Definability Equals Recognizability for k-Outerplanar Graphs*†

Lars Jaffke[1] and Hans L. Bodlaender[2]

1 CWI Amsterdam
Postbus 94079, 1090 GB Amsterdam, The Netherlands
l.jaffke@cwi.nl

2 Department of Information and Computing Sciences,
Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, The Netherlands
Department of Mathematics and Computer Science;
University of Technology Eindhoven, P.O. Box 513, 5600 MB Eindhoven, The Netherlands
h.l.bodlaender@uu.nl

Abstract

One of the most famous algorithmic meta-theorems states that every graph property that can be defined by a sentence in counting monadic second order logic (CMSOL) can be checked in linear time for graphs of bounded treewidth, which is known as Courcelle's Theorem [6]. These algorithms are constructed as finite state tree automata, and hence every CMSOL-definable graph property is recognizable. Courcelle also conjectured that the converse holds, i.e. every recognizable graph property is definable in CMSOL for graphs of bounded treewidth. We prove this conjecture for k-outerplanar graphs, which are known to have treewidth at most $3k-1$ [2].

1998 ACM Subject Classification F.1.1 [Theory of Computation] Models of Computation – Automata, F.4.1 [Mathematical Logic and Formal Languages] Mathematical Logic – Computational logic, G.2.2 [Discrete Mathematics] Graph Theory – Graph algorithms

Keywords and phrases treewidth, monadic second order logic of graphs, finite state tree automata, k-outerplanar graphs

Digital Object Identifier 10.4230/LIPIcs.IPEC.2015.175

1 Introduction

A seminal result from 1990 by Courcelle states that for every graph property P that can be formulated in a language called counting monadic second order logic (CMSOL), and each fixed k, there is a linear time algorithm that decides P for a graph given a tree decomposition of width at most k [6] (while similar results were discovered by Arnborg et al. [1] and Borie et al. [4]). Counting monadic second order logic generalizes monadic second order logic (MSOL) with a collection of predicates testing the size of sets modulo constants. Courcelle showed that this makes the logic strictly more powerful [6]. The algorithms constructed in Courcelle's proof have the shape of a finite state tree automaton and hence we can say that CMSOL-definable graph properties are recognizable (or, equivalently, regular or finite-state). Courcelle's Theorem generalizes one direction of a classic result in automata theory by

* The research was done while the first author was a student at Utrecht University.
† The research of the second author was partially funded by the Networks programme, funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research.

10th International Symposium on Parameterized and Exact Computation (IPEC 2015).
Editors: Thore Husfeldt and Iyad Kanj; pp. 175–186
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Büchi, which states that a language is recognizable, if and only if it is MSOL-definable [5]. Courcelle conjectured in 1990 that the other direction of Büchi's result can also be generalized for graphs of bounded treewidth in CMSOL, i.e. that each recognizable graph property is CMSOL-definable.

This conjecture is still regarded to be open. Its claimed resolution by Lapoire [18] is not considered to be valid by several experts. In the course of time proofs were given for the classes of trees and forests [6], partial 2-trees [7], partial 3-trees and k-connected partial k-trees [16]. A sketch of a proof for graphs of pathwidth at most k appeared at ICALP 1997 [15]. Very recently, one of the authors proved, in collaboration with Heggernes and Telle, that Courcelle's Conjecture holds for partial k-trees without chordless cycles of length at least ℓ [3].

By the results presented in this paper, we add the class of k-outerplanar graphs to this list. In particular, we first prove the conjecture for 3-connected k-outerplanar graphs and then generalize this result to all k-outerplanar graphs, based on the decomposition of a connected graph into its 3-connected components, discovered by Tutte [19] and shown to be definable in monadic second order logic by Courcelle [10].

The rest of the paper is organized as follows. In Section 2 we give the basic definitions and review the concepts involved in our proofs. We present the main result in Section 3 and conclude in Section 4.

For details of the proofs of the results given in this text, the reader is referred to the full version of the paper [14].

2 Preliminaries

Graphs and Tree Decompositions

Throughout the paper, a graph $G = (V, E)$ with vertex set V and edge set E is undirected, connected and simple. We denote the subgraph relation by $G \sqsubseteq H$ and for a set $W \subseteq V$, $G[W]$ denotes the induced subgraph over W in G, so $G[W] = (W, E \cap (W \times W))$. We call a set $C \subset V$ a *cut* of G, if $G[V \setminus C]$ is disconnected. An *ℓ-cut* of G is a cut of size ℓ. A set $S \subseteq V$ is said to be *incident* to an ℓ-cut C, if $C \subset S$. We call a graph *ℓ-connected*, if it does not contain a cut of size at most $\ell - 1$.

We now define the class of k-outerplanar graphs and some central notions used extensively throughout the rest of the paper.

▶ **Definition 1** ((Planar) Embedding). A drawing of a graph in the plane is called an *embedding*. If no pair of edges in this drawing crosses, then it is called *planar*.

▶ **Definition 2** (k-Outerplanar Graph). Let $G = (V, E)$ be a graph. G is called a *planar graph*, if there exists a planar embedding of G. An embedding of a graph G is 1-*outerplanar*, if it is planar, and all vertices lie on the exterior face. For $k \geq 2$, an embedding of a graph G is *k-outerplanar*, if it is planar, and when all vertices on the outer face are deleted, then one obtains a $(k-1)$-outerplanar embedding of the resulting graph. If G admits a k-outerplanar embedding, then it is called a *k-outerplanar graph.*

▶ **Definition 3** (Fundamental Cycle). Let $G = (V, E)$ be a graph with maximal spanning forest $T = (V, F)$. Given an edge $e = \{v, w\}$, $e \in E \setminus F$, its *fundamental cycle* is the cycle which is formed by the unique path from v to w in F together with the edge e.

▶ **Definition 4** (Tree Decomposition, Treewidth). A *tree decomposition* of a graph $G = (V, E)$ is a pair (T, X) of a tree $T = (N, F)$ and an indexed family of vertex sets $(X_t)_{t \in N}$ (called *bags*), such that the following properties hold.

(i) Each vertex $v \in V$ is contained in at least one bag.
(ii) For each edge $e \in E$ there exists a bag containing both endpoints.
(iii) For each vertex $v \in V$, the bags in the tree decomposition that contain v form a subtree of T.

The *width* of a tree decomposition is the size of the largest bag minus 1 and the *treewidth* of a graph is the minimum width of all its tree decompositions. We might sometimes refer to graphs of treewidth at most k as *partial k-trees.*[1]

To avoid confusion, in the following we will refer to elements of N as *nodes* and elements of V as *vertices*. Sometimes, to shorten the notation, we might not differ between the terms *node* and *bag* in a tree decomposition.

We use the following notation. If P denotes a graph property (e.g. a graph contains a Hamiltonian cycle), then by '$P(G)$' we express that a graph G has property P.

Monadic Second Order Logic of Graphs

We now define counting monadic second order logic of graphs $G = (V, E)$, using terminology from [4] and [16]. Variables in this predicate logic are either single vertices/edges or vertex/edge sets. We form predicates by joining *atomic predicates* (vertex equality $v = w$, vertex membership $v \in V$, edge membership $e \in E$ and vertex-edge incidence $\mathrm{Inc}(v, e)$) via negation $\neg$, conjunction $\wedge$, disjunction $\vee$, implication $\rightarrow$ and equivalence $\leftrightarrow$ together with existential quantification $\exists$ and universal quantification $\forall$ over variables in our domain $V \cup E$. To extend this monadic second order logic (MSOL) to *counting* monadic second order logic (CMSOL), one additionally allows the use of predicates $\mathrm{mod}_{p,q}(S)$ for sets S, which are true, if and only if $|S| \bmod q = p$, for constants p and q (with $p < q$).

Let ϕ denote a predicate without unquantified (so-called *free*) variables constructed as explained above and G be a graph. We call ϕ a *sentence* and denote by $G \models \phi$ that ϕ yields a truth assignment when evaluated with the graph G.

▶ **Definition 5** (Definable Graph Properties)**.** Let P denote a graph property. We say that P is *(C)MSOL-definable*, if there exists a (C)MSOL-sentence ϕ_P such that

$$P(G) \Leftrightarrow G \models \phi_P.$$

We distinguish between two types of free variables. Consider a predicate ϕ with free variables $x_1, \ldots, x_p$. A subset of $x_1, \ldots, x_p$, say $x_1, \ldots, x_a$ (where $a \leq p$), can be considered its *arguments*, and the variables $x_{a+1}, \ldots, x_p$ are its *parameters*. We denote this predicate as $\phi(x_1, \ldots, x_a)$, i.e. its parameters do not appear in the notation. We illustrate the difference between arguments and parameters in the following example.

▶ **Example 6.** Let P denote the property that a graph has a k-coloring and $\phi_{col}(v, w)$ a predicate, which is true, if and only if a vertex v has a lower numbered color than w in a given coloring. Then ϕ_{col} has two arguments, vertices v and w, and k parameters, the k color classes. Clearly, the choice of the parameters influences the evaluation of ϕ_{col}, but in most applications of parameters for predicates, it is sufficient to show that one can guess *some* variables of the evaluation graph to define a property.

[1] For several characterizations of graphs of treewidth at most k, see e.g. [2, Theorem 1]

▶ **Definition 7** ((Existentially) Definable Relations). Let $R(x_1, \dots, x_r)$ denote a relation with arguments $x_1, \dots, x_r$. We say that R is *(C)MSOL-definable*, if there exists a parameter-free predicate $\phi_R(x_1, \dots, x_r)$, encoding the relation R. Furthermore we call R *existentially (C)MSOL-definable*, if there exists a predicate $\phi_R(x_1, \dots, x_r)$ with parameters $x_1, \dots, x_p$, which, after substituting the parameters by fixed values in the evaluation graph, encodes the relation R.

A central concept used in this paper is an implicit representation of tree decompositions in monadic second order logic, as we cannot refer to its bags and edges as variables in MSOL directly. We have to define predicates, which encode the construction of a tree decomposition of each member of a given graph class. We require two types of predicates. The **Bag**-predicates will allow us to verify whether a vertex is contained in some bag and whether any vertex set in the graph constitutes a bag in its tree decomposition. Each bag will be associated with either a vertex or an edge in the underlying graph (its *witness*) together with some *type*, whose definition depends on the graph class under consideration. The **Parent**-predicate allows for identifying edges in the tree decomposition, i.e. for any two vertex sets S_p and S_c, this predicate will be true if and only if both S_p and S_c are bags in the tree decomposition and S_p is the bag corresponding to the parent node of S_c.

▶ **Definition 8** (MSOL-definable tree decomposition). A tree decomposition $(T = (N, F), X)$ of a graph $G = (V, E)$ is called *existentially MSOL-definable*, if the following are existentially MSOL-definable (with parameters $x_1, \dots, x_p$ for some constant p).

(i) Each bag $X_p, p \in N$ in the tree decomposition is associated with either a vertex $v \in V$ or an edge $e \in E$ (called its *witness*) and can be identified by one of the following predicates (where $S \subseteq V$ and s and t are constants).
- **(a)** $\mathbf{Bag}_{\tau_1}(v, S), \dots, \mathbf{Bag}_{\tau_t}(v, S)$: The vertex set S forms a bag in the tree decomposition of G, i.e. $S = X_p$ for some $p \in N$, it is of type τ_i $(1 \le i \le t)$ and its witness is v.
- **(b)** $\mathbf{Bag}_{\sigma_1}(e, S), \dots, \mathbf{Bag}_{\sigma_s}(e, S)$: The vertex set S forms a bag in the tree decomposition of G, i.e. $S = X_p$ for some $p \in N$, it is of type σ_i $(1 \le i \le s)$ and its witness is e.

(ii) Each edge in F can be identified with a predicate $\mathbf{Parent}(S_p, S_c)$, where $S_p, S_c \subseteq V$: The vertex sets S_p and S_c form bags in (T, X), i.e. $S_p = X_p$ and $S_c = X_c$ for some $p, c \in N$, and p is the parent node of c in T.

▶ **Lemma 9.** *Let (T, X) be an existentially MSOL-definable tree decomposition with parameters $x_1, \dots, x_p$. There exists a predicate ϕ with zero parameters and p arguments, which is true if and only if the predicates $\boldsymbol{Bag}_{\tau_1}, \dots, \boldsymbol{Bag}_{\tau_t}$, $\boldsymbol{Bag}_{\sigma_1}, \dots, \boldsymbol{Bag}_{\sigma_s}$ and $\boldsymbol{Parent}$ describe a width-k rooted tree decomposition of an evaluation graph G.*

Proof. The proof can be done analogously to the proof of Lemma 4.7 in [16]. ◀

A fundamental result about definable graph properties, which we use extensively throughout our proofs, states that one can define any edge orientation of partial k-trees in MSOL. For an in-depth study of MSOL-definable edge orientations on graphs, see [9].

▶ **Lemma 10** (Lemma 4.8 in [16]). *Any direction over a subset of the edges of an undirected graph of treewidth at most k is existentially MSOL-definable with $k + 2$ parameters.*

Tree Automata for Graphs of Bounded Treewidth

We briefly review the concept of tree automata and recognizability of graph properties for graphs of bounded treewidth. For an introduction to the topic we refer to [13, Chapter 12]. For the formal details of the following notions, the reader is referred to [16].

A tree automaton $\mathcal{A}$ is a finite state machine accepting as an input a tree structure over an alphabet Σ as opposed to words in classical word automata. Formally, $\mathcal{A}$ is a triple $(\mathcal{Q}, \mathcal{Q}_{Acc}, f)$ of a set of states $\mathcal{Q}$, a set of accepting states $\mathcal{Q}_{Acc} \subseteq \mathcal{Q}$ and a transition function f, deriving the state of a node in the input tree $\mathcal{T}$ from the states of its children and its own symbol $s \in \Sigma$. $\mathcal{T}$ is *accepted* by $\mathcal{A}$, if the state of the root node of $\mathcal{T}$ is an element of the accepting states $\mathcal{Q}_{Acc}$.

To recognize a graph property on graphs of treewidth at most k, one encodes a rooted width-k tree decompositions as a labeled tree over a special type of alphabet, in the following denoted by Σ_k (see Definition 3.5, Proposition 3.6 in [16]). We say that a tree automaton over such an alphabet *processes* width-k tree decompositions.

▶ **Definition 11** (Recognizable Graph Properties). Let P denote a graph property. We call P *recognizable* (for graphs of treewidth at most k), if there exists a tree automaton $\mathcal{A}_P$ processing width-k tree decompositions, such that following are equivalent.
(i) (T, X) is a width-k tree decomposition of a graph G with $P(G)$.
(ii) $\mathcal{A}_P$ accepts (the labeled tree over Σ_k corresponding to) (T, X).

Kaller has shown that Courcelle's Conjecture follows immediately from the construction of an MSOL-definable tree decomposition.

▶ **Lemma 12** (Lemma 5.4 in [16]). *Let P denote a graph property, which is recognizable for graphs of bounded treewidth. Suppose that there is an MSOL-definable tree decomposition of width at most k for any partial k-tree G. Then, one can write a CMSOL-sentence Φ, such that $G \models \Phi$ if and only if $P(G)$.*

3 The Main Result

Bodlaender has shown that every k-outerplanar graph has treewidth at most $3k - 1$ [2, Theorem 83], using the following properties of maximal spanning forests of a graph.

▶ **Definition 13** (Vertex and Edge Remember Number). Let $G = (V, E)$ be a graph with maximal spanning forest $T = (V, F)$. The *vertex remember number* of G (with respect to T), denoted by $vr(G, T)$, is the maximum number over all vertices $v \in V$ of fundamental cycles that use v. Analogously, we define the *edge remember number*, denoted by $er(G, T)$.

In particular, Bodlaender gave a constructive proof that the treewidth of a graph is bounded by at most $\max\{vr(G, T), er(G, T) + 1\}$ [2, Theorem 71]. The idea of the proof is to create a bag for each vertex and edge in the spanning tree, containing the vertex itself (or the two endpoints of the edge, respectively) and one endpoint of each edge, whose fundamental cycle uses the corresponding vertex/edge. The tree structure of the decomposition is inherited by the structure of the spanning tree. He then showed, that in a k-outerplanar graph G one can split the vertices of degree $d > 3$ into a path of $d - 2$ vertices of degree three without increasing the outerplanarity index of G (the so-called *vertex expansion step*). In this expanded graph G' one can find a spanning tree of vertex remember number at most $3k - 1$ and edge remember number at most $2k$ [2, Lemmas 81 and 82]. Using [2, Theorem 71], this yields a tree decomposition of width at most $3k - 1$ for G' and by simple replacements

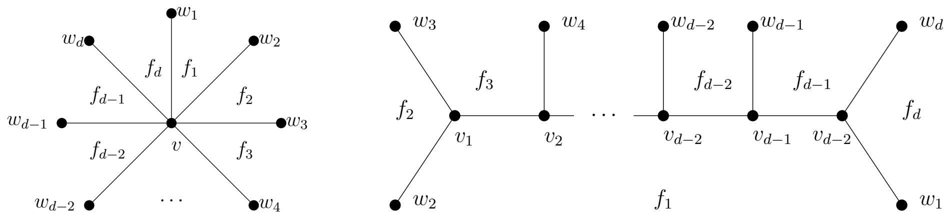

Figure 1 Expanding a vertex v, where f_1 is a layer with lowest layer number.

one finds a tree decomposition for G of the same width. A constructive proof of finding such a spanning tree was given by Katsikarelis [17].

The major challenge of defining such a tree decomposition in MSOL lies in the vertex expansion step, since one cannot use artificially created vertices and edges as variables in MSOL-predicates. We give an implicit representation of this step in Section 3.1. We show how to construct an existentially MSOL-definable tree decomposition of a 3-connected k-outerplanar graph in Section 3.2 and for the general case of k-outerplanar graphs in Section 3.3.

3.1 An Implicit Representation of the Vertex Expansion Step

We first define the partition of the vertex set of a k-outerplanar graph into the layers resulting from repeatedly removing the vertices on the outer face.

▶ **Definition 14** (Stripping Layer of a k-Outerplanar Graph)**.** Let G be a k-outerplanar graph. Removing the vertices on the outer face of an embedding of G is called the *stripping step*. When applied repeatedly, the set of vertices being removed in the i-th stripping step is called the i-th *stripping layer* of G, where $1 \leq i \leq k$.

▶ **Lemma 15.** *Let $G = (V, E)$ be a k-outerplanar graph. The partition of V into the stripping layers of G is existentially MSOL-definable with k parameters.*

To avoid increasing the outerplanarity index of a graph during the expansion of a vertex, we need the notion of *layer numbers*.

▶ **Definition 16** (Layer Number)**.** Let $G = (V, E)$ be a planar graph. The *layer number* of a face is defined in the following way. The outer face gets layer number 0. Then, for each other face, we let the layer number be one higher than the minimum layer number of all its adjacent faces.[2]

The expansion step does not preserve facial adjacency, so in order to not increase the outerplanarity index of the graph, one makes sure that all faces are adjacent to a face with lowest layer number. We illustrate the expansion step of a vertex in Figure 1.

Following the ideas of the proofs given in [2, Section 13], we define another type of remember number.

▶ **Definition 17** (Face Remember Number, Face Remember Set)**.** Let $G = (V, E)$ be a planar graph with a given embedding $\mathcal{E}$ and $T = (V, F)$ a maximal spanning forest of G. The *face*

[2] Unless stated otherwise, we call two faces adjacent, if they share an incident vertex.

remember number of G w.r.t. T, denoted by $fr(G,T)$ is the maximum number of fundamental cycles C of G given T, such that $bd_E(f) \cap E(C) \neq \emptyset$, where $bd_E(f)$ denotes the boundary edges of a face f, over all faces f in $\mathcal{E}$, except the outer face. Given a face $f \in \mathcal{E}$, we call the set of edges, whose fundamental cycles intersect the boundary of f the *face remember set*.

In the following, we denote by G a k-outerplanar graph (before expansion) and by G' the graph obtained by expanding vertices of degree $d > 3$. Consider the vertex v_1 in Figure 1b and let e be an edge whose fundamental cycle C_e uses v_1 in some spanning tree of G'. We observe that C_e intersects with one of the face boundaries of f_1, f_2 or f_3. Since v_1 is a vertex in the expanded graph, we know that in each tree decomposition based on a spanning tree of G' there will be a bag containing one endpoint of each edge, whose fundamental cycle intersects with the face boundary of f_1, f_2 or f_3. Using this observation, we can also show that one can find a tree decomposition of a planar graph, whose width is bounded by the edge and face remember number of one of its spanning trees.

▶ **Lemma 18.** *Let $G = (V,E)$ be a planar graph with spanning tree $T = (V,F)$. The treewidth of G is at most* $\max\{er(G,T)+1, 3 \cdot fr(G,T)\}$.

▶ **Lemma 19.** *Let $G = (V,E)$ be a k-outerplanar graph. There exists a spanning tree $T = (V,F)$ of G with $er(G,T) \leq 2k$ and $fr(G,T) \leq k$.*

Proof. The proof can be done analogously to the proof of Lemma 81 in [2]. ◀

3.2 3-Connected k-Outerplanar Graphs

In the previous section we showed that one can construct a tree decomposition of a k-outerplanar graph G, whose width is bounded by the edge and face remember number of G. We will now use these results to show that the construction of such a tree decomposition is existentially MSOL-definable, if we restrict ourselves to 3-connected k-outerplanar graphs.

A classic result by Whitney states that every 3-connected planar graph has a unique embedding [21] (up to the choice of the outer face). Reconstructing this proof, Diestel has shown that the face boundaries of this embedding can be characterized in strictly combinatorial terms.

▶ **Proposition 20** (Proposition 4.2.7 in [12])**.** *The face boundaries of a 3-connected planar graph are precisely its non-separating induced cycles.*[3]

We immediately find the following.

▶ **Proposition 21.** *The face boundaries of a 3-connected planar graph are MSOL-definable.*

Using these observations, we can define predicates encoding an ordering on the incident edges of each vertex.

▶ **Lemma 22.** *Let $G = (V,E)$ be a 3-connected k-outerplanar graph, $v \in V$ with* $\deg(v) > 3$ *and $e_{\mathcal{A}} \in E$ an incident edge, called its* anchor*. There exists an ordering $nb_<(e,f)$, which mimics a clockwise (or counter-clockwise) traversal (in the unique embedding of G) on all incident edges of v, starting at $e_{\mathcal{A}}$, which is existentially MSOL-definable with two parameters $e_{\mathcal{A}}$ and $e'_{\mathcal{A}} \in E$.*

[3] Let $G = (V,E)$ be a connected graph. A vertex set $W \subseteq V$ is called *non-separating induced cycle*, if $G[W]$ is a cycle and $G[V \setminus W]$ is connected.

Note that one can lead an alternative proof of Lemma 22, using the notion of *rotation systems*, introduced in [11]. Furthermore one can see that the relation $\mathrm{nb}_<(e, f)$ is existentially MSOL-definable for a graph G (and not just a single vertex) by replacing the parameters in the formulation of Lemma 22 with the corresponding edge set equivalents.

The construction of the tree decomposition in the proof of the following lemma can be summarized as follows. Given a k-outerplanar graph $G = (V, E)$, one guesses a directed spanning tree $T = (V, F)$ and then constructs a part of the tree decomposition for each vertex $v \in V$ in the following manner. For each incident edge $e \in E$ of v we create one bag which contains one vertex for each edge, whose fundamental cycle uses e and one endpoint of each edge in the face remember set of one fixed incident face of v with minimum layer number. Additionally, for each edge $f \in F$ in the spanning tree of G one creates a bag which contains one endpoint of each edge whose fundamental cycle uses f. We make two bags adjacent, if they either are created due to the same edge in G or if their corresponding edges $e, f \in E$ are direct neighbors in the ordering $\mathrm{nb}_<(e, f)$. For details of the proof the reader is referred to the full text of the paper [14].

▶ **Lemma 23.** *Let $G = (V, E)$ be a 3-connected k-outerplanar graph. G admits an existentially MSOL-definable tree decomposition of width at most $3k$ and maximum degree 3 with $4k + 3$ parameters.*

3.3 Implications of Hierarchical Graph Decompositions to Courcelle's Conjecture

A *block decomposition* of a connected graph G is a tree decomposition, whose bags contain either the endpoints of a single edge or the vertex set of a maximal 2-connected subgraph[4] of G (called the *blocks* of G) or a cut-vertex of G (called the *cuts*) by making a block-bag adjacent to a cut-bag $\{v\}$ if the block bag contains v (see e.g. Section 2.1 in [12]).

Analogously, Tutte showed that given a 2-connected graph (or a block of a connected graph) one can find a *3-block decomposition* into its *2-cuts* and *3-blocks*, the latter of which are vertex sets of either 3-connected graphs or cycles (but not necessarily subgraphs of G, see below), which can be joined in a tree structure in the same way [19, Chapter 11] [20, Section IV.3]. Courcelle showed that both of these decompositions of a graph are MSOL-definable [10] and also proved that one can find an MSOL-definable tree decomposition of width 2, if all 3-blocks of a graph are cycles [10, Corollary 4.11]. In this section, we will use these methods to prove Courcelle's Conjecture for k-outerplanar graphs by showing that the results of the previous section can be applied to define tree decompositions of 3-connected 3-blocks of a k-outerplanar graph.

We will refer to a block decomposition of a graph G, whose 2-connected blocks are replaced by their 3-block decompositions as the *hierarchical decomposition* of G (cf. [10], for an example see Figure 2). For details of how the bags in the resulting decomposition are connected, the reader is referred to the full text of the paper [14].

▶ **Definition 24** (3-Block)**.** Let $G = (V, E)$ be a 2-connected graph, $\mathcal{S}$ a set of 2-cuts of G and $W \subseteq V$. A graph $H = (W, F)$ is called a *3-block*, if it can be obtained by taking the induced subgraph of W in G and for each incident 2-cut $S = \{x, y\} \in \mathcal{S}$, adding the edge $\{x, y\}$ to F (if it is not already present), plus one of the following holds.

[4] Let $G = (V, E)$ be a graph and $W \subseteq V$. $H = G[W]$ is called a *maximal 2-connected subgraph* of G, if $G[W]$ is 2-connected and for all $W' \supset W$, $G[W']$ is not 2-connected.

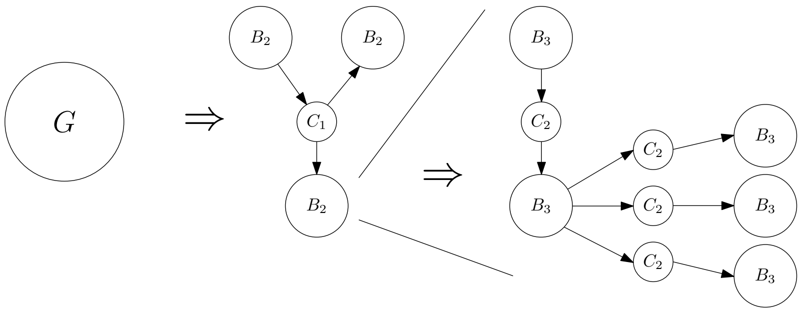

Figure 2 An example hierarchical decomposition of a graph G. A bag labeled C_1 contains a cut-vertex of G, C_2 a 2-cut of G. Bags labeled B_2 contain a 2-block (a single edge or a maximal 2-connected subgraph). If a 2-block contains a maximal 2-connected subgraph of G, it is decomposed further into its 2-cuts and 3-blocks, labeled B_3, which contain either a cycle or a 3-connected 3-block.

(i) H is a cycle of at least three vertices.
(ii) H is a 3-connected graph (referred to as a *3-connected 3-block*).

▶ **Definition 25** (Tutte Decomposition)**.** Let $G = (V, E)$ be a 2-connected graph. A tree decomposition $(T = (N, F), X)$ is called a *Tutte decomposition* of G, if the following hold.
(i) For each $t \in N$, X_t is either a 2-cut (called the *cut bags*) or the vertex set of a 3-block (called the *block bags*).
(ii) Each edge $f \in F$ is incident to precisely one cut bag.
(iii) Each cut bag is adjacent to precisely two block bags.
(iv) Let $t \in N$ denote a cut node with vertex set X_t. Then, t is adjacent to each block node t' with $X_t \subset X_{t'}$.

Tutte has shown that additional restrictions can be formulated on the choice of the set of 2-cuts, such that the resulting decomposition is unique for each graph (for details see the above mentioned literature). In the following, when we refer to *the* Tutte decomposition of a graph, we always mean the one that is unique in this sense, which is also the one that Courcelle defined in his work [10].

▶ **Lemma 26.** *Let $G = (V, E)$ be a 2-connected graph with Tutte decomposition (T, X). G is k-outerplanar, if and only if all 3-connected 3-blocks of (T, X) are at most k-outerplanar.*

The proof of the previous lemma gives the following consequences.

▶ **Corollary 27.** *Let G be a 2-connected graph with Tutte decomposition (T, X).*
(i) *If G is a partial k-tree, then the 3-connected 3-blocks of (T, X) are partial k-trees.*
(ii) *If G is planar, then the 3-connected 3-blocks of (T, X) are planar.*
(iii) *If G is $\mathcal{H}$-minor free, then the 3-connected 3-blocks of (T, X) are $\mathcal{H}$-minor free, where $\mathcal{H}$ is a set of fixed graphs.*

Replacing Edge Quantification by Vertex Quantification

As discussed above, a 3-block is in general not a subgraph of a graph G, as we add edges between the 2-cuts of the Tutte decomposition to turn the 3-blocks into cycles or 3-connected graphs. Since these absent edges cannot be used as variables in MSOL-predicates (which would make our logic non-monadic), we need to find another way to quantify over them.

In [8], Courcelle discusses several structures over which one can define monadic second order logic of graphs, which we will now review.

▶ **Definition 28** (cf. Definition 1.7 in [8]). Let $G = (V, E)$ be a graph. We associate with G two relational structures, denoted by $|G|_1 = \langle V, \text{edg}\rangle$ and $|G|_2 = \langle V \cup E, \text{edg}'\rangle$.

(i) (C)MSOL-predicates over $|G|_1$ only use vertices or vertex sets as variables and we have that $\text{edg}(x, y)$ is true for $x, y \in V$, if and only if there is some edge $e = \{x, y\} \in E$. (C)MSOL-predicates over $|G|_2$ use both vertices and edges and vertex and edge sets as variables. Furthermore, $\text{edg}'(e, x, y)$ is true if and only if $e = \{x, y\}$ and $e \in E$.

(ii) If we can express a graph property in the structure $|G|_1$, we call it *1-definable* and if we can express a graph property in the structure $|G|_2$, we call it *2-definable*.

Clearly, the monadic second order logic we are using throughout this paper is the one represented by the structure $|G|_2$. We use both vertex and edge quantification and one rewrites $\text{Inc}(v, e)$ to $\exists w\ \text{edg}'(e, v, w)$. Since every 1-definable property is trivially also 2-definable, we can conclude that both 1-definability and 2-definability imply (C)MSOL-definability in our sense. Some of the main results of [8] can be summarized as follows.

▶ **Theorem 29** ([8]). *1-Definability equals 2-definability for*

(i) *partial k-trees.*

(ii) *planar graphs.*

(iii) *$\mathcal{H}$-minor free graphs, where $\mathcal{H}$ is a set of fixed graphs.*

Hence, by Theorem 29 we know that we can rewrite each formula using vertex and edge quantification to one only using vertex quantification, if a graph is a member of one of the stated graph classes. We will now show that this result can be used to implicitly quantify over virtual edges of a graph, if these virtual edges can be expressed by an (existentially) MSOL-definable relation. (For a similar application of this result, see [10, Problem 4.10].)

▶ **Lemma 30.** *Let $G = (V, E)$ be a graph which is a member of a graph class $\mathcal{C}$ as stated in Theorem 29 and let P denote a graph property, which is 2-definable by a predicate ϕ_P. Let $E' \subseteq V \times V$ denote a set of virtual edges, such that there is a predicate $edg_{Virt}(v, w)$, which is true if and only if $\{v, w\}$ is a member of E'. Then, P is also 1-definable for the graph $G' = (V, E \cup E')$, if G' is a member of $\mathcal{C}$.*

For the specific case of k-outerplanar graphs, we can now derive the following.

▶ **Corollary 31.** *Let $G = (V, E)$ be a k-outerplanar graph and P a graph property, which is (C)MSOL-definable for 3-connected k-outerplanar graphs. Let B_3 denote a 3-block of G, including the virtual edges between all incident 2-cuts of B_3. Then, P is (C)MSOL-definable for B_3.*

Note that the statements of Lemma 30 and Corollary 31 also hold for existential definability.

Defining the Tree Decomposition of a k-Outerplanar Graph

By Corollary 31 we now know that every graph property, which can be defined for a 3-connected k-outerplanar graph, can also be defined for a 3-block of any k-outerplanar graph G (including its virtual edges). However, there are two more steps we have to take to conclude the proof of our main result, which we will discuss in the current section. First, we have to show that if we are given predicates which encode bounded-width tree decompositions for the 3-connected 3-blocks of a k-outerplanar graph G, then we can define predicates which encode a bounded-width tree decomposition of G.

▶ **Lemma 32.** *Let $G = (V, E)$ be a k-outerplanar graph. G admits an existentially MSOL-definable tree decomposition of width at most $3k + 3$ with a constant number of parameters, if there exist predicates existentially defining width-$3k$ tree decompositions for the 3-connected 3-blocks of G with a constant number of parameters.*

The second obstacle in applying Lemma 23 to define a tree decomposition for G using its (definable) hierarchical graph decomposition is that the number of parameters of all existentially defined predicates has to stay constant. When defining a tree decomposition for a 3-connected k-outerplanar graph in MSOL, one first guesses a rooted spanning tree of G. To avoid guessing a non-constant number of spanning trees, we will find a set of edges $\mathcal{S}_E$, which contains (the edges of) a spanning tree with bounded edge and face remember number for each 3-connected 3-block of G. Furthermore we guess one set $\mathcal{R}_V$, containing one unique vertex for each 3-connected 3-block of G, which we will use as the root of its spanning tree.

▶ **Lemma 33.** *Let $G = (V, E)$ be a planar graph and $G = (V, E \cup E')$ the graph obtained by adding the virtual edges E' of the Tutte decompositions of the 2-connected blocks of G to G. Let $T = (V, F)$ be a spanning tree of G with $er(G, T) \le \lambda$ and $fr(G, T) \le \mu$. Let $B_3 = (V_{B_3}, E_{B_3})$ be a 3-connected 3-block of G' (including virtual edges) and $T_{B_3} = T[V_{B_3}]$. One can construct from T_{B_3} a spanning tree $T^*_{B_3}$ of B_3 with $er(B_3, T^*_{B_3}) \le \lambda$ and $fr(B_3, T^*_{B_3}) \le \mu$ by adding edges from $E \cup E'$ to T_{B_3}.*

We now give an outline of how to complete the proof of our main result. For the technical details, the reader is referred to the full text [14].

We first show that the statement of Lemma 33 also holds for *directed* spanning trees. It is then trivial to derive the set $\mathcal{R}_V$ of roots of the spanning trees of each 3-connected 3-block. Then we prove that both sets are MSOL-definable with a constant number of parameters. Combining these observations with Lemma 32 we can then conclude that k-outerplanar graphs admit existentially definable tree decompositions of width at most $3k + 3$.

Now we can apply Lemmas 9 and 12 to k-outerplanar graphs and in the light of Courcelle's Theorem [6] we then have our main result.

▶ **Theorem 34.** *Definability equals recognizability for k-outerplanar graphs.*

4 Conclusion

In this paper we have shown that recognizability implies definability in counting monadic second order logic for k-outerplanar graphs, resolving a special case of a conjecture by Courcelle [6]. Starting at the more restrictive case of 3-connected k-outerplanar graphs, we proved that one can use hierarchical graph decompositions to define tree decompositions for general k-outerplanar graphs in monadic second order logic. We have also given indications that this technique might be applicable for other graph classes as well (see Corollary 27), depending on how their tree decompositions are defined in MSOL. 3-Connected graphs often have favorable properties when it comes to defining graph properties in MSOL. For example, in our proof we used the fact that the face boundaries of a 3-connected can be expressed in strictly combinatorial terms and are definable in a straightforward way (see Propositions 20 and 21). Hence, we believe that the techniques presented in this paper can be helpful in resolving the conjecture in its general statement.

Acknowledgements. The second author thanks Bruno Courcelle, Mike Fellows, Pinar Heggernes and Jan Arne Telle for inspiring discussions.

References

1 Stefan Arnborg, Jens Lagergren, and Detlef Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12(2):308–340, 1991.

2 Hans L. Bodlaender. A partial k-arboretum of graphs with bounded treewidth. *Theoretical Computer Science*, 209(1-2):1–45, 1998.

3 Hans L. Bodlaender, Pinar Heggernes, and Jan Arne Telle. Recognizability equals definability for graphs of bounded treewidth and bounded chordality. In *Proceedings EUROCOMB 2015*, Electronic Notes in Discrete Mathematics. Elsevier, 2015.

4 Richard B. Borie, R. Gary Parker, and Craig A. Tovey. Automatic generation of linear-time algorithms from predicate calculus descriptions of problems on recursively constructed graph families. *Algorithmica*, 7(1-6):555–581, 1992.

5 J. Richard Büchi. Weak second-order arithmetic and finite automata. *Mathematical Logic Quarterly*, 6(1-6):66–92, 1960.

6 Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Information and Computation*, 85(1):12–75, 1990.

7 Bruno Courcelle. The monadic second-order logic of graphs V: On closing the gap between definability and recognizability. *Theoretical Computer Science*, 80(2):153–202, 1991.

8 Bruno Courcelle. The monadic second order logic of graphs VI: On several representations of graphs by relational structures. *Discrete Applied Mathematics*, 54(2-3):117–149, 1994.

9 Bruno Courcelle. The monadic second-order logic of graphs VIII: Orientations. *Annals of Pure and Applied Logic*, 72(2):103–143, 1995.

10 Bruno Courcelle. The monadic second-order logic of graphs XI: Hierarchical decompositions of connected graphs. *Theoretical Computer Science*, 224(1-2):38–53, 1999.

11 Bruno Courcelle. The monadic second-order logic of graphs XII: Planar graphs and planar maps. *Theoretical Computer Science*, 237(1-2):1–32, 2000.

12 Reinhard Diestel. *Graph Theory*. Number 173 in Graduate Texts in Mathematics. Springer, 4th edition, 2012. Corrected reprint.

13 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

14 Lars Jaffke and Hans L. Bodlaender. Definability equals recognizability for k-outerplanar graphs. *ArXiv e-prints*, 2015. http://arxiv.org/abs/1509.08315.

15 Valentine Kabanets. Recognizability equals definability for partial k-paths. In *Proceedings ICALP 1997*, volume 1256 of *LNCS*, pages 805–815. Springer, 1997.

16 Damon Kaller. Definability equals recognizability of partial 3-trees and k-connected partial k-trees. *Algorithmica*, 27(3-4):348–381, 2000.

17 Ioannis Katsikarelis. Computing bounded-width tree and branch decompositions of k-outerplanar graphs. *ArXiv e-prints*, 2013. http://arxiv.org/abs/1301.5896.

18 Denis Lapoire. Recognizability equals monadic second-order definability for sets of graphs of bounded tree-width. In *Proceedings STACS 1998*, volume 1373 of *LNCS*, pages 618–628. Springer, 1998.

19 William T. Tutte. *Connectivity in Graphs*. University of Toronto Press, 1966.

20 William T. Tutte. *Graph Theory*, volume 21 of *Encyclopedia of Mathematics and its Applications*. Addison-Wesley, 1984.

21 Hassler Whitney. Congruent graphs and the connectivity of graphs. *American Journal of Mathematics*, 54:150–168, 1932.