

On the Parameterised Intractability of Monadic Second-Order Logic

Stephan Kreutzer *

Oxford University Computing Laboratory, kreutzer@comlab.ox.ac.uk

Abstract. One of Courcelle’s celebrated results states that if \mathcal{C} is a class of graphs of bounded tree-width, then model-checking for monadic second order logic (MSO_2) is fixed-parameter tractable (fpt) on \mathcal{C} by linear time parameterised algorithms. An immediate question is whether this is best possible or whether the result can be extended to classes of unbounded tree-width.

In this paper we show that in terms of tree-width, the theorem can not be extended much further. More specifically, we show that if \mathcal{C} is a class of graphs which is closed under colourings and satisfies certain constructibility conditions such that the tree-width of \mathcal{C} is not bounded by $\log^{16} n$ then MSO_2 -model checking is not fpt unless SAT can be solved in sub-exponential time. If the tree-width of \mathcal{C} is not poly-log. bounded, then MSO_2 -model checking is not fpt unless all problems in the polynomial-time hierarchy can be solved in sub-exponential time.

1 Introduction

In 1990, Courcelle proved a fundamental result stating that every property of graphs definable in *monadic second-order logic with edge set quantification* (MSO_2) can be decided in linear time on any class \mathcal{C} of graphs of bounded tree-width. Courcelle’s theorem has important consequences both in logic and in algorithm theory. In the design of efficient algorithms on graphs, it can often be used as a simple way of establishing that a property can be solved in linear time on graph classes of bounded tree-width. Besides being of interest for specific algorithmic problems, results such as Courcelle’s and similar *algorithmic meta-theorems* lead to a better understanding how far certain algorithmic techniques, dynamic programming and decomposition in the case of MSO_2 , range and establish general upper bounds for the parameterised complexity of a wide range of problems. See [9, 10] for recent surveys on algorithmic meta-theorems.

From a logical perspective, Courcelle’s theorem establishes a sufficient condition for tractability of MSO_2 formula evaluation on classes of graphs or structures: whatever the class \mathcal{C} may look like, if it has bounded tree-width, then MSO_2 -model checking is tractable on \mathcal{C} . An obvious question to ask is how tight Courcelle’s theorem is, i.e. whether it can be extended to classes of unbounded tree-width and if so, how large the tree-width of graphs in the class can be in general. Given the considerable interest in Courcelle’s theorem, it is somewhat surprising that not much is known about such limits for MSO_2 model checking. Recently, the question has informally been raised in the community and has led, e.g., to a conjecture by Grohe [9, Conjecture 8.3] that MSO -model

* Research supported by DFG grant KR 2898/1-3. Part of this work was done while the author participated at the workshop “Graph Minors” at Banff Intern. Research Station, October 2008.

checking is not fixed-parameter tractable on any class \mathcal{C} of graphs which is closed under taking subgraphs and whose tree-width is not poly-logarithmically bounded, i.e. there are no constants c, d such that $\text{tw}(G) \leq d \cdot \log^c |G|$ for all $G \in \mathcal{C}$. But to the best of my knowledge, the question has so far not been studied systematically. This is the main motivation for the work reported in this paper.

It follows from the NP-completeness of 3-colourability on planar graphs [8] that MSO-model checking is not fixed-parameter tractable on the class of planar graphs (unless $P = \text{NP}$). Furthermore, it is a simple consequence of the excluded grid theorem that on minor- or topological-minor closed classes of graphs of unbounded tree-width, MSO-model checking is not fpt unless $P = \text{NP}$ (see Section 2). In this paper we establish a strong intractability result by showing that in terms of tree-width, Courcelle's theorem can not be extended much further to classes of unbounded tree-width. Throughout the paper, we will work with coloured graphs, i.e. we will fix a set Γ of edge and vertex colours. A class \mathcal{C} of graphs is said to be closed under Γ -colourings if whenever $G \in \mathcal{C}$ and G' is obtained from G by recolouring, i.e. the underlying undirected graphs are isomorphic, then $G' \in \mathcal{C}$. We will mostly consider classes of graphs closed under colourings. An alternative characterisation is to consider relational structures over a signature σ with at most binary relation symbols. We can then fix a class \mathcal{C}' of graphs and consider the class of all finite σ -structures whose Gaifman-graphs are in \mathcal{C} . However, in this paper we prefer to work with coloured graphs rather than Gaifman-graphs of structures. Given a class \mathcal{C} of graphs, we write $\text{MC}(\text{MSO}_2, \mathcal{C})$ for the model-checking problem for MSO_2 on \mathcal{C} (see Section 2 for details).

Definition 1.1. The tree-width of a class \mathcal{C} of graphs is *strongly unbounded* by a function $f : \mathbb{N} \rightarrow \mathbb{N}$ if there is a polynomial $p(x)$ such that for all $n \geq 1$ there is a graph $G \in \mathcal{C}$ of tree-width between n and $p(n)$ whose tree-width is not bounded by $f(|G|)$ and 2) given n , G_n can be constructed in time $2^{(|n|_u)^\varepsilon}$, for some $\varepsilon < 1$, where $|n|_u$ denotes the unary encoding of n . The tree-width of \mathcal{C} is *strongly unbounded poly-logarithmically* if it is strongly unbounded by $\log^c n$, for all c .

Essentially, *strongly* means that a) there are not too big gaps between the tree-width of graphs witnessing that the tree-width of \mathcal{C} is not bounded by $f(n)$ and b) we can compute such witnesses efficiently. We will see below why this condition is needed. The following is the main result of the paper. Let Γ be a set of colours with at least one edge and two vertex colours.

Theorem 1.2. *Let \mathcal{C} be a constructible class of Γ -coloured graphs closed under colourings.*

1. *If the tree-width of \mathcal{C} is strongly unbounded poly-logarithmically then $\text{MC}(\text{MSO}_2, \mathcal{C})$ is not in XP, and hence not fpt, unless all problems in NP (in fact, all problems in the polynomial-time hierarchy) can be solved in sub-exponential time.*
2. *If the tree-width of \mathcal{C} is strongly unbounded by $\log^{16} n$ then $\text{MC}(\text{MSO}_2, \mathcal{C})$ is not in XP unless SAT can be solved in sub-exponential time.*

See Section 2 for a definition of FPT and XP. We refer to Definition 3.6 for a precise definition of constructible classes but will explain the concept informally below. Let us give some applications of the theorem. For $c > 0$ let \mathcal{C}_c be the class of all graphs

G of tree-width at most $\log^c |G|$. This class is constructible and hence its closure under colourings has intractable MSO_2 model-checking, if $c > 16$. Similarly, the class of planar graphs of tree-width at most $\log^c n$ is constructible. Finally, all (topological) minor-closed classes of unbounded tree-width are constructible and rich. All these examples show that Courcelle's theorem can not be extended to classes of graphs with only poly-logarithmic or a $\log^c n$ bound on the tree-width, for $c > 16$.

High level description of the proof. Let us give an intuitive account of the proof of the previous theorem. Clearly, with today's methods we cannot hope to prove that MSO_2 -model-checking is fixed-parameter intractable for a class of graphs without relating it to assumptions in complexity theory. Consequently, we prove that $\text{MC}(\text{MSO}_2, \mathcal{C})$ is fixed-parameter intractable for a class \mathcal{C} by reducing an NP-complete problem P to $\text{MC}(\text{MSO}_2, \mathcal{C})$ such that if there is an fpt-algorithm for $\text{MC}(\text{MSO}_2, \mathcal{C})$, then P can be solved in sub-exponential time $2^{o(n)}$. More precisely, for each language $P \in \text{NP}$ we construct a formula φ_P and then, given a word w , we construct a graph $G_w \in \mathcal{C}$ such that $G_w \models \varphi_P$ if, and only if, $w \in P$. We will see that the number of vertices of G_w can be bounded by $2^{|w|^{\frac{1}{y}}}$, for some $y > 1$, so that if there was an algorithm for $\text{MC}(\text{MSO}_2, \mathcal{C})$ with running time $f(|\varphi|) \cdot |G|^c$ then this would imply that $w \in P$ could be decided in time $\mathcal{O}(2^{c|w|^{\frac{1}{y}}}) = 2^{o(|w|)}$. Here, Condition 1) of Definition 1.1 ensures that \mathcal{C} contains a graph G_w the word w can be reduced to and Condition 2) ensures that we can compute it in time sub-exponential in the length of $|w|$. For this reduction to work, we need some intermediate steps.

It is well-known that MSO_2 -model checking is fixed-parameter intractable on the class of coloured grids (see Figure 1), which can be seen as follows: suppose P can be solved by a non-deterministic Turing-machine M in time n^c , where n is the length of the input. Given a word w of length n , we choose a $(n^c \times n^c)$ -grid G_w and label its top-most row by w from left to right. From the Turing-machine M deciding P we can compute an MSO_2 -formula φ_M depending only on M such that $G_w \models \varphi_M$ if, and only if, w is accepted by M and hence $w \in P$. Essentially, the MSO formula uses the grid to guess the computation table of a successful run of M on w . Hence, an fpt-algorithm for MSO -model checking on grids yields a polynomial time algorithm for P .

Clearly, if we are just given a class of graphs of tree-width not bounded poly-logarithmically, then there is no guarantee that it contains any grids. But we will show that we can define grids in graphs of this class by MSO_2 -formulas. Adapting a recent proof by Reed and Wood, we first show that if G is a graph of large tree-width then it contains a large structure which we call *coloured pseudo-wall*. Pseudo-walls do not actually occur as minors or subgraphs of G but as topological minors of certain intersection graphs formed by sets of disjoint paths in G . However, it turns out that this is enough to define coloured grids in coloured pseudo-walls by MSO_2 -formulas. It follows that if the tree-width of G is not bounded by k then we can define an $(l \times l)$ -grid in G in MSO_2 , where l is roughly $\sqrt[10]{k}$ (see Theorem 3.5 for details), and this grid can be coloured. We call a class *constructible* if we can construct these pseudo-walls in graphs $G \in \mathcal{C}$ in polynomial time and it is such classes with which we work in this paper (see Definition 3.6 for details).

The important aspect here is that the size of the grids we define is polynomially related to the tree-width of the graph, in contrast to the grids guaranteed by the excluded grid theorem (see Theorem 3.1), where the tree-width is exponentially larger than the grids we are guaranteed to find. Hence, using pseudo-walls, if the tree-width of a graph is $\log^c n$ then we can define $(l \times l)$ -grids in G for $l \cong \log^{\frac{c}{10}} n$ and this is enough for the reduction sketched above to work.

Obtaining sub-exponential time algorithms for problems such as TSP or SAT is an important open problem in the algorithms community and the common assumption is that no such algorithm exists. This has led to the *exponential-time hypothesis* (ETH) which says that there is no such sub-exponential time algorithm for SAT, a hypothesis widely believed in the community.

Let us briefly comment on the restrictions imposed on the classes \mathcal{C} we study here. While every graph of large enough tree-width contains large pseudo-walls, we do not yet know if we can always compute these structures in polynomial time (and hence we impose the additional restriction to constructible classes). It is conceivable that large pseudo-walls can indeed be computed in polynomial time. This would essentially show that all classes are constructible and effectively remove this condition from our main result. We pose this question as an open problem.

As mentioned above, Grohe conjectured that MSO-model checking is not fpt on any class of graphs closed under subgraphs and whose tree-width is not poly-logarithmically bounded. The statement of the conjecture and the main result of this paper are incomparable as I require closure under colourings whereas Grohe does not. On the other hand, the conjecture requires closure under subgraphs which I do not. Note that while tree-width is preserved by taking subgraphs, logarithmic tree-width is not, i.e. a graph whose tree-width is bounded by $\log n$ may contain a subgraph of order m whose tree-width is not bounded by $\log m$. Closure under subgraphs therefore does rule out natural examples of graph classes. For instance, the class of all graphs of tree-width at most $\log n$ is not closed under subgraphs. On the other hand, Grohe's conjecture does not require colours or constructibility conditions and refers to classes of plain graphs.

Note that it is important for our results that we work with MSO_2 and allow quantification over sets of edges as well as over sets of vertices. If we only consider vertex set quantification, i.e. deal with MSO_1 , then the theorem is false, as for instance, MSO_1 -model checking is fpt on the class of cliques.

Following Courcelle's theorem, a series of algorithmic meta-theorems for first-order logic on planar graphs [7], (locally) minor-free graphs [6, 2] and various other classes have been obtained. Again, no deep lower bounds, i.e. intractability conditions, are known (see [10] for some simple bounds and [9, 10] for recent surveys of the topic). The aim of this paper is to initiate a thorough study of sufficient conditions for intractability in terms of structural properties of input instances.

Organisation. We recall monadic second-order logic and what we need about its parameterised complexity in Section 2. The main result is then proved as follows. To show that $\text{MC}(\text{MSO}_2, \mathcal{C})$ is hard on classes \mathcal{C} of tree-width not poly-logarithmically bounded, we first use a result by Reed and Wood [12] to show that any graph of large enough tree-width contains a structure that is grid-like enough for our purposes. This is proved in Section 3. While these structures do not exist as minors in the graphs, they turn out

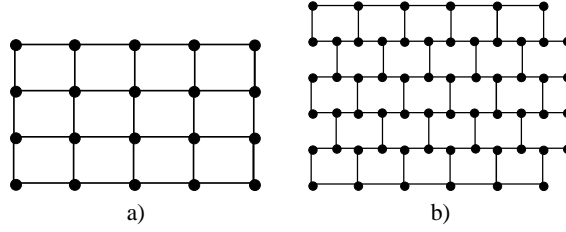


Fig. 1. a) A (4×5) -grid and b) an elementary wall of order 5.

to be MSO_2 -definable, which is shown in Section 4. To use the MSO_2 -definability for our result, we introduce a new kind of interpretations between structures, called MSO_2 – MSO_2 -*transductions* (see Section 5). Finally, in Section 6, we combine all this to show the main result of the paper. Due to space restrictions, some proofs have been removed from this abstract. See <http://arxiv.org/abs/0904.1302> for a full version.

Acknowledgements. I would like to thank Mark Weyer for pointing out that the result proved here readily extends to problems in the polynomial time hierarchy.

2 Complexity of Monadic Second-Order Logic

We first need some notation and a few concepts from graph theory. We refer to [3] for background on graphs. For $k \geq 1$, we define $[k] := \{1 \dots, k\}$. All graphs in this paper are finite and undirected. We write $V(G)$ for the set of vertices and $E(G)$ for the set of edges in a graph G . A graph H is a *sub-division* of G (a *1-subdivision*) if H is obtained from G by replacing edges in G by paths of arbitrary length (of length 2, resp.). H is a *topological minor* of G if a subgraph $G' \subseteq G$ is isomorphic to a sub-division of H .

An *elementary wall* W is a graph as in Figure 1b). The cycles of minimal length in W are called *bricks*. A *wall* is a subdivision of an elementary wall. The *height* of a wall is the number of rows of bricks and its *width* the number of columns of bricks. An $l \times k$ -wall is a wall of height l and width k and a wall of *order* l is an $l \times l$ -wall. Finally, the *nails* of a wall are the vertices of degree 3 in it together with the 4 corners. Hence, in an elementary wall all vertices are nails whereas in a general wall only the vertices of the underlying elementary wall are nails.

For the purpose of this paper, it might be easier to think of $k \times k$ -grids instead of $k \times k$ -walls and everything would go through with grids also. The important property of walls is that their maximum degree is 3. And if a graph H of degree ≤ 3 is a minor of G , then H is also a topological minor of G (see [3, Prop. 1.7.2]). Hence, a sub-division of H actually occurs as subgraph of G . Defining topological minors in MSO_2 is much easier than defining minors as we do not need contraction. We will therefore work with walls instead of grids in this paper.

I assume familiarity with basic notions of mathematical logic (see e.g. [4]). In this paper we will only consider signatures $\sigma := \{E, B_1, \dots, B_s, C_1, \dots, C_t\}$ of coloured graphs, where E denotes the edge relation, B_i the colours of edges and C_i the colours of vertices. We allow multiple colours per edge or vertex. We denote σ -structures by

Roman letters A, G, H, \dots . If $R \in \sigma$ is a relation symbol and A a σ -structure, we write $R(A)$ for the interpretation of R in A .

The class of formulas of *monadic second-order logic with edge set quantification* over a signature σ , denoted $\text{MSO}_2[\sigma]$, is defined by the rules for first-order logic with the following additional rules: if X is a second-order variable either ranging over sets of vertices or over sets of edges and $\varphi \in \text{MSO}_2[\sigma \dot{\cup} \{X\}]$, then $\exists X \varphi \in \text{MSO}_2[\sigma]$ and $\forall X \varphi \in \text{MSO}_2[\sigma]$ with the obvious semantics where, e.g., a formula $\exists F \varphi$, F being a variable over sets of edges, is true in a structure A if there is a subset $F \subseteq E(A)$ such that $(A, F) \models \varphi$. If $\varphi(x)$ is a formula with a free variable x and A is a structure, we write $\varphi(A)$ for the set $\{a : A \models \varphi[a]\}$. See [11] for more on MSO.

In [15], Vardi proved that model checking for MSO_2 is PSPACE-complete on the class of all graphs. The complexity of model-checking problems can elegantly be studied in the framework of *parameterised complexity* (see [5] for background on parameterised complexity). If \mathcal{C} is a class of σ -structures, we define the *parameterised model-checking problem* $\text{MC}(\text{MSO}_2, \mathcal{C})$ for MSO_2 on \mathcal{C} as the problem to decide, given $G \in \mathcal{C}$ and $\varphi \in \text{MSO}_2[\sigma]$, if $G \models \varphi$. The *parameter* is $|\varphi|$. The problem is *fixed-parameter tractable* (fpt), or in the class FPT, if there is a computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ and $k \in \mathbb{N}$, such that for all $G \in \mathcal{C}$ and $\varphi \in \text{MSO}_2[\sigma]$, $G \models \varphi$ can be decided in time $f(|\varphi|) \cdot |G|^k$. The problem is in the class XP, if it can be decided in time $|G|^{f(|\varphi|)}$. FPT in the parameterised world corresponds to polynomial-time in the classical framework as the class of problems that can be solved efficiently. XP can be seen as the parameterised exponential-time and is obviously a much larger class of problems than FPT.

Tree-width is a global connectivity measure of graphs that was introduced by Robertson and Seymour in their graph minor series. We refer the reader to [3] for a definition of tree-width. Let $f : \mathbb{N} \rightarrow \mathbb{N}$ be a function and \mathcal{C} be a class of graphs. The tree-width of \mathcal{C} is *bounded by f* , if $\text{tw}(G) \leq f(|G|)$ for all $G \in \mathcal{C}$. \mathcal{C} has *bounded tree-width* if its tree-width is bounded by a constant. Many natural classes of graphs, for instance series-parallel graphs, are found to have bounded tree-width. The following lemma, whose proof is standard, will be needed below.

Lemma 2.1. *Let M be a non-deterministic Turing-machine. There is a formula $\varphi_M \in \text{MSO}_2$ such that for all words $w \in \Sigma^*$, if G is a $k \times k$ -wall whose top-most row is coloured by w from the left, then $G \models \varphi_M$ if, and only if, M accepts w in at most k steps. Furthermore, the formula φ_M can be constructed effectively from M . The same holds if M is an alternating Turing-machine with a bounded number of alternations, as they are used to define the polynomial-time hierarchy.*

Lemma 2.1 together with Theorem 3.1 shows that if \mathcal{C} is closed under (topological) minors and has unbounded tree-width, then $\text{MC}(\text{MSO}_2, \mathcal{C})$ is not fpt unless $P = \text{NP}$.

3 Pseudo-Walls in Graphs

One of the fundamental results of Robertson and Seymour's theory of graph minors is the excluded grid theorem [14], saying that there is a computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that every graph of tree-width at least $f(k)$ contains a $k \times k$ -grid as a minor. The best explicit bound known for the function f is given by the following theorem.

Theorem 3.1 (Robertson, Seymour, Thomas [13]). *Every graph of tree-width at least $20^{2 \cdot k^5}$ contains a $k \times k$ grid as a minor.*

Robertson et al. [13] also proved that there are graphs of tree-width proportional to $k^2 \log k$ that do not contain $G_{k \times k}$ as a minor. So far this is the best lower bound known for the function f above. In particular it is open whether $f(k)$ above can be bounded by a polynomial. In [12] Reed and Wood consider a different type of obstructions to small tree-width, called *grid-like minors*. A grid-like minor of order l in a graph G is a set \mathcal{P} of paths in G such that the intersection graph $\mathcal{I}(\mathcal{P})$ contains a K_l -minor, where K_l denotes the complete graph on l vertices. Here, the *intersection graph* of a set \mathcal{P} of paths is the graph with vertex set \mathcal{P} and an edge between two paths $P, Q \in \mathcal{P}$ if $P \cap Q \neq \emptyset$. If \mathcal{P}, \mathcal{Q} are sets of paths in G , we write $\mathcal{I}(\mathcal{P}, \mathcal{Q})$ for $\mathcal{I}(\mathcal{P} \cup \mathcal{Q})$, the intersection graph of their disjoint union.

Theorem 3.2 (Reed, Wood [12]). *Every graph of tree-width at least $ck^4 \sqrt{\log k}$ contains a grid-like minor of order k , for some constant c . Conversely, every graph that contains a grid-like minor of order l has tree-width at least $\lceil \frac{l}{2} \rceil - 1$.*

While I do not yet know how to use this result directly, we can use its proof to find the structures in G we need.

Definition 3.3. A *pseudo-wall* of order l in G is a pair $(\mathcal{P}, \mathcal{Q})$ of sets of disjoint paths in G such that $\mathcal{I}(\mathcal{P}, \mathcal{Q})$ is a wall of order l .

We will see below that every graph of large enough tree-width contains a large pseudo-wall and that these can be defined in MSO_2 . Essentially, to show that MSO_2 model-checking is fixed-parameter intractable on a class \mathcal{C} of large enough tree-width, we will use pseudo-walls in a similar way as walls are used in Lemma 2.1. In particular, we want to label the top-most row of the pseudo-wall by a word w over a finite alphabet. However, pseudo-walls do not occur as subgraphs of the graphs G , which makes labelling them somewhat more difficult. Instead, we have to colour the graph G so that this colouring induces the labelling of the pseudo-wall it contains. The main difficulty is that the colouring of G must induce a unique labelling of the pseudo-wall and that both the pseudo-wall as well as its labelling can be defined inside G by MSO_2 -formulas. Unfortunately, this makes the definition of a coloured pseudo-wall technically somewhat more complicated. Let Σ be a set of colours and let B be an additional colour for edges and R an additional colour for vertices. Let $\Gamma := \{B, R\} \cup \Sigma$.

A Σ -coloured pseudo-wall of order l in a Γ -coloured graph G is a triple $(\mathcal{P}, \mathcal{Q}, A)$ such that one of the following holds:

Simple pseudo-walls. $\mathcal{I}(\mathcal{P}, \mathcal{Q})$ is a 1-subdivision of an elementary wall W of order l such that the vertices of W (which we called *nails* above) are exactly the paths in \mathcal{P} . See Figure 2 for an illustration. Figure 2 a) shows the pseudo-wall, where the solid black circles are the vertices from \mathcal{P} and squares denote the vertices from \mathcal{Q} . Figure 2 b) shows how (a part of) this pseudo-wall corresponds to paths in G , where dashed lines represent paths in \mathcal{Q} and solid lines paths in \mathcal{P} . Note, though, that in general the paths could intersect in much more complicated ways than displayed and that paths can intersect more than 3 other paths although walls have maximal degree 3.

Let $\mathcal{P} := \{P_1, \dots, P_k\}$ be such that $P_1 \dots P_l$ form the nails of the top-most row of W in order from left to right. Recall that each P_i is a path in G . Then A is the path

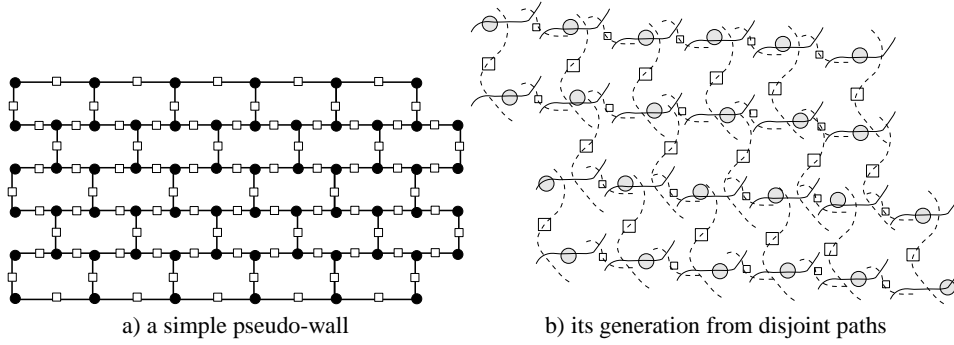


Fig. 2. A simple pseudo-wall and the paths \mathcal{P} (solid) and \mathcal{Q} (dashed) generating it.

in G obtained from the “concatenation” $P_1 \cdot P_2 \cdots P_k$, i.e. $V(A) := \bigcup_{1 \leq i \leq k} V(P_i)$ and $E(A)$ consists of $\bigcup_{1 \leq i \leq k} E(P_i)$ together with additional edges connecting one endpoint of P_i to an endpoint of P_{i+1} , for $1 \leq i < k$, so that this results in a path.

Furthermore, the edges in $E(A)$ are coloured by colour $B \in \Gamma$. The two endpoints of each P_i are coloured by R and the vertices in the paths P_1, \dots, P_l carry colours from Σ so that all vertices in a path P_i are coloured by the same colour from Σ .

This colouring of G induces a labelling of the wall of order l where the nails v_1, \dots, v_l in the top-most row are labelled so that v_i is labelled by the colour $C_i \in \Sigma$ of the path P_i . If $w := C_1 \cdots C_l$ is the sequence of colours on P we say that $(\mathcal{P}, \mathcal{Q}, A)$ encodes the word $w \in \Sigma^*$.

The motivation behind simple pseudo-walls is as follows. If we find this structure in a graph G then the path A tells us what the top-most row of the wall is and it also gives us an order on the vertices of the top-most row. Colouring A by B will enable us to define this coloured pseudo-wall in MSO_2 . If we want to encode a word $w := w_1, \dots, w_l \in \Sigma^*$ in the wall then we can simply label the paths P_1, \dots, P_l in G which form the top-most row of the wall by w_1, \dots, w_l and this induces the correct labelling of the wall $\mathcal{I}(\mathcal{Q}, \mathcal{P})$.

Complex pseudo-walls. Complex walls are structures as illustrated in Figure 3. Essentially, they consist of a subdivision W' of a wall W in $\mathcal{I}(\mathcal{P}, \mathcal{Q})$. To define the colouring of the wall, there will be additional paths in $\mathcal{I}(\mathcal{P}, \mathcal{Q})$ connecting some of the vertices of the top-most row of W' to the path A so that the order is preserved, i.e. the paths do not “cross”. We can then colour the path A and thereby induce a colouring of the top-most row.

Formally, for complex coloured pseudo-walls, A is a path in G such that each $U \in \mathcal{P}$ has exactly one endpoint in A and no path in \mathcal{Q} has an endpoint in A . Furthermore, there are subsets $\mathcal{P}' \subseteq \mathcal{P}$ and $\mathcal{Q}' \subseteq \mathcal{Q}$ such that $\mathcal{I}' := \mathcal{I}(\mathcal{P}', \mathcal{Q}')$ is a wall of order l .

Let $T \subseteq \mathcal{I}'$ be the top-most row of the wall and let $x_1 \dots x_k$ be the vertices of T in order from left to right. Let $I := \{i_1, \dots, i_l\}$ be the index set such that x_{i_1} is the top-left corner, x_{i_l} is the top-right corner and $(x_{i_j})_{1 < j < l}$ lists the vertices in T of degree 3 in order from left to right. For $1 < s < t \leq l$ let $T(s, t)$ be the segment of T between x_{i_s} and x_{i_t} including the latter but not the former. We define $T[0, 1]$ to be the segment

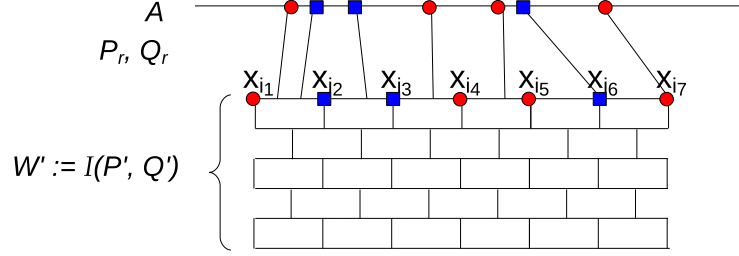


Fig. 3. A complex pseudo-wall.

containing the vertices x_{i_1}, \dots, x_{i_l} . Now, the sets $\mathcal{P}_r := \mathcal{P} \setminus \mathcal{P}'$ and $\mathcal{Q}_r := \mathcal{Q} \setminus \mathcal{Q}'$ induce disjoint paths $P_1 \dots P_l$ in $\mathcal{I} := \mathcal{I}(\mathcal{P}, \mathcal{Q})$ (i.e. each P_i consists of a set of paths in G) such that 1) one endpoint of each path P_i in \mathcal{I} is incident to a vertex x_i of the top-most row of the wall so that each $T(s, t)$, for $1 < s < t \leq l$, contains exactly one x_i and $T[0, 1]$ contains 2 and 2) for the other endpoint u_i of P_i in $\mathcal{I}(\mathcal{P}, \mathcal{Q})$ (which is a path in G) we have $\{v_i\} = u_i \cap A$, where $v_i \in V(G)$, and 3) v_1, \dots, v_l occur in this order on A .

Now suppose $v_1 \dots v_l$ are coloured by $C_1 \dots C_l$ respectively. Then this colouring induces the labelling of $\mathcal{I}(\mathcal{P}', \mathcal{Q}')$ where x_{i_s} gets colour C_s , $1 \leq s \leq l$. We say that $(\mathcal{P}, \mathcal{Q}, A)$ encodes the word $w := C_1 \dots C_l$.

A crucial feature of pseudo-walls in coloured graphs is that they are unique in the sense that if G is a graph coloured by $\{B, R\} \dot{\cup} \Sigma$, then every pseudo-wall $(\mathcal{P}, \mathcal{Q}, A)$ in G encodes the same word w (there may be no coloured pseudo-wall in G).

This is obvious for simple pseudo-walls, as the path A is uniquely determined by its colouring (B -edges and the P_i 's separated by R -vertices) and this uniquely determines the colouring of the wall and hence the encoded word. For complex walls, the path A is again determined by its colouring and this fixes the order of the colours occurring on A and hence on the wall. Here we use that the paths connecting A to the wall preserve the order.

Definition 3.4. We say that a graph G encodes $w \in \Sigma^*$ if it contains a Σ -coloured pseudo-wall encoding w . We say that G encodes w with power k , for some $k \geq 1$, if G contains a Σ -coloured pseudo-wall of order $|w|^k$ encoding w .

The proof of the next theorem is essentially the proof of Theorem 3.2 with some modifications to get coloured pseudo-walls instead of grid-like minors.

Theorem 3.5. *There is a constant c such that if G is a graph of tree-width at least $c \cdot m^8 \cdot \sqrt{\log(m^2)}$, then G contains a Σ -coloured pseudo-wall of order m .*

We can now give a formal definition of *constructible* classes of graphs.

Definition 3.6. Let \mathcal{C} be a class of graphs closed under Γ -colourings. \mathcal{C} is called *constructible* if in every graph $G \in \mathcal{C}$ of tree-width at least $c \cdot m^8 \cdot \sqrt{\log(m^2)}$, where c is from Theorem 3.5, we can compute in polynomial time a coloured pseudo-wall of order m .

It is conceivable that the large pseudo-walls whose existence we proved above can always be computed in polynomial time. This would imply that all classes of graphs are constructible. We leave this for future research.

Theorem 3.5 shows that in any graph of sufficiently large tree-width we find a large pseudo-wall. We will show below that this is enough to define large walls in graphs of large tree-width by means of MSO_2 -formulas. It follows from Theorem 3.5 above that if \mathcal{C} is a class of graphs of unbounded tree-width which is closed under colouring then for each $w \in \Sigma^*$, \mathcal{C} contains a graph encoding w . In fact, for each $c \geq 1$, G contains a graph encoding w with power c . The following lemma summarises what we will need about colourings in the following sections.

Lemma 3.7. *Let \mathcal{C} be a class of graphs closed under Γ -colourings and let $w \in \Sigma^*$ be a word of length m . If there is a graph $G \in \mathcal{C}$ of tree-width $c \cdot (m^k)^8 \cdot \sqrt{\log(m^k)^2}$, where c is the constant from Theorem 3.5, whose tree-width is not bounded by $\log^{8k} |G|$ then there is a graph $G \in \mathcal{C}$ encoding w with power k such that $|G| < 2^{c' \cdot m^{\frac{1}{y}}}$, for some constants $y > 1$ and $c' := c(k)$ depending on k but not on w .*

4 Defining coloured pseudo-walls in graphs of large tree-width

In this section we aim at defining Σ -coloured pseudo-walls in graphs of large enough tree-width in MSO_2 . Fix a set Σ of colours and let $\Gamma := \Sigma \dot{\cup} \{B, R\}$ be as defined in Section 3. Let G be a Γ -coloured graph and $\mathcal{P}, \mathcal{Q}, A \subseteq E(G)$ be sets of edges. For $(\mathcal{P}, \mathcal{Q}, A)$ to be a Σ -coloured pseudo-wall in G , we first need to say that \mathcal{P} and \mathcal{Q} are sets of pairwise disjoint paths in G . Note that \mathcal{P} induces a set of pairwise disjoint paths if, and only if, *i*) every vertex $v \in G$ is incident to at most two edges in \mathcal{P} and *ii*) the subgraph of G induced by the edges in \mathcal{P} is acyclic. This can easily be defined in MSO_2 and we will see the formulas below for the more complicated case of paths and acyclicity in $\mathcal{I}(\mathcal{P}, \mathcal{Q})$. Furthermore, we have to say that the edges of A , and only those, are coloured by B . Now, we have to distinguish between simple and complex coloured pseudo-walls. This can easily be done in MSO_2 as in the first case $\bigcup_{P \in \mathcal{P}} P \subseteq A$ (at least in the pseudo-walls generated in the previous section, in general pseudo-walls this is only true for the paths in the top-most row, but that could equally be used to distinguish the two types of walls) whereas this fails in the second. We will present the case for simple pseudo-walls explicitly. The other case follows using the same ideas.

We first need a few auxiliary formulas. To ease the presentation we assume that no path P occurs in both \mathcal{P} and \mathcal{Q} . This is guaranteed by the pseudo-walls generated in Section 3 but we could also easily modify the formulas below to avoid this assumption (see also Section 5).

In what follows we will use MSO_2 -formulas, interpreted in G , to speak about the intersection graph $\mathcal{I} := \mathcal{I}(\mathcal{P}, \mathcal{Q})$. To increase readability of formulas we agree on the following convention: lower case letters are used for first-order variables, variables P, Q, \dots range over sets of edges and variables E, F, H range over sets of vertices. It may seem bizarre to use F, H for a set of vertices. The reason will become clear below as we will be using variables E for sets of vertices in G which represent sets of edges in \mathcal{I} . As a final piece of notation, we write “ $P \in \mathcal{P}$ ” to say that P is a component of

\mathcal{P} , i.e. one of the paths in \mathcal{P} , and analogously for $Q \in \mathcal{Q}$. Furthermore, we will write $x \in V(P)$ for the formula $\exists y Pxy$ to say that x is adjacent to an edge in P .

Recall that for two paths $P \in \mathcal{P}$ and $Q \in \mathcal{Q}$ there is an edge $\{P, Q\} \in E(\mathcal{I})$ if $P \cap Q \neq \emptyset$ in G . As \mathcal{P} and \mathcal{Q} are sets of disjoint paths, there are no three distinct paths in $\mathcal{P} \cup \mathcal{Q}$ intersecting in a single vertex. Hence, we can represent edges $\{P, Q\} \in E(\mathcal{I})$ by a vertex $v \in V(P \cap Q)$. However, in MSO_2 we cannot pick a single vertex from $V(P \cap Q)$ and therefore will represent the edge $\{P, Q\}$ by the set $V(P \cap Q)$. Let $\varphi_E(x) := \exists P \in \mathcal{P} \exists Q \in \mathcal{Q} x \in V(P \cap Q)$, $\text{inc}(x, P) := x \in V(P)$ and $x \sim y := \exists P \in \mathcal{P} \exists Q \in \mathcal{Q} x, y \in V(P \cap Q)$ be MSO_2 -formulas, where we will usually write $\sim(x, y)$ in infix notation. \sim defines an equivalence relation on the set of vertices satisfying $\varphi_E(x)$ and we can represent edges in \mathcal{I} by equivalence classes of \sim in G . Hence, \mathcal{I} is isomorphic to the graph $\mathcal{J} := (V, E, \sigma)$ with vertex set $V := \mathcal{P} \cup \mathcal{Q}$ and edge set $E := \{[x]_{\sim} : x \in \varphi_E(G)\}$, where a vertex $P \in V$ is incident with an edge $e \in E$ if there is a vertex $v \in e \cap P$ (and hence $e \subseteq P$). \mathcal{J} is MSO_2 -definable in G , by the formulas φ_E , inc and \sim with parameters \mathcal{P} , \mathcal{Q} and we can represent variables over elements of \mathcal{J} by variables ranging over sets of edges in G by enforcing that these are interpreted by a path from either \mathcal{P} or \mathcal{Q} . Variables X over sets of elements of \mathcal{J} can be represented in G by pairs X_P, X_Q of variables ranging over sets of edges so that a set $X \subseteq V(\mathcal{J})$ is represented by the pair of sets $X_P := X \cap \mathcal{P}$ and $X_Q := X \cap \mathcal{Q}$. Finally, sets $F \subseteq E(\mathcal{J})$ of edges can be represented by sets $F' \subseteq \varphi_E(G)$ closed under \sim so that if $\{P, Q\} \in F$ then $V(P \cap Q) \subseteq F'$. Using this idea we can then say about \mathcal{J} , and hence about \mathcal{I} , that \mathcal{J} is a wall as follows: 1) There are two sets $\mathcal{H}, \mathcal{V} \subseteq E(\mathcal{I})$ of edges, each of which induces a set of pairwise vertex disjoint paths in \mathcal{I} (which we will think of as horizontal and vertical paths in a wall). 2) For all $P \in \mathcal{H}$ and $Q \in \mathcal{V}$, $P \cap Q$ is connected and $V(P \cap Q) \cap V(H) = \emptyset$ for all $H \in (\mathcal{V} \cup \mathcal{H}) \setminus \{P, Q\}$. 3) There is a path $L \in \mathcal{V}$ such that the intersection of L with each $Q \in \mathcal{H}$ contains an endpoint of Q (we think of L as the left-most vertical path in the wall). Once we have L , we can give the horizontal paths $P \in \mathcal{H}$ a direction, where we say that $p \in V(P)$ is to the *left* of $p' \in V(P)$, if the subpath of P containing p' and a vertex in L also contains p . 4) There is a path $T \in \mathcal{H}$ such that the intersection of T with each $P \in \mathcal{V}$ contains an endpoint of P (T is the top-most horizontal path in the wall). We can now use T to give the vertical paths $P \in \mathcal{V}$ a direction and say that $p \in V(P)$ is *above* $p' \in V(P)$, if the subpath of P containing p' and a vertex in T also contains p . 5) For each path $P \in \mathcal{V}$ except L there is a path $P' \in \mathcal{V}$ (the path immediately to the left of P) such that for all $Q \in \mathcal{H}$: if $p \in V(P \cap Q)$ and $p' \in V(P' \cap Q)$ are vertices in the intersection of Q and P, P' resp., then p' is to the left of p in Q and there is no $S \in \mathcal{H}$ such that any $s \in V(S \cap Q)$ lies in the subpath of Q between p and p' . The analogue condition for horizontal paths.

To demonstrate the idea of the MSO_2 -formalisation we give precise formulas for the set \mathcal{H} in Condition 1. It will be clear that the other conditions can be formalised analogously. We have to say that there is a set $\mathcal{H} \subseteq E(\mathcal{J})$ of edges inducing a set of pairwise disjoint paths in \mathcal{J} . To define this in G , we first need a formula $\text{Path}(P, Q, H)$ saying that there is a path from $P \in \mathcal{P} \cup \mathcal{Q}$ to $Q \in \mathcal{P} \cup \mathcal{Q}$ using only edges from H , where H is a subset of $\varphi_E(G)$, closed under \sim , representing edges in \mathcal{J} . The usual way of expressing that two vertices x, y in a graph are connected within a set H of edges is to

say that all sets U of vertices which contain x and are closed under the edge relation H also contain y . In our case, sets of vertices of \mathcal{J} are represented by pairs of sets $U_P \subseteq \mathcal{P}$ and $U_Q \subseteq \mathcal{Q}$ consisting of connected components of \mathcal{P} and \mathcal{Q} . Hence, the idea above is expressed by the formula $\text{Path}(P, Q, H)$ defined as

$$\forall U_P \subseteq \mathcal{P} \forall U_Q \subseteq \mathcal{Q} \left((P \in U_P \cup U_Q \wedge \forall X, Y \in \mathcal{P} \cup \mathcal{Q} [X \in U_P \cup U_Q \wedge \exists e (e \in H \wedge \text{inc}(e, X) \wedge \text{inc}(e, Y)) \rightarrow Y \in U_P \cup U_Q]) \rightarrow Q \in U_P \cup U_Q \right)$$

where we write $X \in U_P \cup U_Q$ to say that X is a component either of U_P or U_Q and $U_P \subseteq \mathcal{P}$ to say that U_P is a set of components of \mathcal{P} . Now, we can say that \mathcal{H} induces a set of pairwise disjoint paths as follows. We first say that every vertex in \mathcal{H} has degree at most 2: $\forall P \in \mathcal{P} \cup \mathcal{Q} (\exists \leq 2 f \in \mathcal{H} \text{ inc}(f, P))$, where $\exists \leq 2 f \dots$ is an abbreviation for: there are at most 2 edges f such that To say that \mathcal{H} induces an acyclic graph we say that for all $P \in \mathcal{P} \cup \mathcal{Q}$, if P is incident to an edge $e := \{P, Q\} \in \mathcal{H}$ then there is no path from P to Q in $\mathcal{H} - e$. The latter can be expressed using the formula Path above. This concludes the formalisation of Condition 1).

Clearly, if \mathcal{V} and \mathcal{H} satisfy the conditions above, then they generate a wall in \mathcal{J} and conversely, the disjoint horizontal and vertical paths of a wall satisfy the conditions. Hence, \mathcal{J} is a wall if such \mathcal{V} and \mathcal{H} exist containing all vertices and edges of \mathcal{J} . Formalising all this gives us a formula which says of \mathcal{P}, \mathcal{Q} that the pair $(\mathcal{P}, \mathcal{Q})$ is a pseudo-wall. Note that so far we have not used the additional path A . Hence, if we are not interested in coloured pseudo-walls but simply in pseudo-walls we can use this formula.

We now proceed to define coloured walls and the induced colouring of $\mathcal{I}(\mathcal{P}, \mathcal{Q})$. From the formalisation above we now have sets \mathcal{H}, \mathcal{V} containing the horizontal and vertical paths of the wall as well as two paths L, T giving the top-most row and left-most column. The left-most row gives us an ordering on the top-most row and all we have to do is to define the colours of the vertices on the top-most row from the additional path A , which is easily done. Hence, we can write formulas $\varphi_C(P)$, for $C \in \Sigma$ which are true for the vertices in the wall coloured by C . Complex pseudo-walls can be defined analogously. Taken together, we have a formula $\varphi_U(\mathcal{P}, \mathcal{Q}, A)$ which says that $(\mathcal{P}, \mathcal{Q}, A)$ is a coloured pseudo-wall. Here, the sets \mathcal{P} and \mathcal{Q} define the vertices of the pseudo-wall whereas A is an additional parameter used in the formulas. It will be convenient to take the sets T, L defining the top- and left-most row and column as parameters also rather than defining them. Hence, we have a formula $\varphi_U(\mathcal{P}, \mathcal{Q}, A, L, T)$ which says that $(\mathcal{P}, \mathcal{Q}, A)$ is a Σ -coloured pseudo-wall with left-most column L and top-most row T , formulas $\varphi_E(x, \mathcal{P}, \mathcal{Q}, A, L, T)$, $\text{inc}(x, P, \mathcal{P}, \mathcal{Q}, A, L, T)$ and $\sim(x, y, \mathcal{P}, \mathcal{Q}, A, L, T)$ defining the edge relation of the pseudo-wall and formulas $\varphi_B(x, \mathcal{P}, \mathcal{Q}, A, L, T)$ and $\varphi_C(P, \mathcal{P}, \mathcal{Q}, A, L, T)$, where $C \in \Sigma \cup \{R\}$, defining the colours.

All formulas together define, in graphs of large enough tree-width coloured properly, a large wall whose top-most row is labelled by a word over Σ . Hence, if \mathcal{C} is a class of graphs of unbounded tree-width, closed under colourings, we can define arbitrarily large coloured walls in \mathcal{C} . We know already that (presumably) MSO_2 -model checking is not fixed-parameter tractable on the class of coloured walls. To prove the main result of this paper we need a way to translate MSO_2 -formulas φ over walls to MSO_2 -formulas φ^* over the graphs in which we define the walls. We could do this in an ad-hoc way

and modify the formulas $\varphi_U \dots$ for each given formula φ . We find it more convenient, though, to treat these modifications uniformly within the framework of interpretations. In the next section we therefore introduce a new form of interpretations which simplifies dealing with the intersection graphs we have to define and which might also be useful elsewhere.

5 MSO₂ – MSO₂-transductions

In this section we introduce a class of interpretations, called MSO₂–MSO₂-*transductions*, between classes of graphs which allow us to define one class \mathcal{B} of graphs inside another class \mathcal{C} so that we can translate MSO₂-formulas over \mathcal{B} to MSO₂-formulas saying the same over the graphs in \mathcal{C} . Unlike first-order interpretations, MSO₂–MSO₂-transductions associate with every structure a class of structures and in this sense resemble MSO-transductions as, e.g., studied by Courcelle. Let $\sigma := \{E, B_1, \dots, B_t, C_1, \dots, C_s\}$ be a signature of coloured graphs as defined in Section 2. Let τ be a signature.

Definition 5.1 (MSO₂–MSO₂-transduction). Let $\overline{U} := U_1, \dots, U_k$ and $\overline{X} := X_1, \dots, X_l$ be tuples of binary relation symbols. An MSO₂ – MSO₂-transduction of σ in τ with parameters $\overline{U}, \overline{X}$ is a tuple $\Theta := (\varphi_U(U_1, \dots, U_k, \overline{X}), (\varphi_E^{i,j}(x), \text{inc}_E^{i,j}(x, P, Q), \sim^{i,j}(x))_{1 \leq i < j \leq k}, (\varphi_F^{i,j}(x))_{1 \leq i < j \leq k, F \in \{B_1, \dots, B_t\}}, (\varphi_C(P)^i)_{C \in \sigma, 1 \leq i \leq k})$, where P, Q are unary second-order variables and x is a first-order variable, such that for all τ -structures A and sets $\overline{U}, \overline{X} \subseteq E(A)$ with $(A, \overline{U}, \overline{X}) \models \varphi_U$:

- $\sim^{i,j}$ defines an equivalence relation on $\varphi_E^{i,j}(A)$
- for all $x \in V(A)$ and $1 \leq i < j \leq k$, if $(A, \overline{U}, \overline{X}) \models \varphi_E^{i,j}(x)$ then there are exactly two sets $P_i \subseteq U_i$ and $P_j \subseteq U_j$ such that $(A, \overline{U}, \overline{X}) \models \text{inc}_E^{i,j}(x, P_i, P_j)$ and if $(A, \overline{U}, \overline{X}) \models x \sim^{i,j} y$ then $(A, \overline{U}, \overline{X}) \models \text{inc}_E^{i,j}(y, P_i, P_j)$
- for all $F \in \{B_1, \dots, B_t\}$, $\varphi_F(A) \subseteq \varphi_E(A)$.

We abbreviate MSO₂ – MSO₂-transductions of σ in τ as σ - τ -transductions. Let Θ be a σ - τ -transduction. To every τ -structure A , Θ associates a class $\Theta(A)$ of σ -structures defined as follows. If $U_1, \dots, U_k, X_1, \dots, X_l \subseteq E(A)$ are sets of edges such that $(A, \overline{U}, \overline{X}) \models \varphi_U$, then we define the structure $B := \Theta(A, \overline{U}, \overline{X})$ as follows:

- $V(B) := \dot{\bigcup}_{1 \leq i \leq k} V_i$ where $V_i := \{V \subseteq U_i : V \text{ is a connected component of } U_i\}$
- $E(B) := \dot{\bigcup}_{1 \leq i < j \leq k} E^{i,j}$ where $E^{i,j} := \{[v]_{\sim^{i,j}} : v \in \varphi_E^{i,j}(A)\}$ and the $E^{i,j}$ are taken to be disjoint.
- an edge $e \in E^{i,j}$ is incident to vertices $P \in V_i$ and $Q \in V_j$ if $A \models \text{inc}_E^{i,j}(e, P, Q)$ for some $Q \in V_j$ and likewise for $P \in V_i$.
- an edge $e \in E^{i,j}$, for $1 \leq i < j \leq k$, is coloured by F , where $F \in \sigma$ is binary, if $A \models \varphi_F^{i,j}(e)$.
- a vertex $P \in V_i$ is coloured by $C \in \sigma$, where C is unary, if $(A, \overline{U}, \overline{X}) \models \varphi_C^i(P)$.

Hence, with every structure A and satisfying assignment $U_1, \dots, U_k, \overline{X}$ of φ_U the transduction Θ associates structures whose universes consist of the connected components of the U_i . For classes \mathcal{A} of τ -structures we define $\Theta(\mathcal{A}) := \{B : B \in \Theta(A) \text{ for } A \in \mathcal{A}\}$.

some $A \in \mathcal{A}$. The definition of the edge relation may seem to be overly complicated, as we define the edges and their incidence by different formulas and furthermore do it separately for each pair i, j . The reason is that we want to use MSO_2 -formulas over the structures $\Theta(A)$ and hence have to be able to quantify over sets of edges in $B \in \Theta(A)$. As MSO_2 does not allow quantification over arbitrary binary relations, we have to encode edges by individual elements of A and then use sets over vertices to encode sets of edges.

As all interpretations, MSO_2 - MSO_2 -transductions define a way of transforming one class of structures into another and on the other hand, provide a translation of MSO_2 -formulas φ over σ -structures into MSO_2 -formulas φ^* over τ -structures so that if φ is a formula with free variables $F_1, \dots, F_l, X_1, \dots, X_s, y_1, \dots, y_r$, where the F_i 's are binary, the X_i 's are unary and the y_i 's are individual variables, then φ^* is a formula with free variables $(F_i)_1^*, \dots, (F_i)_k^*, (X_i)_1^*, \dots, (X_i)_k^*$, and $(Y_i)_1^*, \dots, (Y_i)_k^*$, where the $(F_i)_j^*$'s are binary and all other unary. In addition, the parameters $\overline{U}, \overline{X}$ of the transduction occur free in φ^* . We refer to the full version for details.

Lemma 5.2 (interpretation lemma). *Let A be a τ -structure and $\overline{U}, \overline{X} \subseteq E(A)$ be such that $(A, \overline{U}, \overline{X}) \models \varphi_U$. Let $B := \Theta(A, \overline{U}, \overline{X})$. For all $\varphi \in \text{MSO}_2[\sigma]$, $(A, \overline{U}, \overline{X}) \models \Theta(\varphi)$ if, and only if, $B \models \varphi$.*

Corollary 5.3. *Let $\varphi \in \text{MSO}_2[\sigma]$ and $\psi := \exists \overline{U} \exists \overline{X} \varphi^* \in \text{MSO}_2[\tau]$. For all τ -structures A , $A \models \psi$ iff there is a $B \in \Theta(A)$ such that $B \models \varphi$.*

6 Putting it all together

In this section we prove Theorem 1.2. Let $\Sigma := \{C_1, \dots, C_l\}$, with $l \geq 2$, be a set of colours and $\Gamma := \Sigma \dot{\cup} \{B, R\}$, where B is a binary and R a unary relation symbol. Let \mathcal{C} be a constructible class of Γ -coloured graphs closed under colourings such that the tree-width of \mathcal{C} is strongly unbounded by $\log^{\delta k} n$, for some $k \geq 1$. We first observe that the formulas $\varphi_U(\mathcal{P}, \mathcal{Q}, A, L, T)$, $\varphi_E, \text{inc}, \sim, \varphi_B, \varphi_C$ as constructed in Section 4 can be used to define an MSO_2 - MSO_2 -transduction Θ such that $\Theta(\mathcal{C})$ is the class of coloured walls in graphs $G \in \mathcal{C}$. Here, we take $\overline{U} := \mathcal{P}, \mathcal{Q}$ as the parameters defining the vertex set of the resulting graphs and $\overline{X} := A, L, T$ as additional parameters used in the transduction.

By Lemma 3.7, $\Theta(\mathcal{C})$ contains for each $w \in \Sigma^*$ a wall encoding w with power k , i.e. there is a $(|w|^k \times |w|^k)$ -wall in $\Theta(\mathcal{C})$ whose top-most row is labelled by w from the left. In particular, as SAT can be solved in time quadratic in the size of the input by a non-deterministic Turing-machine, if $k \geq 2$ then for each CNF formula w of length m , $\Theta(\mathcal{C})$ contains a wall of size $m^2 \times m^2$ labelled by w .

Now take a formula φ_{CNF} which, on a wall W encoding w , checks whether w correctly encodes a CNF-formula and whether the order of W is at least $|w|^2$. This can be done by simulating a non-deterministic Turing machine doing this test. Let $\psi_{\text{CNF}} := \exists \mathcal{P} \mathcal{Q} \text{ALT}(\varphi_U \wedge \Theta(\varphi_{\text{CNF}}))$ and let $\mathcal{C}_{\text{CNF}} := \{A \in \mathcal{C} : A \models \psi_{\text{CNF}}\} \subseteq \mathcal{C}$. By the interpretation Lemma 5.2, \mathcal{C}_{CNF} contains for each CNF-formula w a graph $G \in \mathcal{C}$ encoding w with power 2 and conversely each graph $G \in \mathcal{C}_{\text{CNF}}$ encodes a CNF-formula with power 2.

Now, let φ be the MSO_2 -sentence from Section 2 which, by simulating an appropriate Turing-machine, is true in a wall of order $|w|^2$ encoding a CNF-formula w if,

and only if, w is satisfiable and let $\vartheta := \Theta(\varphi)$. It follows that if the tree-width of \mathcal{C} is strongly unbounded by $\log^{16} n$, then model-checking $\vartheta := \Theta(\varphi)$ in \mathcal{C}_{CNF} is equivalent to solving SAT. If in addition \mathcal{C} is constructible then this allows us to formally define a subexponential time reduction from SAT to \mathcal{C} as follows. Given a CNF-formula w , we construct a graph $G \in \mathcal{C}$ such that G encodes w with power 2 and $|G| < 2^{c \cdot |w|^{\frac{1}{y}}}$, for some $y > 1$ and $c > 0$. By definition of constructibility and strongly unboundedness, such a graph G exists in \mathcal{C} and can be constructed in time $|G|^r$, for some fixed $r > 0$, and hence in time $< (2^{|w|^{\frac{1}{y}}})^r = 2^{r \cdot |w|^{\frac{1}{y}}}$. Now suppose $\text{MC}(\text{MSO}_2, \mathcal{C})$ was in XP, i.e. for some computable function f , given a graph $G \in \mathcal{C}$ and $\varphi \in \text{MSO}_2$, $G \models \varphi$ could be decided in time $|G|^{f(|\varphi|)}$. Hence, we could decide if $G \models \vartheta$, where ϑ is the formula defined above, in time $|G|^{f(|\vartheta|)} < 2^{f(|\vartheta|) \cdot |w|^{\frac{1}{y}}}$. Taken together, we could decide if w is satisfiable in time $< 2^{(r+f(|\vartheta|)) \cdot |w|^{\frac{1}{y}}}$, for fixed $r, y > 1$ and a fixed formula ϑ . Hence, SAT would be decidable in sub-exponential time.

The same argument shows that if \mathcal{C} is a rich and constructible class of Γ -coloured graphs closed under colourings whose tree-width is effectively not bounded by $\log^{8 \cdot k} n$ and \mathcal{L} is a problem that can be decided by a non-deterministic Turing-machine in time n^k , then $\text{MC}(\text{MSO}_2, \mathcal{C})$ is not in XP unless \mathcal{L} can be solved in sub-exponential time. This implies Theorem 1.2. The extension to the polynomial time hierarchy follows as we can simulate alternating Turing-machines with bounded number of alternations in MSO_2 in the same way as non-deterministic Turing-machines.

References

1. B. Courcelle. Graph rewriting: An algebraic and logic approach. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume 2, pages 194 – 242. Elsevier, 1990.
2. A. Dawar, M. Grohe, and S. Kreutzer. Locally excluding a minor. In *Logic in Computer Science (LICS)*, pages 270–279, 2007.
3. R. Diestel. *Graph Theory*. Springer-Verlag, 3rd edition, 2005.
4. H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic*. Springer, 1994.
5. J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.
6. Jörg Flum and Martin Grohe. Fixed-parameter tractability, definability, and model checking. *SIAM Journal on Computing*, 31:113 – 145, 2001.
7. M. Frick and M. Grohe. Deciding first-order properties of locally tree-decomposable structures. *Journal of the ACM*, 48:1148 – 1206, 2001.
8. M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified NP-complete problems. In *ACM Symposium on Theory of Computing (STOC)*, pages 47–63, 1974.
9. M. Grohe. Logic, graphs, and algorithms. In T. Wilke, J. Flum, E. Grädel, editor, *Logic and Automata – History and Perspectives*. Amsterdam University Press, 2007.
10. S. Kreutzer. Algorithmic meta-theorems. to appear. See <http://arxiv.org/abs/0902.3616>.
11. L. Libkin. *Elements of Finite Model Theory*. Springer, 2004.
12. Bruce Reed and David Wood. Polynomial treewidth forces a large grid-like minor. unpublished. Available at arXiv:0809.0724v3 [math.CO], 2008.
13. N. Robertson, P. Seymour, and R. Thomas. Quickly excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 1994.
14. N. Robertson and P. D. Seymour. Graph minors V. Excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 41(1):92–114, 1986.
15. M. Vardi. On the complexity of relational query languages. In *Proc. of the 14th Symposium on Theory of Computing (STOC)*, pages 137–146, 1982.