

# Recognizable sets of graphs, hypergraphs and relational structures : a survey

*Bruno Courcelle*

Université Bordeaux 1, LaBRI  
(joint work with P. Weil and A. Blumensath)

## Summary

1. Recognizability, an algebraic notion.
2. Recognizable and equational (context-free) sets.
3. Recognizability : dependency upon the algebraic structure.
4. The graph algebras VR and HR.
5. VR- versus HR-recognizability.
6. Monadic Second-Order Transductions as generalized homomorphisms
7. Open questions

# 1. Recognizability : an algebraic notion

The recognizability of a set of finite words or trees (actually terms) can be defined in several ways :

- a) By finite automata ; deterministic ones provide linear time recognition algorithms;
- b) By finite saturating congruences.

For finite graphs, there is no automaton model, except in very special cases. Algebraic definitions via finite congruences can be given.

Finite graphs and hypergraphs can be equipped with several algebraic structures, based on operations generalizing the concatenation of words. The notion of recognizability depends on the algebraic structure under consideration.

On graphs we have **two robust algebraic structures, VR and HR**, originating from algebraic descriptions of context-free graph grammars.

On hypergraphs (or relational structures), we have **HR** as for graphs and **QF, based on quantifier-free definable transductions**.

## Recognizable sets : algebraic definition

$F$  : a finite set of operations with (fixed) arity.

$\mathbf{M} = \langle M, (f_M)_{f \in F} \rangle$  : an  $F$ -algebra.

*Definition* :  $L \subseteq M$  is *(F-)recognizable* if it is a union of equivalence classes for a finite congruence  $\approx$  on  $\mathbf{M}$  (*finite* means that  $M / \approx$  is finite).

Equivalently,  $L = h^{-1}(D)$  for a homomorphism  $h : \mathbf{M} \rightarrow \mathbf{A}$ , where  $\mathbf{A}$  is a finite  $F$ -algebra,  $D \subseteq A$ .

**Rec( $\mathbf{M}$ )** is the set of recognizable subsets of  $M$ ,  
with respect to the algebra  $\mathbf{M}$ .

*Closure properties* : **Rec( $\mathbf{M}$ )** contains  $M$  and  $\emptyset$ , and is closed under union, intersection and difference.

*Finite automata* : 1) The recognizable subsets of  $\mathbf{T}(F)$  ( the *initial* F-algebra on the set of finite ground terms on F ), are definable by finite (deterministic) tree-automata.

2) Membership of a term in a set  $L \in \text{Rec}(\mathbf{T}(F))$  can be checked in linear time.

*Finitely generated algebras.*

If the unique homomorphism  $\text{Val}_M : \mathbf{T}(F) \rightarrow \mathbf{M}$  (which evaluates a term into an element of M) is surjective, i.e., if F *generates* M:

$$L \text{ is recognizable} \Leftrightarrow \text{Val}_M^{-1}(L) \in \text{Rec}(\mathbf{T}(F))$$

Hence  $\text{Rec}(\mathbf{M})$  is finite or countable.

*Consequence for graphs.*

If M is a set of graphs, F a set of *graph operations* (generalizing the concatenation of words), if L is an F-recognizable subset of M, then the membership of a graph G in L can be checked as follows :

- find *any* term t such that  $\text{Val}_M(t) = G$ , (*can be difficult*)
- check whether  $t \in \text{Val}_M^{-1}(L)$  (can be done in time  $O(|t|)$ ).

*The many-sorted case with infinitely many sorts*

$S$  : the countable set of sorts.

$F$  : an  $S$ -signature (means that each  $f$  in  $F$  has a type

$$s_1 s_2 \dots s_k \rightarrow s, \text{ with } s, s_i \in S)$$

$\mathbf{M} = \langle (M_s)_{s \in S}, (f_{\mathbf{M}})_{f \in F} \rangle$   $F$ -algebra,  $M_s \cap M_t = \emptyset$ , if  $s \neq t$

$$\text{where } f_{\mathbf{M}} : M_{s_1} \times M_{s_2} \times \dots \times M_{s_k} \rightarrow M_s$$

*Definition* :  $L \subseteq M_s$  is *(F-) recognizable* if it is a union of equivalence classes for a congruence  $\approx$  on  $\mathbf{M}$  such that equivalent elements are of the same sort and there are finitely many classes of each sort.

## Importance of the notion of recognizability for language theory, graph grammars and algorithmics

1) We can obtain linear algorithms for wide classes of graph properties, for graphs belonging to certain finitely generated graph algebras.

2) Monadic second-order (MS) logic specifies these graph properties in a uniform way.

3) Every set of graphs (or graph property) definable by an MS formula is recognizable (resp. admits such algorithms) for appropriate graph algebras.

4) The intersection of a context-free set of graphs and a recognizable one is context-free (in the appropriate algebraic frameworks). This gives many closure properties for context-free sets of graphs via the use of MS logic as a specification language for graph properties. Recognizability is a basic tool for graph grammars, as for words and trees (terms).

5) Recognizability makes it possible to construct in a uniform way graph rewriting rules by which one can recognize sets of graphs of bounded tree-width by graph reduction.

Hence, its use for sets of graphs, in connexion with graph operations is worth being investigated.

MS logic yields easy ways to specify recognizable sets of graphs, as do finite automata and regular expressions for sets of words and trees.

## 2. Recognizable and equational (context-free) sets

**Equational sets** : Context-Free (Graph) Grammars in an algebraic setting.

In the case of words the set of context-free rules

$$S \rightarrow a S T ; \quad S \rightarrow b ; \quad T \rightarrow c T T T ; \quad T \rightarrow a$$

is equivalent to the system of two set equations:

$$S = a S T \quad \cup \quad \{ b \}$$

$$T = c T T T \quad \cup \quad \{ a \}$$

where  $S$  is the language generated by  $S$  (idem for  $T$  and  $T$ ).

For graphs we consider similarly systems of equations like:

$$S = f( k( S ), T ) \quad \cup \quad \{ b \}$$

$$T = f( T , f( g(T ), m( T ))) \cup \{ a \}$$

where  $f$  is a binary operation,  $g, k, m$  are unary operations on graphs,  $a, b$  are basic graphs.

An **equational set** is a component of the least (unique) solution of such an equation system. This is well-defined in any algebra.



# Closure properties and algebraic characterizations

## General algebraic properties

Algebraic notions	Algebraic characterizations	Closure properties
EQ		union, $\cap$ Rec
	equation systems	homomorphisms
	$Val(REC(Terms))$	MS-transductions
REC		Boolean operations
	congruences	homomorphisms <sup>-1</sup>
		MS-transductions <sup>-1</sup>

## Signatures for graphs and hypergraphs :

*HR* : graphs, hypergraphs (equivalently, **relational structures**) with “sources” : distinguished vertices handled as values of constants in relational structures  
*To be defined later*

*VR* : graphs with vertex labels  
*To be defined later*

*QF* : hypergraphs, i.e., relational structures : disjoint union and quantifier-free transductions.

### 3. Recognizability : dependency on the algebraic structure

Several algebraic structures on a same set.

What are the corresponding recognizable sets ?

*Examples :*

$\langle \mathbf{N}, 0, 1, 2 \rangle$ ,  $\langle \mathbf{N}, \text{suc}(\cdot), 0 \rangle$ ,  $\langle \mathbf{N}, \text{suc}(\cdot), \text{pred}(\cdot), 0 \rangle$

*Facts :* Let  $F$  and  $G$  be sets of operations on  $M$  giving two algebras  $M_F$  and  $M_G$ .

1) If  $F \subseteq G$  then  $\text{Rec}(M_G) \subseteq \text{Rec}(M_F)$ .

*Example of strict inclusion :*

$M_F = \langle \mathbf{N}, 0, 1, 2 \rangle$  ;  $M_G = \langle \mathbf{N}, \text{suc}(\cdot), 0, 1, 2 \rangle$

$\text{Rec}(\langle \mathbf{N}, 0, 1, 2 \rangle) =$  every subset of  $\mathbf{N}$ ;

$\text{Rec}(\langle \mathbf{N}, \text{suc}(\cdot), 0, 1, 2 \rangle) =$  the semi-linear sets of integers.

2) If  $G = F \cup$  some functions expressible by finite F-terms (F-derived operations) then :

$$\text{Rec}(M_G) = \text{Rec}(M_F).$$

*Example* :  $M_F = \langle X^*, *, a, b, \varepsilon \rangle$  ;  $M_G = \langle X^*, *, f, a, b, \varepsilon \rangle$

where for  $u, v$  in  $X^*$  , we let  $f(u,v) = a*u*v$

3) Adding non-derived operations :

*Example* :

$M_F = \langle X^*, *, a, b, \varepsilon \rangle$  ;  $M_G = \langle X^*, *, \mu, a, b, \varepsilon \rangle$

where  $\mu$  the mirror-image.

$\mu$  is not F-derived, however :  $\text{Rec}(M_G) = \text{Rec}(M_F).$

*Proof sketch* :

Let L be F-recognizable with congruence  $\sim$ . We define a G-congruence:

$$u \equiv v \text{ if and only if } u \sim v \text{ and } \mu(u) \sim \mu(v)$$

The verifications use :  $\mu(\mu(u)) = u$  and  $\mu(u.v) = \mu(v).\mu(u)$

If  $\sim$  has  $n$  classes, then  $\equiv$  has  $\leq n^2$  classes and saturates L.

## A generalization

Proposition (*Conservative extension of a signature*) :

If  $G = F \cup U$  where  $U$  is a *finite* set of unary operations *closed under composition*, such that for every  $f$  in  $F$  and every  $h$  in  $U$ , there exists a term  $t$  over  $F$  and elements  $h_1, \dots, h_k$  of  $U$  such that, for all  $x_1, \dots, x_k$  :

$$h(f(x_1, \dots, x_k)) = t(h_1(x_1), \dots, h_k(x_k))$$

then :

$$\text{Rec}(M_G) = \text{Rec}(M_F).$$

*Proof* : Let  $L$  be  $F$ -recognizable with congruence  $\sim$ . We define the  $G$ -congruence

$$u \equiv v \text{ if and only if } u \sim v \text{ and } h(u) \sim h(v)$$

for all  $h$  in  $U$ .

More complicated formulations can be given for many sorted algebras.

## Degenerated cases of recognizability

M the base set, F the set of operations.

1) If the algebraic structure is "too weak", every set is recognizable. If M is countable, then  $\text{Rec}(\mathbf{M})$  is **uncountable**.

*Case 1* : F has only nullary symbols and unary operations generating by composition a finite monoid  $\mathbf{U}$  :

For every  $L \subseteq M$ , the equivalence relation :

$$u \approx v \Leftrightarrow \text{for every } h \text{ in } \mathbf{U}, h(u) \text{ and } h(v) \text{ are both in } L, \\ \text{or both not in } L$$

is a saturating congruence with  $\leq 2^{\text{Card}(\mathbf{U})}$ .

*Note* : If  $\mathbf{U} = \{\text{identity}\}$  and the two classes are L and  $M - L$ .

*Example of  $\mathbf{U}$*  : For graphs with n sources, numbered from 1 to n, the group of permutations generated by the circular shift and the exchange of the first two sources.

Case 2 : There is a binary concatenation but it is "too weak".

*Example* : Every set of complete graphs  $K_n$  is HR-recognizable, because the operation of parallel composition of the HR signature cannot split large cliques. *To be discussed later.*

---

2) If the algebraic structure is "too rich", the only recognizable sets are  $M$  and  $\emptyset$ .

*Example* :  $\langle \mathbf{N}, \text{suc}(\cdot), \text{pred}(\cdot), 0 \rangle$  where  $\text{pred}(0) = 0$  and  $\text{pred}(n+1) = n$ .

Let  $L \subseteq \mathbf{N}$ ,  $L \neq \mathbf{N}$ ,  $L \neq \emptyset$ .

Without loss of generality,  $0 \in L$ ,  $p$  is the least integer  $\notin L$ .

Assume  $\approx$  is a congruence saturating  $L$ .

Consider  $x \approx y$  with  $x \leq y = x + p + q$ . Then  $\text{pred}^{x+q}(x) = 0$ ,  $\text{pred}^{x+q}(y) = p$  but  $0$  is not  $\approx$ -equivalent to  $p$ . Contradiction.

Two equivalent integers have difference at most  $p$ , and  $\approx$  has infinitely many classes.  $L$  is not recognizable.

*Another example :*

$$M_F = \langle X^*, *, a, b, \varepsilon \rangle ; M_G = \langle X^*, *, \mathbf{sh}(\cdot), a, b, \varepsilon \rangle$$

where **sh** is the shift operation defined by :

$$\mathbf{sh}(\varepsilon) = \varepsilon, \mathbf{sh}(au) = ua, \mathbf{sh}(bu) = ub$$

for every  $u \in X^*$ .

We have  $\text{Rec}(M_G) \subset \text{Rec}(M_F)$  with strict inclusion.

*Proof sketch :*  $a^*b \notin \text{Rec}(M_G)$ . Otherwise let  $\equiv$  be a finite congruence, we have  $a^n b \equiv a^{n+p} b$  for some  $n, p > 1$ . Hence, using  $n+1$  shifts,  $a^n b \equiv a^{p-1} b a^{n+1}$ . But  $a^n b$  belongs to  $L$  and  $a^{p-1} b a^{n+1}$  does not. Contradiction.

**Fact :** Every commutative regular language  $L$  is in  $\text{Rec}(M_G)$ .

Hence,  $\text{Rec}(M_G)$  is not trivial.

*Proof sketch :* Let  $\equiv$  be a finite congruence for  $L$ . The least  $F$ -congruence containing  $\equiv$  and closed under commutations of letters, is also a congruence for shift, has less classes than  $\equiv$ , and saturates  $L$ .

## 4. The graph algebras VR and HR

### *HR operations*

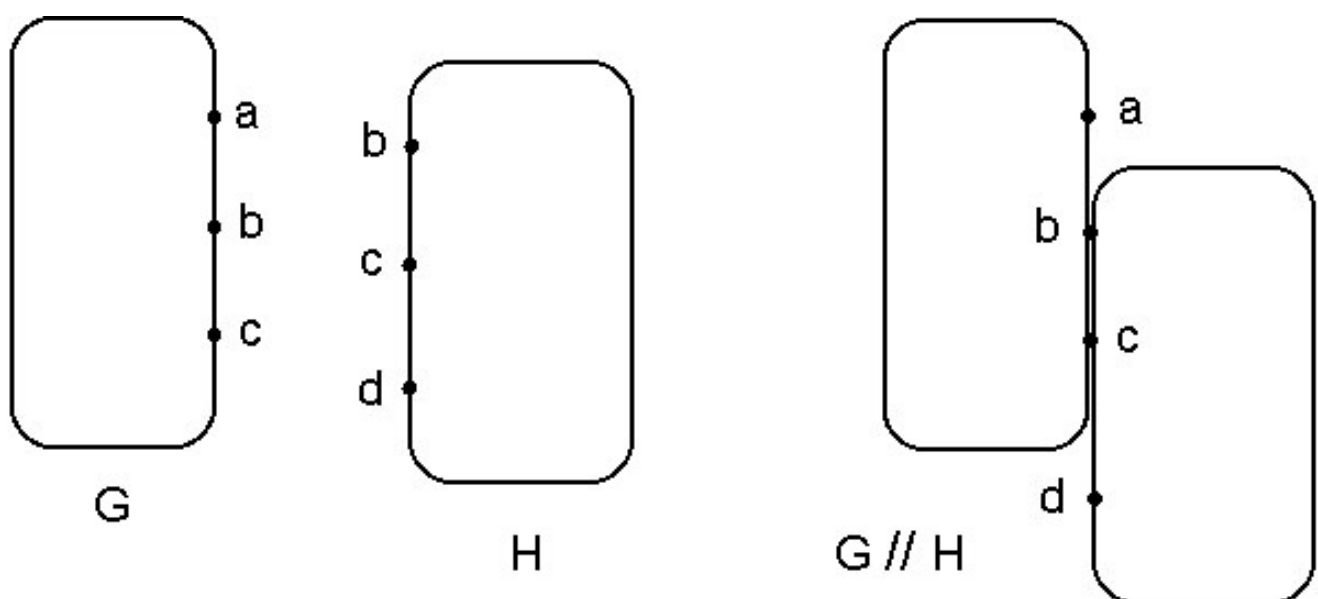
(Origin : **H**yperedge **R**eplacement hypergraph grammars ;  
associated complexity measure : **tree-width**)

Graphs have distinguished vertices called *sources*,  
pointed to by labels from a set of size  $k$  :  $\{a, b, c, \dots, h\}$ .

Binary operation(s) : *Parallel composition*

$G // H$  is the disjoint union of  $G$  and  $H$   
and sources with same  
label are **fused**.

(If  $G$  and  $H$  are not disjoint, one  
first makes a copy of  $H$  disjoint  
from  $G$ .)





Unary operations :

*Forget a source label*

$Forget_a(G)$  is  $G$  without  $a$ -source :  
the source is no longer distinguished ; it is made "internal".

*Source renaming :*

$Rena_{a,b}(G)$  renames  $a$  into  $b$  (assuming  $b$  is not a source label in  $G$ ).

Nullary operations denote *basic graphs*, the connected graphs with one or two vertices.

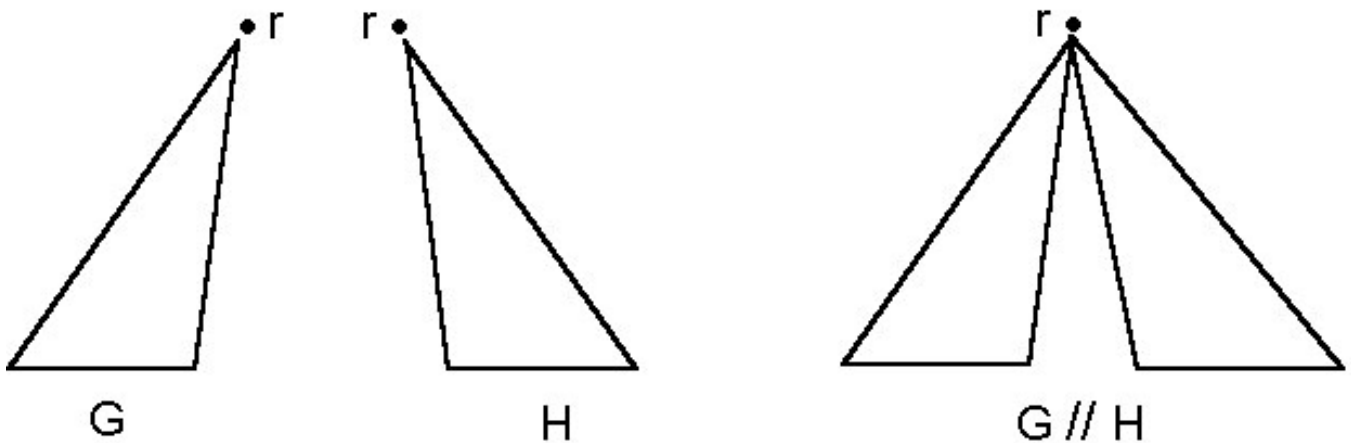
*For dealing with hypergraphs one takes more nullary symbols for denoting hyperedges.*

The proper algebraic framework : a **many sorted algebra** where **each finite set of source labels is a sort**. The above operations are overloaded.

*Proposition:* A graph has tree-width  $\leq k$  if and only if it can be constructed from basic graphs with  $\leq k+1$  labels by using the operations  $//$ ,  $Ren_{a,b}$  and  $Forget_a$ .

*Example :* Trees are of tree-width 1, constructed with two source labels,  $r$  (root) and  $n$  (new root):

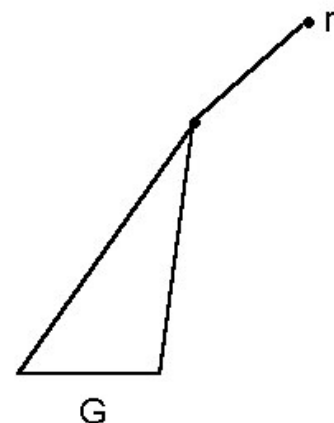
Fusion of two trees at their roots :



Extension of a tree by parallel composition with a new edge, forgetting the old root, making the "new root" as current root :

$$E = r \bullet \text{---} \bullet n$$

$$Ren_{n,r}(Forget_r(G // E))$$



Non-derived unary operation that can be added without changing the HR-equational and the HR-recognizable sets :

$Fuse_{a,b}$  fuses the  $a$ -source and the  $b$ -source and forgets  $b$ .

Proof : By the proposition on **conservative extensions of signatures** and using in particular the following equalities :

$$Fuse_{a,b}(G // H) = Fuse_{a,b}(G) // Fuse_{a,b}(H)$$

$$Fuse_{a,b}(Forget_c(G)) = Forget_c(Fuse_{a,b}(G))$$

$$Fuse_{a,b}(Ren_{c,a}(G)) = Ren_{c,a}(Fuse_{c,b}(G))$$

Technical details concerning the many sorts are omitted.

# VR operations

(origin : **V**ertex **R**eplacement graph grammars ; associated complexity measure : **clique-width** )

Graphs are simple, directed or not.

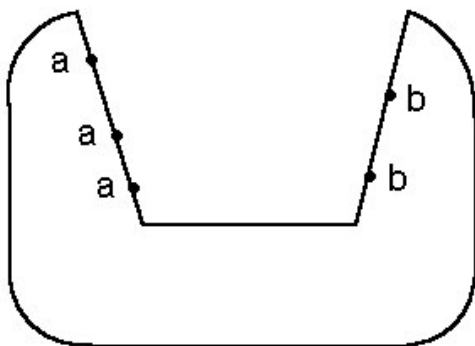
k labels :  $a, b, c, \dots, h$ .

Each vertex has one and only one label ; a label  $p$  may label several vertices, called the  $p$ -ports.

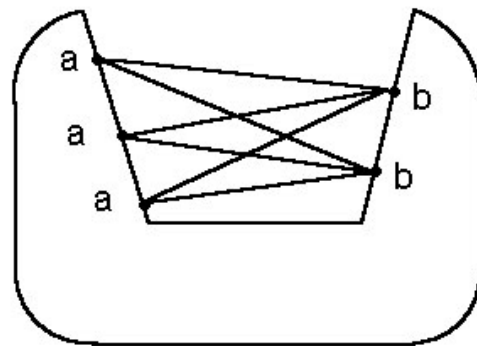
Binary operation: disjoint union  $\oplus$

Unary operations: Edge addition denoted by  $Add\text{-}edg_{a,b}$

$Add\text{-}edg_{a,b}(G)$  is  $G$  augmented with (un)directed edges from every  $a$ -port to every  $b$ -port.



G



$Add\text{-}edg_{a,b}(G)$

$Relab_{a,b}(G)$  is  $G$  with every vertex labelled by  $a$  relabelled into  $b$

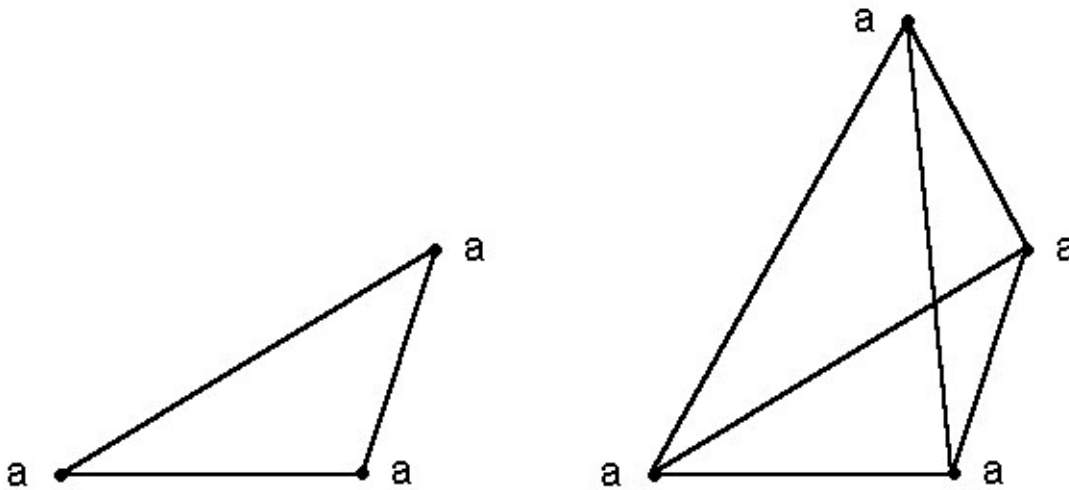
*Basic graphs* are those with a single vertex.

Definition: A graph  $G$  has clique-width  $\leq k$

$\Leftrightarrow$  it can be constructed from basic graphs by means of  $k$  labels and the operations  $\oplus$ ,  $Add-edg_{a,b}$  and  $Relab_{a,b}$ .

Its (exact) clique-width,  $cwd(G)$ , is the minimum such  $k$ .

*Example* : Cliques have clique-width 2.



$K_n$  is defined by  $t_n$

$$t_{n+1} = \text{Relabb,a}(\text{Add-edga,b}(t_n \oplus \mathbf{b}))$$

*Another example* : *Cographs*

Cographs are generated by  $\oplus$  and  $\otimes$  defined by :

$$G \otimes H = \text{Relabb,a}(\text{Add-edga,b}(G \oplus \text{Relaba,b}(H)))$$

=  $G \oplus H$  with “all edges” between  $G$  and  $H$ .

## More operations

We can enrich VR into  $VR^+$  by adding unary operations of several types.

### Quantifier-free definable operations

A typical example is the *edge complement* for loop-free undirected graphs :

If  $G = \langle V_G, \text{edg}_G(\dots) \rangle$  then  $\text{Compl}(G) = G' = \langle V_{G'}, \text{edg}_{G'}(\dots) \rangle$

where :  $\text{edg}_{G'}(x,y) : \Leftrightarrow x \neq y \wedge \neg \text{edg}_G(x,y)$ .

With the **conservative extension of signatures**, we get that

**VR-recognizable**  $\Leftrightarrow$  **VR<sup>+</sup>-recognizable**.

*Fusion, a non-quantifier-free definable operation :*

$Fuse_a(G)$  is obtained from  $G$  by fusing all  $a$ -ports  
into a single vertex.

*Questions :* For graphs, is it true that  $(VR^+ + \text{Fusion})\text{-recognizable} = \text{VR-recognizable}$  ? Yes (A. Blumensath and B.C. 2004)

Is it true that  $(VR^+ + \text{Fusion})\text{-equational} = \text{VR-equational}$  ?

Yes (B.C. and J.Makowsky).

For hypergraphs, the second result extends. And the first one ????

## 5. VR- and HR-recognizable sets.

### Bounded density functions and collapsing

Let  $f$  be a function from finite graphs to integers.

A set  $L$  is  $f$ -bounded if  $\{f(G) \mid G \in L\}$  is bounded.

Useful functions :

Degree( $G$ ) = maximum degree of a vertex.

Clique-Minor( $G$ ) = maximum  $n$  such that  $K_n$  is a minor of  $und(G)$

( $und(G)$ ) = the undirected graph associated with  $G$ ,  
or  $G$  if undirected).

Tw( $G$ ) = tree-width of  $G$ .

Cwd( $G$ ) = clique-width of  $G$ .

Biclique( $G$ ) = maximum  $n$  such that  $K_{n,n}$  is  
a subgraph of  $und(G)$ .

DirectedBiClique( $G$ ) = maximum  $n$  such that  $\overrightarrow{K}_{n,n}$  is a subgraph of  $G$ .

Density( $G$ ) =  $\lceil |E(G)| / |V(G)| \rceil$

UnifDensity( $G$ ) =  $\text{Max}\{ \text{Density}(H) \mid H \text{ subgraph of } G \}$ .

Density-bounded = Sparse

UnifDensity-bounded = Uniformly sparse = Bounded arboricity



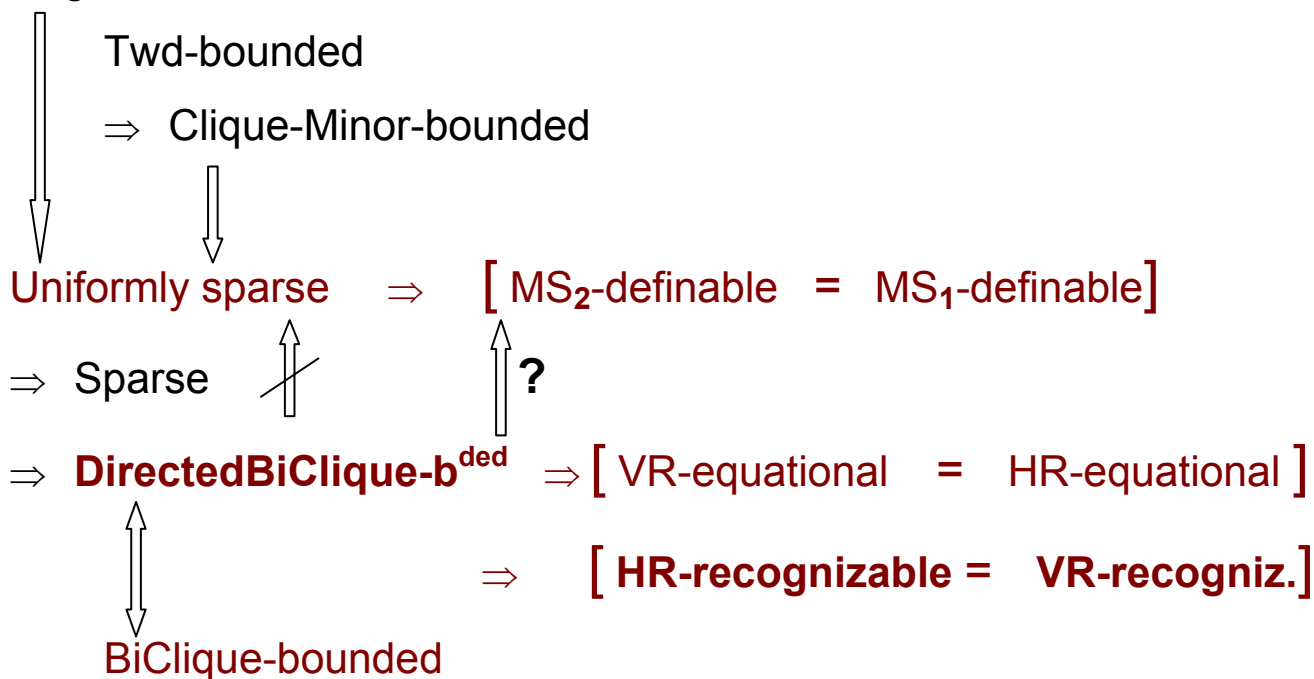
*General situation*

HR-equational  $\Rightarrow$  VR-equational

VR-recognizable  $\Rightarrow$  HR-recognizable

*Consequences of boundedness conditions*

Degree-bounded



## Some combinatorial lemmas :

1) BiClique-bounded  $\Leftrightarrow$  DirectedBiClique-bounded

because for  $G$  directed :

a)  $G$  without  $\overrightarrow{K_{p,p}}$  implies  $und(G)$  without  $K_{m,m}$  where  $m = p2^{2p}$

b)  $und(G)$  without  $K_{p,p}$  implies  $G$  without  $\overrightarrow{K_{p,p}}$

2) Uniformly sparse  $\Rightarrow$  BiClique-bounded but not vice-versa.

$\Rightarrow$  is clear.

$\Leftarrow$  is false because graphs without  $K_{2,2}$  may have  $\alpha \cdot n^{3/2}$  edges for  $n$  vertices by Bollobas (Extremal graph theory, 1978).

By Thm 2.10 (Chap. VI, p.316) :

For each  $t > 2$ , there is  $\alpha$ , and for each  $n$ , there is a graph with  $n$  vertices,  $\alpha \cdot n^{1+\varepsilon}$  edges and without  $K_{t,t}$ , where  $\varepsilon = 1 - 2/(t+1) > 0$ .

## The strict inclusion

$$\text{VR-recognizable} \subset \text{HR-recognizable}$$

*Inclusion* : If a set is VR-recognizable, it is  $\text{VR}^+$ -recognizable, hence HR-recognizable because the HR operations are  $\text{VR}^+$ -derived. (The operation  $Fuse_{a,b}$  is quantifier-free definable on relational structures with nullary symbols denoting sources.)

### *Strictness* :

*Fact 1* : Every set of complete graphs  $K_n$  is HR-recognizable, because the operation of parallel composition of the HR signature **cannot split large cliques**.

*Proof* : Consequence of a more general proposition.

*Fact 2* : Let  $P \subseteq \mathbf{N}$  (positive integers), not  $\mathbf{N}$ -recognizable (e.g., the set of prime numbers, not semi-linear).

Consider the terms  $t_n$  of width 2, constructed previously, to define  $K_n$ .

Consider  $K_P = \{K_n \mid n \in P\}$ . It is HR-recognizable by Fact 1. If it would be VR-recognizable, so would be  $\{t_n \text{ 's} \mid n \in P\}$ , and  $P$  would be  $\mathbf{N}$ -recognizable.

We let  $\Gamma_k$  be the set of graphs with sources  $1, \dots, k$ .

For  $G, H \in \Gamma_k$ , we let  $G \& H = \text{Forget}_{1, \dots, k}(G // H)$ .

Proposition : Let  $L \subseteq \Gamma_0$  (a set of graphs without sources).

It is HR-recognizable if and only if,

for each  $k$ , there exists a finite equivalence relation  $\cong$

on  $\Gamma_k$  such that :

for all  $G \cong G', H \cong H'$  in  $\Gamma_k$ , we have :

$G \& H \in L$  if and only if  $G' \& H' \in L$ .

*Proof* : "only if" is clear because  $\&$  is HR-derived.

"If" One lets  $\cong_k$  be the finite congruence on  $\Gamma_k$  defined by  $G \cong_k G'$  iff for every  $H$  in  $\Gamma_k$

$G \& H \in L$  if and only if  $G' \& H \in L$ .

For  $G, G'$  with sources we let  $G \equiv G'$  iff they have the same sets  $\sigma(G)$  and  $\sigma(G')$  of source labels and for every bijection  $h : \sigma(G) \rightarrow \{1, \dots, k\}$ , we have  $\text{Ren}_h(G) \cong_k \text{Ren}_h(G')$ . It is a congruence. The verification uses the following facts :

$$\text{Forget}_{All}(\text{Ren}_h(\text{Forget}_a(G)) // H) = \text{Forget}_{All}(\text{Ren}_{h'}(G) // (H \oplus \mathbf{k+1}))$$

$$\text{Forget}_{All}(\text{Ren}_h(G // K) // H) = \text{Forget}_{All}(\text{Ren}_{h'}(G) // (\text{Ren}_{h'}(K) // H))$$

Proposition : Let  $L \subseteq \Gamma_0$ . If for each  $k$  there is a finite subset  $B_k$  of  $\Gamma_k$  such that, whenever  $G \& H \in L$ , at least one of  $G$  and  $H$  is in  $B_k$ , then  $L$  is HR-recognizable.

Proof : For every  $G \in \Gamma_k - B_k$ , we let  $A(G) = \{ H \in \Gamma_k / G \& H \in L \}$ . It is a subset of the finite set  $B_k$ .

We let  $G \approx G'$  iff  
 either  $G = G' \in B_k$   
 or  $G, G' \in \Gamma_k - B_k$  and  $A(G) = A(G')$ .

This equivalence is finite.

For all  $G, G', H$  with  $G \approx G'$  :

$G \& H \in L$  if and only if  $G' \& H \in L$ .

This gives the desired fact.

Corollary : Every set of cliques is HR-recognizable.

Proof :

One takes for  $B_k$  the set of graphs all vertices of which are sources. (If  $G$  and  $H$  have *internal* (non-source) vertices, then  $G \& H$  cannot be complete.)

## A similar more general result

*Proposition* : Let  $\mathbf{M}$  be an F-algebra,  $L \subseteq M$ .

Let  $U$  be the set of all *unary derived operators* with parameters  $m_1, m_2, \dots$  in  $M$ , i.e., defined as :  $x \mapsto t(x, m_1, \dots, m_k)$ , where  $t$  is a term with a single occurrence of  $x$  (parameters  $m_1, m_2, \dots$  need not be defined by finite terms).

If there exist  $M_{\text{small}} \subseteq M$  and  $U_{\text{small}} \subseteq U$ , both finite, such that :

for all  $m \in M$  and  $u \in U$  :

$u(m) \in M_{\text{small}} \Rightarrow u \in U_{\text{small}} \text{ and } m \in M_{\text{small}}$

$u(m) \in L \Rightarrow u \in U_{\text{small}} \text{ or } m \in M_{\text{small}} \text{ (or both)}$

then  $L$  is recognizable.

*Proof sketch* : For  $m \in M$ , let  $E(m) = \{ u \in U \mid u(m) \in L \}$ .

Define  $m \approx m'$  iff

either  $m = m' \in M_{\text{small}}$  or  $m, m' \notin M_{\text{small}}$  and  $E(m) = E(m')$ .

In the later case  $E(m) = E(m') \subseteq U_{\text{small}}$  (which is finite).

Hence  $\approx$  is a finite equivalence. The identity belongs to  $U$ , hence  $\approx$  saturates  $L$ .

It is a *congruence* :

We check that :

$$m \approx m' \text{ and } u \in U \Rightarrow u(m) \approx u(m').$$

Clear if  $m = m' \in M_{\text{small}}$  .

Otherwise  $m, m' \notin M_{\text{small}} \Rightarrow u(m), u(m') \notin M_{\text{small}}$

If  $u' \in E(u(m))$ , then  $u'(u(m)) \in L$ .

Hence  $u' \circ u \in E(m)$ ,  $u' \circ u \in E(m')$ , and  $u' \in E(u(m'))$ .

Hence  $E(u(m)) = E(u(m'))$  and thus,  $u(m) \approx u(m')$ .

This implies that  $\approx$  is a congruence. Because, if  $m \approx m'$  and  $p \approx p'$  and  $f$  is a binary operation,  $f(m,p) \approx f(m',p)$  and  $f(m',p) \approx f(m',p')$ . Hence  $f(m,p) \approx f(m', p')$ , so  $\approx$  is a congruence.

*Theorem* : If a set of directed graphs is  
 DirectedBiClique-bounded and HR-recognizable,  
 $\Rightarrow$  it is VR-recognizable.

*Proof sketch.*

Let  $m$  be the largest integer such that  $\overrightarrow{K_{m,m}}$  is a subgraph of a graph in a set  $L$ , assumed HR-recognizable.

*Some facts :*

- 1) Several vertices may have the same *port* label.
- 2) The operation  $Add-edg_{a,b}$  adds edges. We get a subgraph  $\overrightarrow{K_{p,q}}$  where  $p$  is the number of  $a$ -ports and  $q$  is the number of  $b$ -ports.
- 3) If  $p$  remains bounded by  $k$  (in the process of constructing graphs in a class  $C$ ), then the  $a$ -ports can be replaced by (at most  $k$  sources) with source labels  $(a,1)$ , ...,  $(a,k)$ .
- 4) If we apply  $Add-edg_{a,b}$  in a case where  $p$  and  $q > m$ , we go outside of  $L$ .

In order to construct a VR congruence, we represent VR operations (limited by 4) ) by HR operations on *expansions*.  
**Expansions anticipate "authorized" edge additions.**



# Idea of the construction of a congruence saturating a set L

Start from an equivalence  $\sim$ , for example :

$$G \sim G' \text{ defined by } G \in L \Leftrightarrow G' \in L$$

Refine it into  $\equiv$  such that

$$G \equiv G' \Rightarrow \text{for everything happening to } G \text{ and } G', \text{ i.e. whenever}$$
$$H = t(G, M_1, \dots, M_p)$$
$$H' = t(G', M_1, \dots, M_p)$$
$$\text{we have } H \sim H'$$
$$\text{(this implies } H \equiv H')$$

## Transfer of congruence from HR to VR

$$\sim \text{ on HR} \qquad \qquad \qquad \equiv \text{ on VR}$$

$$\text{Future}(G) \longleftarrow G$$

contains information  
on the effects of all  
possible applications to  
G of VR operations ;  
( a set of graphs )

Define  $G \equiv G'$  as :  $\text{Future}(G) \sim \text{Future}(G')$

*Expansions* : For  $G$ , graph with **ports** :

A port label is *void* or *small* or *large*

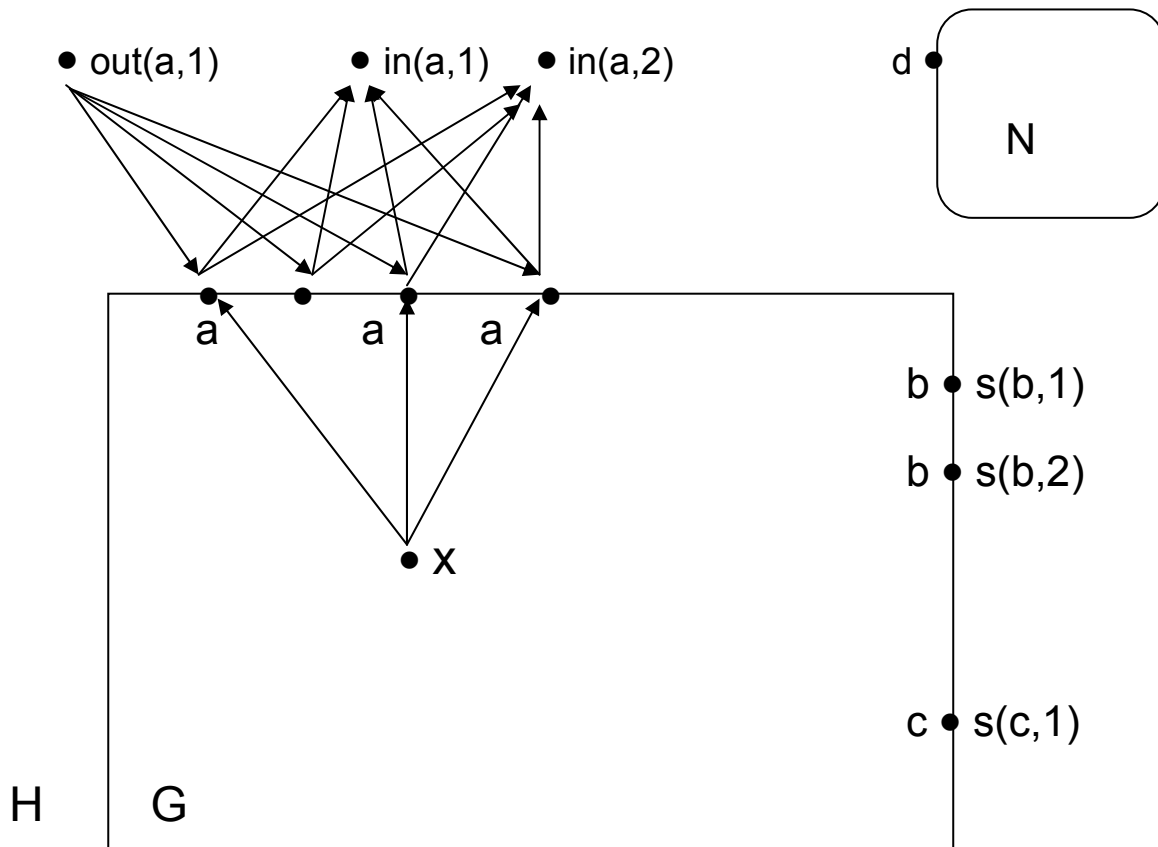
if it labels **no** vertex, **or at most  $m$** , or **at least  $m+1$** .

An *expansion* of  $G$  is a supergraph  $H$  with **sources** and without  $\overrightarrow{K_{m,m}}$  as a subgraph.

*See example next page with  $m = 2$ .*

These "new" edges are "potential" edges used in the following way :

*Add- $edg_{a,d}$*  ( $G \oplus N$ ) (where  $N$  is a graph with a single  $d$ -port), can be replaced by the fusion of the *in(a,2)*-source of  $H$  (expansion of  $G$ ), and the  $d$ -port of  $N$ , handled as a source. Source fusion is an HR-operation. Hence, the typical VR operation is performed by an HR-operation on expansions.



An expansion  $H$  of a graph  $G$

*Example with  $m = 2$ .*

Ports labelled by small labels  $b, c$  are made into sources.

Ports labelled by large label  $a$  are equipped with directed edges from  $/$  to **new** source vertices ; source labels are  $out(a,1), in(a,1), in(a,2)$ .

Because of edges from an existing vertex  $x$  to 3  $a$ -ports, at most one source  $out(a,.)$  is introduced. Two  $in(a,.)$ -sources can be introduced but no more, otherwise one creates a  $\overrightarrow{K_{3,3}}$ .

**Definition :** From a given HR-congruence  $\sim$  we let the VR-congruence  $\equiv$  be :  $G \equiv G'$  iff

- 1)  $G$  and  $G'$  both contain  $\overrightarrow{K_{m+1,m+1}}$  or both do not, and we have :
- 2)  $G$  and  $G'$  satisfy the same *small* FO formulas (*i.e.*, First Order formulas of quantifier rank at most  $2m+2$ ), and for every expansion  $H$  of  $G$ , there is an expansion  $H'$  of  $G'$  such that  $H' \sim H$ , and  $H, H'$  satisfy the same *small* FO formulas, and *vice versa*.

This *equivalence is finite* because there are finitely many small FO formulas and every graph with  $p$  port labels has finitely many expansions up to isomorphism (and the number of expansions is upper bounded in terms of  $p$  and  $m$ ).

It *saturates*  $L$  because a graph without ports and without  $\overrightarrow{K_{m+1,m+1}}$  is its unique expansion.  $\sim$  and  $\equiv$  coincide on these graphs.

It remains to prove that  $\equiv$  is a *VR-congruence*.

Let  $G \equiv G'$ , we want to prove that :

$$M = \text{Forget}_a(G) \equiv M' = \text{Forget}_a(G').$$

The first verifications are easy.

Let  $N$  be an expansion of  $M$ . There is an expansion  $H$  of  $G$  such that  $N = t(H)$  for some (fixed) HR-term  $t$ .

There exists a corresponding expansion  $H'$  of  $G'$  (since  $G \equiv G'$ ) with  $H \sim H'$ , and the desired expansion  $N'$  of  $M'$  is defined as  $t(H')$ .

Since  $\sim$  is an HR-congruence,  $N' = t(H') \sim t(H) = N$ .

Since  $\text{Forget}_a$  is quantifier-free definable, if  $H$  and  $H'$  satisfy the same small formulas, so do  $N$  and  $N'$ .

The same technique applies to  $\text{Relab}_{a,b}$  and to the binary operations denoted by  $\otimes_J$  such that :

$$G \otimes_J H = \text{Add-edg}_{a,b}(\dots(\text{Add-edg}_{c,d}(G \oplus H)\dots))$$

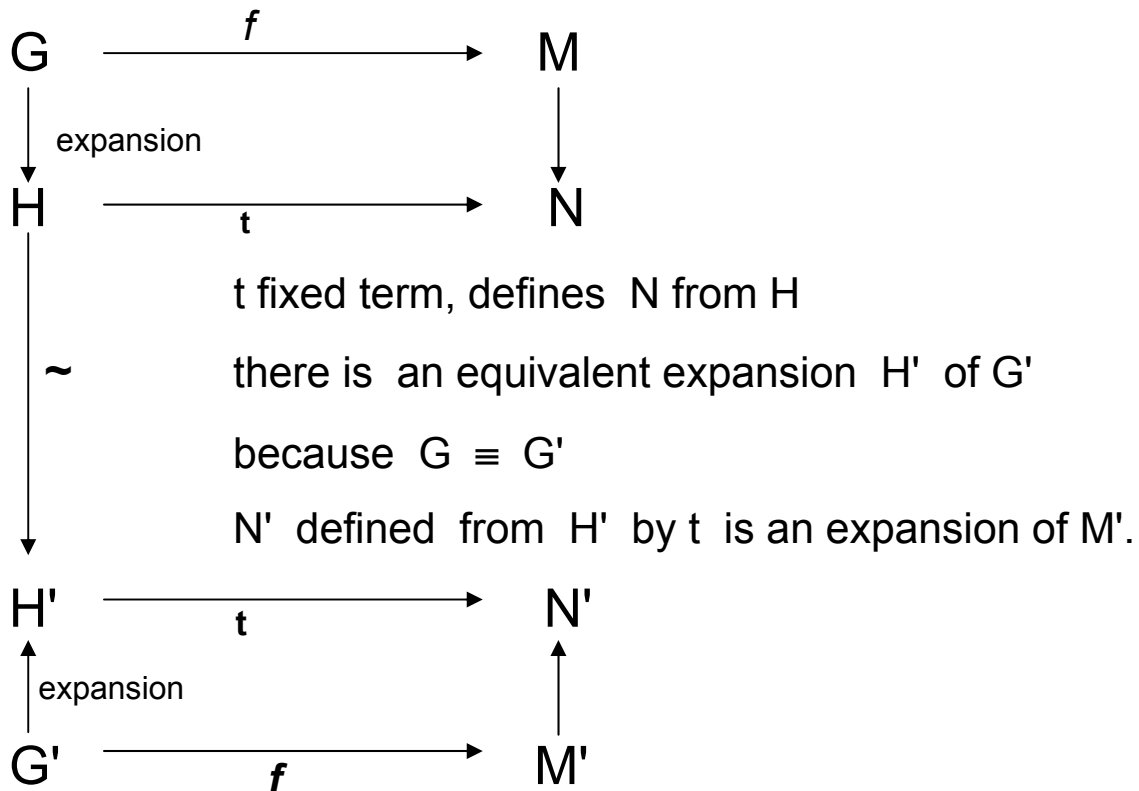
that add edges between  $G$  and  $H$  (and not inside  $G$  and  $H$ ) ; restrictions on labels can insure that.

We obtain recognizability for the *NLC algebra*, a variant of the VR algebra, introduced by E. Wanke (1994).

*Lemma* : NLC-recognizability = VR-recognizability.

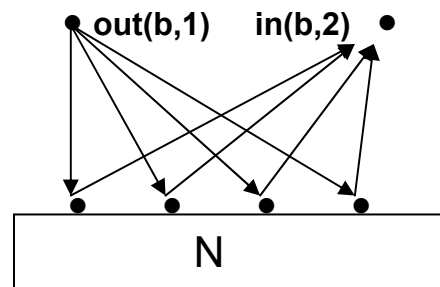
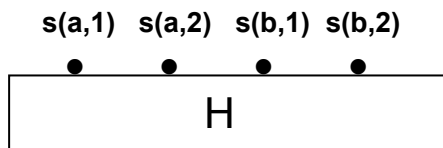
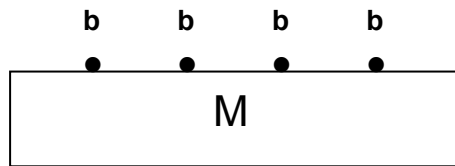
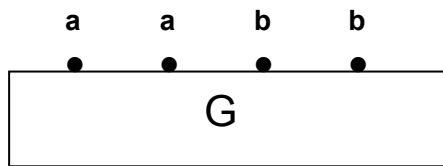
A picture : Let  $f$  be a unary operation.

For every expansion  $N$  of  $M$ , there exists an expansion  $H$  of  $G$  such that :

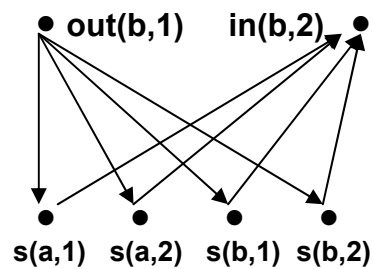


$N'$  is  $\sim$ -equivalent to  $N$ . This gives  $M' \equiv M$ .

Example for operation  $f = Relab_{a,b}$ ,  $m = 2$  :



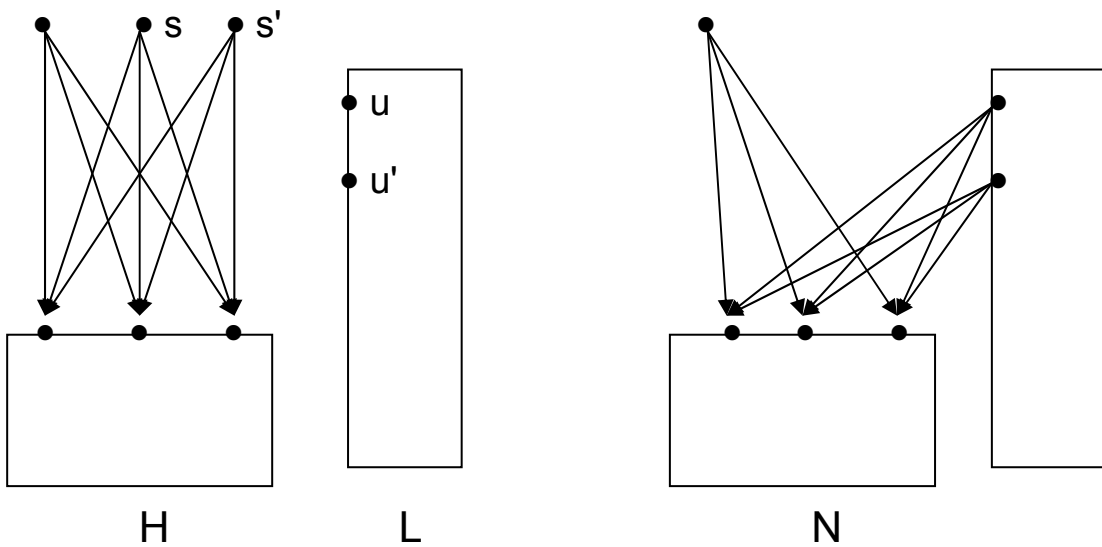
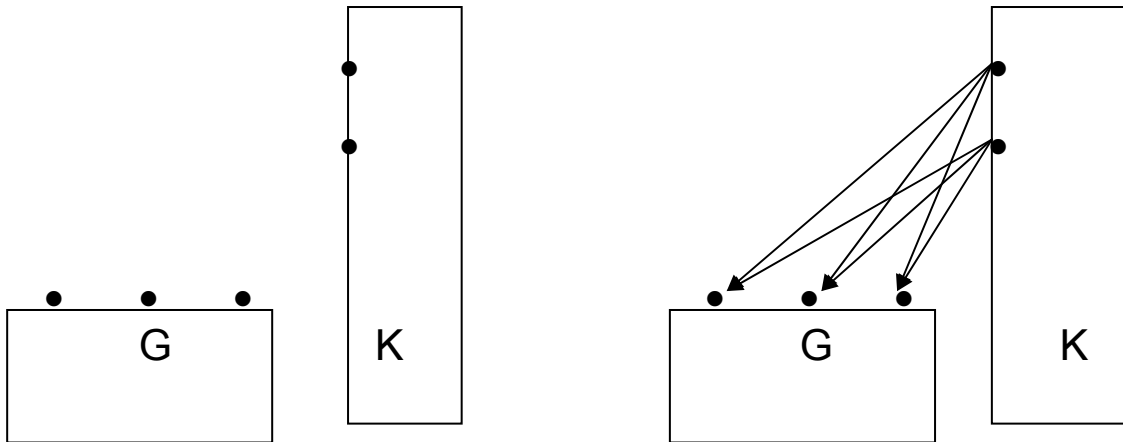
$E =$



$$N = t(H) = \text{Forget}_{\{s(a,1), s(a,2), s(b,1), s(b,2)\}} (H // E)$$

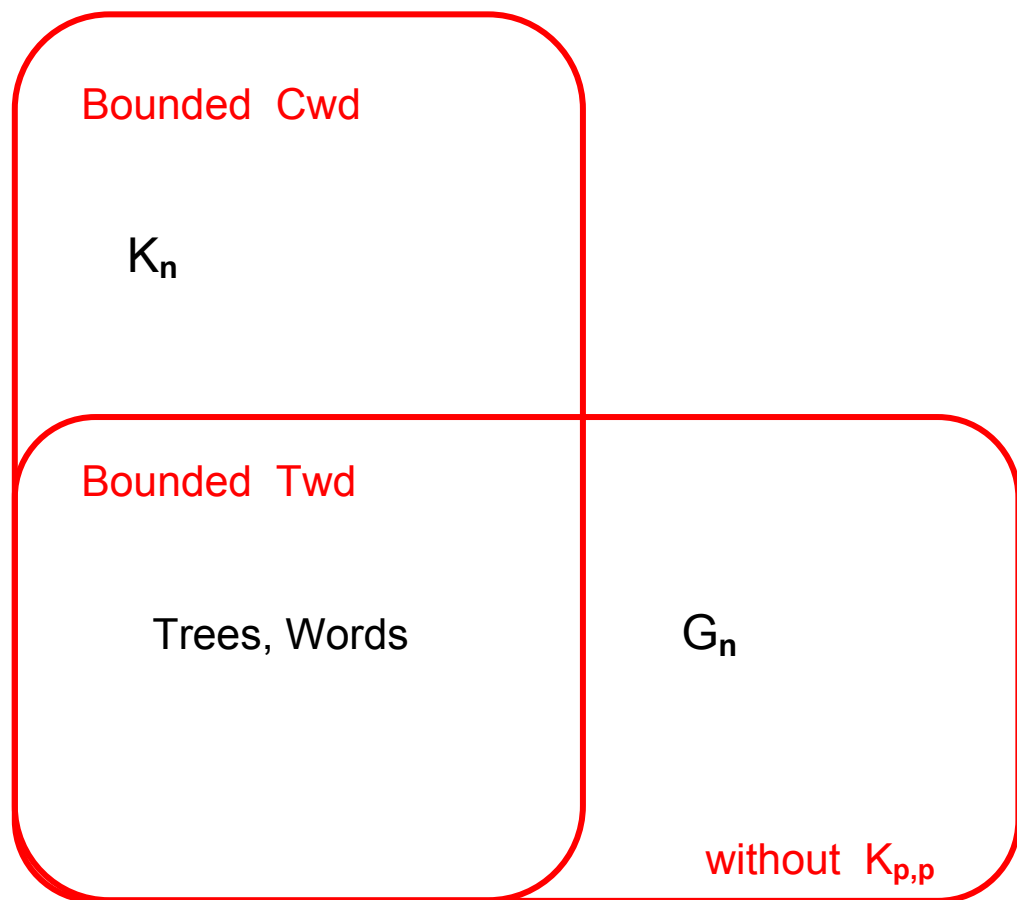
An example with  $m = 3$  for :

$$G \otimes_J K = \text{Add-edg}_{b,a}(K \oplus G)$$



$$N = t(H,L) = \text{Forget}_{\{s,s'\}}(H // \text{Ren}_{u,s}(\text{Ren}_{u',s'}(L)))$$

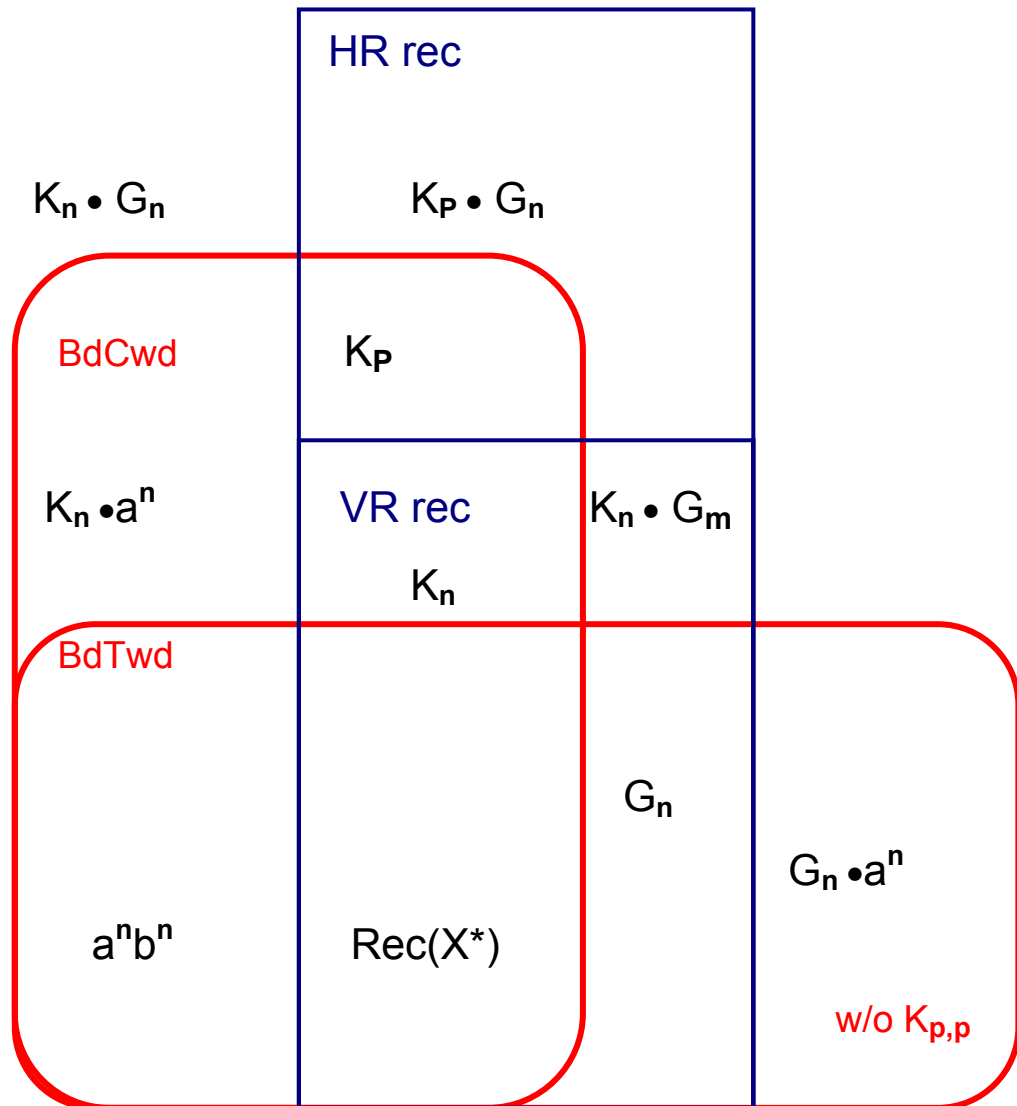




$K_n$  the set of all cliques

$G_n$  the set of all  $n \times n$  grids

A set of graphs has bounded tree-width if and only if it has bounded clique-width and no  $K_{p,p}$  for some  $p$ .



$K_n$  = the set of all cliques

$G_n$  = the set of all  $n \times n$  grids

$K_n \bullet a^n$  =  $n$ -cliques glued by one vertex to an  $n$ -string

$G_n \bullet a^m$  =  $n \times n$  grids glued by one vertex to any string

*Questions* : Is it true that every HR-recognizable set of unbounded tree-width has uncountably many HR-recognizable subsets ?

Is it true that every VR-recognizable set of unbounded clique-width has uncountably many VR-recognizable subsets ?

## 6. Monadic Second-Order Transductions as generalized homomorphisms

### Monadic Second-Order (MS) Logic

= First-order logic on power-set structures

= First-order logic extended with (quantified) variables denoting subsets of the domains.

### MS properties :

transitive closure,

properties of paths, connectivity,

planarity (via Kuratowski, uses connectivity),

k-colorability.

*Notation* :  $\Sigma$  : a finite set of relation symbols ;

MS-def( $\Sigma$ ) = the sets of finite  $\Sigma$ -relational structures characterized as  $\{ S / S \models \varphi \}$  for an MS formula  $\varphi$

**Main result** :  $\text{MS-def}(\Sigma) \subseteq \text{REC}(\text{QF})_{\Sigma}$

For graphs :

$\text{MS-def}(\{\text{edg}\}) \subseteq \text{REC}(\text{VR}) = \text{REC}(\text{VR}^+)$

## MS transductions

$\text{STR}(\Sigma)$  : the set of finite  $\Sigma$ -relational structures.

Multivalued mappings  $\tau : \text{STR}(\Sigma) \rightarrow \text{STR}(\Gamma)$

$$S \longmapsto T = \tau(S)$$

where  $T$  is :

a) defined by MS formulas

b) inside the structure:  $S \oplus S \oplus \dots \oplus S$   
(fixed number of disjoint "marked" copies of  $S$ )

c) in terms of "parameters" i.e. subsets  
 $X_1, \dots, X_p$  of the domain of  $S$

*Proposition* : The composition of two

MS transductions is an MS transduction.

*Remark* : For each tuple of parameters  $X_1, \dots, X_p$

satisfying an MS property,  $T$  is uniquely defined.

$\tau$  is multivalued by the different choices of parameters.



## The fundamental property of MS transductions :

$$\begin{array}{ccc} S & \longrightarrow & \tau(S) \\ \tau^\#(\psi) & \longleftarrow & \psi \end{array}$$

Every MS formula  $\psi$  has an effectively computable *backwards translation*  $\tau^\#(\psi)$ , an MS formula, such that :

$$S \models \tau^\#(\psi) \quad \text{iff} \quad \tau(S) \models \psi$$

The verification of  $\psi$  in the **object structure**  $\tau(S)$  **reduces** to the **verification** of  $\tau^\#(\psi)$  in the given structure  $S$ .

$S$  describes  $\tau(S)$ ; the MS properties of  $\tau(S)$  are described by MS properties of  $S$ .

### Main results :

- 1)  $L \subseteq \text{STR}(\Sigma)$  is QF-equational iff  $L = \tau(\text{Trees})$ , for an MS transduction  $\tau$ .
- 2) QF- equational sets are preserved under MS transductions
- 3) (A.B. and B.C.,2004) : **QF-recognizable sets are preserved under *inverse* MS transductions.**

## Relationships between algebraic and logical notions

Algebraic notions	Algebraic characterizations	Logical characterizations	Closure properties
EQ			union, $\cap$ Rec
	equation systems	MS-trans(Trees)	homo
	$Val(REC(Terms))$		MS-trans
REC			Boolean opns
	congruences	MS-def $\subset$ REC	homo <sup>-1</sup>
			MS-trans <sup>-1</sup>

### Signatures for graphs and hypergraphs :

$HR$  : graphs and hypergraphs with “sources”

$VR$  : graphs with vertex labels (“ports”)

$VR^+$  :  $VR$  with quantifier-free operations  
(ex. edge complement)

$QF$  : hypergraphs, i.e., relational structures  
(disjoint union  $\oplus$  and quantifier-free definable  
transductions)

Theorem (A. Blumensath, B.C.,2004) : QF-recognizable sets are preserved under *inverse* MS transductions.

This is already known for MS definable sets and the recognizable sets of graphs of bounded tree-width.

**Proof sketch** : Every MS transduction is the composition of MS transductions of 3 types :

- Copy<sub>k</sub>
- Guessing of unary relations
- Parameterless noncopying

1) Copy<sub>k</sub> :

$$S \quad | \rightarrow \quad S \oplus S \oplus \dots \oplus S \quad (\text{k times})$$

augmented with binary relations  $Y_{i,j}$  for  $1 \leq i < j \leq k$  defined as

$\{(x,y) \mid x \text{ is the } i\text{-copy, } y \text{ is the } j\text{-copy of some } u \text{ in } D_S\}$

Facts : a)  $\text{Copy}_k(S \oplus T) = \text{Copy}_k(S) \oplus \text{Copy}_k(T)$

b) For  $f$  quantifier-free, there is a quantifier-free  $g$  :

$$\text{Copy}_k(f(S)) = g(\text{Copy}_k(S))$$

Hence Copy<sub>k</sub> is “almost” a homomorphism, and REC is preserved under inverse homomorphisms.



## 2) Closure under forgetting unary relations :

We let  $\text{fgt}_{\Pi} : \text{STR}(\Sigma \cup \Pi) \rightarrow \text{STR}(\Sigma)$  just « forget »  $\Pi$ ,  
 $\Pi$  is a finite set of unary relations.

Its inverse consists in « guessing » the relations in  $\Pi$ , this can be done by means of **parameters** of an MS transduction.

*Lemma* : If  $L \in \text{REC}$ , then  $\text{fgt}_{\Pi}(L) \in \text{REC}$ .

*Proof* : From a congruence  $\equiv$  for  $L$ , we define the equivalence :

$$S \cong T \text{ iff } \{ [U]_{\equiv} / \text{fgt}_{\Pi}(U) = \mathbf{S} \} = \{ [U]_{\equiv} / \text{fgt}_{\Pi}(U) = \mathbf{T} \}$$

It is finite, saturates  $\text{fgt}_{\Pi}(L)$ .

It is a congruence : we use the fact that

$$\begin{aligned} S \oplus T = \text{fgt}_{\Pi}(W) \text{ iff} \\ \text{there exist } S', T' \text{ such that } W = S' \oplus T', \\ S = \text{fgt}_{\Pi}(S'), \text{ and } T = \text{fgt}_{\Pi}(T'), \end{aligned}$$

and a similar observation for unary operations.

### 3) Parameterless noncopying MS transductions.

We let  $L \in \text{REC}(\Gamma)$

$\tau$  be a parameterless noncopying MS transduction :  
 $\text{STR}(\Sigma) \rightarrow \text{STR}(\Gamma)$  of quantifier-height  $k$

We prove that  $\tau^{-1}(L) \in \text{REC}(\Sigma)$

From a congruence  $\equiv$  for  $L$ , we define, on each set  $\text{STR}(\Delta)$ , the equivalence :

$S \cong T$  iff

$\text{tp}_k(S) = \text{tp}_k(T)$  and for every parameterless noncopying MS transduction  $\mu : \text{STR}(\Delta) \rightarrow \text{STR}(\Gamma)$  of quantifier-height at most  $k$ , we have  $\mu(S) \equiv \mu(T)$ .

( $\text{tp}_k(S)$  is the MS theory of  $S$  of quantifier height at most  $k$ .)

It is finite, saturates  $\tau^{-1}(L)$  (because  $\tau$  is one of the considered transductions  $\mu$ ).

It is a congruence. The less easy case is that of  $\oplus$ .

## 7. Open questions

*Question 1* : Which operations, quantifier-free definable or not, yield conservative extensions of VR, HR, STR ?

*Question 2* : Under which operations, quantifier-free definable or not, are  $\text{Rec}(\text{VR})$  and  $\text{Rec}(\text{HR})$  closed ?

(The case of  $\text{Rec}(\text{HR})$  is considered in B.C.: *(HR-)Recognizable sets of graphs, equivalent definitions and closure properties*, 1994. It is not hard to see that  $\text{Rec}(\text{VR})$  is closed under  $\oplus$  (disjoint union) but not under the operations  $\text{Add-} \text{edg}_{a,b}$ .)

*Question 3* : Can one find a complexity measure on hypergraphs and relational structures that generalize clique-width and has good properties with respect to operations like the restriction to an induced subhypergraph ?

*Question 4* : Is it true that every HR-recognizable set of unbounded tree-width has uncountably many HR-recognizable subsets ?

Is it true that every VR-recognizable set of unbounded clique-width has uncountably many VR-recognizable subsets ?