

On Seese's Conjecture

Bruno Courcelle

Université Bordeaux 1, LaBRI

Summary

1. Graphs, Languages, Theories
2. MS-compatible structure transformations and
MS-transductions
3. Seese's Conjecture
4. Tree-width and Clique-width
5. Proved cases of the conjecture
6. Reductions between specific cases
7. Description of MS-compatible structure
transformations
8. Other open questions

1. Logical expression of graph properties

Graphs are simple, directed, finite.

(Extension is easy to undirected graphs, to hypergraphs,
and to relational structures)

$G = \langle V, \text{edg}(\dots) \rangle$

Vertices, edge relation

φ logical formula

$G \models \varphi$ is a property of G

$G \models \varphi(x,y)$ is a property of a pair (x,y) of vertices of G

Languages : FO : First-Order logic

MS : Monadic Second-Order logic

SO : Second-Order logic

Incidence graphs

$\text{Inc}(G) = \langle V \cup E, \text{src}(\cdot, \cdot), \text{tgt}(\cdot, \cdot) \rangle$

Vertices and edges, source relation, target relation :

$\text{src}(e, x)$ and $\text{tgt}(e, y)$ iff $e : x \rightarrow y$

Using incidence graphs :

φ logical formula (with src , tgt) :

$\text{Inc}(G) \models \varphi$ is a property of G expressed via
its incidence graph.

Languages : FO_2 , MS_2 , SO_2

Remark : $\text{MS}_2 = \text{GSO}$, Guarded second-order logic

(Graedel et al.)

Expressive powers :

$\text{FO} = \text{FO}_2 < \text{MS} < \text{MS}_2 < \text{SO} = \text{SO}_2$

Satisfiability, Theories

Let \mathcal{C} be a class of finite graphs, let L be a logical language

$\text{Th}(L, \mathcal{C})$ is the set of formulas in L that are true in
all graphs in \mathcal{C}

$\text{Sat}(L, \mathcal{C})$ is the set of formulas in L that are true in
some graph in \mathcal{C}

Are these sets recursive ?

Is the L -theory of \mathcal{C} decidable ? Is the L -satisfiability
problem for \mathcal{C} decidable ?

\mathcal{C}	FO	MS	MS ₂	SO
All (finite)	Und	Und	Und	Und
Grids	Dec	Und	Und	Und
Directed Cliques	Dec	Dec	Und	Und
Forests (Discrete graphs)	Dec	Dec	Dec	Und
Planar		Und	Und	Und

2. MS-compatible transformations

Definition of an **MS-transduction** (sometimes called an interpretation):

A transformation τ of structures such that :

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

where T is **defined by MS formulas**

inside the structure:

$$S \oplus S \oplus \dots \oplus S$$

(fixed number of disjoint copies of S)

Example of an MS-transduction

The *square* mapping δ on words: $u \mid \rightarrow uu$

We let $u = \text{aac}$

S	$\cdot \rightarrow \cdot \rightarrow \cdot$
	a a c

$S \oplus S$	$\cdot \rightarrow \cdot \rightarrow \cdot$	$\cdot \rightarrow \cdot \rightarrow \cdot$
	a a c	a a c
	p1 p1 p1	p2 p2 p2

$\delta(S)$	$\cdot \rightarrow \cdot \rightarrow \cdot \rightarrow \cdot \rightarrow \cdot \rightarrow \cdot$
	a a c a a c

In $\delta(S)$ we redefine Suc as follows :

$$Suc(x,y) : \Leftrightarrow p1(x) \& p1(y) \& Suc(x,y) \\ \vee p2(x) \& p2(y) \& Suc(x,y) \\ \vee p1(x) \& p2(y) \& \text{"x has no successor"} \\ \& \text{"y has no predecessor"}$$

We also remove the "marker" predicates $p1, p2$.

MS-compatible graph transformations.

Definition : A transformation of graphs

(represented by structures):

$$S \longrightarrow \tau(S)$$

$$\tau^{\#}(\psi) \longleftarrow \psi$$

such that every MS formula ψ 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 ψ in the object structure $\tau(S)$ reduces to the verification of $\tau^{\#}(\psi)$ in the given structure S .

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

Consequence: If a set of structures L has a decidable MS satisfiability problem, then so has $\tau(L)$.

The set $\text{Sat}(\text{MS}, \tau(L))$ many-one reduces to the set $\text{Sat}(\text{MS}, L)$.

Proposition: Every MS-transduction is MS-compatible.

3. Monadic Second-Order Problems

Theorem (Seese): If $\text{Sat}(\text{MS}_2, \mathcal{C})$ is recursive, then $\text{Inc}(\mathcal{C}) = \tau(T)$ for some MS - transduction τ and some set T of finite trees, equivalently, \mathcal{C} has bounded **tree-width**.

Conjecture (Seese) : If $\text{Sat}(\text{MS}, \mathcal{C})$ is recursive, then $\mathcal{C} = \tau(T)$ for some MS - transduction τ and some set T of finite trees, equivalently \mathcal{C} has bounded **clique-width**.

Stronger forms :

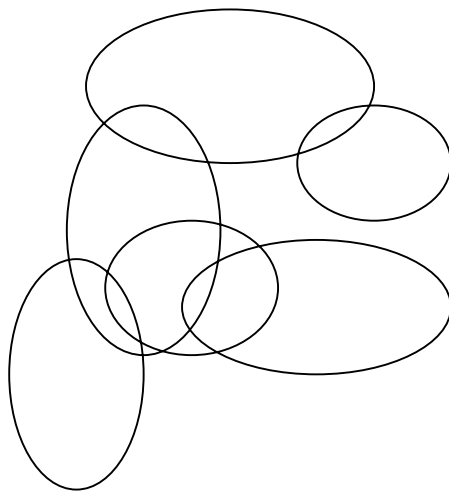
Theorem (Lapoire): If $\text{Sat}(\text{MS}_2, \mathcal{C})$ is recursive, then $\text{Inc}(\mathcal{C}) = \tau(T)$ and $T = \sigma(\text{Inc}(\mathcal{C}))$ for some MS - transductions τ and σ and some set T of finite trees. Hence, $\text{Sat}(\text{MS}, T)$ is **recursive** and $\text{Sat}(\text{MS}_2, \mathcal{C})$ reduces to $\text{Sat}(\text{MS}, T)$.

Conjecture : If $\text{Sat}(\text{MS}, \mathcal{C})$ is recursive, then $\mathcal{C} = \tau(T)$ and $T = \sigma(\mathcal{C})$ for some MS - transductions τ and σ and some set T of finite trees. Hence, **$\text{Sat}(\text{MS}, T)$ is recursive.**

4. Tree-width and clique-width

4.1 Tree-width

Tree-decompositions of width k :



Boxes have size at most $k + 1$

Tree-width of G , $\text{twd}(G) =$ minimal width of a tree-decomposition.

A syntax for tree-decompositions

We use graphs with distinguished vertices pointed to by labels from a set of size k :

$$\{a, b, c, \dots, h\}$$

Operations are *Parallel composition*

$G // H$ is the disjoint union of G and H with distinguished vertices labelled in the same way fused.

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

and *Forget a label*

$\text{Forget}_a(G)$ is G without label a (the distinguished vertex labelled by a is made "ordinary").

Basic graphs are the connected graphs with one or two vertices

Proposition: A graph has tree-width $\leq k$ iff it can be constructed from basic graphs with $\leq k$ labels by using the operations $//$ and Forget_a .

Theorem :

(1) For fixed k , the mapping :

$$\text{term } t \mapsto \text{Inc}(G)$$

where G is denoted by t (where t is a term using k labels, whence G has tree-width at most k) is an MS-transduction.

(2) The set $\text{Inc}(G)$ for G of tree-width at most k is $\tau(\text{FiniteTrees})$ for some MS-transduction τ .

(3) Conversely, if C is a set of graphs and

$\text{Inc}(C) = \tau(\text{FiniteTrees})$ for some MS-transduction τ ,

then C has bounded tree-width. [B.C. and J. Engelfriet].

4.2 (Bi)clique-decompositions and clique-width

Clique-width is another complexity measure on graphs that defines a more powerful hierarchy.

Notation: $cwd(G)$

Facts: 1. If $twd(G) \leq k$, then $cwd(G) \leq 2^{O(k)}$

2. Cliques (and cographs) have clique-width 2
(and unbounded twd).

3. Planar graphs have unbounded clique-width.

4. Deciding if $cwd(G) \leq k$ is NP,

polynomial if $k \leq 3$,

perhaps NP-complete for $k = 4$.

Operations defining clique-width

We use k labels: a, b, c, \dots, h .

Each vertex has one and only one label.

Operations are the *disjoint union* \oplus ,

and 2 (types of) *unary operations*:

Add- $edg_{a,b}(G) = G$ augmenting G with edges from every vertex labelled by a to every vertex labelled by b

Relab $_{a,b}(G) = G$ relabelling in G every vertex labelled by a into b

Basic graphs are those with a single vertex.

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

it can be constructed from basic graphs with k

labels by using the operations \oplus ,

Add- $edg_{a,b}$ and Relab $_{a,b}$.

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

Theorem (comparison with a previous theorem):

(1) For fixed k , the mapping :

$$\text{term } t \mapsto G \text{ (resp. Inc}(G))$$

where G is denoted by t (term using k labels) is an MS-transduction; G has clique-width (resp. tree-width) at most k .

(2) The set of graphs G of clique-width at most k

(of graphs $\text{Inc}(G)$ for G of tree-width at most k)

is $\tau(\text{FiniteTrees})$ for some MS-transduction τ .

(3) Conversely, if C is a set of graphs $= \tau(\text{FiniteTrees})$

(resp. $\text{Inc}(C) = \tau(\text{FiniteTrees})$) for some MS-

transduction τ , then C has bounded clique-width

[J.Engelfriet] (resp. tree-width, [B.C. and J. Engelfriet]).

These results yield the equivalent forms of Seese's Theorem and Conjecture.

5. Proofs: Theorem and special cases of the conjecture.

5.1 Proof Sketch of the Theorem (for MS_2 and tree-width).

It uses the result of Robertson and Seymour (Graph Minors V, 1986) that graphs without a fixed planar graph as a minor have bounded tree-width.

1. The transformation

$$\text{Minors} : (\text{Inc}(G), X, Y) \mapsto \text{Inc}(H)$$

for G a graph,

X a set of edges to be contracted,

Y a set of vertices and edges to be deleted

H : the resulting minor,

is an MS-transduction.

2. If $\text{Sat}(MS_2, C)$ is decidable, so is $\text{Sat}(MS_2, \text{Minors}(C))$,

3. and $\text{Minors}(C)$ can only contain finitely many square grids (otherwise, $\text{Sat}(MS_2, C)$ is undecidable because arbitrary long Turing Machine computations can be encoded).

4. Hence C has bounded tree-width by R&S, GM V.

5.2 Cases where the conjecture reduces to the theorem :

Planar graphs (Seese, 1991), graphs of bounded degree, graphs without a fixed graph H as a minor, and, subsuming all these cases :

For the class U_k of uniformly k -sparse graphs G , i.e. those such that every (finite) subgraph M of G has at most $k \cdot |V(M)|$ edges.

Theorem (Courcelle, to app.) : For each k , the mapping :

$$G \mapsto \text{Inc}(G)$$

is an MS-transduction on U_k .

Corollary : For graphs in U_k , MS_2 formulas can be translated into equivalent MS formulas ; hence if $\mathcal{C} \mu U_k$ and $\text{Sat}(MS, \mathcal{C})$ is decidable, so is $\text{Sat}(MS_2, \mathcal{C})$ hence \mathcal{C} has bounded twd.

No hint for a general proof of the conjecture !

Corollary : For every k , for every class of graphs \mathcal{C}

having uniformly k -sparse edge-complements,

if $\text{Sat}(\text{MS}, \mathcal{C})$ is decidable, then \mathcal{C} has bounded clique-width.

Hint : Because then, $\text{Sat}(\text{MS}, \mathbf{K})$ is decidable where \mathbf{K} is the class of edge-complements of the graphs in \mathcal{C} ,

hence has bounded tree-width (because $\mathbf{K} \mu U_k$),
hence has bounded clique-width,

hence so has \mathcal{C} because bounded clique-width is preserved under edge-complement.

Observation :

The conjecture holds for sets of graphs that are :

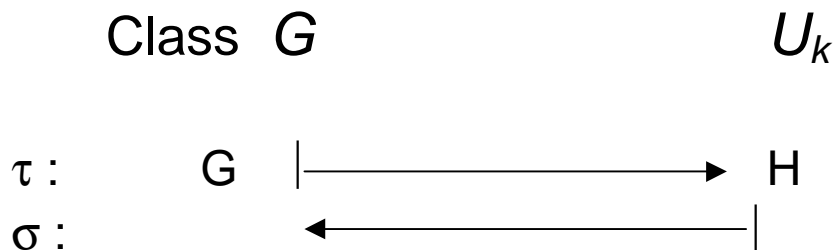
either very sparse (uniformly k -sparse),

or very dense (with uniformly k -sparse edge-complements)

What happens in the middle ?

5.3 Classes of unbounded tree-width

Method :



where τ, σ are MS-transductions,

H is uniformly k-sparse and « describes » G

τ maps G to H and σ maps H to G ;

If C is a subclass of G :

Sat(MS, C) decidable \Rightarrow Sat(MS, $\tau(C)$) decidable

$\Rightarrow \tau(C)$ has bounded **tw**

$\Rightarrow \tau(C)$ has bounded **cw**

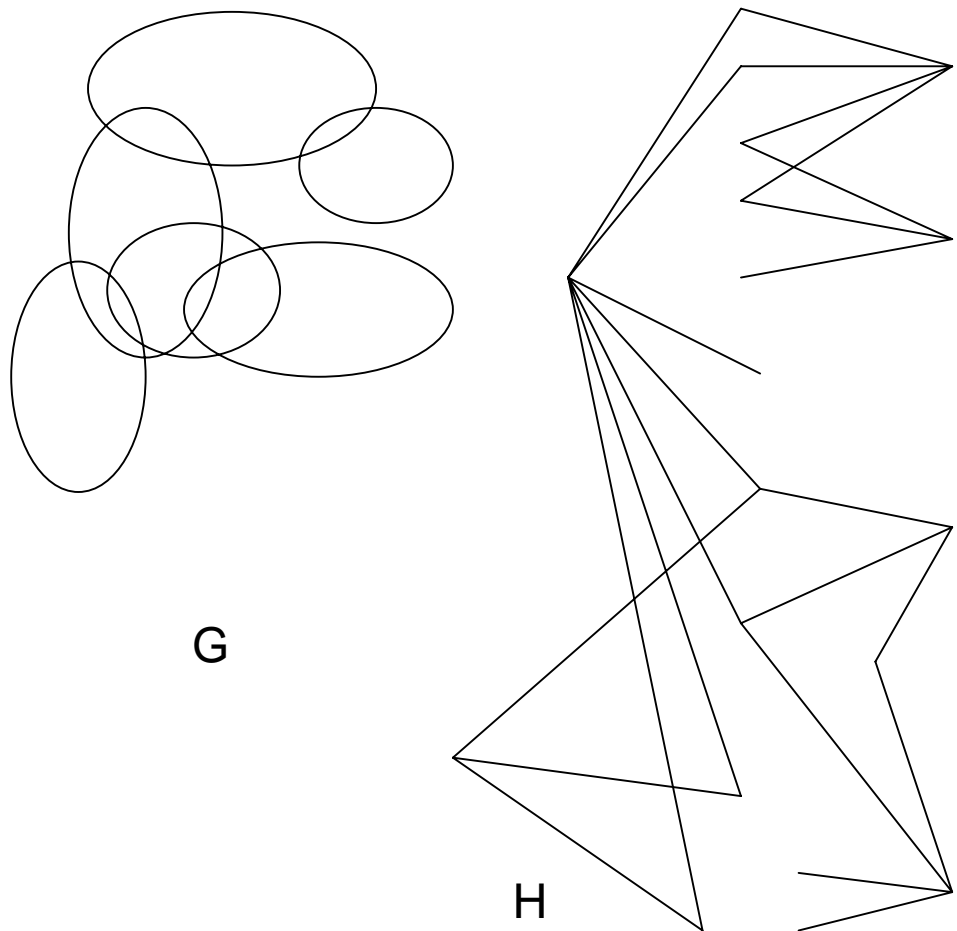
$C = \sigma(\tau(C))$ has bounded cw.

The strong form of the conjecture holds.

Cases where this method applies.

1. Chordal graphs of clique-degree at most d .

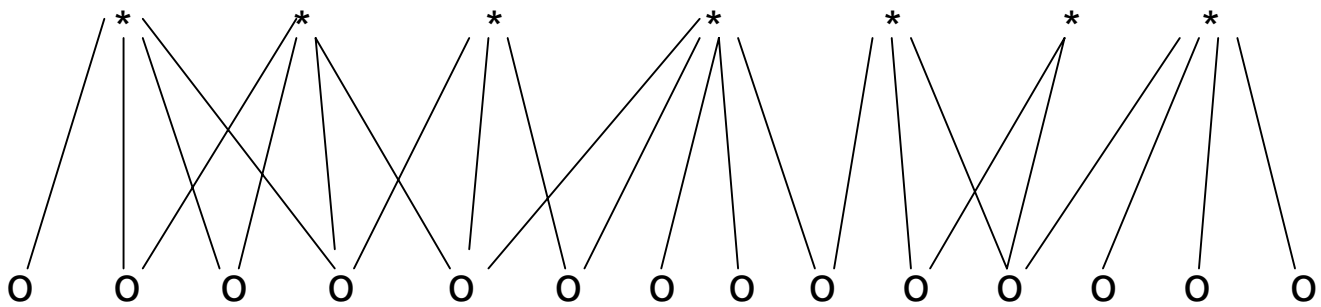
Tree-decompositions with cliques as boxes, each vertex belongs to at most d maximal cliques :



H is a bipartite graph; its vertices corresponding to those of G have degree at most d . The transduction σ creates cliques from stars.

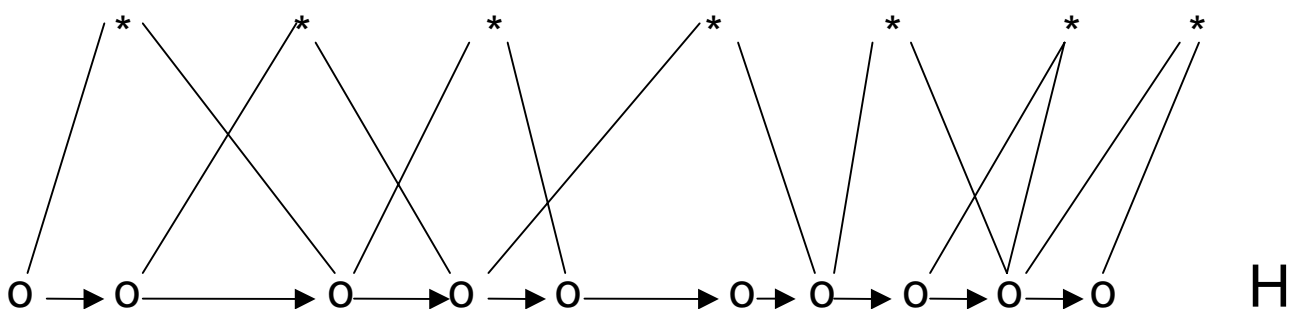
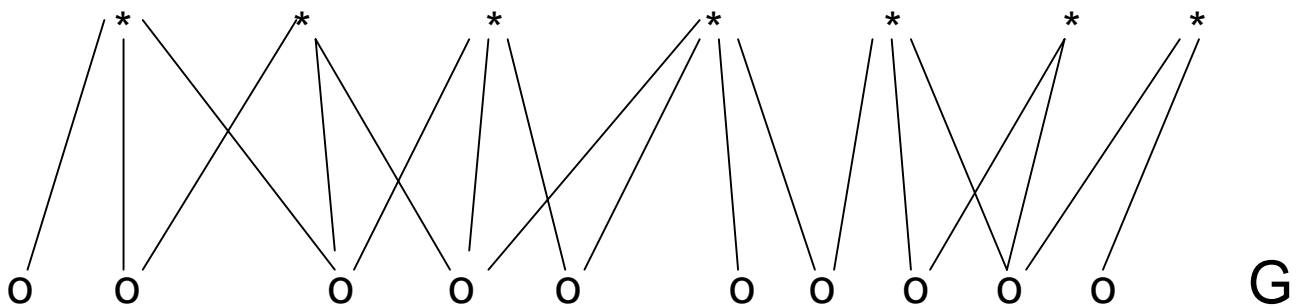
2. Convex bipartite graphs (or hypergraphs)

A hypergraph is convex if there exists a linear order on vertices such that every hyperedge is an interval



Order is unique only for **prime** convex hypergraphs, and can be defined in MS logic.

Convex nonprime hypergraphs can be built from prime ones by vertex substitution. There exists a unique canonical modular decomposition.

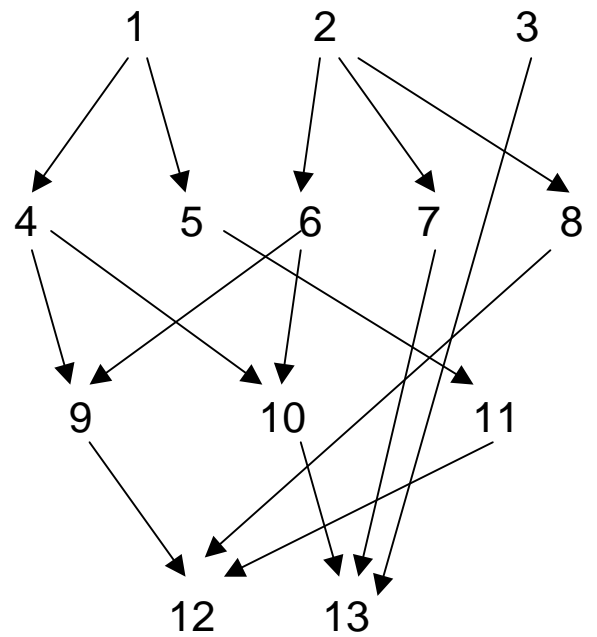
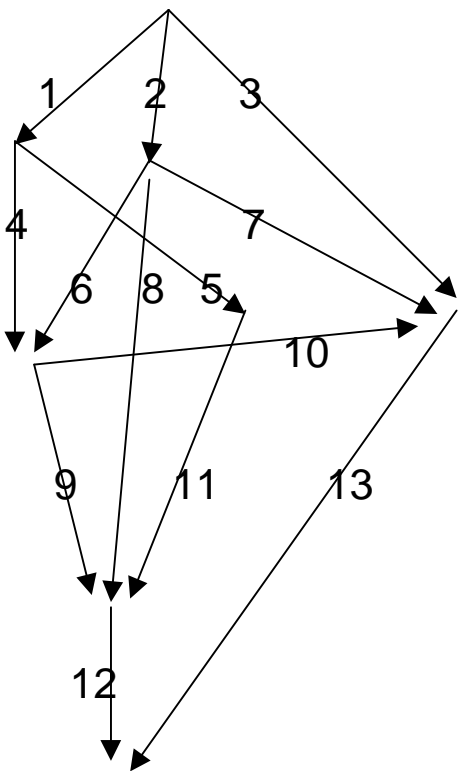
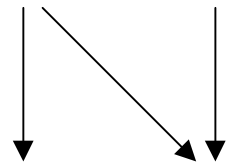


3. N-free Hasse diagrams

We use *directed line graphs*:

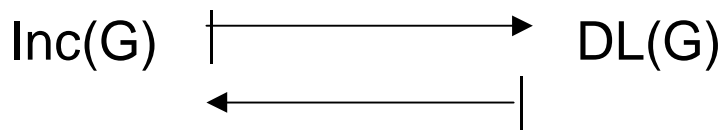
Simple DAG with source and sink
(bipolar orientation)

N-free Hasse diagram



Fact (Heuchenne, 1964): The mapping DL is a bijection.

The mapping δ (directed line graph) is bijective:



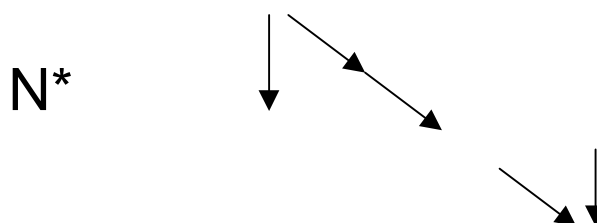
δ and its inverse are MS transductions.

Hence, the conjecture holds for N-free Hasse diagrams,
and for the corresponding posets.

(Because the bijective correspondence between a finite posets and its Hasse diagram is an MS transduction in both directions.)

This class of posets properly includes that of N-free posets, also called quasi-series-parallel because they can be generated by certain operations on posets, among which are series composition and disjoint union (equiv. parallel composition).

The corresponding Hasse diagrams are N^* -free, not necessarily N-free.



What would help for the proof of the general case.

We have:

G large tree-width $\Rightarrow G$ contains large grid minor

We need :

G large clique-width $\Rightarrow \tau(G)$ is a large grid

for some fixed MS-transduction τ

hence a knowledge of the structure of graphs of large clique-width.

6. Reductions between specific cases

Let C and D be two classes.

We write $C \Rightarrow D$, if one can prove the conjecture for all subsets of D assuming it is proved for all subsets of C .

Questions :

1. ? Finite graphs \Rightarrow countable graphs ?

(For countable graphs, the same proof as for finite graphs works for tree-width and MS_2 satisfiability).

2. ? Finite graphs \Rightarrow finite relational structures ?

3. Theorem (for classes of finite graphs):

Directed graphs

⇔ Directed acyclic graphs

⇔ Partial orders

⇔ Undirected graphs

⇔ Bipartite graphs

⇔ Chordal graphs

⇔ Vertex/edge labeled such classes.

Proof sketches :

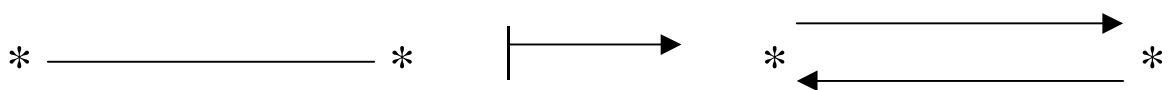
1. Class $C \Rightarrow k\text{-color-vertex}(C)$.

If L included in C has decidable MS-satisfiability problem,
 so has $L' = L$ with colors removed,
 hence L' has bounded clique-width,
 so has L .

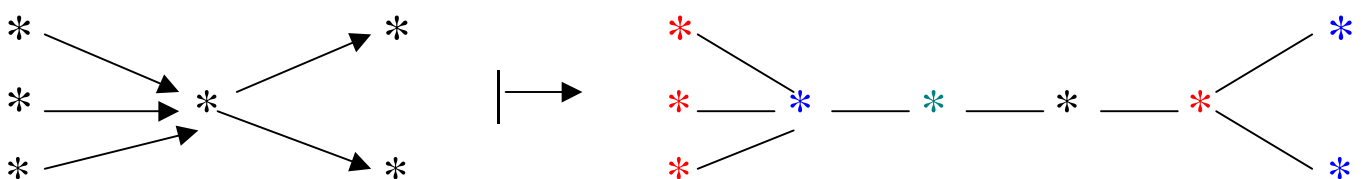
2. General tool to prove $C \Rightarrow D$:

encode G of D into H of C by MS-transduction, so that its inverse is also an MS-transduction.

Directed \Rightarrow Undirected :



4-vertex-colored bipartite undirected \Rightarrow Directed

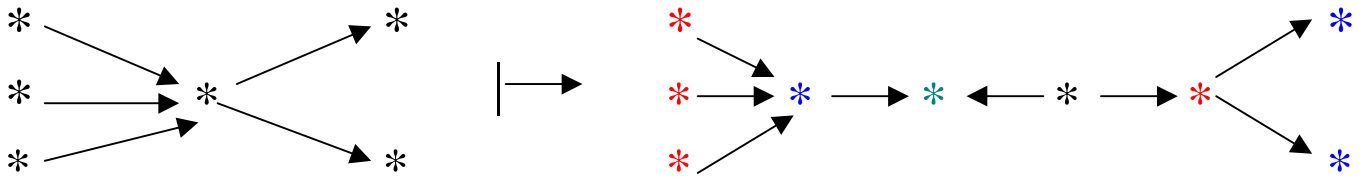


Decoding : * — * | → * → *, other edges contracted

4-vertex-colored bipartite Hasse diagrams

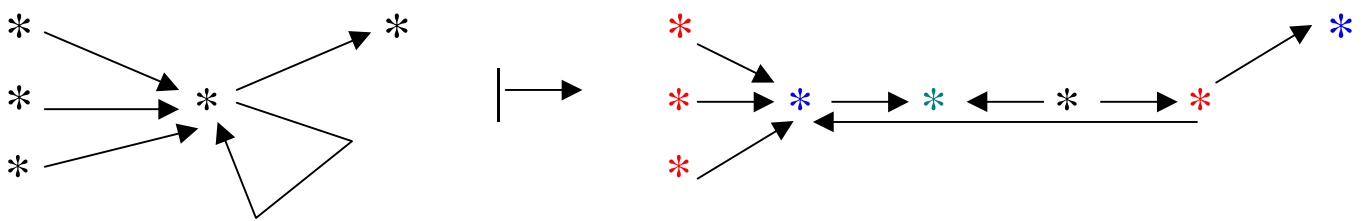
\Rightarrow

Directed

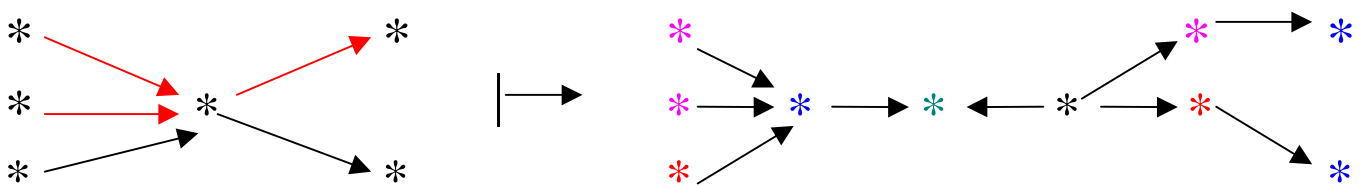


Decoding : $* \rightarrow * \mid \rightarrow * \rightarrow *$, other edges contracted

Case of a loop :

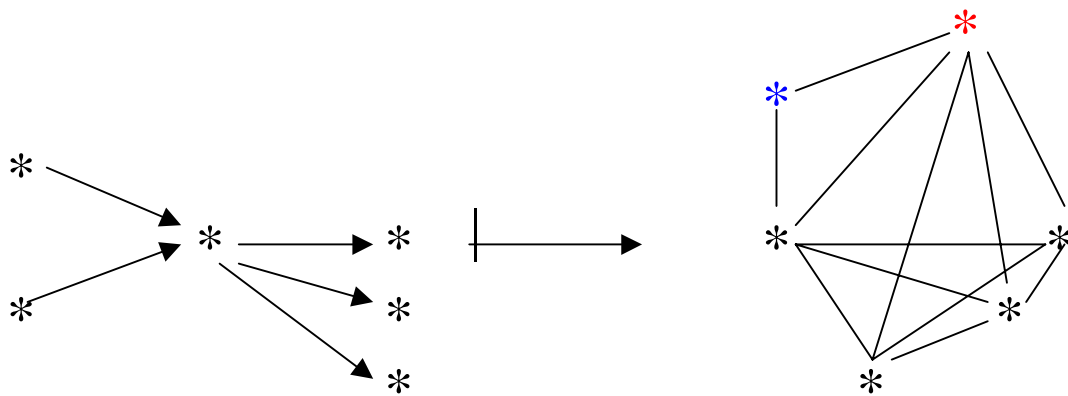


For edge labels, the technique is similar :



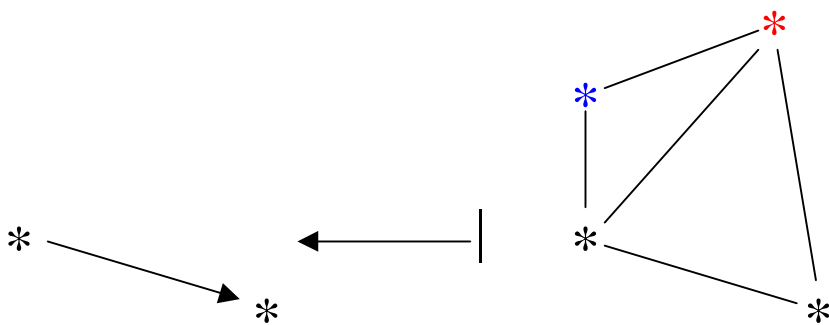
Decoding : $* \rightarrow * \mid \rightarrow * \rightarrow *$,
 $* \rightarrow * \mid \rightarrow * \rightarrow *$,
 other edges contracted

3-vertex-colored chordal graphs \Rightarrow Directed



Directed graph, n vertices,

K_n + vertices, edges



Directed edges are created from this pattern ; blue and red vertices are deleted

7. Description of MS compatible structure transformations ?

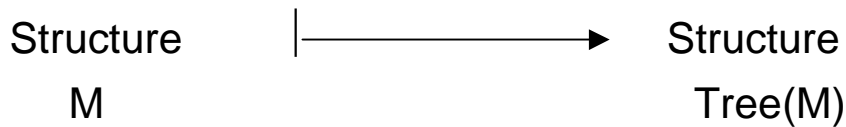
Which structure transformations are MS-compatible ?

MS compatible transformations of structures :

1. MS-transductions
2. The composition of two MS compatible transformations (clear from the definition)
3. Unfolding
4. The Shelah-Stupp-Muchnik tree expansion (proved by Walukiewicz).

Can every MS-compatible transformation be obtained from these cases ?

The Shelah-Stupp-Muchnik Construction



$$M = \langle V, \text{edg}(\dots) \rangle$$

$$\text{Tree}(M) = \langle \text{Seq}(V), \text{edg}^{\text{Seq}}(\dots), \text{son}(\dots), \text{clone}(\dots) \rangle$$

$$\text{Seq}(V) = \text{nonempty sequences over } V,$$

$$\text{edg}^{\text{Seq}} = \{ (wx, wy) \mid w \in V^*, (x, y) \in \text{edg} \}$$

$$\text{son} = \{ (w, wx) \mid w \in \text{Seq}(V), x \in V \},$$

$$\text{clone} = \{ wxx \mid w \in V^*, x \in V \},$$

(clone = copy of its father).

Theorem (Walukiewicz): The mapping $M \xrightarrow{\quad} \text{Tree}(M)$ is MS-compatible.

Observation: For a graph G , the tree $\text{Unf}(G)$ is *MS-definable inside* $\text{Tree}(G)$ (whence, definable from $\text{Tree}(G)$ by an MS-transduction).

Consequence: The mapping Unf is MS-compatible, as composition of two MS-compatible mappings.

8. Other open questions

8.1 For a Noetherian and confluent term rewriting system, the normal form mapping goes from finite terms to finite terms .

When is it an MS-transduction ?

When is it MS-compatible?

8.2 Operations on finite graphs such that the value mapping has good algorithmic behaviour but is not an MS transduction ?

(Might give counter-examples to Seese's conjecture.)