

The Monadic Second-Order Logic of Graphs XV: On a Conjecture by D. Seese

Bruno Courcelle
LaBRI, CNRS and Bordeaux 1 University,
351 cours de la Libération
F-33405 Talence
courcell@labri.fr

March 27, 2005

Abstract

A conjecture by D. Seese states that if a set of graphs has a decidable monadic second-order theory, then it is the image of a set of trees under a transformation of relational structures defined by monadic second-order formulas, or equivalently, has bounded clique-width. We prove that this conjecture is true if and only if it is true for the particular cases of bipartite undirected graphs, of directed graphs, of partial orders and of comparability graphs. We also prove that it is true for line graphs, for interval graphs and for partial orders of dimension 2. Our treatment of certain countably infinite graphs uses a representation of countable linear orders by binary trees that can be constructed by monadic second-order formulas. By using a counting argument, we show the intrinsic limits of the methods used here to handle this conjecture.

1 Introduction

A conjecture by D. Seese [31] states that if a set of graphs has a *decidable monadic second-order theory*, then it is the image of a set of trees under a *monadic second-order transduction*, in other words under a transformation of relational structures defined by monadic second-order formulas. We will say that such a set is *tree-definable*, where, implicitly, the relevant language is monadic second-order logic. By results of Courcelle, Engelfriet and van Oostrom ([13], [20]), this is equivalent to saying that it has *bounded clique-width*. This means that the graphs can be constructed from isolated vertices taken as basic graphs by means of certain graph operations : disjoint union, addition of edges based on

labellings of the vertices where labels belong to fixed finite sets C , replacement of labels by other labels in C . This applies to countable graphs as well as to finite ones because one can define them as values of infinite terms written with these operations (Courcelle [12]). The *clique-width* of a graph is the minimum cardinality of a set of labels C making it possible to construct this graph. The syntax tree of an algebraic expression that uses these operations and denotes a graph is a tree-structuring of this graph, and the mapping from this tree to the corresponding graph is a monadic second-order transduction. From the graph theoretical point of view, the graphs of a tree-definable set can be very far from trees : the set of cliques (i.e., of complete graphs) is tree-definable, and even "path-definable" because the corresponding syntax trees are paths. However, cliques have such a "regular" structure that they can be defined by algebraic terms. The graphs in a tree-definable set also have a certain form of "regularity", not always easily visible.

Seese's Conjecture is still open, however, *relativizations* of it to particular graph classes have been established. We say that a class \mathcal{C} of graphs or of relational structures *satisfies Seese's Conjecture*, which we denote by $SC(\mathcal{C})$, if every subset L of \mathcal{C} having a decidable monadic second-order theory is tree-definable. In particular, Seese has proved in [31] that $SC(\text{Planar})$ holds, where Planar denotes the class of planar graphs. We call a statement of the form $SC(\mathcal{C})$ the relativization to \mathcal{C} of Seese's Conjecture. Weakenings have also been established. A set of graphs for which monadic second-order logic *with edge set quantifications* is decidable, which is a stronger requirement than just assuming the decidability of monadic second-order logic, has bounded tree-width (Seese [31]). This latter result is actually a seminal one since all proved relativizations of the Conjecture reduce to it, as we now explain informally. (All the necessary definitions are given in Section 2 ; we abbreviate "monadic second-order" into MS and "monadic second-order with edge set quantifications" by MS₂).

We sketch its proof. Assume that a set of graphs L has a decidable MS₂ *satisfiability problem*, which means that one can decide whether an MS₂ formula is satisfied by some graph in L (this is an equivalent form of the hypothesis). So has the set M of all their *minors* (because the minors of a graph are definable in this graph by MS₂ formulas). Hence M does not contain square grids of all sizes (because otherwise, one could represent on large grids terminating Turing Machine computations, and the Halting Problem would be decidable). Hence the graphs in L have bounded tree-width by a result of Robertson and Seymour [30] (also proved in Diestel et al. [19] and in the book by Diestel [18]). Since every set of graphs of bounded tree-width has bounded clique-width and is tree-definable, we get the result.

It entails the validity of the Conjecture for planar graphs, for graphs of degree $\leq d$ for any fixed d , for graphs without any fixed graph as a minor, because for each such class, MS₂ formulas can be effectively translated into equivalent MS formulas (Courcelle [5]), hence the MS₂ satisfiability problem is decidable if the MS satisfiability problem is. This equivalence of MS₂ and MS formulas has been extended by Courcelle in [11] to the class \mathcal{U}_k of *uniformly k -sparse* graphs, defined as the class of finite or countable graphs, every finite subgraph

of which has a number of edges bounded by k times the number of vertices. To summarize, every subset of \mathcal{U}_k having a decidable monadic second-order theory has bounded tree-width.

The validity of $SC(\mathcal{C})$ has also been established for classes \mathcal{C} which are not uniformly k -sparse for any k . The method, which will also be used in the present article, is the following one : a bijection is defined between a class \mathcal{C} and a class $\mathcal{D} \subseteq \mathcal{U}_k$ for some k , and the mapping that encodes \mathcal{C} into \mathcal{D} and its inverse are definable by MS formulas. (To take a very simple example, the set of cliques with a distinguished vertex is in bijection in this way with the set of stars, i.e., trees of diameter at most 2, and stars are uniformly 1-sparse). This bijection makes it possible to deduce $SC(\mathcal{C})$ from $SC(\mathcal{U}_k)$. This coding technique is described in Section 3. It has been used by Courcelle to establish relativizations of the Conjecture to chordal graphs where every vertex belongs to a bounded number of maximal cliques ([7]) and to certain "convex" bipartite graphs ([10]). In these two cases and in those we consider in Sections 4,5 and 7, we obtain sets of graphs of unbounded tree-width having a decidable monadic second-order theory. Hence, we go out of the classes \mathcal{U}_k . However, this encoding technique has intrinsic limits that we discuss in Section 8.

A major progress towards the proof of Seese's Conjecture has been done by Courcelle and Oum [17]. They establish the weakening of the Conjecture where the hypothesis "has a decidable MS satisfiability problem" is replaced by the stronger hypothesis "has a decidable C_2MS satisfiability problem" where C_2MS refers to monadic second-order formulas that may use the set predicate $Even(X)$ expressing that a set X has even cardinality. This predicate is useful for writing a logical formula expressing the *vertex-minor inclusion relation between graphs*, a notion that parallels minor inclusion. This expression uses computations in the finite field $GF(2)$ that can be simulated with $Even$. The proof scheme of this result follows that for sets of graphs having a decidable MS_2 satisfiability problem. It is based on a result on vertex-minors and bipartite undirected graphs of bounded clique-width analogous to the result by Robertson and Seymour of [30].

In the present article, we establish the equivalences between relativizations of Seese's Conjecture to directed graphs, to bipartite undirected graphs, to partial orders and to comparability graphs. Some of these equivalences are used in [17]. Furthermore, we establish the validity of $SC(\mathcal{C})$ for the classes \mathcal{C} of line graphs, of directed line graphs, of finite partial orders of dimension 2, of finite interval graphs and also for a few related classes. For all these proofs we use codings by MS formulas of the considered graphs by uniformly k -sparse graphs. These codings are interesting on their own because they show how certain graph definitions can be handled by MS formulas. For certain countable graphs, we prove a weak version of the Conjecture where the hypothesis is the decidability of the satisfiability problem for *monadic second-order order-invariant* formulas, i.e., for monadic second-order formulas using an auxiliary linear order, but written in such a way that their validity does not depend on the chosen linear order.

This article is organized as follows. Section 2 consists of definitions and basic results about monadic second-order logic, monadic second-order transductions, clique-width and Seese's Conjecture. In Section 3 we prove the equivalences of certain relativizations of Seese's Conjecture. In Section 4, we prove the Conjecture for line graphs and for directed line graphs. In Section 5 we consider comparability graphs. We prove that the unique transitive orientation of a prime comparability graph is definable by an MS formula. This yields a proof of the Conjecture for finite partial orders of dimension 2. In order to extend it to countable partial orders of dimension 2, we prove in Section 6 that the classical representation of a countable linear order by the in-order on the nodes of a binary tree is definable by first-order formulas in terms of an arbitrary linear ordering of order type ω . In Section 7 we apply these results to interval graphs. Section 8 shows the intrinsic limits of the method of MS definable codings into uniformly k -sparse graphs. Section 9 is a conclusion and shows a table summarizing the main results.

2 Definitions and background results

Definitions and notation concerning monadic second-order formulas and monadic second-order transformations of relational structures are, up to minor details, those of the articles by the author on "The monadic second-order logic of graphs" and of the book chapter [9]. Subsection 2.3 reviews clique-width, and Subsection 2.4 presents Seese's Conjecture.

2.1 Structures and Monadic Second-order logic

A *relational signature* is a finite set $R = \{A, B, C, \dots\}$ of relation symbols, each of them given with an *arity* $\rho(A)$ in \mathbf{N}_+ . We denote by $STR(R)$ the set of finite or countable R -structures $S = \langle D_S, (A_S)_{A \in R} \rangle$ where $A_S \subseteq D_S^{\rho(A)}$ if $A \in R$. A *binary structure* is a structure using relations of arity at most 2. *Monadic Second-order logic* (*MS logic* for short) is first-order logic enriched with variables denoting subsets of the domains of the considered structures, and new atomic formulas of the form $x \in X$ expressing the membership of x in a set X . We will denote by $MS(R, W)$ the set of MS formulas written with the set R of relation symbols and having their free variables in a set W consisting of individual as well as set variables.

The *MS satisfiability problem for a class \mathcal{C}* of structures consists in deciding, for a given closed MS formula whether it is satisfied by *some* structure in \mathcal{C} . Since MS logic is closed under negation this problem is equivalent to that of *deciding the monadic theory of \mathcal{C}* , i.e., of deciding whether a given formula is true in *all* structures of \mathcal{C} . However, proofs are easier to write in terms of satisfiability than in terms of theories. The objective proposed by Seese in [31] consists in understanding the structure of the sets of graphs having a decidable MS satisfiability problem.

A simple graph G will be defined as an $\{edg\}$ -structure $G = \langle V_G, edg_G \rangle$ where V_G is the set of vertices of G and $edg_G \subseteq V_G \times V_G$ is a binary relation representing the edges. For undirected graphs, the relation edg_G is symmetric.

This definition is inappropriate for graphs with multiple edges. For representing them, but also for representing more properties of simple graphs, we will use their *incidence structures*. For a graph G we let $Inc(G) = \langle V_G \cup E_G, inc_G \rangle$ where E_G is the set of edges, inc_G is the ternary incidence relation such that $(e, x, y) \in inc_G$ if and only if e is an edge linking x to y (or linking x and y , if the graph is undirected).

A set of simple graphs (resp. of finite simple graphs) is *MS-definable* if it is (up to isomorphism) the set of finite or countable models (resp. of finite models) of an MS formula, written with the binary edg relation. A set L of graphs (resp. of finite graphs) is *MS₂-definable* if $\{Inc(G) \mid G \in L\}$ is (up to isomorphism) the set of finite or countable models (resp. of finite models) of an MS formula, written with the ternary inc relation. The subscript 2 refers to the two possible types of quantifications, over vertices and over edges, arising from this representation.

All graphs and structures will be finite or countable. What we call a *class* of graphs is actually a set of isomorphism classes of finite or countable graphs with a characteristic property. In the sense of set theory, we only deal with finite and countable sets.

2.2 Transductions of relational structures

We will use MS formulas to define certain transformations of graphs and more generally of relational structures. Let \mathcal{A} and \mathcal{B} be classes of relational structures. As in language theory, a binary relation $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{B}$ will be called a *transduction* : $\mathcal{A} \rightarrow \mathcal{B}$. It will be considered as a multivalued partial mapping associating with certain elements of \mathcal{A} one or more elements of \mathcal{B} .

An *MS transduction* is a transduction specified by MS formulas. It transforms a structure S , given with an n -tuple of subsets of its domain called the *parameters*, into a structure T , the domain of which is a subset of $D_S \times [k]$ (where $[k]$ denotes $\{1, \dots, k\}$). Furthermore, each such transduction, has an associated *backwards translation*, a mapping that transforms effectively every MS formula φ relative to T , possibly with free variables, into one, say $\varphi^\#$, relative to S having free variables corresponding to those of φ (k times as many actually) together with those denoting the parameters. This new formula expresses in S the property of T defined by φ . We now give some details. See also Courcelle [6,9].

We let R and Q be two relational signatures and W be a finite set of set variables, called *parameters*. A (Q, R) -*definition scheme* is a tuple of formulas of the form :

$$\begin{aligned} \Delta &= (\varphi, \psi_1, \dots, \psi_k, (\theta_w)_{w \in Q * k}) \\ \text{where } k > 0, Q * k &:= \{(A, \vec{j}) \mid A \in Q, \vec{j} \in [k]^{\rho(A)}\}, \\ \varphi \in MS(R, W), \psi_i &\in MS(R, W \cup \{x_1\}) \text{ for } i = 1, \dots, k, \end{aligned}$$

and $\theta_w \in MS(R, W \cup \{x_1, \dots, x_{\rho(A)}\})$, for $w = (A, \vec{j}) \in Q * k$.

These formulas are intended to define a Q -structure T from an R -structure S . Let $S \in STR(R)$, let γ be a W -assignment in S . A Q -structure T with domain $D_T \subseteq D_S \times [k]$ is defined in (S, γ) by Δ if :

- (i) $(S, \gamma) \models \varphi$
- (ii) $D_T = \{(d, i) \mid d \in D_S, i \in [k], (S, \gamma, d) \models \psi_i\}$
- (iii) for each A in Q : $A_T = \{((d_1, i_1), \dots, (d_t, i_t)) \in D_T^t \mid (S, \gamma, d_1, \dots, d_t) \models \theta_{(A, \vec{j})}\}$, where $\vec{j} = (i_1, \dots, i_t)$ and $t = \rho(A)$.

By $(S, \gamma, d_1, \dots, d_t) \models \theta_{(A, \vec{j})}$, we mean $(S, \gamma') \models \theta_{(A, \vec{j})}$, where γ' is the assignment extending γ , such that $\gamma'(x_i) = d_i$ for all $i = 1, \dots, t$; a similar convention is used for $(S, \gamma, d) \models \psi_i$. Since T is associated in a unique way with S, γ and Δ whenever it is defined, i.e., whenever $(S, \gamma) \models \varphi$, we can use the functional notation $def_\Delta(S, \gamma)$ for T .

The *transduction defined by Δ* is the relation $\mathcal{D}_\Delta := \{(S, T) \mid T = def_\Delta(S, \gamma) \text{ for some } W\text{-assignment } \gamma \text{ in } S\} \subseteq STR(R) \times STR(Q)$. A transduction $f \subseteq STR(R) \times STR(Q)$ is an *MS (definable) transduction* if it is equal to \mathcal{D}_Δ for some (Q, R) -definition scheme Δ (equal up to isomorphisms of structures). In the case where $W = \emptyset$, we say that \mathcal{D}_Δ is *definable without parameters* (note that it is functional). We will refer to the integer k by saying that Δ and \mathcal{D}_Δ are *k-copying*; if $k = 1$ we will say that they are *noncopying*. A noncopying definition scheme can be written more simply : $\Delta = (\varphi, \psi, (\theta_A)_{A \in Q})$.

For an example we recall from [14], Lemma 2.1, that if, on the structures S in $STR(R)$ we have an MS-definable equivalence relation \approx , then the transduction that maps S to its quotient structure S/\approx is an MS transduction. If $S = \langle D_S, (A_S)_{A \in R} \rangle$, then $S/\approx = \langle D_{S/\approx}, (A_{S/\approx})_{A \in R} \rangle$ where $A_{S/\approx}$ is the set of tuples $([a_1], \dots, [a_n])$ such that (a_1, \dots, a_n) belongs to A_S and $[a]$ is the equivalence class of a . This will be used in cases where the equivalence relation \approx is the reflexive, transitive and symmetric closure of an MS definable relation. It is MS definable because the transitive closure of an MS definable relation is MS definable. See Courcelle [9] and related articles for these basic facts.

The definitions concerning MS transductions of structures apply to graphs. However, since we have two representations of graphs by logical structures, we must be more precise. We say that an MS transduction \mathcal{D} on graphs is *(i,j)-definable*, where i and j belong to $\{1, 2\}$ if and only if the transduction of structures $\{(G_i, G'_j) \mid (G, G') \in \mathcal{D}\}$ is an MS transduction, where for every graph G , we let $G_1 = \langle V_G, edg_G \rangle$ and $G_2 = Inc(G)$. (If G has multiple edges, they are not distinguished in G_1 .) If we do not specify that an MS transduction is (i, j) -definable, this means that it is $(1,1)$ -definable.

For an example the mapping that associates with a graph G and a set X of edges, the graph obtained by contracting the edges of X is a $(2,2)$ -definable MS transduction with parameter X . The resulting graph is H with $E_H = E_G - X$,

$V_H = V_G/\approx$ and $inc_H = inc_G/\approx$ where \approx is the equivalence relation such that $x \approx y$ if and only if x and y are linked by an undirected path, all edges of which are in X .

For every class of structures $\mathcal{C} \subseteq STR(R)$ and every integer m , we denote by \mathcal{C}^{m-col} the corresponding class of *m-colored structures*. It is a subclass of $STR(R \cup P)$ where $P = \{p_1, \dots, p_m\}$ is a set of new unary relation symbols. The structures of \mathcal{C}^{m-col} are the expansions of those of \mathcal{C} with m unary relations p_1, \dots, p_m . We will say that an element x of a structure in \mathcal{C}^{m-col} has color i if $p_i(x)$ holds. An element may have one, no or several colors.

Consider a (Q, R) -definition scheme using m parameters. For every R -structure S , for every assignment γ of subsets of the domain of S to the parameters, we make (S, γ) into an m -colored structure S_γ defined as the expansion of S with m colors represented by the values of the unary predicates p_1, \dots, p_m . Hence, the definition scheme Δ can be converted into a parameter-less $(Q, R \cup P)$ -definition scheme Δ' that defines a transduction of $STR(R \cup P)$ into $STR(Q)$. Hence, if L is a subset of $STR(R)$ and $K \subseteq \mathcal{D}_\Delta(L)$, we also have $K = \mathcal{D}_{\Delta'}(L')$ where L' is a subset of $STR(R \cup P) = STR(R)^{m-col}$. One defines L' by taking the structures S_γ associated with the structures S and the assignments γ such that $def_\Delta(S, \gamma)$ is defined.

The fundamental property of MS transductions

The following proposition says that if $T = def_\Delta(S, \gamma)$, then the monadic second-order properties of T can be expressed as monadic second-order properties of (S, γ) . The usefulness of definable transductions is based on this proposition.

Let $\Delta = (\varphi, \psi_1, \dots, \psi_k, (\theta_w)_{w \in Q^*k})$ be a (Q, R) -definition scheme, written with a set of parameters W . Let V be a set of set variables disjoint from W . For every variable X in V , for every $i = 1, \dots, k$, we let X_i be a new variable. We let $V' := \{X_i \mid X \in V, i = 1, \dots, k\}$. Let S be a structure in $STR(R)$ with domain D . For every mapping $\eta : V' \rightarrow \mathcal{P}(D)$, we let $\eta^k : V \rightarrow \mathcal{P}(D \times [k])$ be defined by $\eta^k(X) = \eta(X_1) \times \{1\} \cup \dots \cup \eta(X_k) \times \{k\}$. With this notation we can state :

Proposition 2.1 : For every formula β in $MS(Q, V)$ one can construct a formula $\beta^\#$ in $MS(R, V' \cup W)$ such that, for every S in $STR(R)$, for every assignment $\gamma : W \rightarrow S$ for every assignment $\eta : V' \rightarrow S$ we have :

$$\begin{aligned} (S, \eta \cup \gamma) \models \beta^\# & \text{ if and only if :} \\ def_\Delta(S, \gamma) \text{ is defined, } \eta^k & \text{ is a } V\text{-assignment in } def_\Delta(S, \gamma), \\ \text{and } (def_\Delta(S, \gamma), \eta^k) & \models \beta. \end{aligned}$$

Note that, even if $T = def_\Delta(S, \gamma)$ is well-defined, the mapping η^k is not necessarily a V -assignment in T , because $\eta^k(X)$ may not be a subset of the domain of T which is a possibly proper subset of $D_S \times [k]$. We call $\beta^\#$ the

backwards translation of β relative to the transduction \mathcal{D}_Δ . The reader will find a more complete construction in [4], Proposition 2.5, p. 166. Here are some important consequences.

Proposition 2.2 : The inverse image of an MS-definable set of structures under an MS transduction is MS-definable. The composition of two MS transductions is an MS transduction.

Proof : Let \mathcal{D}_Δ be an MS transduction from $\mathcal{STR}(R)$ to $\mathcal{STR}(Q)$ and let L be a subset of $\mathcal{STR}(Q)$ defined as the set of models of a closed MS formula β . Its inverse image under \mathcal{D}_Δ is the set of structures S in $\mathcal{STR}(R)$ such that, for some γ , we have $def_\Delta(S, \gamma) \in L$. It is thus characterized by the formula $\exists W_1, \dots, W_n. \beta^\#$, where W_1, \dots, W_n is the list of parameters. See Courcelle [4] for a proof of the second statement. \square .

Proposition 2.3 : If a class of structures has a decidable MS satisfiability problem, then so has its image under an MS transduction.

Proof : Assume that a subset $L \subseteq \mathcal{STR}(R)$ has a decidable MS satisfiability problem, and that $L' = \mathcal{D}_\Delta(L)$. An MS formula β holds in some structure in L' if and only if the formula $\exists W_1, \dots, W_n. \beta^\#$ (where $\beta^\#$ is as in Proposition 2.1) holds in some structure in L . This is decidable since L has a decidable MS satisfiability problem. \square

From an MS transduction, one obtains thus a reduction between theories called an *interpretation*. This notion is used to relate decidability properties of different theories (Rabin [29]). In our constructions, the main notion is the transformation of graphs and of relational structures. The corresponding reduction of theories is just a by-product. Algorithmic applications of MS definable transformations of relational structures are also presented by Seese in [32].

2.3 Operations on graphs and on relational structures.

We review some operations on relational structures and on graphs. We do not recall the definition of tree-width, thoroughly studied by Bodlaender [1]. We also assume that the notion of a graph minor is known. The books by Diestel [18], Spinrad [33], and Brandstädt et al. [2] are good references for the basic notions of graph theory and the definitions and basic properties of many graph classes.

Disjoint union

Let S_1 and $S_2 \in \mathcal{STR}(R)$. We define $S_1 \oplus S_2$ as the structure S_3 built as the union of S_1 and of an isomorphic copy S'_2 of S_2 such that $D_{S_1} \cap D_{S'_2} = \emptyset$. We let $D_{S_3} = D_{S_1} \cup D_{S'_2}$ and $A_{S_3} = A_{S_1} \cup A_{S'_2}$ for each $A \in R$. (We are

interested by properties of structures up to isomorphism, hence we can freely replace structures by isomorphic copies.)

Quantifier-free definable operations

We denote by $QF(R, \{x_1, \dots, x_n\})$ the set of quantifier-free formulas over R with free-variables in $\{x_1, \dots, x_n\}$. A *quantifier-free definition scheme* is a noncopying (Q, R) -definition scheme without parameters $\Delta = (\varphi, \psi, (\theta_A)_{A \in Q})$ such that the formula φ is the Boolean constant *true*, the formulas ψ and θ_A for $A \in Q$ are quantifier-free formulas. The formula θ_A is said to *define* A . With such Δ is thus associated a total mapping $\mathcal{D}_\Delta: STR(R) \rightarrow STR(Q)$. We call \mathcal{D}_Δ a *quantifier-free definable operation* or *transduction*.

We apply these definitions to graphs. The disjoint union, denoted by \oplus is as for structures. It is the same for the two representations of graphs we have defined because $Inc(G \oplus H) = Inc(G) \oplus Inc(H)$.

The *edge-complement* for simple, loop-free undirected graphs can be defined as the quantifier-free definable operation such that $\theta_{edg}(x_1, x_2)$ is the formula $\neg(x_1 = x_2) \wedge \neg edg(x_1, x_2)$. We will denote by \bar{G} the edge-complement of a graph G . (Another notion of edge-complement can be defined for graphs with loops by deleting $\neg(x_1 = x_2)$ in the above formula). The *join* operation can be defined by $G_1 \otimes G_2 = (\bar{G}_1 \oplus \bar{G}_2)$, as a combination of edge-complements and disjoint-union. The *cographs* are the finite graphs generated from $\mathbf{1}$ (the graph with a single vertex and no edge) by disjoint union and edge-complement, or, equivalently by disjoint union and join. (See the book [2] for other characterizations of cographs.)

In order to generate larger families of graphs, it is convenient to use *labelled graphs*. We let $P = \{p, q, r, \dots\}$ be a finite set of *labels* handled as unary relation symbols. We let $R = \{edg\} \cup P$. A *P-graph* is a structure $G = \langle V_G, edg_G, (p_G)_{p \in P} \rangle$ where $\langle V_G, edg_G \rangle$ is a simple graph (directed or not) and the sets $(p_G)_{p \in P}$ form a partition of V_G (some sets p_G may be empty). A *P-graph* is thus a structure in $STR(R)$ satisfying some particular conditions. As operations on *P-graphs* we will use the disjoint union \oplus , and operations indexed by the labels in P . For distinct labels p, q , we let $add_{p,q}$ be the quantifier-free definable operation defined by letting $\theta_{edg}(x_1, x_2)$ be the formula :

$$edg(x_1, x_2) \vee (p(x_1) \wedge q(x_2))$$

This operation adds a new directed edge from any vertex labelled by p to any vertex labelled by q unless there exists already one (we deal with simple graphs). For adding undirected edges we use $add_{p,q} \circ add_{q,p}$ which can also be written as a single quantifier-free definable operation. We will also use the quantifier-free definable operation $ren_{p \rightarrow q}$ which changes everywhere into q each label p . For each p we let \mathbf{p} be the constant denoting the graph with one vertex labelled by p and no edge. For generating graphs with loops, it suffices to take also as basic graphs the graphs \mathbf{p}^{loop} with a loop incident to a single vertex labelled by p .

For each set P , we let \mathcal{C}_P denote $\{\mathbf{p}, \mathbf{p}^{loop}, \oplus, add_{p,q}, ren_{p \rightarrow q} \mid p, q \in P, p \neq q\}$. We let \mathcal{C}_n denote $\mathcal{C}_{\{1, \dots, n\}}$. Every P -graph defined by a term $t \in T(\mathcal{C}_P)$ (the set of finite terms written with the symbols of \mathcal{C}_P) is finite, simple and loop-free. Every finite, simple loop-free P -graph is the value of some $t \in T(\mathcal{C}_P)$ for some large enough set $P' \supseteq P$.

The *clique-width* of G denoted by $cwd(G)$ is defined as the smallest cardinality of P such that G is the value of some $t \in T(\mathcal{C}_P)$. See [15,16] about clique-width and [12] for the case of countable graphs. Trees have clique-width at most 3 and the cographs are the finite, simple, undirected graphs of clique-width at most 2. The clique-width of a graph is not modified by the addition or deletion of loops. We now recall some results from [9,13,20] :

Proposition 2.4 : A set of finite graphs has bounded clique-width (resp. bounded tree-width) if and only if it is a subset of the image of a set of finite trees under an MS transduction (resp. a (1,2)-definable MS transduction).

Note the differences between the two statements. In the second one, the output graphs are represented by their incidence structures, which means that the considered transduction constructs their edges as elements of the domain of the output structures. In the first one, the output graphs are represented by their sets of vertices and binary edge relations. The edges are not defined as elements of the domain, but as pairs of vertices. Tree-width and clique-width can also be defined for countable graphs. Proposition 2.4 extends to countable graphs and trees (Courcelle [3,12]). Since the composition of two MS transductions is an MS transduction, we get for finite as well as for countable graphs :

Corollary 2.5 : 1) The image of a set of graphs of bounded clique-width under an MS transduction has bounded clique-width.

2) The image of a set of graphs of bounded clique-width under a (1,2)-definable MS transduction has bounded tree-width.

3) The image of a set of graphs of bounded tree-width under a (2,2)-definable (resp. by a (2,1)-definable) MS transduction has bounded tree-width (resp. has bounded clique-width).

In the statements of Proposition 2.4 and their extensions to countable graphs, one can replace "is a subset of the image of a set of (finite) trees " by "is the image of a set of (finite) m -colored trees". This follows from the observation made in Subsection 2.2 on the transformation of parameters into colors of the elements of the input structures.

We will say that a set of structures is *tree-definable* if it is the image of a set of m -colored trees under an MS transduction. This definition makes no assumption on the effectiveness of the definition of the set of input trees. A tree-definable set of finite graphs need not be recursively enumerable.

Conjecture 1 [31] : *If a set of finite or countable graphs has a decidable MS satisfiability problem, then it is tree-definable, or equivalently, has bounded clique-width.*

Conjecture 2 : *If a set of finite or countable relational structures has a decidable MS satisfiability problem, then it is tree-definable.*

We say that a class \mathcal{C} of graphs or of relational structures *satisfies Seese's Conjecture* if all its subsets having a decidable MS satisfiability problem are tree-definable. We will write this shortly as property $SC(\mathcal{C})$ of the class \mathcal{C} . In particular, $SC(\text{Planar})$ holds as proved by Seese in [31]. Here, we make precise some results informally presented in the introduction.

Proposition 2.6 [7,31] : *If a set L of finite or countable graphs has a decidable MS_2 satisfiability problem, i.e., is such that $Inc(L)$ has a decidable MS satisfiability problem, then it has bounded tree-width.*

Its proof rests on the result of Robertson and Seymour [30] (also proved in Diestel et al. [19] and in the book [18]). The hypothesis is stronger than that of Conjecture 1, and the conclusion is also stronger since, for a set of graphs, bounded tree-width implies bounded clique-width but not vice-versa.

For every integer k , we let \mathcal{U}_k denote the class of simple, finite or countable graphs G that are *uniformly k -sparse* i.e., such that every finite subgraph H has a number of edges at most k times the number of vertices. (The following characterization will be useful : a graph is uniformly k -sparse iff it has an orientation such that each vertex has indegree at most k .) It is proved in [11] that every MS_2 definable subset of \mathcal{U}_k is MS definable, and more precisely, that every MS formula using edge and/or edge set quantifications (hence that is intended to be evaluated in a structure $Inc(G)$) can be translated into an MS formula (not using edge and edge set quantifications) that is equivalent to the given one in the graphs of \mathcal{U}_k . Of course, the translation depends on k . In technical terms, for every k , the identity on \mathcal{U}_k is a (1,2)-definable transduction. It follows then from Proposition 2.6 :

Corollary 2.7 : *If a set of simple, uniformly k -sparse graphs has a decidable MS satisfiability problem, then it has bounded tree-width, whence also bounded clique-width.*

As a consequence, we obtain that Seese's Conjecture holds for planar graphs, for graphs of degree at most d , because these classes are subclasses of \mathcal{U}_k for large enough k . We will be able to establish weak versions of the Conjecture relative to a stronger language than MS logic. We review some definitions from Courcelle [8]. An *ordered structure* is a pair (S, \leq) consisting of a structure S and an ω -order \leq of its domain, i.e., a linear order which is isomorphic to the ordinal ω if the domain is countable. A property of ordered structures is *order-invariant* if for any two ω -orders \leq and \leq' , the property holds for (S, \leq) if and

only if it holds for (S, \leq') . A property of R -structures S is *MS-OI-expressible* if it is of the form "there exists an ω -order \leq such that $P(S, \leq)$ holds" where P is an MS expressible order-invariant property of ordered structures. The *MS-OI-satisfiability problem* is decidable for a set L of R -structures if for every formula in $MS(R \cup \{\leq\}, \emptyset)$ expressing an order-invariant property, one can decide whether it is satisfied in some structure of L . (The notation $MS(\leq)$ is used in [8] for MS-OI restricted to finite structures). One cannot decide whether a given MS formula is order-invariant ([8]). However, we will use this notion for formulas which will be order-invariant by construction.

3 Equivalent relativizations of Seese's Conjecture

A class of graphs or of relational structures \mathcal{C} *satisfies Seese's Conjecture* if all its subsets having a decidable MS satisfiability problem are tree-definable, hence have bounded clique-width if \mathcal{C} is a class of graphs. This property is thus trivial for a class of graphs having bounded clique-width.

For two classes of graphs or of structures, we say that \mathcal{D} *reduces to \mathcal{C} with respect to Seese's Conjecture*, written $\mathcal{D} \leq_S \mathcal{C}$, if $SC(\mathcal{C}) \implies SC(\mathcal{D})$ that is, if we can prove $SC(\mathcal{D})$ by assuming $SC(\mathcal{C})$. We say that \mathcal{C} and \mathcal{D} are *equivalent with respect to Seese's Conjecture*, written $\mathcal{D} \equiv_S \mathcal{C}$ if $\mathcal{C} \leq_S \mathcal{D}$ and $\mathcal{D} \leq_S \mathcal{C}$.

If $\mathcal{C} \subseteq \mathcal{D} \subseteq STR(R)$ then, the reduction $\mathcal{C} \leq_S \mathcal{D}$ is trivial, and the reduction $\mathcal{D} \leq_S \mathcal{C}$ yields the equivalence of \mathcal{C} and \mathcal{D} .

We present some tools that will help to establish reductions.

3.1 Monadic second-order codings and colorings

Let \mathcal{A} and \mathcal{B} be two classes of structures. An *MS coding* of \mathcal{A} into \mathcal{B} is a pair (γ, δ) , where γ is the *coding transduction*, an MS transduction from \mathcal{A} into \mathcal{B} that is total on \mathcal{A} (each structure in \mathcal{A} has one or more images under γ) and δ is the *decoding transduction* also an MS transduction that is functional and is the inverse of γ . This implies that if H belongs to $\gamma(G)$ then $\delta(H) = G$, up to isomorphism. An MS coding (γ, δ) is *bijective* if γ is one-to-one.

Proposition 3.1 : If there exists an MS coding (γ, δ) of \mathcal{D} into \mathcal{C} , then $\mathcal{D} \leq_S \mathcal{C}$. If there exists an MS coding of \mathcal{C} into \mathcal{U}_k for some k , then $SC(\mathcal{C})$ holds.

Proof : Assume that we have an MS coding (γ, δ) of \mathcal{D} into \mathcal{C} and $SC(\mathcal{C})$ holds. Let L be a subset of \mathcal{D} having a decidable MS satisfiability problem, then so has $\gamma(L)$, by Proposition 2.3, hence $\gamma(L) \subseteq \theta(T)$ where T is a set of trees and θ is an MS transduction, and thus $L \subseteq \delta(\theta(T))$. Hence L is tree-definable

because the composition of two MS transductions is an MS transduction by Proposition 2.2. The second assertion follows since $SC(\mathcal{U}_k)$ holds. \square

MS codings into \mathcal{U}_k are used by Courcelle in [7] for proving the Conjecture for the chordal graphs such that every vertex belongs to a bounded number of maximal cliques, and in [10] for certain "convex" bipartite graphs.

For every class of structures \mathcal{C} , we denote by \mathcal{C}^{k-col} (cf. Subsection 2.2) the class of their expansions by k unary relations, also called their colorings with k colors.

Lemma 3.2 : For every k and every class $\mathcal{C} \subseteq STR(R)$, we have $\mathcal{C} \equiv_S \mathcal{C}^{k-col}$.

Proof: The quantifier-free definable transduction associating with S in \mathcal{C} its expansion S^* by k empty unary relations is an MS coding. Its inverse is the quantifier-free transduction fg_P which "forgets" the relations in P . It follows from Proposition 3.1 that $\mathcal{C} \leq_S \mathcal{C}^{k-col}$. (We do not have $\mathcal{C} \subseteq \mathcal{C}^{k-col}$ since \mathcal{C} and \mathcal{C}^{k-col} have distinct relational signatures. Hence the reduction $\mathcal{C} \leq_S \mathcal{C}^{k-col}$, although easy is not trivial.)

We now consider the other direction. We assume $SC(\mathcal{C})$ and we let $L \subseteq \mathcal{C}^{k-col}$ having a decidable MS satisfiability problem. We let $L' = fg_P(L)$. An MS formula β is satisfied in some structure S in L' if and only if it is satisfied in some structure in L (in this case the color predicates play no rôle). This is decidable by the hypothesis. Hence L' is tree-definable by the hypothesis on \mathcal{C} . But the transduction that associates with a structure in \mathcal{C} the set of all its k -colorings is an MS transduction, say γ , that uses k parameters for specifying the colors of the elements. Hence L is a subset of $\gamma(L')$ which is tree-definable since L' is and the class of MS transductions is closed under composition. \square

3.2 Equivalent relativizations

The following results concern countable graphs as well as finite graphs.

Proposition 3.3 : Undirected Graphs \leq_S Directed Graphs.

Proof : We have Undirected Graphs \subseteq Directed Graphs since an undirected edge is handled in the representing structures as a pair of opposite directed edges. The reduction is thus immediate. \square

Proposition 3.4 : Directed Graphs \leq_S Bipartite Undirected Graphs and Directed Graphs \leq_S Undirected Graphs.

Proof : We have trivially : Bipartite Undirected Graphs \leq_S Undirected Graphs. Let \mathcal{D} be the class (Bipartite Undirected Graphs) $^{4-col}$. By Lemma 3.2

: $\mathcal{D} \leq_S$ **Bipartite Undirected Graphs**. Hence, we need only prove that **Directed Graphs** $\leq_S \mathcal{D}$, i.e., we need only find an MS coding of directed graphs into certain bipartite undirected graphs, the vertices of which are colored with four colours.

The encoding is as follows : Let G be a directed graph. For every vertex v of this graph we define four vertices, $(v, 1), (v, 2), (v, 3), (v, 4)$ with respective colors *Target*, *Middle₁*, *Middle₂*, *Source*. We define undirected edges between $(v, 1)$ and $(v, 2)$, $(v, 2)$ and $(v, 3)$, $(v, 3)$ and $(v, 4)$. For every edge $v \rightarrow w$ in G , we create an edge linking $(v, 4)$ and $(w, 1)$. We may have $v = w$, in this case we get a cycle with four edges. We obtain thus a graph $\gamma(G)$ in \mathcal{D} , and the mapping γ is a 4-copying MS transduction.

Its image is characterized by the condition that every vertex colored by *Middle₁* (resp. *Middle₂*) has degree 2 and has one neighbour colored by *Target* (resp. *Source*), and another by *Middle₂* (resp. *Middle₁*).

The inverse of τ is obtained by the following operations : first every edge with ends x and y colored respectively by *Source* and *Target* is directed from x to y , and then all other edges are contracted. Edge contractions can be handled by taking a quotient graph. This mapping is thus an MS transduction. \square

Our aim is now to prove the following proposition:

Proposition 3.5 : **Partial Orders** \equiv_S **Directed Graphs**

The *Hasse diagram* $H(P)$ of a strict partial order $P = (V, <)$ is the simple directed acyclic graph (V, \rightarrow) such that $x \rightarrow y$ if and only if $x < y$ and there is no z such that $x < z < y$. A *Hasse diagram* is a simple directed acyclic graph such that if $x \rightarrow y$, there is no other path from x to y . (We leave out the conventions regarding drawings of Hasse diagrams ; they are explained in the book by Trotter [34] where (V, \rightarrow) is called the *cover graph* of P).

If P is finite, then P is the transitive closure of $H(P)$. The *height* of P (cf. [34]) is the maximal length n of a chain $x_1 < x_2 < \dots < x_n$. If P is infinite but has finite height it is the transitive closure of $H(P)$. (This condition is sufficient but not necessary).

The mapping from P to $H(P)$ is an MS transduction, and so is the mapping from a graph to its transitive closure. Finite partial orders and finite Hasse diagrams are in bijection by a pair of MS transductions. So are for each n , the finite and countable partial orders of height at most n and the corresponding Hasse diagrams.

Lemma 3.6 : **Directed Graphs** \leq_S **(Bipartite Hasse Diagrams)^{4-col}**

Proof : We use the same coding as in Proposition 3.4 with the following modifications : Edges in the graph $\gamma(G)$ are directed from $(v, 1)$ to $(v, 2)$, from $(v, 3)$ to $(v, 2)$, from $(v, 3)$ to $(v, 4)$, and finally from $(v, 4)$ to $(w, 1)$ whenever there is in G an edge from v to w .

The directed graph $\gamma(G)$ associated in this way with a directed graph G is acyclic because of the edges from $(v, 3)$ to $(v, 2)$ which forbid circuits. There is no edge from x to y such that there exists a directed path from x to y of length at least 2. Hence $\gamma(G)$ is a Hasse diagram with directed paths of length at most 3. (Their transitive closures are partial orders of height at most 4). The mapping γ is an MS-transduction, and so is its inverse, as in the proof of Proposition 3.4. \square

Proof of Proposition 3.5 : Since every partial order is a directed graph, we have the reduction $\text{Partial Orders} \leq_S \text{Directed Graphs}$.

Conversely we have $\mathcal{PO}' \leq_S \text{Partial Orders}$ where \mathcal{PO}' is the class of strict partial orders of height at most 4 that are the transitive closures of the Hasse diagrams of the special form constructed in Lemma 3.6. These Hasse diagrams form a subclass, call it \mathcal{HD} , of $(\text{Bipartite Hasse Diagrams})^{4-col}$. We have thus a bijective MS coding of \mathcal{HD} into \mathcal{PO}' . Hence $\mathcal{HD} \leq_S \mathcal{PO}'$. By Lemma 3.6, we have $\text{Directed Graphs} \leq_S \mathcal{HD}$. Hence, by the transitivity of reduction, we have $\text{Directed Graphs} \leq_S \text{Partial Orders}$. \square

Next we consider directed graphs with edge labels. A *k-edge-labelled directed graph* is one such that each edge has a label in the set $\{1, \dots, k\}$. We may have two parallel edges with different labels. Such a graph is represented by a relational structure with k binary edge relations, edg_1, \dots, edg_k .

Lemma 3.7 : For each k we have $k\text{-Edge-Labelled Directed Graphs} \leq_S (\text{Bipartite Hasse Diagrams})^{k+3-col}$

Proof : The proof is as in Lemma 3.6, with the following differences. For every vertex v of the graph G that we encode, we define $k + 3$ vertices, $(v, 1), \dots, (v, k + 3)$ with respective colors *Target*, *Middle₁*, *Middle₂*, *Source₁*, \dots , *Source_k*. We define directed edges from $(v, 1)$ to $(v, 2)$, from $(v, 3)$ to $(v, 2)$ and from $(v, 3)$ to (v, i) for each $i = 4, \dots, k + 3$. For every directed edge in G from v to w , colored by i , we create an edge linking $(v, 3 + i)$ to $(w, 1)$. The transduction γ is here $(k + 3)$ -copying. Its inverse is defined by the following operations : first every edge with ends x and y colored respectively by *Source_i* and *Target* is directed from x to y and gets color i and then all other edges are contracted. The verifications are easy. \square

We now consider particular undirected graphs. A *split graph* is a connected graph consisting of a clique (a complete loop-free undirected graph) and other vertices linked to the clique by one or more edges.

Proposition 3.8 : $\text{Split Graphs} \equiv_S \text{Bipartite Undirected Graphs}$

Proof : A split graph is a bipartite graph with partition (A, B) of its vertex set augmented with edges between any two vertices of A . From this observation, we get a bijective MS coding of the class of Split Graphs into (Bipartite Undirected

Graphs)^{2-col}. We get the reduction \leq_S using also Lemma 3.2. The other direction is similar. \square

Since split graphs have diameter at most 3, bounding the diameter of graphs does not yield classes satisfying Conjecture (as does on the opposite a bound on degree as a consequence of Corollary 2.7). *Chordal graphs* are usually defined as finite graphs. They can be characterized as the graphs having a tree-decomposition all "boxes" of which are cliques. (See Diestel [18] or Brandstädt et al.[2]). This characterization can be used as a definition for countable chordal graphs. In the following theorem, the class of chordal graphs can be replaced by any class of undirected graphs containing the split graphs. A *pomset* or *partially ordered multiset* is a partial order, each element of which has a label taken from an alphabet X . We let $\text{Pom}(X)$ denote the class of pomsets over a finite alphabet X .

Theorem 3.9 : 1) The following classes of graphs are pairwise equivalent with respect to Seese's Conjecture :

- (i) Undirected Graphs,
- (ii) Bipartite Undirected Graphs,
- (iii) Chordal Graphs,
- (iv) Split Graphs,
- (v) Directed Graphs,
- (vi) Directed Acyclic graphs.

2) They are equivalent to the corresponding classes of vertex and/or edge labelled graphs.

3) The class Undirected Graphs is equivalent to the class Partial Orders and to the class $\text{Pom}(X)$ for each finite alphabet X .

4) All the above equivalences hold for the corresponding classes of finite graphs and of finite partial orders.

Proof : By Propositions 3.3, 3.4 and inclusions, we have (i) \leq_S (v) \leq_S (ii) \leq_S (i).

Because of inclusions of classes we have (iv) \leq_S (iii) \leq_S (i), and by Proposition 3.8, we have (ii) \leq_S (iv).

By inclusions we have Bipartite Hasse Diagrams \leq_S (vi) \leq_S (v).

By Lemma 3.2, (Bipartite Hasse Diagrams)^{4-col} \leq_S Bipartite Hasse Diagrams and by Lemma 3.7, (v) \leq_S (Bipartite Hasse Diagrams)^{4-col}.

This achieves the proof of assertion 1)

2) Consider any k . We have proved in Lemma 3.7 that k -Edge-Labelled Directed Graphs \leq_S Directed Graphs.

By inclusion of classes we have :

k -Edge Labelled Undirected Graphs \leq_S k -Edge Labelled Directed Graphs,

and, by the same proof as in the first part of Lemma 3.2 :

Undirected Graphs \leq_S k -Edge Labelled Undirected Graphs.

Together with implications resulting from inclusions, we obtain that all k -edge labelled classes are equivalent. For vertex labellings for all these classes, we can use Lemma 3.2.

3) This follows from Proposition 3.5, and with Lemma 3.2, we get the equivalence with pomsets.

4) Clear because all constructions transform finite structures into finite structures. \square

A table in Section 9 collects these results.

3.3 Prime graphs

We review some definitions. The reader will find more details in the survey by Möhring and Radermacher [28], in the book by Spinrad [33] or in Courcelle [8]. Let G be a directed or undirected simple graph. A *nontrivial module* in G is a set M of at least two vertices such that $V_G - M$ is not empty and for every x, x' in M and every y in $V_G - M$, we have

$$edg_G(x, y) \iff edg_G(x', y) \text{ and } edg_G(y, x) \iff edg_G(y, x').$$

This means that every vertex outside of M is linked in the same way to all vertices of M . If the edges are labelled, then the edge labellings must be the same between y and all vertices of M . A graph is *prime* if it has at least three vertices and no nontrivial module. For a class \mathcal{C} we denote by $Prime(\mathcal{C})$ the class of prime graphs in \mathcal{C} .

Proposition 3.10 : For every class \mathcal{C} of directed or undirected graphs closed under taking induced subgraphs, we have : $\mathcal{C} \equiv_S Prime(\mathcal{C})$.

Proof: Since \mathcal{C} is closed under taking induced subgraphs, we have $Prime(\mathcal{C}) \subseteq \mathcal{C}$, hence $Prime(\mathcal{C}) \leq_S \mathcal{C}$.

Let us now assume that $SC(Prime(\mathcal{C}))$ holds. For every set of graphs $L \subseteq \mathcal{C}$ the set $PSub(L)$ of prime induced subgraphs of the graphs in L is obtained from L by an MS transduction, and is a subset of $Prime(\mathcal{C})$ since \mathcal{C} is closed under taking induced subgraphs. Hence, if L has a decidable MS-satisfiability problem, so has $PSub(L)$. Since $SC(Prime(\mathcal{C}))$ holds, the graphs in $PSub(L)$ have clique-width at most k for some k . The finite induced subgraphs of the graphs in L have thus clique-width at most k (because the clique-width of a finite graph is the maximum clique-width of its prime induced subgraphs). Hence, the countable graphs in L have clique-width bounded by a constant depending on k by the compactness result of Courcelle [12], because their finite induced subgraphs have clique-width at most k . \square

4 Line graphs

In the previous section, we have established a number of equivalences between relativizations of Seese's Conjecture to classes of graphs. We now prove the validity of this conjecture for directed line graphs and for line graphs. The techniques for these two apparently related classes are actually different.

4.1 Directed line graphs

We consider in this subsection directed graphs, possibly with loops and multiple edges, but without isolated vertices. A graph G is handled through its incidence structure $Inc(G) = \langle V_G \cup E_G, inc_G \rangle$ where $inc_G \subseteq E_G \times V_G \times V_G$, as defined in Subsection 2.1. The *directed line graph of G* , denoted by $DL(G)$ is the simple directed graph with set of vertices E_G and edges $e \rightarrow f$ whenever e and f are edges of G such that the target of e is the source of f . We say that H is a *directed line graph* if it is $DL(G)$ for some graph G . We denote by \mathcal{DLG} the class of directed line graphs, and by \mathcal{D} be the class of directed graphs having at most one vertex of indegree 0 and at most one vertex of outdegree 0.

Proposition 4.1 : 1) The mapping DL is a (2,1)-definable MS transduction of directed graphs onto the class \mathcal{DLG} .

2) It is a bijection of \mathcal{D} onto \mathcal{DLG} and its inverse $DL_0^{-1} : \mathcal{DLG} \rightarrow \mathcal{D}$ is a (1,2)-definable MS transduction.

3) A simple directed graph is a directed line graph if and only if whenever we have vertices u, x, y, z with $u \leftarrow x \rightarrow y \leftarrow z$, we have also $z \rightarrow u$ (vertices u, x, y, z are not necessarily distinct).

Before doing the proof we note that the two graphs $\{x \leftarrow y \rightarrow u\}$ and $\{x \rightarrow y \leftarrow u\}$ have the same directed line graph, consisting of two isolated vertices but are not isomorphic. Hence the mapping DL is not a bijection of the class of all directed graphs onto \mathcal{DLG} .

Proof : 1) Clear from the definition.

2) For a graph G not in \mathcal{D} , let G' be obtained by fusing all vertices of indegree 0 into a single vertex, and all vertices of outdegree 0 also into a single vertex. Then $G' \in \mathcal{D}$ and $DL(G') = DL(G)$, hence DL maps \mathcal{D} onto \mathcal{DLG} . We now prove it is a bijection.

Let H be $DL(G)$, where G is in \mathcal{D} . The edges of G are the vertices of H . For every vertex e of H we let $(e, 1)$ and $(e, 2)$ denote the source and the target of e in G . We let W be the set of all such pairs $(e, 1)$ and $(e, 2)$. A vertex of G has several denotations in W . We let \approx be the equivalence relation on W generated by the set of pairs $((e, 2), (f, 1))$ for all edges $e \rightarrow f$ of H , together with the set of pairs $((e, 1), (f, 1))$ such that e and f have indegree

0 in H , and the set of pairs $((e, 2), (f, 2))$ such that e and f have outdegree 0. From the hypothesis that G is in \mathcal{D} , we see that \approx -equivalent elements denote the same vertex and that any two denotations for a same vertex are \approx -equivalent. Hence, G can be reconstructed in a unique way from H and DL is one-to-one on \mathcal{D} . Furthermore, DL_0^{-1} is a (1,2)-definable MS transduction, since quotient structures by MS definable equivalence relations can be defined by MS transductions, as recalled in Subsection 2.2. This completes the proof of 2).

3) This is a known result by Heuchenne [24]. We prove it for completeness.

Let $H = DL(G)$. If $u \leftarrow x \rightarrow y \leftarrow z$ in H , then the target of x (as edge of G) is equal to the sources of u and y , the target of z is equal to the source of y and thus also to the source of u . Hence we must have also $z \rightarrow u$ in H .

Let us conversely assume that a simple directed graph H satisfies this condition. We let $W = V_H \times \{1, 2\}$ and we define on W a binary relation \approx by the following conditions :

$(v, i) \approx (w, j)$ iff

- either $i = j = 1$, and for every x , $x \rightarrow v$ if and only if $x \rightarrow w$,
- or $i = j = 2$, and for every x , $v \rightarrow x$ if and only if $w \rightarrow x$,
- or $i = 1, j = 2$ and $w \rightarrow v$,
- or $i = 2, j = 1$ and $v \rightarrow w$.

We claim that \approx is an equivalence relation. It is reflexive and symmetric, we check transitivity. Let $(v, i) \approx (w, j) \approx (z, k)$. We want to prove $(v, i) \approx (z, k)$.

The cases where $i = j = k$ are clear from the definitions.

We consider the case $(v, 1) \approx (w, 1) \approx (z, 2)$. We have $z \rightarrow w$, hence $z \rightarrow v$ since $(v, 1) \approx (w, 1)$, hence $(v, 1) \approx (z, 2)$.

We now consider the case $(v, 1) \approx (w, 2) \approx (z, 1)$. We have $w \rightarrow v$ and $w \rightarrow z$. Let x be any vertex in H such that $x \rightarrow v$. We have $x \rightarrow z$ by the hypothesis on H . Similarly, $x \rightarrow z$ implies $x \rightarrow v$. Hence $(v, 1) \approx (z, 1)$. All other cases are similar.

We let G' be the graph with set of vertices W/\approx , set of edges V_H and incidence relation $x : [(x, 1)] \rightarrow [(x, 2)]$, where $[w]$ is the equivalence class of w with respect to \approx . It is easy to check that G' is in \mathcal{D} , and that $DL(G')$ is isomorphic to H . \square

In technical terms, this proposition says that (DL_0^{-1}, DL) is an MS coding of \mathcal{DLG} into $Inc(\mathcal{D})$, the class of incidence structures of the graphs in \mathcal{D} .

Corollary 4.2 : If a set of directed graphs L has bounded tree-width then $DL(L)$ has bounded clique-width. The converse holds if $L \subseteq \mathcal{D}$.

Proof : Follows from Proposition 4.1 and Corollary 2.5. \square

The converse is not true without the hypothesis $L \subseteq \mathcal{D}$. For a counter-example consider the graph G_n with vertices $1, \dots, n, (i, j)$ for $1 \leq i < j \leq n$, edges $i \rightarrow i + 1$, $(i, j) \rightarrow i$, $(i, j) \rightarrow j$. This graph has K_n as a minor, hence tree-width at least $n - 1$. In the corresponding graph G'_n (cf. the proof

of Proposition 4.1), the vertices (i, j) are fused into a single vertex and G'_n has tree-width 2. We have $DL(G'_n) = DL(G_n)$. The graphs G_n have unbounded tree-width but their directed line graphs have bounded clique-width.

It follows from Corollary 4.2 that the graphs in \mathcal{DLG} have unbounded clique-width. Hence the following theorem is not trivially true.

Theorem 4.3 : The class \mathcal{DLG} of directed line graphs satisfies Seese's Conjecture.

Proof : Let L be a subset of \mathcal{DLG} for which the MS satisfiability problem is decidable. Then the MS_2 satisfiability problem is decidable for $DL_0^{-1}(L)$ (because DL_0^{-1} is (1,2)-definable). Hence, by Proposition 2.6, $DL_0^{-1}(L)$ has bounded tree-width, and L has bounded clique-width (by Corollary 2.5.3) since it is the image under the (2,1)-definable MS transduction DL of a set of bounded tree-width. \square

A directed graph is *N-free* if it has no induced subgraph isomorphic to $1 \longleftarrow 2 \longrightarrow 3 \longleftarrow 4$.

Corollary 4.4 : Seese's Conjecture holds for N-free Hasse diagrams and for the corresponding partial orders.

Proof : It suffices to prove that an N-free Hasse diagram belongs to \mathcal{DLG} , hence satisfies Condition 3 of Proposition 4.1.

Let G be an N-free Hasse diagram not in \mathcal{DLG} , if any exists. There are vertices u, x, y, z with $u \longleftarrow x \longrightarrow y \longleftarrow z$ with no edge $z \longrightarrow u$. Assume first they are all distinct. Can we have an edge from u to z ? No because then we have a path of length 3 from x to y , contradicting the definition of Hasse diagrams.

Can we have an edge between x and z , or between u and y ? No, again by the definition of a Hasse diagram. None of the edges of $u \longleftarrow x \longrightarrow y \longleftarrow z$ has an opposite edge, otherwise we get a circuit. Hence $u \longleftarrow x \longrightarrow y \longleftarrow z$ is an induced subgraph, hence G is not N-free.

Assume now u, x, y, z are not pairwise distinct. We cannot have $u = z$ otherwise we get a circuit. If $x = z$ or $u = y$ then we have $z \longrightarrow u$, contradicting the initial assumption. Hence, G is a directed line graph, and the result follows from Theorem 4.3. \square

For two partial orders P and Q , we let A be a set of maximal elements of P and B be a set of minimal elements of Q . We define the partial order $R = P \otimes_{A,B} Q$ as the union of P and Q (assumed to be disjoint ; if they are not one takes disjoint isomorphic copies) with order relation \leq_R defined as follows : $x \leq_R y$ if and only if $x \leq_P y$ or $x \leq_Q y$ or $x \in P, y \in Q, x \leq_P a$ for some $a \in A$ and $b \leq_Q y$ for some $b \in B$. If A is empty, we obtain the disjoint union (also called *parallel composition*). If A is the set all maximal elements of P and

B is the set of all minimal elements of Q we obtain the *series-composition*. The *quasi-series-parallel partial orders* are the *finite* partial orders generated by the operations $\otimes_{A,B}$ from the singletons.

Corollary 4.5 : Seese's Conjecture holds for the class of quasi-series-parallel partial orders.

Proof : This is a consequence of Corollary 4.4 because the quasi-series-parallel partial orders are the finite partial orders having an N-free Hasse diagrams, a result by Habib and Jegou [23] (also in Möhring [27]) \square

4.2 Line graphs

We consider here undirected graphs, possibly with loops and multiple edges but without isolated vertices. A graph G is represented by $Inc(G)$. We have $(e, x, y) \in inc_G$ if and only if $(e, y, x) \in inc_G$. The *line graph of G* , denoted by $L(G)$ is the simple, loop-free, undirected graph with set of vertices E_G and edges $e - f$ whenever e and f are incident edges of $G, e \neq f$. We say that H is a *line graph* if it is $L(G)$ for some graph G . We let \mathcal{LG} be the class of line graphs. Our objective is to establish $SC(\mathcal{LG})$. The following is clear from the definition.

Proposition 4.6 : The mapping L is a (2,1)-definable MS transduction.

Several graphs may have the same line graph. For example K_3 is the line graph of both K_3 and $K_{1,3}$. A theorem by Whitney [36] states that they are the only two non-isomorphic, connected, simple, loop-free graphs having isomorphic line graphs. However, this does not mean that the other connected, simple, loop-free graphs can be reconstructed from their line graphs. For a counter-example, consider G with edges $1 : a - b, 2 : a - c, 3 : b - c, 4 : d - b, 5 : d - c$, and H with edges $1 : a - b, 4 : a - c, 3 : b - c, 2 : d - b, 5 : d - c$. We have $L(G) = L(H)$ with vertices $\{1, 2, 3, 4, 5\}$. The graphs G and H are different but isomorphic by an isomorphism that does not preserve the "names" of edges.

Since \mathcal{LG} is closed under taking induced subgraphs, it is enough by Proposition 3.10 to establish $SC(Prime(\mathcal{LG}))$. Let \mathcal{S} be the class of connected, simple, loop-free graphs.

Proposition 4.7 : Every prime line graph H is $L(G)$ for some $G \in \mathcal{S}$. The mapping associating with every connected, simple, loop-free graph H the set $L^{-1}(H) \cap \mathcal{S}$ is a (1,2)-definable MS transduction λ .

Proof : If a graph $L(G)$ is prime, then it is connected and G is connected. The graph G is simple because, if it has at least three edges and two of them e, f

with same end vertices, then $\{e, f\}$ is a nontrivial module of $L(G)$, and $L(G)$ is not prime. If G has a loop e incident to a vertex v , then this loop can be made into a pending edge incident to v without changing the line graph. (A *pending edge* has one end of degree 1, called a *pending vertex*.)

For proving the second assertion, we first consider the special case where G is a tree T with at least two edges and L is its line graph. It is clear that every vertex of L belongs to one or two maximal cliques. We denote by K_L is the set of maximal cliques of L , by V_L^1 the set of vertices of L belonging to a single maximal clique. Hence, V_L^1 is the set of pending edges of T . The set of vertices of T is in bijection with $V_L^1 \cup K_L$: a pending vertex of T corresponds to a vertex in V_L^1 , namely the corresponding pending edge of T , and a vertex of degree >1 with incident edges e_1, \dots, e_p corresponds to the maximal clique $\{e_1, \dots, e_p\}$. We let $K'_L = K_L \cup \{\{v\} \mid v \in V_L^1\}$.

We now consider a connected graph G and $H = L(G)$. Let E be a set of vertices of H , i.e. of edges of G , such that $G - E$ is a spanning tree of G . Then $L(G - E) = H[V_H - E]$, the induced subgraph of H with set of vertices $V_H - E$. A vertex of H belonging to E is adjacent to all vertices of $A \cup B$ where A and B are disjoint sets in $K'_{H[V_H - E]}$. The sets A and B are disjoint because G has no multiple edges.

We now show how these notions can be represented in MS logic. We first consider a tree T and $L = L(T)$. We choose a pending vertex of T as root. Its incident edge r is called the *root edge* and is colored by 0. The edges incident with it are colored by 1. The uncolored edges incident with those colored by 1 are colored by 2. The still uncolored edges incident with those colored by 2 are colored by 0, and then we repeat by using colors 1,2,0,1, etc... Hence, on every path in T starting from the root, the edges are colored successively by 0, 1, 2, 0, 1, 2, 0...

This defines in L a particular vertex r and a vertex coloring with colors 0,1,2. The graph L , r and its coloring satisfy the following properties :

- (1) r is colored by 0 and its neighbours are colored by 1 ;
- (2) the binary relation defined by : $edg_U(x, y) : \iff x$ and y are adjacent in L and for some i , x has color i and y has color $(i + 1) \bmod 3$, defines a directed tree U , the root of which is r ;
- (3) if $edg_U(x, y)$ and $edg_U(x, z)$ then y and z are adjacent in L , hence x and its sons in U form a clique ;
- (4) every maximal clique of L is of the form described in (3) ; it has one vertex colored by some i , (it will be called the *leader* of the clique), and all others are colored by $(i + 1) \bmod 3$.

We add a new vertex to U , we link it by an edge to its root r and we obtain a tree isomorphic to T up to edge directions (T is undirected). Hence, a given graph is the line graph of a tree having at least two vertices if and only if it has a vertex r and a coloring satisfying conditions (1)-(4). From such a coloring, one can construct U and T as explained above, and $L = L(T)$.

We now consider a graph H , for which we want a graph $G \in \mathcal{S}$ such that $L(G) = H$. One can construct an MS formula $\varphi(E, r, X_0, X_1, X_2)$ expressing in a graph H that E, X_0, X_1, X_2 is a partition of V_H , X_0, X_1, X_2 define a coloring

of $V_H - E$ with colors 0,1,2, this coloring and the vertex r satisfy conditions (1)-(4) and each vertex of E is adjacent to all vertices of $A \cup B$ where A and B are disjoint sets in $K'_{H[V_H-E]}$. If $H = L(G)$ for some $G \in \mathcal{S}$ there exist E, r, X_0, X_1, X_2 satisfying φ . From such sets one can construct a graph G as follows :

$V_G = \{\bar{x} \mid x \in V_{H[V_H-E]}^1\} \cup \{\tilde{x} \mid x \text{ is the leader of a maximal clique}\}$, ("leader" is meant with respect to the coloring defined by X_0, X_1, X_2),

$E_G = V_H$ and the incidences are as follows :

if $e = r$, then $r : \bar{r} - \tilde{r}$,

if $e \in V_{H[V_H-E]}^1 - \{r\}$, then $e : \bar{e} - \tilde{y}$ where $edg_U(y, e)$ in the directed tree U constructed from r, X_0, X_1, X_2 ,

if $e \in V_H - (V_{H[V_H-E]}^1 \cup E)$, then $e : \tilde{e} - \tilde{y}$ where $edg_U(y, e)$ in the directed tree U constructed from r, X_0, X_1, X_2 ,

if $e \in E$, then $e : v(A) - v(B)$ where e is adjacent in H to all vertices of $A \cup B$ and A and B are disjoint sets in $K'_{H[V_H-E]}$; we denote by $v(A)$ the vertex \bar{x} (of G) if $A = \{x\}$, the vertex \tilde{x} if A is a maximal clique with leader x .

It is easy to check that $L(G) = H$. Hence G is constructed from H by a (1,2)-definable MS transduction λ with parameters $E, \{r\}, X_0, X_1, X_2$. It is (1,2)-definable because the vertices of H are used to specify the edges of G as individual objects. \square

Remarks : In technical terms, Propositions 4.6 and 4.7 say that (λ, L) is an MS coding of the class $L(\mathcal{S})$ into $Inc(\mathcal{S})$ the class of incidence structures of the graphs in \mathcal{S} .

The mapping L^{-1} is not a (1,2)-definable MS transduction on \mathcal{LG} . Assume it is, call it τ . Let M be the set of graphs S consisting of the vertices $0, 1, \dots, n$ and two parallel edges between 0 and each $i > 0$. It is definable by an MS_2 formula. We have $L(S) = K_m$ iff $K_m \in \tau^{-1}(S)$. Hence the set of cliques of the form $L(S)$ for some S in M is MS definable. But $K_m \in L(M)$ if and only if m is even. We get a contradiction because the set of even cliques is not MS definable (Courcelle [7],[9]). This proves the claim.

Corollary 4.8 : If M is a set of graphs of bounded tree-width then $L(M)$ has bounded clique-width. If $M \subseteq \mathcal{S}$ and $L(M)$ has bounded clique-width then M has bounded tree-width.

Proof : This is a consequence of Propositions 4.6 and 4.7 and of Corollary 2.5. \square

This result is also proved by Gurski and Wanke [22] who give precise bounds.

Theorem 4.9 : The class of line graphs satisfies Seese's Conjecture.

Proof : We prove $SC(L(\mathcal{S}))$ and this yields $SC(\mathcal{LG})$ by Proposition 3.10 since $Prime(\mathcal{LG}) \subseteq L(\mathcal{S})$ and \mathcal{LG} is closed under taking induced subgraphs. If a set $M \subseteq L(\mathcal{S})$ has a decidable MS satisfiability problem, then $L^{-1}(M) \cap \mathcal{S}$

has a decidable MS_2 satisfiability problem by Propositions 4.7 and 2.3, hence has bounded tree-width. Hence $M = L(L^{-1}(M) \cap \mathcal{S})$ has bounded clique-width by Proposition 4.6 and our basic results. \square

5 Comparability graphs

The *comparability graph* of a partial order $P = (V, \leq)$ is the simple, loop-free, undirected graph $\text{Comp}(G)$ with set of vertices V and where two vertices x and y are adjacent if and only if $x < y$ or $y < x$. A graph is a comparability graph if and only if it has a *transitive orientation*, i.e., an orientation such that if there are edges $x \rightarrow y$ and $y \rightarrow z$, then there is an edge $x \rightarrow z$.

Gallai has proved that a finite simple, loop-free, undirected graph H is a comparability graph if and only if it does not contain as an induced subgraph any of the graphs in an infinite set \mathcal{F} , described in the article by Trotter and Moore [35] and in the book by Trotter [34]. This characterization is valid for countable graphs : if a graph contains an induced subgraph of \mathcal{F} , then it is not a comparability graph, because the class of comparability graphs is closed under taking induced subgraphs. If, conversely, it does not contain any such subgraph, one can use Koenig's Lemma to extend to the considered countable graph some transitive orientations that exist for its finite subgraphs. (See Courcelle [11] for this type of use of Koenig's Lemma.)

The set \mathcal{F} consists of ten finite graphs and 8 infinite families of graphs $\mathcal{A}, \overline{\mathcal{B}}, \mathcal{C}, \mathcal{D}, \mathcal{E}, \overline{\mathcal{G}}, \overline{\mathcal{H}}, \mathcal{J}$ where $\overline{\mathcal{X}}$ is the set of edge-complements of the graphs in \mathcal{X} and :

$$\mathcal{A} = \{C_{2n+1} \mid n \geq 2\} \text{ where } C_n \text{ is the undirected cycle with } n \text{ vertices,}$$

$$\mathcal{B} = \{C_n \mid n \geq 6\},$$

$\mathcal{C} = \{(a \otimes P_{2n}^{b,c}) + \{b-d, c-e\} \mid n \geq 2\}$ where $P_m^{b,c}$ denotes the path with m vertices, $m-1$ edges and two ends named b and c ; the notation $H + \{x-y, z-t, \dots\}$ means that we add to a graph H the edges $x-y, z-t, \dots$ and their end vertices if they are not already in H ; the operations \otimes and \oplus are defined in Subsection 2.3,

$$\mathcal{D} = \{((a \oplus d) \otimes P_{2n}^{b,c}) + \{b-e, d-e, a-f, c-f\} \mid n \geq 2\},$$

$$\mathcal{E} = \{((a \otimes d) \otimes P_{2n+1}^{b,c}) + \{b-e, d-e, a-f, c-f\} \mid n \geq 1\},$$

$$\mathcal{G} = \{(a \otimes P_n^{b,c}) + \{b-d, c-e, a-f\} \mid n \geq 2\},$$

$$\mathcal{H} = \{((a \otimes d) \otimes P_n^{b,c}) + \{a-e, a-g, b-e, c-f, d-f, d-g\} \mid n \geq 2\},$$

$$\mathcal{J} = \{((a \oplus d) \otimes P_n^{b,c}) + \{a-e, a-g, b-e, c-f, d-f, d-g\} \mid n \geq 2\}.$$

Lemma 5.1 : The class of comparability graphs is MS-definable.

Proof : We first prove that the set \mathcal{F} is MS-definable (up to isomorphism). The ten finite graphs form an MS-definable set. Since, whenever a set is MS definable, so is its edge-complement, it suffices to observe that each set $\mathcal{A}, \overline{\mathcal{B}}, \mathcal{C}, \mathcal{D}, \mathcal{E}, \overline{\mathcal{G}}, \overline{\mathcal{H}}, \mathcal{J}$ is MS-definable. This is easy from the above descriptions.

We only give the proof for the set \mathcal{C} . The other proofs are similar. The formula expressing that a graph is in \mathcal{C} is the conjunction of the following conditions :

- (i) there exist a set of vertices X inducing a graph P_m for m even and at least 4, the ends of which are vertices b and c ,
- (ii) there are exactly three vertices not in X , say a, d, e ,
- (iii) in addition to the edges of the path induced by X , there are edges between b and d , between c and e , and between a and each vertex of X .

These conditions characterize the graphs in \mathcal{C} . They are expressible by an MS formula. Hence, by using similar descriptions for the other infinite sets, we obtain that \mathcal{F} can be defined by an MS formula α that is the finite disjunction of the MS formulas describing the various types of graphs. One can then construct an MS formula β expressing that a given graph G has no set of vertices Y such that the induced subgraph $G[Y]$ satisfies α . This formula defines the class of comparability graphs. \square

Gallai has also proved that if a finite comparability graph is prime, then it has only two transitive orientations. (See the book by Golumbic [21] or the article by Kelly [25]). This means that one can choose the orientation of one edge arbitrarily, and that this choice determines in a unique way the orientations of all other edges.

Consider for an example the graph $u - v - x - y - z$ (i.e., the path with 5 vertices u, v, x, y, z and the edges $u - v$, etc...). The orientation $u \rightarrow v$ forces $v \leftarrow x$ (otherwise the edge between u and x is missing) which forces $x \rightarrow y$, and then also $y \leftarrow z$ (each time with the same argument).

The proof given in [21] works for countable graphs as well as for finite graphs.

Proposition 5.2 : There exists an MS formula $\gamma(x, y, u, v)$ such that for every graph G and vertices x, y, u, v , $G \models \gamma(x, y, u, v)$ if and only if G is a prime comparability graph with edges $x - y$ and $u - v$, such that $x \neq u, y \neq u, v \neq x$ and, in the two transitive orientations of G , either $x \rightarrow y$ and $v \rightarrow u$, or $x \leftarrow y$ and $v \leftarrow u$.

Proof : We need a technical construction. Let G be a comparability graph with transitive orientation \vec{G} . Let x, u be two distinct vertices, let $X = \{a \in V_G \mid a \rightarrow x\}$ and $U = \{b \in V_G \mid u \rightarrow b\}$ where edge directions are relative to \vec{G} . Let x' and u' be two new vertices and $\vec{G}(x, u, X, U)$ be the directed graph consisting of \vec{G} and the following edges :

- (a) $x' \leftarrow u'$,
- (b) $a \rightarrow x'$ for every $a \in X \cup \{x\}$,
- (c) $u' \rightarrow b$ for every $b \in U \cup \{u\}$.

We let $G(x, u, X, U)$ be the corresponding undirected graph.

Claim 1: $\vec{G}(x, u, X, U)$ is transitive.

Proof : Let w, y, z be vertices such that $w \rightarrow y \rightarrow z$. We must prove that $w \rightarrow z$. We distinguish several cases.

Case 1 : w, y, z are all in G . Then, we have $w \longrightarrow z$ because \vec{G} is transitive and the edges $w \longrightarrow y$ and $y \longrightarrow z$ are in \vec{G} .

Case 2 : One of w, y, z is u' or x' . Since x' and u' are of outdegree and indegree 0 respectively the only possibilities are $z = x'$ and $w = u'$. We consider the first one and we distinguish subcases.

Subcase 1 : $x = y$

We have $w \longrightarrow x$ in \vec{G} hence $w \in X$, and $w \longrightarrow x' = z$.

Subcase 2 : $x = w$

We have $w = x \longrightarrow x' = z$ by the definition of $\vec{G}(x, u, X, U)$.

Subcase 3 : x is neither y nor w .

We have $y \longrightarrow x' = z$ in $\vec{G}(x, u, X, U)$ hence $y \in X$, and $y \longrightarrow x$. Since $w \longrightarrow y$ is in \vec{G} , transitive, we have $w \longrightarrow x$ in \vec{G} . Hence $w \in X$ and thus $w \longrightarrow x' = z$.

The case $w = u'$ is fully similar.

Case 3 : Two of w, y, z are u' and x' .

Considering the indegree and the outdegree of u' and x' , the only possibility is $w = u'$ and $z = x'$. But we have $u' \longrightarrow x'$ by definition of $\vec{G}(x, u, X, U)$, hence $w \longrightarrow z$. This completes the proof of the claim. \square

Note that $G(x, u, X, U)$ can be defined by the above conditions (a)-(c) and by omitting edge directions from any two distinct vertices x, u and any two sets X and U . With this definition, we have :

Claim 2 : Let G be a prime comparability graph, $x - y$ and $u - v$ be two edges of G and $x \neq u, y \neq u, v \neq x$. We have $v \longrightarrow u$ in the unique transitive orientation of G such that $x \longrightarrow y$ if and only if there exist two sets X and U such that $y \notin X, v \notin U$ and $G(x, u, X, U)$ is a comparability graph.

Proof : "Only if". Let \vec{G} be a transitive orientation of G with $x \longrightarrow y$ and $v \longrightarrow u$. We let $X = \{a \in V_G \mid a \longrightarrow x\}$ and $U = \{b \in V_G \mid u \longrightarrow b\}$ and the conclusion follows from Claim 1.

"If" Let X and U be such that $y \notin X, v \notin U$ and $G(x, u, X, U)$ has a transitive orientation \vec{H} such that $x \longrightarrow y$. Its restriction to G , an induced subgraph of $G(x, u, X, U)$ is a transitive orientation, and it is the unique one for G such that $x \longrightarrow y$ since G is prime. Since we have in $G(x, u, X, U)$ a path or a cycle $y - x - x' - u' - u - v$, (we can have $y = v$) without edges $y - x', x - u', x' - u, u' - v$ the orientation $x \longrightarrow y$ forces $v \longrightarrow u$. Hence we have $v \longrightarrow u$ in the unique transitive orientation of G such that $x \longrightarrow y$. \square

By using the MS formula β expressing that a graph is a comparability graph, one can build an MS formula $\delta(x, u, X, U)$ which expresses for a graph G , for vertices x, u and for sets of vertices X and U that $x \neq u$ and that $G(x, u, X, U)$ is a comparability graph. It follows then from Claim 2 that the desired formula $\gamma(x, y, u, v)$ can be taken as the MS formula expressing the following :

(i) G is a prime comparability graph,

- (ii) $x \neq u \wedge y \neq u \wedge v \neq x \wedge \text{edg}(x, y) \wedge \text{edg}(u, v)$
- (iii) $G \models \exists X, U \{y \notin X \wedge v \notin U \wedge \delta(x, u, X, U)\}$.

That G is prime is easily expressible by an MS formula translating the definition. This completes the proof. \square

Hence, the "unique" transitive orientation of a prime comparability graph is MS definable, where "unique" is meant up to the arbitrary choice of orientation for one edge.

Corollary 5.3 : There exists an MS transduction that associates with a prime comparability graph its two transitive orientations.

Proof : The transduction uses two parameters $\{x\}$ and $\{y\}$ such that $x - y$ in the given graph G . The direction of an edge $u - v$ in the unique transitive orientation of G such that $x \rightarrow y$ is obtained as follows.

If $x \neq u, y \neq u, v \neq x$, it is directed $v \rightarrow u$ iff $\gamma(x, y, u, v)$ holds. If $v = x$ and y and u are not adjacent, it is directed $v \rightarrow u$. If $v = x$ and $y - u$, it is directed $u \rightarrow v$ if and only if $\gamma(y, x, u, x)$ holds. (This implies also the orientation $u \rightarrow y$). The other transitive orientation is obtained by exchanging x and y . \square

Corollary 5.4 :1) Comparability Graphs \equiv_S Undirected Graphs.

2) The equivalence also holds for the corresponding classes of finite graphs.

Proof : The reduction \leq_S follows from the inclusion of classes. For the other direction, we prove that **Partial Orders** \leq_S **Comparability Graphs**. It is enough by Proposition 3.10 to consider prime partial orders ("prime" is meant with respect to the directed graph of the corresponding strict partial order). The comparability graph of a prime partial order is a prime undirected graph, by [28] (Theorem 1.5.1 page 287). If L is a set of prime partial orders having a decidable MS satisfiability problem, so has the set $C(L)$ of their comparability graphs. This set is tree-definable by the hypothesis. By Corollary 5.3, and since the graphs in $C(L)$ are prime, one can obtain L as the image $C(L)$ by an MS transduction. Hence L is tree-definable. This proves the result for prime partial orders. The case of **Partial orders** follows then from Proposition 3.10. \square

The edge-complement of a comparability graph is called a *cocomparability graph*. We let **CiCC** be the class $(\text{Comparability Graphs}) \cap (\text{Cocomparability Graphs})$, and **FCiCC** be its restriction to finite graphs. A partial order P has *dimension* k if k is the minimum number of linear orders, the intersection of which is P .

The Hasse diagram of a finite linear order is a directed path, hence a directed graph of indegree 1. A finite partial order P of dimension at most k can be represented by the union U of k directed paths representing k finite linear orders of which it is the intersection. The edges of the different paths are labelled by

$1, \dots, k$ in order to be distinguished from one another. It is then easy to build an MS transduction transforming U into P . We obtain a representation of finite partial orders of dimension at most k by uniformly k -sparse directed graphs.

If we could construct such a representing graph U from P by an MS transduction, we would prove that for each k , the class of the finite partial orders of dimension at most k satisfies the Conjecture. We can only do this for $k = 2$.

Proposition 5.5 : The class FCiCC and the class of finite partial orders of dimension at most 2 satisfy Seeses's Conjecture.

Proof : We will use the fact proved by Golumbic in [21], Theorem 5.38 page 138, that a partial order has dimension at most 2 if and only if the edge-complement of its comparability graph is also a comparability graph, hence belongs to the class CiCC.

Since the classes of comparability and of cocomparability graphs are closed under taking induced subgraphs, it is enough to prove the Conjecture for the prime graphs in FCiCC. Let G be a prime graph in FCiCC. It is not complete and its edge-complement is not either. Let us choose two adjacent vertices x and y . One can define by an MS formula a transitive orientation H of G such that $x \longrightarrow y$.

Let u be adjacent in G to x and not to y . Since the edge-complement \overline{G} of G is also prime, one can define by an MS formula a transitive orientation K of \overline{G} such that $u \longrightarrow y$. The opposite one K^{-1} is also MS definable. Each of the binary relations $H \cup K$ and $H \cup K^{-1}$ is a strict linear order on V_G , and their intersection is a partial order P of dimension 2, the comparability graph of which is G (we use here the argument by Golumbic).

Because they are finite, we can represent each linear order by a directed path. Hence, these two linear orders can be represented by an edge-labelled directed graph U that is uniformly 2-sparse. Then U is definable from G by an MS transduction. We obtain the validity of $SC(Prime(FCiCC))$, whence of $SC(FCiCC)$ by Proposition 3.10.

For finite partially ordered sets of dimension at most 2, it is enough to consider those which are prime when considered as directed graphs. As recalled above, the comparability graph of a prime partial order is a prime undirected graph ([28], Theorem 1.5.1), and so is its edge-complement (because the notion of a module is invariant under taking edge-complements). Hence we can use the argument of the first assertion to complete the proof. \square

6 A monadic second-order coding of countable linear orders

The statement of Proposition 5.5 is limited to finite graphs and to finite partial orders because the proof uses the representation of linear orders by their Hasse

diagrams, which does not always work for infinite linear orders. In particular, the Hasse diagram of the ordered set of rational numbers is empty. We overcome this difficulty by representing countable linear orders by sets of nodes of binary ordered trees, and by showing that this representation is definable by an MS transduction using an auxiliary and arbitrary linear ordering of type ω of the given set.

By a *tree*, we mean here a binary tree defined as the simple directed edge-labelled graph $T = (N_T, lson_T, rson_T)$ where N_T is the finite or countable set of nodes, $lson_T$ and $rson_T$ are two binary functional relations defining for each node its *left son* and its *right son*. A node may have no son, two sons, or just a right son or a left son. The *root* is the unique node of indegree 0. Whether a structure T is a tree can be expressed in MS logic.

We will write $x \rightarrow_l y$ if y is the left son of x , $x \rightarrow_r y$ if y is the right son of x , and $x \rightarrow y$ if y is the left or the right son of x . A linear order, the *in-order*, on N_T can be defined as follows :

$x \sqsubseteq_T y$ if and only if :

$x = y$ or $x \rightarrow_r z \rightarrow^* y$ or $y \rightarrow_l z \rightarrow^* x$ for some z , or $u \rightarrow_l z \rightarrow^* x$ and $u \rightarrow_r z' \rightarrow^* y$ for some u, z, z' .

We let $\Omega(T)$ be the linearly ordered set (N_T, \sqsubseteq_T) . It clear that Ω is an MS transduction. Our objective is to construct T from $\Omega(T)$ by an MS transduction. An ω -order is a linear order which is finite or isomorphic to the ordinal ω .

Proposition 6.1 : There exists an MS transduction γ that defines from a structure (N, \sqsubseteq, \leq) such that \sqsubseteq is a linear order and \leq is an ω -order on N , a tree T such that $\Omega(T) = (N, \sqsubseteq)$.

Proof : Let (N, \sqsubseteq, \leq) be given as in the statement. We leave out the case where N is finite, for which the construction is immediate, without using \leq .

We take r , the \leq -least element of N as root of T . For every x in N , $x \neq r$, we let :

$m(x)$ be the \sqsubseteq -largest element y such that $y \sqsubseteq x$ and $y < x$,

$M(x)$ be the \sqsubseteq -smallest element y such that $x \sqsubseteq y$ and $y < x$.

We have $m(x) \sqsubseteq x$ and $x \sqsubseteq M(x)$ whenever $m(x)$ or $M(x)$ is defined.

a) If $M(x)$ is undefined, we let $(m(x), x)$ belong to $rson$.

b) If $m(x)$ is undefined, we let $(M(x), x)$ belong to $lson$.

If $M(x)$ and $m(x)$ are both defined, we have $m(x) \sqsubseteq x \sqsubseteq M(x)$ and

c) if $m(x) < M(x)$ we let $(M(x), x)$ belong to $lson$ and finally

d) if $M(x) < m(x)$ we let $(m(x), x)$ belong to $rson$.

This can be expressed by first-order formulas λ and ρ defining $lson$ and $rson$.

For an example consider the sequence :

$$9 \sqsubseteq 8 \sqsubseteq 1 \sqsubseteq 6 \sqsubseteq 0 \sqsubseteq 7 \sqsubseteq 2 \sqsubseteq 5 \sqsubseteq 3 \sqsubseteq 4,$$

where the order \leq is the natural one : $0 < 1 < 2 \dots$. The associated tree is

$$\{[(9)8]1[6]\}0\{[7]2[(5)3(4)]\}.$$

The sequence of nodes is the original \sqsubset -sequence. Parentheses and brackets define the binary tree structure. Node 8 has 9 as left son and has no right son.

We make some observations to help the understanding and the forthcoming proof. For every pair (y, x) in $lson$ or in $rson$, y is before x in the enumeration defined by $<$. This guarantees the absence of circuits. The construction consists in putting in a tree the elements of N in the order defined by $<$. There are four ways to add a node x :

as right son of the rightmost node $m(x)$, by clause a) above ; in the above example this is the case of nodes 2 ($m(2) = 0$), 3 ($m(3) = 2$), 4 ($m(4) = 3$) ;

as left son of the leftmost node $M(x)$, by clause b) above ; in the above example this is the case of nodes 1 ($M(x) = 0$), 8 ($M(8) = 1$), 9 ($M(9) = 8$) ;

in cases c) and d) the node x must be placed between $m(x)$ and $M(x)$ which are the elements of N before x with respect to $<$, which are closest to x with respect to \sqsubseteq (they already exist in the tree). Depending on whether $m(x) < M(x)$ or $M(x) < m(x)$, x is defined as left son of $M(x)$ (case c)), or right son of $m(x)$ (case d)) ; in the example, case c) applies to nodes 5 ($m(5) = 2, M(5) = 3$) and 7 ($m(7) = 0, M(7) = 2$) and case d) applies to node 6 ($m(6) = 1, M(6) = 0$).

It remains to prove that $(N, lson, rson)$ is actually a tree T such that $\Omega(T) = (N, \sqsubseteq)$. For every x in N , we let $T(x)$ be the restriction of the structure $(N, lson, rson)$ to the set $A(x) = \{y \mid y \leq x\}$.

Claim : $T(x)$ is a finite tree and $\Omega(T(x)) = (A(x), \sqsubseteq)$.

Proof of claim. By induction on $<$. The least element of N is r . The tree $T(r)$ is reduced to r and the assertion holds.

Consider $x \neq r$ and x' , its predecessor with respect to $<$. Hence $T(x')$ satisfies the property. From the definitions, x is the second component of a unique pair (y, x) either in $lson$ or in $rson$, and furthermore, $y < x$. We review the different cases.

In Case a), $y = m(x)$, $M(x)$ is undefined, we have $m(x) \sqsubset x$. Then y has no right son in $T(x')$ because otherwise, if it had one say z , then either $x \sqsubset z$ and $M(x)$ would be defined or $z \sqsubset x$ and $m(x)$ would not be y (y would not be the \sqsubset -predecessor of x in $A(x)$). Hence by setting x as right son of y , we get a tree $T(x)$ satisfying $\Omega(T(x)) = (A(x), \sqsubseteq)$.

In Case b) the argument is the same by exchanging left and right, and $m(x)$ and $M(x)$.

In the next two cases $M(x)$ and $m(x)$ are both defined and we have $m(x) \sqsubset x \sqsubset M(x)$. This means that x must be inserted "between" $m(x)$ and $M(x)$ which are consecutive in $(A(x'), \sqsubseteq)$. Furthermore, in $T(x')$, $m(x)$ is an ancestor of $M(x)$ or vice-versa, because otherwise, they have a common ancestor, say z , $m(x) \sqsubset z \sqsubset M(x)$ hence they are not consecutive in $(A(x'), \sqsubseteq)$.

If $m(x) < M(x)$ (case c)), we have $y = M(x)$, x is set as left son of y , $m(x)$ is an ancestor of $M(x)$. (This cannot be the reverse because $m(x) < M(x)$).

Assume y has already a left son, say z , in $T(x')$. Either $x \sqsubset z$ and $M(x)$ would not be y (because $x \sqsubset z \sqsubset y$) or $z \sqsubset x$ but then $m(x) \sqsubset z$ (because $m(x)$ is an ancestor of $M(x)$, so that $M(x)$ is, or is below, the right son of $m(x)$) but this contradicts the definition of $m(x)$. Hence we can set x as left son of y . And we get thus a tree $T(x)$ satisfying $\Omega(T(x)) = (A(x), \sqsubseteq)$.

If $M(x) < m(x)$ (case d)) the proof is fully similar. \square

We now complete the proof of Proposition 6.1. We take for T the union of the trees $T(x)$ which extend one another. Every x in N is a node of this tree, because the isomorphism type of $<$ is ω , hence x is added at some step. The tree T is thus $(N, lson, rson)$. We have noted that $lson$ and $rson$ are definable by first-order formulas. Hence we have the desired transduction γ . \square

Proposition 6.2 : If a set of partial orders of dimension 2 or a subset of CiCC has a decidable MS-OI satisfiability problem, then it has bounded clique-width.

Proof : The proof of Proposition 5.5 is restricted to finite graphs because it uses a representation of linear orders by directed paths. By Proposition 6.1, and by using an auxiliary ω -order, we can represent the countable linear orders by binary trees, hence by graphs of degree 3. Hence, we get an MS-coding into graphs of bounded degree. The argument goes on as in Proposition 5.5. \square

7 Interval graphs

Let \mathcal{V} be a finite or countable set of intervals of the real line. Let $G(\mathcal{V})$ be the graph with set of vertices \mathcal{V} and such that two vertices (i.e., intervals) I and J are adjacent if and only if they *meet*, i.e., if and only if $I \cap J \neq \emptyset$. We denote by $N(I)$ the set of intervals that meet I . We say that a graph $G(\mathcal{V})$ is an *interval graph*.

The set \mathcal{V} is a *standard* if the following conditions are satisfied :

- (1) The intervals are closed.
- (2) No real number is an end of two intervals.
- (3) For every two intervals I, J , if $N(I) = N(J)$, then either $I \subset J$ or $J \subset I$.
- (4) For every two intervals I, J , if $N(I) \subset N(J)$, then $I \subset J$.

Lemma 7.1 : Every finite or countable set of intervals can be transformed into a standard one defining the same interval graph.

Proof : For the case of a finite set, this can be done by easy geometrical transformations. Let \mathcal{V} be a countable set of intervals defined as the union of an increasing sequence of finite sets \mathcal{V}_i . For each i we transform \mathcal{V}_i into a standard set \mathcal{V}'_i where, in conditions (3) and (4), the sets $N(I)$ are understood

with respect to the set \mathcal{V} and not only with respect to \mathcal{V}_i . There are finitely many ways to replace \mathcal{V}_i by a standard set \mathcal{V}'_i . By using Koenig's Lemma (as in [11]), one can select an infinite sequence \mathcal{V}'_i of intervals that can be merged to form the desired transformation of \mathcal{V} . The details are routine. \square

This shows that instead of using the real line, one can represent an interval graph by intervals of a countable linearly ordered set. Our effort will consist in constructing such a set from $G(\mathcal{V})$, by an MS transduction.

The following observation will be crucial. Let \mathcal{V} be a set of intervals (not necessarily standard). Let \prec be the strict partial order on \mathcal{V} defined by $I \prec J$ if and only if every element of I is strictly smaller than every element of J . It is a transitive orientation of $\overline{G(\mathcal{V})}$, the edge-complement of $G(\mathcal{V})$. Hence $\overline{G(\mathcal{V})}$ is a comparability graph.

Let \mathcal{V} be a standard set of intervals. Let an interval x be written $[a(x), b(x)]$. Let $H(\mathcal{V})$ be the directed graph with set of vertices $\mathcal{V} \cup \mathcal{V}'$ where $\mathcal{V}' = \{a(x), b(x) \mid x \in \mathcal{V}\}$, and with the following directed edges:

- edges labelled by 0 representing the successor relation on \mathcal{V}' ,
- edges labelled by 1 directed from x to $a(x)$ for each x in \mathcal{V} ,
- edges labelled by 2 directed from x to $b(x)$ for each x in \mathcal{V} .

This graph is uniformly 2-sparse (because it has an orientation of indegree at most 2). The graph $G(\mathcal{V})$ can be obtained from $H(\mathcal{V})$ as follows :

its vertices are those of $H(\mathcal{V})$ which are the sources of edges labelled by 1 and 2,

there is an edge between two vertices x and y if and only if there exist in $H(\mathcal{V})$: edges labelled by 1 from x and y to x' and y' respectively, edges labelled by 2 from x and y to x'' and y'' respectively, and a vertex z that belongs to the path in $H(\mathcal{V})$ from x' to x'' , and to the path from y' to y'' .

This transformation, call it δ , is an MS transduction. It is clear that it produces $G(\mathcal{V})$ from $H(\mathcal{V})$.

Proposition 7.2 : There is an MS transduction associating with every finite prime interval graph G a uniformly 2-sparse graph H such that $G = \delta(H)$.

Proof : Let G be a finite or countable simple undirected graph, that we assume to be $G(\mathcal{V})$ for some standard set of intervals \mathcal{V} on a countable linear order $<$. (The main part of the proof works for countable as well as finite graphs.) Let us also assume that G is prime with at least 3 vertices. This implies that for any two vertices x and y of G , we have $N(x) \neq N(y)$, (we denote by $N(x)$ the set of vertices adjacent to x) otherwise $\{x, y\}$ is a nontrivial module and G is not prime.

Let us associate with every vertex x two pairs $(x, 1)$ and $(x, 2)$ intended to represent $a(x)$ and $b(x)$. We let \mathcal{V}' be the set of such pairs.

Let us also choose two non adjacent vertices u and v such that $u \prec v$ where \prec is the order on the set \mathcal{V} of intervals. Since G is prime, so is its edge-complement which is a comparability graph, and has thus a unique transitive orientation such that $u \longrightarrow v$. Furthermore, this orientation can be defined by an MS formula

by Proposition 5.2. By the uniqueness of the transitive orientation (because G is assumed to be prime), its transitive closure coincides with the order \prec on \mathcal{V} .

We have thus available some information on the relative ordering of the ends of our intervals. We now prove that this information can be completed so as to give a reconstruction of the linear order on \mathcal{V}' , corresponding to $<$ by the bijection of \mathcal{V}' onto the set of ends of the intervals. We define W as the set of all pairs of the following types :

- (i) $((x, 1), (x, 2))$ for x in \mathcal{V} ,
- (ii) $((x, 2), (y, 1))$ for x, y in \mathcal{V} and $x \longrightarrow y$,
- (iii) $((x, 1), (y, 1))$ and $((y, 2), (x, 2))$ for adjacent vertices x, y in \mathcal{V} , such that $N(y) \subset N(x)$,
- (iv) $((x, 1), (y, 1)), ((y, 1), (x, 2)), ((x, 2), (z, 1)), ((z, 1), (y, 2))$, for x, y, z in \mathcal{V} such that y is adjacent to x and to z , $x \prec z$, and neither $N(x) \subset N(y)$ nor $N(y) \subset N(x)$,
- (v) $((y, 1), (z, 2)), ((z, 2), (x, 1)), ((x, 1), (y, 2)), ((y, 2), (x, 2))$, for x, y, z in \mathcal{V} such that y is adjacent to x and to z , $z \prec x$, and neither $N(x) \subset N(y)$ nor $N(y) \subset N(x)$.

We now prove that the transitive closure of W yields the linear order on the ends of the intervals of \mathcal{V} . Each pair (w, w') in W satisfies $w < w'$ where w, w' are the corresponding ends of the considered intervals. This is obvious for each of cases (i)-(v).

Let us now consider two interval ends. The two ends of a same interval are ordered by (i).

The ends of two disjoint intervals are ordered by (ii), (i) and transitivity.

The ends of two adjacent intervals x and y such that $N(x) \subset N(y)$ or $N(y) \subset N(x)$ are ordered by (iii), (i) and transitivity.

Consider now two overlapping intervals x and y (the previous case does not apply). Since G is prime there is some z which is adjacent to x and not to y or vice versa.

If z is adjacent to y and $x \prec z$, we must have the ordered pairs listed in (iv) because y intersects x and z and does not contain x .

If z is adjacent to y and $z \prec x$, we must have the ordered pairs listed in (v) for the same reason.

If z is adjacent to x and not to y , we can exchange the roles of x and y in (iv) and (v) and use them to order the ends of x and y .

Hence, for any two intervals, conditions (i)-(v) together with transitivity give the linear order of the ends of the intervals as given in \mathcal{V} . This linear order can be thus defined by an MS formula.

For each x in \mathcal{V} , we let $I(x)$ be the interval $[(x, 1), (x, 2)]$ of this linear order on \mathcal{V}' . It is clear that the intersection graph of this family is G .

From now on we assume that G is finite. The above define objects can be encoded into a graph H with set of vertices $\mathcal{V} \cup \mathcal{V}'$, having three types of directed edges :

- edges labelled by 0 representing the successor relation on \mathcal{V}' ,

edges labelled by 1 directed from x to $(x, 1)$ for each x in \mathcal{V} ,
edges labelled by 2 directed from x to $(x, 2)$ for each x in \mathcal{V} .

This graph is uniformly 2-sparse, and can be defined from G (assumed to be a prime interval graph) by an MS transduction.

This transduction works properly under the following assumptions:

a) that G is prime : this is easily MS expressible,

b) that G is an interval graph : this is MS expressible by means of the forbidden induced subgraphs characterizing interval graphs given by Trotter and Moore in [35], which consist of two finite graphs, the cycles C_n for $n \geq 4$ and the graphs of the families \mathcal{G} and \mathcal{H} used to describe comparability graphs ; the proof is the same as for Lemma 5.1,

c) that $u \prec v$ in the given set of intervals : this condition is actually not important. If it does not hold, we have $v \prec u$ and the opposite linear order on the elements forming the intervals is constructed.

Hence only the verifications of a) and b) must be incorporated as preliminary tests to the definition of MS transduction constructing H from G . \square

Theorem 7.3 : The class of finite interval graphs satisfies the Conjecture.

Proof : Immediate consequence of Proposition 7.2 since we have an MS coding of finite prime interval graphs into uniformly 2-sparse graphs. The general case follows from Proposition 3.10. \square

With the tools of Section 6, we can handle the case of countable interval graphs.

Theorem 7.4: If a set of interval graphs has a decidable MS-OI satisfiability problem, then it has bounded clique-width.

Proof : The proof of Proposition 7.2 only holds for *finite* interval graphs because it uses a representation of linear orders by directed paths, which does not hold for infinite linear orders. With the results of Section 6, we can encode a countable linear order given with an auxiliary ω -linear order by a binary tree, i.e., by a graph of outdegree 2. Hence, the graph H can be replaced by a graph of degree 3, hence a graph which is uniformly 3-sparse. We obtain thus an MS coding as desired. \square

8 Intrinsic limits of the methods used so far

Our main technique to establish that a class of graphs \mathcal{C} satisfies the Conjecture is to find an MS coding $\mathcal{C} \longrightarrow \mathcal{U}_k$ where \mathcal{U}_k is the class of uniformly k -sparse graphs. (The constructions of Section 4 for line graphs are also essentially of

this form because the incidence structures $Inc(G)$ are equivalent to uniformly 2-sparse edge labelled directed graphs.) Is there any hope to prove the Conjecture by this technique ? Our answer is no because of the following.

Proposition 8.1 : There does not there exist an MS coding (γ, δ) of Finite Undirected Graphs into \mathcal{U}_k , for any k .

Proof : Let $\mathcal{C} = \text{Finite Undirected Graphs}$. We assume the existence of MS transductions $\gamma : \mathcal{C} \rightarrow \mathcal{U}_k, \delta : \mathcal{U}_k \rightarrow \mathcal{C}$ such that for every G in \mathcal{C} , for every H in $\gamma(G)$ we have $\delta(H) = G$. We will get a contradiction by means of a counting argument.

Let us consider a *labelled graph* G with n vertices labelled from 1 to n (this is a "labelled object" in the sense of combinatorics, see [37]). Let us assume that γ is p -copying. Then it produces graphs with at most np vertices. We may consider that all the graphs of $\gamma(G)$ have np vertices, with a special label marking the vertices that should be considered as absent. We consider that the graphs in $\gamma(G)$ are directed, vertex and edge labelled : each vertex has one label taken from a set of cardinality q , each edge has a direction and one label taken from a set of cardinality r (without loss of generality, we assume we do not have two edges with different labels and/or directions between two vertices).

How many graphs labelled in this way, with set of vertices $\{1, \dots, np\}$ can be in \mathcal{U}_k ? We first bound the number of labelled graphs that can be constructed from G by γ (in terms of parameters). The vertices of these graphs are pairs of a vertex of G and an integer in $\{1, \dots, p\}$. The number of labelled rooted forests with m vertices is $(m+1)^{m-2}$ (see Wilf [37]). A uniformly k -sparse graph is the union of k rooted forests. Hence with set of vertices $\{1, \dots, np\}$, one can build at most $(np+1)^{k(np-2)}$ such graphs, each having at most knp edges. (This is not a tight upper bound because some edges may be present in several of the k forests).

The number of labellings of the vertices is q^{np} and that of labellings of edges is at most r^{knp} . Concerning edge directions, we have at most 2^{knp} possibilities to modify the directions given by the covering forests.

Hence, the domain of the decoding transduction δ has cardinality at most $(np+1)^{k(np-2)}q^{np}(2r)^{knp}$. We can bound this number by $2^{a \cdot n \log(n)+b}$ for some a and b depending on p, q, k , and r . Hence, the class \mathcal{C} may have at most $2^{a \cdot n \log(n)+b}$ labelled graphs with n vertices. It cannot be equal to the class of all undirected simple graphs with n vertices labelled from 1 to n which has cardinality $2^{\binom{n-1}{2}}$.

Actually this comparison is based on the interpretation of $\delta(H) = G$ as an real equality and not as an equality *up to isomorphism*. This real equality is actually the case in all the concrete constructions used in our proofs. However, one must not exclude the case of an MS coding for which $\delta(H) = G$ holds only up to isomorphism. Hence we should compare $2^{a \cdot n \log(n)+b}$ with the number of unlabelled graphs with n vertices. Consider a labelled complete graph. Every

permutation of its vertices leaves it invariant. This means that it is counted only for one among the $2^{n(n-1)/2}$ graphs. On the opposite, a graph without automorphism is produced $n!$ times by $n!$ different labelled graphs. It follows that the number of unlabelled graphs with n vertices is larger than $2^{n(n-1)/2}/n!$, which is itself of the order $2^{n^2/2 - c \cdot n \log(n) + d'}$ (since $n!$ is of order $2^{c \cdot n \log(n) + d}$) hence is larger than $2^{a \cdot n \log(n) + b}$. \square

Is it possible to improve the situation by constructing an MS coding of \mathcal{C} into a class \mathcal{D} for which we already have an MS coding into \mathcal{U}_k ? No because in such a case, we can compose these MS codings and we obtain an MS coding of \mathcal{C} into \mathcal{U}_k . Hence, even if we pile up ingenious constructions, we are always facing the same problem. We conclude that something else must be invented to prove the conjecture. This is done in Courcelle and Oum [17].

9 Conclusion and open questions

We have proved that many relativizations of Seese's Conjecture turn out to be equivalent. We have also established a few new cases of validity of the Conjecture. The main results are summarized in Table 1 and 2

Undirected graph classes	Directed graph classes
Undirected	Directed
Bipartite	Directed Acyclic
Chordal	
Split	
Comparability	Partial Orders

Table 1 : Equivalent relativizations

Undirected graph classes	Directed graph classes
Uniformly k-sparse	Uniformly k-sparse
Line graphs	Directed line graphs
	Quasi-series-parallel partial orders
Finite interval graphs	Finite partial orders of dimension 2
Interval graphs (for MS-OI)	Partial orders of dimension 2 (for MS-OI)

Table 2 : Proved relativizations.

Table 1 shows the equivalent relativizations. One could add the extensions of these classes by vertex and edge labellings. Table 2 shows the main established relativizations. Here are some open questions.

Question 9.1 : Do we have Finite Relational Structures \leq_S Finite Graphs ?

Question 9.2 : Do we have Countable Graphs \equiv_S Finite Graphs ?

A countable graph is *locally finite* if every vertex has finite degree. A countable graph has clique-width at most $f(k)$ if its finite induced subgraphs have clique-width at most k , for some function f . By using this fact, one obtains the following result (Corollary 11.3 of [12]) :

Proposition 9.3 : Locally Finite Graphs \equiv_S Finite Graphs.

A variant of MS logic is *weak monadic second-order logic*, which has the same syntax but for which set variables range on finite sets. Seese's Conjecture for finite graphs is equivalent to the corresponding conjecture for countable graphs relatively to satisfiability of weak monadic second-order formulas. (Courcelle [12], Theorem 11.1).

Finally, one can strengthen the definition of a tree-definable set as follows. Let us say that a set of graphs L is *strongly tree-definable* if there exists an MS coding (γ, δ) of L into a set of labelled trees T . In such a case, the MS satisfiability problems for L and for the subset $\gamma(L)$ of T are interreducible by Proposition 2.1.

Question 9.4 : Is it true that if set of graphs has a decidable MS satisfiability problem, then it is strongly tree-definable ?

Lapoire has proved [26] that for each k , one can define an MS coding of the finite graphs of tree-width at most k into a set T of finite labelled trees which encode tree-decompositions of width k of the input graphs. (The trees in T are unordered and of unbounded degree). The proofs of relativizations of the Conjecture shown in Table 2 reduce, via effectively constructible MS transductions, to that of Proposition 2.6 for graphs of bounded tree-width. Hence for all these cases restricted to finite graphs, the strong form of the Conjecture holds. (The result of [26] is not known to hold for infinite graphs).

We have shown in Proposition 8.1 that the class of finite graphs has no MS coding into any class \mathcal{U}_k .

Question 9.5 : What are the common structural properties of the classes of graphs which have MS codings in \mathcal{U}_k for some k ?

Acknowledgements : Thanks to A. Blumensath and D. Seese for helpful remarks.

10 References

- [1] H. Bodlaender, A partial k -arboretum of graphs with bounded treewidth. *Theor. Comput. Sci.* 209 (1998): 1-45.
- [2] A. Brandstädt, Van Bang Le, J. Spinrad, Graph classes, a survey, SIAM monographs on discrete mathematics, 1999.
- [3] B. Courcelle, The monadic second-order logic of graphs II: Infinite graphs of bounded width, *Mathematical Systems Theory*, 21(1989)187-221.
- [4] B. Courcelle, The monadic second-order logic of graphs V: On closing the gap between definability and recognizability. *Theoretical Computer Science*, 80 (1991) 153-202.
- [5] B. Courcelle, The monadic second-order logic of graphs VI: On several representations of graphs by relational structures, *Discrete Applied Mathematics* 54 (1994) 117-149. Corrections in *Discrete Applied Mathematics* 63 (1995) 199-200.
- [6] B. Courcelle. Monadic second-order graph transductions: A survey. *Theoretical Computer Science*, 126:53–75, 1994.
- [7] B. Courcelle, The monadic second-order logic of graphs VIII: Orientations, *Annals of Pure Applied Logic* 72 (1995) 103-143.
- [8] B. Courcelle, The monadic second-order logic of graphs X: Linear orders, *Theoretical Computer Science*, 160 (1996) 87-143.
- [9] B. Courcelle. The expression of graph properties and graph transformations in monadic second-order logic. In G. Rozenberg, editor, *Handbook of graph grammars and computing by graph transformations, Vol. 1: Foundations*, pages 313–400. World Scientific, 1997.
- [10] B. Courcelle, A monadic second-order definition of the structure of convex hypergraphs, *Information and Computation* 178 (2002) 391-411
- [11] B. Courcelle, The monadic second-order logic of graphs XIV: Uniformly sparse graphs and edge set quantifications. *Theoretical Computer Science* 299 (2003) 1-36.
- [12] B. Courcelle, Clique-width of countable graphs: a compactness property. *Discrete Mathematics* 276 (2004) 127-148.
- [13] B. Courcelle and J. Engelfriet. A logical characterization of the sets of hypergraphs defined by hyperedge replacement grammars. *Mathematical Systems Theory*, 28:515–552, 1995.
- [14] B. Courcelle and J. Makowsky, Fusion in relational structures and the verification of monadic second-order properties, *Math. Struct. in Comp. Sci.* 12 (2002) 203-235.
- [15] B. Courcelle, J. Makowsky, U. Rotics, Linear time solvable optimization problems on graphs of bounded clique-width, *Theory Comput. Systems* 33(2000) 125-150.
- [16] B. Courcelle and S. Olariu, Upper bounds to the clique-width of graphs, *Discrete Applied Maths*, 101 (2000) 77-114.
- [17] B. Courcelle and S. Oum, Vertex-minors, monadic second-order logic and a conjecture by Seese, July 2004, submitted, <http://www.labri.fr/~courcell>

- [18] R. Diestel, Graph Theory, Springer, 2000, freely readable on <http://www.math.uni-hamburg.de/home/diestel/books/graph.theory/>.
- [19] R. Diestel, T. Jensen, K. Gorbunov, C. Thomassen, Highly connected sets and the excluded grid theorem, *J. Comb. Th. Series B*, 75 (1999) 61-73.
- [20] J. Engelfriet and V. van Oostrom, Logical description of context-free graph-languages, *Journal of Computer and System Sciences*, 55:489-503, 1997.
- [21] M. Golumbic, Algorithmic graph theory and perfect graphs, Academic Press, 1980.
- [22] F. Gurski and E. Wanke, Line graphs of bounded clique-width, Preprint, 2004.
- [23] M. Habib, R. Jegou, N-free posets as generalizations of series-parallel posets, *Discrete Applied Maths* 12(1985), 279-291.
- [24] C. Heuchenne, Sur une certaine correspondance entre graphes, *Bull. Soc. Roy. Sci.*, Liège, Belgium, 33 (1964) 743-753.
- [25] D. Kelly, Comparability graphs, in I. Rival ed., *Graphs and order*, Reidel, Dordrecht, 1985, pp. 3-40.
- [26] D. Lapoire, Recognizability equals monadic second-order definability for sets of graphs of bounded tree-width, Proceedings of STACS 1998, *Lec. Notes Comp. Sci.*, Vol. 1373, pp.618-628.
- [27] R. Möhring, Computationally tractable classes of ordered sets, in I. Rival ed., *Algorithms and Order*, Kluwer Acad. Publish., Dordrecht, 1989, pp. 105-194.
- [28] R. Möhring and F. Radermacher, Substitution decomposition for discrete structures and connections with combinatorial optimization, *Annals of Discrete Mathematics* 19 (1984) 257-356.
- [29] M. Rabin, Decidability of second-order theories and automata on infinite trees, *Transactions of the AMS*, 141 (1969) 1-35.
- [30] N. Robertson, P. Seymour, Graph Minors V, Excluding a planar graph, *J. Comb. Th. Series B*, 41 (1986) 92-114.
- [31] D. Seese. The structure of the models of decidable monadic theories of graphs. *Annals of Pure Applied Logic*, 53:169-195, 1991.
- [32] D. Seese, Interpretability and tree-automata: a simple way to solve algorithmic problems on graphs closely related to trees, in *Tree Automata and Languages*, M. Nivat and A. Podelski (eds.), 1992, Elsevier Science Pub., pp. 83 - 114.
- [33] J. Spinrad, Efficient Graph Representations, Fields Institute Monographs 19, American Mathematical Society, Providence, 2003
- [34] W. Trotter, Combinatorics and partially ordered sets - Dimension theory, J. Hopkins University Press, Baltimore, London, 1992.
- [35] W. Trotter, J. Moore, Characterization problems for graphs, partially ordered sets, lattices, and families of sets, *Discrete Mathematics* 16(1976) 361-381.
- [36] H. Whitney, Congruent graphs and the connectivity of graphs, *Amer. J. Math.* 54 (1932) 150-168.
- [37] H. Wilf, Generating Functionology, Academic Press, 1994, freely readable on <http://www.math.upenn.edu/~wilf/DownldGF.html>.