote the resulting tree by $\mathcal{B}[r \downarrow \mathcal{A}]$.
- If $r$ has at least two children, the *$r_\beta$-child-hand-shaking compound* of $\mathcal{B}$ by $\mathcal{A}$ at $r$ is the rooted tree obtained by replacing $r$ by a copy of $\mathcal{A}$, connecting the leaf of $\mathcal{A}$ to the child $r_\beta$ of $r$, connecting all the other children of $r$ to the root of $\mathcal{A}$ and, if $r$ is not the root, connecting the root of $\mathcal{A}$ to the parent of $r$. We denote the resulting tree by $\mathcal{B}[r \downarrow_{r_\beta} \mathcal{A}]$.

In both types of compounds, we rename each node $s$ of the path to $(r, s)$ in the resulting tree where $r \in V_{\mathcal{B}}$ is the node at which occurs the compound. If no compound occurs at $r$, it is renamed to $(r, -)$ in the resulting tree.

## 3 Treewidth and pathwidth for cartesian product of graphs

If $(T, (X_t)_{t \in V_T})$ is a tree-decomposition of $G$ then $(T, (X_t \times V_H)_{t \in V_T})$ is a tree-decomposition of $G \otimes H$. Therefore :

**Fact 3.1.**

*i.* $\mathsf{twd}(G \otimes H) \leq \mathsf{min}\ \{\ (\mathsf{twd}(G) + 1).|V_H|\ ,\ (\mathsf{twd}(H) + 1).|V_G|\ \} - 1.$

*ii.* $\mathsf{pwd}(G \otimes H) \leq \mathsf{min}\ \{\ (\mathsf{pwd}(G) + 1).|V_H|\ ,\ (\mathsf{pwd}(H) + 1).|V_G|\ \} - 1.$

In the following, we prove a better upper bound for pathwidth and, under certain conditions, a better upper bound for treewidth.

**Fact 3.2.** *A tree has a rooted fundamental tree-decomposition in which all bags are relax-introduce.*

*Proof.* We root the tree at a fixed leaf. In the notation $[u, v]$ used for the edges of the rooted tree, the extremity $u$ is the parent of $v$. At each edge $[u, v]$ of the tree, we associate a bag $\{u, v\}$. Two bags are connected iff the corresponding edges are of the form $[u, v]$, $[v, w]$. The tree-decomposition is rooted by considering each bag $[u, v]$ the parent of all the bags of the form $[v, w]$. It is easy to see that such a rooted tree-decomposition is fundamental and that its bags are all relax-introduce. □

**Theorem 3.3.**

*i.* *If $G$ has a rooted fundamental tree-decomposition with no blocking bag then, for any graph $H$,* $\mathsf{twd}(G \otimes H) \leq \mathsf{twd}(G).|V_H| + \mathsf{pwd}(H)$*. Moreover, there exists an infinite set of pairs of graphs $\{(G_k, H_k), k \geq 3\}$ such that $G_k$ has no rooted fundamental tree-decomposition with no blocking bag and such that* $\mathsf{twd}(G_k \otimes H_k) > \mathsf{twd}(G_k).|V_{H_k}| + \mathsf{pwd}(H_k)$.

*ii.* *If $G$ has a rooted fundamental tree-decomposition whose bags of size* $\mathsf{twd}(G) + 1$ *are all relax-introduce or child-hand-shaking then, for any graph $H$,* $\mathsf{twd}(G \otimes H) \leq \mathsf{twd}(G).|V_H|$.

*Proof.* The graph $G_k$ is shown below. We let $\alpha = (3k^2 - 3k - 2)/2$. The graph $H_k$ is any connected graph on $k \geq 3$ vertices and with pathwidth less than $|V_{H_k}| - 1$. The graph $G_k$ is a 2-tree (hence of treewidth 2). One can check that in any rooted fundamental tree-decomposition of $G_k$, the bag containing the triangle $\{a, b, c\}$ is blocking. The graph $G_k \otimes H_k$ has a minor isomorphic to $K_{3k}$. To see that, first consider the $(k^2 - k)/2$ pairs of vertices of the form $(a, r), (a, s)$ with $r \neq s$. For one of these pairs use a path in $\{a\} \otimes H_k$, and for each of the others use consecutively a path in $\{x_i\} \otimes H_k$, $1 \leq i \leq (k^2 - k - 2)/2$. All the used paths are disjoint because their internal vertices are contained in pairwise distinct copies of $H_k$. Now, consider all the pairs of vertices of the form $(a, r), (b, s)$ with $r \neq s$. For each of these $(k^2 - k)$ pairs use consecutively a path in $\{x_i\} \otimes H_k$, $(k^2 - k)/2 \leq i \leq \alpha$. We do the same for all pairs of vertices of the form $(b, r), (b, s)$ and $(b, r), (c, s)$ with $r \neq s$ using the following copies of $H_k$: $\{b\} \otimes H_k$ and $\{y_i\} \otimes H_k$, $1 \leq i \leq \alpha$. Finally, for each of the pairs of vertices of the form $(c, r), (c, s)$ and of the form $(c, r), (a, s)$ with $r \neq s$, we use the following copies of $H_k$: $\{c\} \otimes H_k$ and $\{z_i\} \otimes H_k$, $1 \leq i \leq \alpha$. Now, since $G_k \otimes H_k$ has a minor isomorphic to $K_{3k}$, it cannot have a treewidth less than $3k - 1$. The pair $(G_k, H_k)$ is then as claimed.

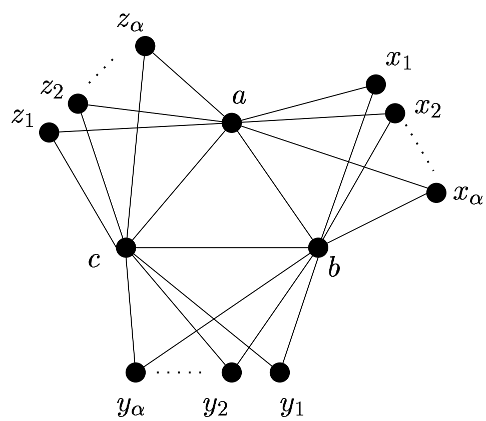


Figure 1: The graph $G_k$

Now, assume that we are given a rooted fundamental tree-decomposition $(\mathcal{B}, (Y_r))$ for $G$ with no blocking bag, and a path-decomposition $(\mathcal{A}, (X_s))$ for $H$. Let $r_0$ denotes the root of $\mathcal{B}$. We denote by $0, 1, \ldots, l$, the nodes of $\mathcal{A}$ from the root down to the leaf. We give a tree-decomposition $(\mathcal{T}, (Z_t))$ of $G \otimes H$. Let $P$ be a rooted path with $q = |V_H|$ nodes denoted by $0, 1, \ldots, (q-1)$ from the root down to the leaf. The tree-decomposition of $G$ is processed in a prefix-order by the procedure `Construct`. This procedure takes as input the given tree-decomposition and path-decomposition of $G$ and $H$ respectively. It then decides, at each node, which type of compound has to be applied according to the type of the current bag. At the end of this first procedure, one obtains a tree which is the base of a tree-decomposition for the cartesian product. A second procedure `Fill-bags` processes this tree and decides, again according to the bag types of the tree-decomposition of $G$, what is the content of the current bag.
Start by setting $T := \mathcal{B}$ and by denoting each node $r$ of $T$ by $(r, -)$. The structure of $T$ and its nodes' name are updated in a prefix-order by examining the type of each bag in $(\mathcal{B}, (Y_r))$.

```
Construct((B,(Y_r)), T, r, A) {
begin
    If r is such that |Y_r| ≤ twd(G) then
    begin Rename (r,−) to (r,0); end
```

```
    elsif Y_r is relax-introduce then
    begin T := T[(r,−) ↓ P];
         {this renames the nodes of P to (r,0),...,(r,(q−1))}.
    end
    elsif Y_r is r_β-child-hand-shaking then
    begin T := T[(r,−) ↓_{r_β} P];
         {this renames the nodes of P to (r,0),...,(r,(q−1))}.
    end;
    else {this is furtive-introduce }
    begin T := T[(r,−) ↓ 𝒜] ;
         {this renames the nodes of 𝒜 to (r,0),...,(r,l))}.
    end;
    If r is not a leaf
    begin
      For each child r_γ of r
       begin Construct((ℬ,(Y_r)), T, r_γ, 𝒜) end;
    end;
end; }
```

$\mathcal{T}$ is the tree $T$ obtained at the end of the execution performed by calling Construct$((\mathcal{B},(Y_r)),\ T,\ r_0,\ \mathcal{A})$. Every node of $\mathcal{T}$ has a name of the form $(r,k)$. The root is named $(r_0,0)$.

Now, we give a procedure that, given a tree $\mathcal{T}$ produced by application of the previous procedure, tells what is the content of each bag $Z_t$, $t \in V_{\mathcal{T}}$ according to the type of each bag of $(\mathcal{B},(Y_r))$. The vertices of $H$ are denoted by $1,\ldots,q$. The notation $[i]$ denotes the set $\{1,2,\ldots,i\}$ if $i \geq 1$ and, $[0]=\emptyset$.

```
Fill-bags((ℬ,(Y_r)), 𝒯, r, (𝒜,(X_s)) {
begin
    If r is such that |Y_r| ≤ twd(G) then
    begin
      Z_(r,0) = Y_r × V_H;
    end
    elsif Y_r is relax-introduce then
    begin
      pick a synchronized couple (y,z);
      For i := 1 to q do
       begin
        Z_(r,i−1) = (Y_r \ {y,z}) × V_H ∪ {y} × (V_H − [q−i]) ∪ {z} × [q−i+1];
       end;
    end
    elsif Y_r is r_β-child-hand-shaking then
    begin
      pick a synchronized couple (y,x)
      For i := 1 to q do
       begin
        Z_(r,i−1) = (Y_r \ {x,y}) × V_H ∪ {y} × (V_H − [q−i]) ∪ {x} × [q−i+1];
       end;
```

```
    end
    else {Y_r is furtive-introduce }
    begin
      pick a vertex y that is furtive;
      For s := 1 to l+1 do
        begin
          Z_(r,s-1) = (Y_r \ {y}) × V_H ∪ {y} × X_(s-1);
        end;
    end;
    If r is not a leaf
    begin
      For each child r_γ of r
        begin Fill-bags((B,(Y_r)), T, r_γ, (A,(X_s))) end;
    end;
end; }
```

**Property 3.4.** *For every $r$, let $G_r$ be the subgraph of $G$ induced by all vertices in $Y_r$. The subgraph of $\mathcal{T}$ induced by all the bags $Z_t$ where $t$ is of the form $(r,*)$ and $*$ stands for any valid integer is a path-decomposition of the graph $G_r \otimes H$. This decomposition is of width at most* $\mathsf{twd}(G).|V_H| + \mathsf{pwd}(H)$.

*Proof.*

*Checking condition (P1) of the definition of a tree-decomposition.*- Let $(u,i)$ be a vertex of $G_r \otimes H$. We want to prove that $(u,i)$ is in some bag $Z_{(r,*)}$. If $Y_r \leq \mathsf{twd}(G)$ then, $\{u\} \times V_H \subset Z_{(r,0)}$. If $Y_r$ is relax-introduce and $(u,z)$ (resp. $(y,u)$) is the synchronized couple picked by the procedure then $\{u\} \times V_H \subset Z_{(r,(q-1))}$ (resp. $Z_{(r,0)}$). If $u$ is not involved in the synchronized couple then $\{u\} \times V_H$ is entirely included in all $Z_{(r,k)}$, $0 \leq k \leq (q-1)$. If $Y_r$ is $r_\beta$-child-hand-shaking and $u$ is not involved in the synchronized couple $(y,x)$ picked by the procedure then, $\{u\} \times V_H$ is contained in all $Z_{(r,k)}$, $0 \leq k \leq (q-1)$. If $u = x$ then, $(u,i)$ is in all $Z_{(r,k)}$, $0 \leq k \leq (q-i)$. If $u = y$, $(u,i)$ is in all $Z_{(r,k)}$, $(q-i) \leq k \leq (q-1)$. If $Y_r$ is furtive-introduce, $y$ the furtive vertex picked by the procedure, and $u \neq y$ then, $\{u\} \times V_H$ is entirely contained in all $Z_{(r,k)}$ with $0 \leq k \leq l$. If $u = y$, let $s$ be such that $i \in X_s$ then, $(u,i) \in Z_{(r,s)}$.

*Checking condition (P2) of the definition of a tree-decomposition.*- Let $e$ be an edge of $G_r \otimes H$. We want to check that it has both ends in some $Z_{(r,*)}$. If it is of the form $[(u,i),(u,i')]$ with $[i,i'] \in E_H$, then the only case that needs a check is that when $u$ matches the furtive vertex of a furtive-introduce bag. Since $(\mathcal{A},(X_s))$ is a path-decomposition for $H$, there exists $s$, $0 \leq s \leq l$ such that $[i,i'] \in X_s$. Therefore, $e \in Z_{(r,s)}$. Now, assume that $e$ is of the form $[(u,i),(u',i)]$ with $[u,u'] \in E_{G_r}$. If $Y_r$ is relax-introduce or child-hand-shaking, $(u,i)$ and $(u',i)$ are together in $Z_{(r,(q-i))}$. If $Y_r$ is furtive-introduce, $(u,i)$ and $(u',i)$ are together in $Z_{(r,s)}$ where $s$ is such that $i \in X_s$.

The last condition in the definition of a tree-decomposition is clearly satisfied.

If $Y_r$ is relax-introduce or child-hand-shaking, every bag of the form $Z_{(r,*)}$ is of size $\mathsf{twd}(G).|V_H| + 1$. If $Y_r$ is furtive-introduce, every bag of the form $Z_{(r,*)}$ is of size at most $\mathsf{twd}(G).|V_H| + \mathsf{pwd}(H) + 1$. This ends the proof of property 3.4.

Claim.- The decomposition $(\mathcal{T},(Z_t))$ obtained at the end of the execution performed by calling `Fill-bags`$((\mathcal{B},(Y_r)),\ \mathcal{T},\ r_0)$ is a tree-decomposition of $G \otimes H$.

By means of the previous property, only the last condition of the definition of a tree-decomposition has to be checked. So, let $(u,h) \in Z_{t_1} \cap Z_{t_2}$ and let $t$ be a node of $\mathcal{T}$ on the path between $t_1$

and $t_2$. We want to prove that $(u,h) \in Z_t$. Also by means of property 3.4 this has to be checked only in the case $t_1$ of the form $(r_1,*)$ and $t_2$ of the form $(r_2,*)$, with $r_1 \neq r_2$. Since $(\mathcal{B},(Y_r))$ is a tree-decomposition $u$ in in all bags $Y_\alpha$ with $\alpha$ on the path $P[r_1,r_2]$ connecting $r_1$ and $r_2$.

*Case $r_1$ is an ancestor of $r_2$ (or inversely).*- In this case, $u$ cannot be furtive-introduce for any bag $Y_\alpha$ with $\alpha \in P[r_1,r_2]$. Also, $u$ cannot be involved in a synchronized couple of the form $(u,z)$ for any bag $Y_\alpha$ with $\alpha \in P[r_1,r_2]-\{r_1\}$. If $Y_{r_1}$ is $r_\beta$-child-hand-shaking, $r_2$ and $r_\beta$ from the same branch and $u$ matches the single-branch-leaving vertex then, $\{u\} \times V_H$ is entirely contained in all bags $Z_{(\alpha,*)}$ with $\alpha \in P[r_1,r_2]-\{r_1,r_2\}$ and $(u,h)$ is contained in all bags $Z_t$, with $t$ higher then $t_2$ and $(u,h)$ is in all bags $(r_1,i)$ with $(q-h) \leq i \leq (q-1)$. This constitutes the path in $\mathcal{T}$ between $t_1$ and $t_2$. If $u$ does not match the single-branch-leaving vertex then, $\{u\} \times V_H$ is entirely contained in all bags $Z_{(\alpha,*)}$ with $\alpha \in P[r_1,r_2]-\{r_2\}$ and $(u,h)$ is contained in all bags $Z_t$, with $t$ higher then $t_2$. This constitutes the path in $\mathcal{T}$ between $t_1$ and $t_2$.
If $Y_{r_1}$ is $r_\beta$-child-hand-shaking, and $r_2$ and $r_\beta$ are not from the same branch, and $u$ in not involved in the form $(y,u)$ in the picked synchronized couple then, $\{u\}\times V_H$ is entirely contained in all bags $Z_{(\alpha,*)}$ with $\alpha \in P[r_1,r_2]-\{r_2\}$, $(u,h)$ is in all bags $Z_t$, with $t$ higher then $t_2$. If the synchronized couple is of the form $(y,u)$ then, $\{u\} \times V_H$ is entirely contained in all bags $Z_{(\alpha,*)}$ with $\alpha \in P[r_1,r_2]-\{r_1,r_2\}$, and in $Z_{(r_1,0)}$ and, $(u,h)$ is contained in all bags $(r_1,i)$ with $0 \leq i \leq (q-h)$. This constitutes the path in $\mathcal{T}$ between $t_1$ and $t_2$.

*Case $r_1$ and $r_2$ are not related by the ancestor relation.*- Let $a$ be their lowest common ancestor (*lca*). Since $(\mathcal{B},(Y_r))$ is a tree-decomposition $u \in Y_a$. If $Y_a$ is furtive-introduce, the *lca* of $t_1$ and $t_2$ is $(a,l)$ and we have $\{u\} \times V_H \subset Z_{(a,l)}$. If $Y_a$ is relax-introduce, the *lca* of $t_1$ and $t_2$ is $(a,(q-1))$ and we have $\{u\} \times V_H \subset Z_{(a,(q-1))}$. Then, we finish by applying the previous case. If $Y_a$ is such that $|Y_a| \leq \mathsf{twd}(G)$ then the *lca* of $t_1$ and $t_2$ is $(a,0)$ and $\{u\} \times V_H \subset Z_{(a,0)}$. Then, we also finish by applying the previous case. If $Y_a$ is child-hand-shaking then, the *lca* of $t_1$ and $t_2$ is $(a,0)$ and $u$ cannot be the single-branch-leaving vertex: $y$. Since $(Y-\{y\}) \times V_H \subset Z_{(a,0)}$, we have $\{u\} \times V_H \subset Z_{(a,0)}$. We finish by applying the previous case. This ends the proof of the claim and of theorem 3.3 □

The following corollary gives an infinite class of graphs for which the bound in Fact 3.1.i is improved.

**Corollary 3.5.**

*i.* *If $G$ is a tree (resp. a path) then, for any graph $H$,*

$$\mathsf{twd}(G \otimes H) \leq |V_H| \text{ (resp. } \mathsf{pwd}(G \otimes H) \leq |V_H|).$$

*ii.* *Let $H$ be a fixed graph that has a hamiltonian path. If $\mathcal{K}$ is the class of all trees that have a path (resp. of all paths) of length at least $|V_H|-1$ then,*

$$\{G \otimes H,\ G \in \mathcal{K}\} \subset \mathsf{TWD}(=|V_H|) \text{ (resp. } \{G \otimes H,\ G \in \mathcal{K}\} \subset \mathsf{PWD}(=|V_H|)).$$

*iii.* *For any graphs $G$ and $H$,*

$$\mathsf{pwd}(G \otimes H) \leq \min\ \{\mathsf{pwd}(G).|V_H| + \mathsf{pwd}(H)\ ,\ \mathsf{pwd}(H).|V_G| + \mathsf{pwd}(G)\}$$

**Proof.** The first assertion is immediate from Fact 3.2 and ii. of theorem 3.3. The second one is due to the fact that, $G \otimes H$ has a subgraph isomorphic to the square grid $|V_H| \times |V_H|$. The last assertion is due to vi. of remark 2.4. □

**Remainig questions**

*Question 1*: Can one prove a bound on $\mathsf{twd}(G \otimes H)$ similar to that in theorem 3.3.i but with $\mathsf{twd}(H)$ instead of $\mathsf{pwd}(H)$? If the hypotheses are not strengthened (that is, if one uses only conditions on bag types) then, the only case where such a bound might be better is when the rooted fundamental tree-decompositions of $G$ all have at least one furtive-introduce bag that is not relax-introduce and $H$ is such that $\mathsf{twd}(H) < \mathsf{pwd}(H)$.

*Question 2*: Can one obtain a characterization for the graphs that have a rooted tree-decomposition with no blocking bag?

# 4 Cartesian products by a finite set

**Theorem 4.1.** *Let $\mathcal{J}$ be a finite set of graphs and $\mathcal{I}$ be a class of graphs. We denote by $\otimes_{\mathcal{J}}(\mathcal{I})$ the set of graphs $\{G \otimes H,\ G \in \mathcal{I},\ H \in \mathcal{J}\}$.*

*i. If $\mathcal{I}$ is HR-equational then so is $\otimes_{\mathcal{J}}(\mathcal{I})$.*

*ii. Assume now that the graphs in $\mathcal{J}$ are all connected. If $\mathcal{I}$ is MS-definable then $\otimes_{\mathcal{J}}(\mathcal{I})$ is $MS_2$-definable.*

*Proof.* For each graph $H$ in $\mathcal{J}$, we denote by $q_H$ the number of vertices of $H$.

i. For each graph $H$ in $\mathcal{J}$ we consider a set $\mathcal{K}_H = \{1_H, 2_H, \ldots q_H\}$ of source labels.

Let $M$ (resp., $\otimes_{\mathcal{J}} M$) be the algebra of all graphs with source labels in a countable set $\mathcal{C}$ (resp., $\mathcal{C} \times \bigcup_{H \in \mathcal{J}} \mathcal{K}_H$). For each $H \in \mathcal{J}$, we enlarge the set of constant symbols of $\otimes_{\mathcal{J}} M$ by all the graphs $\otimes_H \mathsf{c}$, $c \in \mathcal{C}$, $H \in \mathcal{J}$, denoting the graph $H$ where every vertex $i_H$ is the $(c, i_H)$-source, and all the graphs $\otimes_H \mathsf{cc}'$, $c, c' \in \mathcal{C}$, $H \in \mathcal{J}$, obtained from the cartesian product of $H$ by an edge and labelling in a copy of $H$, each vertex $i_H$ by $(c, i_H)$ and in the other copy each vertex $i_H$ by $(c', i_H)$.

Let us define, for each $H \in \mathcal{J}$, a mapping $f_H : M \to \otimes_{\mathcal{J}} M$, inductively on the form of the terms of $M$. This mapping is used to transform the polynomial system from which $\mathcal{I}$ is obtained to a polynomial system $S_H$ for which a component has $\otimes_{\{H\}}(\mathcal{I})$ as the least solution. If $t$ is a constant symbol then, we let $f_H(t) = \otimes_H t$. If $t = r //_{C,C'} s$ then, $f_H(t) = f_H(r) //_{C \times \mathcal{K}_H, C' \times \mathcal{K}_H} f_H(s)$.

Finally, if $t = fg_c(r)$ then, $f_H(t) = fg_{(c,1_H)}(fg_{(c,2_H)}(\ldots(fg_{(c,q_H)}(f_H(r)))\ldots)$.

There exists a finite subset $\mathcal{K}$ of $\mathcal{C}$ such that $\mathcal{I}$ is obtained as a component of the least solution of a polynomial system $S$ over $F_{HR}(\mathcal{K})$. Let $S =< u_1 = p_1, \ldots, u_n = p_n >$, and assume w.o.l.g., that $\mathcal{I} = L((S, M), u_1)$.

Let $S_H =< u_1 = \otimes_H p_1, \ldots, u_n = \otimes_H p_n >$ be the polynomial system over $F_{HR}(\mathcal{K} \times \mathcal{K}_H)$ where each $\otimes_H p_j$ is obtained from $p_j$ by replacing each monomial $t_\alpha$ of $p_j$ by $f_H(t_\alpha)$.

Note that, in each $S_H$ the sort of $u_1$ (in the algebra $\otimes_{\mathcal{J}} M$) is $C \times \mathcal{K}_H$ where $C$ is the sort (in the algebra $M$) of the unknown $u_1$ of $S$.

Let us denote by $fg_{C \otimes \mathcal{K}_H}$ the composition of mappings $\bigcirc_{c \in C}\{fg_{(c,1_H)} \circ fg_{(c,2_H)} \circ \ldots \circ fg_{(c,q_H)}\}$, and consider the monomial: $p_0^H = fg_{C \otimes \mathcal{K}_H}(u_1)$. It is of the sort: $\emptyset$.

Consider any sequence of the systems $S_H$ and rename the unknowns of the second system in the sequence to $u_{n+1}, \ldots, u_{2n}$ respectively, and the unknowns of the third system to $u_{2n+1}, \ldots, u_{3n}$ respectively, etc.

Now, let $S'$ be the polynomial system over $F_{HR}(\mathcal{K} \times \bigcup_{H \in \mathcal{J}} \mathcal{K}_H)$ obtained as follows. First, merge all the equations of all the sytems $S_H$. Then, add the equation $u_0 = \underset{H \in \mathcal{J}}{+_\emptyset} p_0^H$.

We have, $\otimes_{\mathcal{J}}(\mathcal{I}) = L((S', \otimes_{\mathcal{J}} M), u_0)$.

ii. We add $2|\mathcal{J}|$ unary predicates in the signature of the relational structures. These predicates "encode" the vertex set and the edge set of all the fixed graphs of the set $\mathcal{J}$. Let $\varphi$ be the (closed) formula defining $\mathcal{I}$. For a graph $G$ and a subset $X \subseteq V_G$, we denote by $G_X$ the subgraph of $G$ induced by $X$. We prove that, given a fixed connected graph $H$, the following property for a graph $G'$ is expressible in $MS_2$ logic. "$G'$ is a cartesian product by the graph $H$".

There exists a partition $X_1, X_2, \ldots X_{q_H}$ of the set of vertices of $G'$ such that all the following is satisfied:

(a) $\bigvee_{1 \leq i \leq q_H} (G'_{X_i} \models \varphi)$.

(b) For all pairs $\{i, j\}$,
edges between $X_i$ and $X_j$ exist iff $[i, j]$ is an edge of $H$,
and
edges between $X_i$ and $X_j$ are the arcs of an isomorphism between $X_i$ and $X_j$.

(c) For every $i, 1 \leq i \leq q_H$, for every pair $\{x, y\}$ of vertices in the same $X_i$, if $x$ and $y$ are connected by a path using only edges with ends in different $X_j's$, then $x = y$.

Since $H$ is connected, if the two first conditions are both satisfied, the $X_i's$ induce pairwise isomorphic subgraphs all satisfying $\varphi$. □

The property of being a cartesian product by a fixed template cannot be expressed in the weaker $MS_1$-logic. indeed, the condition of "the isomorphism between $X_i$ and $X_j$" cannot be expressed in $MS_1$-logic since the property "$X_i$ and $X_j$ have the same cardinality" is not $MS_1$ [Cou97].

# 5 Conclusion

We proved that the composition of an HR set by a finite set of graphs using the cartesian product yields an HR set. We give a concrete construction of a set of "production rules" for the resulting class. Can one obtain an analogous result for VR sets?

The notion of *clique-width* is related to VR grammars (see, e.g., [Cou97]) in the same way that treewidth is related to HR grammars. Treewidth and clique-width are also comparable measures of complexity. Indeed, a property expressible in $MS_1$-logic is decidable in linear time on every graph class with bounded clique-width. It is interesting to establish upper bounds on the clique-width of two graphs expressed in terms of related parameters of the two involved graphs as done in section 3. Furthermore, other graph operations can be investigated relatively to these aspects, particularly the cartesian sum that we believe to be more suitable with VR grammars. Given two graphs $G$ and $H$, their cartesian sum has vertex set the cartesian product of the two sets $V_G$ and $V_H$ and edge set $\{\{(x,y),(x',y')\},\ [x,x\prime] \in E_G$ or $[y,y\prime] \in E_H\}$.

**Acknowledgements:**

We thank Bruno Courcelle for stimulating discussions and for the ideas that help us to define the example of the graph $G_k$ in the proof of theorem 3.3 and to simplify the proof of expressibility in $MS$-logic.

## References

[BC87] M. Bauderon and B. Courcelle. Graph expressions and graph rewritings. *Math. System Theory*, 20:83–127, 1987.

[Bod96] H. L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.*, 25:1305–1317, 1996.

[Chl92] J. Chlebíková. On the treewidth of a graph. *Acta Math. Univ. Comenianae*, 16(2):225–236, 1992.

[Cou90] B. Courcelle. The monadic second-order logic of graphs I: Recognizable sets of finite graphs. *Information and Computation*, 85:12–75, 1990. Presented at the International Workshop on Graph Grammars and their application to computer Science, Warrenton, Virginia, 1986. LNCS Comp. Sci. vol-291, 133-146.

[Cou92] B. Courcelle. The monadic second-order logic of graphs III: Tree-decompositions, minors and complexity issues. *Theoretical Informatics and Applications*, 26(3):257–286, 1992.

[Cou93] B. Courcelle. *Graph grammars, monadic second-order logic and the theory of graph minors,* in "Graph Structure Theory", Contemporary Mathematics 147 American Mathematical Society, pages 565–590. 1993.

[Cou96] B. Courcelle. Tutorial. basic notions of universal algebra for language theory and graph grammars. *Theoretical Computer Science*, 163:1–54, 1996.

[Cou97] B. Courcelle. *The expression of graph properties and graph transformations in monadic second-order logic.,* Handbook of graph grammars and computing by graph transformation: Vol. I : Foundations, pages 313–400. G. Rozenberg ed., World Scientific Publishing Co., Inc., 1997.

[DF97] R. G. Downey and M. R. Fellows. *Parameterized complexity.* Springer, 1997.

[FG01] M. Frick and M. Grohe. Deciding first-order properties of locally tree-decomposable structures. *JACM*, 48:1184–1206, 2001.

[FV59] S. Feferman and R. Vaught. The first order properties of algebraic systems. *Fund. Math.*, 47:57–103, 1959.

[IK00] W. Imrich and S. Klavẑar. *Product graphs: Structure and Recognition.* John Wiley and Sons, Inc., 2000.

[See91] D. Seese. The structure of the models of decidable monadic theories of graphs. *Annals of Pure Applied Logic*, 53:169–195, 1991.

# Appendix

## A Relationships between tree-decompositions and graph expressions

Graphs and more generally hypergraphs can be obtained as values of terms from well defined (many *sorted*) *algebras* (see for example, [Cou96, Cou92, BC87]). Let $\mathcal{C}$ be a countable set of *source labels*. A graph with *sources* is a pair $< G, src_G >$ where $G$ is a graph, $src_G : C \to V_G$ is a total mapping and $C$ is a finite subset of $\mathcal{C}$. The set $C$ is called the type of $G$ and is denoted by $\tau(G)$. We denote by $\mathcal{GS}(C)$ the set of all graphs of type $C$. The set $\mathcal{S}$ of finite subsets of $\mathcal{C}$ is used as the set of sorts. We denote by $\mathcal{GS}$ the set of all graphs with sources. We consider the following $\mathcal{S}$-signature:

**(1) Parallel composition**

If $G \in \mathcal{GS}(C)$ and $G' \in \mathcal{GS}(C')$ we let $H = G//_{C,C'}G'$ be the (isomorphism class of the ) graph in $\mathcal{GS}(C \cup C')$ obtained by first taking the union of two disjoint graphs $K$ and $K'$ respectively isomorphic to $G$ and $G'$ and then by fusing any two vertices $v$ and $v'$ that are respectively the $c$-source of $K$ and the $c$-source of $K'$ for some $c \in C \cap C'$.

**(2) forgetting a source label**

If $G \in \mathcal{GS}(C)$ and $c \in C$, we let $H = fg_c(G)$ the graph in $\mathcal{GS}(C \setminus \{c\})$ with source mapping $src_H = src_G/_{(C\setminus\{c\})}$. In words, the vertex with label $c$ in $G$ becomes "anonymous" in $H$.

**(3) constant operators**

For every source label $c$, we denote by $\mathsf{c}$ the graph with a single vertex which is the $c$-source, $\mathsf{c}^{\mathsf{l}}$ the graph consisting of a single vertex which is the $c$-source and at which there is a loop and, for every pair $\{c, c'\}$ of distinct source labels, the graph $\mathsf{cc}'$ consisting of an egde whose extremities are respectively the $c$-source and the $c'$-source.

We let $F_{HR}$ be the $\mathcal{S}$-signature consisting of $//_{C,C'}$, $fg_c$, $\mathsf{c}$, $\mathsf{c}^{\mathsf{l}}$ and $\mathsf{cc}'$ for all relevant $C$, $C'$ subsets of $\mathcal{C}$ and all source labels $c$, $c'$. We obtain an $F_{HR}$-algebra. The terms in the free algebra $T(F_{HR})$ are called $HR$(graph)-expressions. Every $t \in T(F_{HR})$ denotes a graph $val(t)$ called the value of $t$. A term $t$ in the free algebra can be viewed as a labeled rooted tree constructed recursively as follows. The root is labeled with the operation $f$ such that $t = f(t_1, \ldots, t_{\rho(f)})$, where $\rho(f)$ is the arity of $f$. For each $t_i$, we construct a labeled rooted tree in the same way and so on. Hence the leaves are the constants.
For every finite subset $C$ of $\mathcal{C}$, every finite graph $G$ in $\mathcal{GS}(C)$ is the value of some $HR$-expression. Hence $\mathcal{GS}$ is homorphic to the $F_{HR}$-algebra so defined.

For $\mathcal{K} \subseteq \mathcal{C}$, we denote by $F_{HR}(\mathcal{K})$ the subsignature consisting of the above symbols with $C$, $C' \subseteq \mathcal{K}$, $c$, $c' \in \mathcal{K}$.

Other sets of $HR$-operations have also been considered (see [Cou96, Cou97] for example). It may happen in some proofs, that certain signatures provide more facilities.

The following result has been proved in [Cou93].

$$\mathsf{twd}(G) = \min \{ \ |C|, C \subseteq \mathcal{C} / \ G = val(t) \text{ for some } t \in T(F_{HR}(C)) \ \} - 1.$$

# B Equational sets of graphs

*Equational* sets have been defined by Courcelle in the general context of Universal Algebra in a way comparable to context-free languages obtained by concatenation of letters from a fixed alphabet. In order to simplify the readability of this paper, we give the definition in a less general context by considering only what is needed to deal with *HR-equational* sets of graphs. Let $F$ be the $F_{HR}$ signature relative to a countable set of labels $\mathcal{C}$. We define a signature $F^+$ that allows to obtain the power-set of $\mathcal{GS}$ as a (many sorted)-algebra. First, for every $f \in F$, we correspond an $f_{\mathcal{P}(\mathcal{GS})}$ of the same type as follows.

- If $f$ is of type $C \times C' \to C \cup C'$ then, for any $A_C \subseteq \mathcal{GS}(C)$, $A_{C'} \subseteq \mathcal{GS}(C')$, we let $f_{\mathcal{P}(\mathcal{GS})}(A_C, A_{C'}) = \{t//_{C,C'}t',\ t \in A_C, t' \in A_{C'}\}$.
- If $f$ is of type $C \to C \setminus \{c\}$ $(c \in C)$ then, for any $A_C \subseteq \mathcal{GS}(C)$, $f_{\mathcal{P}(\mathcal{GS})}(A_C) = \{fg_c(t), t \in A_C\}$.
- For each constant operator $f$, if $t$ is the term produced by $f$ in $\mathcal{GS}$ then, the term produced by the corresponding constant operator in $\mathcal{P}(\mathcal{GS})$ is the singleton $\{t\}$.

We add to the family of operators $(f_{\mathcal{P}(\mathcal{GS})})_{f \in F}$ two new symbol operators, for every sort $C \subseteq \mathcal{C}$: a symbol $+_C$ of type $C \times C \to C$ and a constant operator $\Omega_C$ of sort $C$. For every $C \subseteq \mathcal{C}$, $\Omega_C := \emptyset$, and $A +_C A' := A \cup A'$, for any $A, A'$ subsets of $\mathcal{GS}(C)$. This ends the definition of the signature $F^+$.

Let $F$ denotes again the $F_{HR}$ signature relative to a countable set $\mathcal{C}$ of source labels. A *polynomial system* over $F$ is a sequence of equations $S =< u_1 = p_1, \ldots, u_n = p_n >$, where $U = \{u_1, \ldots u_n\}$ is an $\mathcal{S}$-sorted set of variables called the set of the *unknowns* of $S$. Each term $p_i$ is a *polynomial*, that is a term of the form $\Omega_C$ or of the form $t_1 +_C \ldots +_C t_m$, where each term $t_j$ is a *monomial* over $F \cup U$ (that is a term in $T(F \cup U)$) of the same sort as $u_i$.
Let $C_1, \ldots, C_n$ be the sorts of $u_1, \ldots, u_n$ respectively. A mapping $S_{\mathcal{P}(\mathcal{GS})}$ from $\mathcal{P}(\mathcal{GS}(C_1)) \times \ldots \times \mathcal{P}(\mathcal{GS}(C_n))$ into itself is associated to $S$ as follows. For any $A_1 \subseteq \mathcal{GS}(C_1), \ldots, A_n \subseteq \mathcal{GS}(C_n)$,

$$S_{\mathcal{P}(\mathcal{GS})}(A_1, \ldots, A_n) := (p_{1\mathcal{P}(\mathcal{GS})}(A_1, \ldots, A_n), \ldots, p_{n\mathcal{P}(\mathcal{GS})}(A_1, \ldots, A_n))$$

A *solution* of $S$ in $\mathcal{P}(\mathcal{GS})$ is an $n$-tuple $(A_1, \ldots, A_n)$, such that for each $i, 1 \leq i \leq n$, $A_i \subseteq \mathcal{GS}(C_i)$, and $(A_1, \ldots, A_n) = S_{\mathcal{P}(\mathcal{GS})}(A_1, \ldots, A_n)$. A solution of $S$ is also called a *fixed-point* of $S_{\mathcal{P}(\mathcal{GS})}$.

For set-inclusion, the least solution (in $\mathcal{P}(\mathcal{GS})$) of such a system $S$ is denoted by $(L((S, \mathcal{GS}), u_1), \ldots, L((S, \mathcal{GS}), u_n))$. An *HR-equational* set of graphs is a component of such a least solution. It is obtained as an infinite set-union:

$$L((S, \mathcal{GS}), u_i) = \bigcup_{l \geq 0} A_i^l,$$

where $A_i^0 = \emptyset$, and $(A_1^{l+1}, \ldots, A_n^{l+1}) = S_{\mathcal{P}(\mathcal{GS})}(A_1^l, \ldots, A_n^l)$.