Universität Augsburg

A Lower Bound for the Treewidth of k-Outerplanar Graphs

Frank Kammer und Torsten Tholey

Report 2009-7 April 2009

Institut für Informatik
D-86135 Augsburg

A Lower Bound for the Treewidth of k-Outerplanar Graphs

Frank Kammer and Torsten Tholey

Institut für Informatik, Universität Augsburg, D-86135 Augsburg, Germany
{kammer,tholey}@informatik.uni-augsburg.de

Abstract. Many optimization problems can be solved efficiently if a tree-decomposition of small width is given. Unfortunately, all known algorithms computing, for general graphs, a tree decomposition of width k, if one exists, have a running time exponential in k. However, Bodlaender observed that each k-outerplanar graph has a tree decomposition of width at most $3k-1$ and his analysis implicitly leads to an $O(kn)$-time algorithm for computing such a tree-decomposition. In this paper we show that the bound $3k-1$ is tight, i.e., for every $k \in \mathbb{N}$, there are k-outerplanar graphs having treewidth $3k-1$.

1 Introduction

The treewidth of a graph is one of the classical complexity parameters studied in graph theory. Its importance is based on the fact that graphs of bounded treewidth have a treelike structure that allows to generalize efficient algorithms for hard problems on trees to graphs of bounded treewidth. In particular, all decision problems that can be expressed in monadic second-order logic can be solved by polynomial time algorithms on graphs of bounded treewidth [2, 4, 7]. In practice, the efficiency of such algorithms depends on the fact whether a tree-composition of small width exists and whether it can be computed efficiently. Arnborg, Corneil and Proskurowski [1] have shown that determining the treewidth of a graph is in general an $\mathcal{NP}$-hard problem. Bodlaender [4] could show that, for every fixed k, there is linear-time algorithm that tests whether a given graph has treewidth at most k and, if so, computes a tree-decomposition of this width. However, his algorithm is practically infeasible since the constants depending on k are of enormous size. Therefore, one of the best algorithms today is the algorithm of Reed [9] which computes, for any fixed k and any graph of treewidth at most k, a tree-decomposition of width $\leq 4k$ in $O(n \log n)$ time. Nevertheless, even the running time of Reeds algorithm is exponential in k. For this reason, it seems appropriate to take other complexity parameters into account that allow the construction of tree decompositions of small treewidth.

For planar graphs, one of such other complexity parameters is the outerplanarity index. Baker [3] presented many efficient algorithms for k-outerplanar graphs and used them for constructing approximation schemes for many optimization problems on planar graphs. Bodlaender [4] proved that the treewidth of a k-outerplanar graph is bounded by $3k-1$. His analysis can be transformed into an algorithm computing a tree-decomposition of width at most $3k-1$ in $O(kn)$ time. In this paper we show that the upper bound of the treewidth given by Bodlaender is tight, i.e., we show that for every $k \in \mathbb{N}$ there are k-outerplanar graphs having treewidth $3k-1$. Surprisingly, we could not find this result explicitly being published in literature.

2 Treewidth and the k-Cops-and-Robber-Game

The treewidth of a graph is defined as follows:

Definition 1 (tree decomposition, bag, (tree)width). *A* tree decomposition *for a graph $G=(V,E)$ is a pair (T,B), where $T=(V_T,E_T)$ is a tree and B is a mapping that maps each node w of T to a subset of V—called the* bag *of w—such that*

1. $\bigcup_{w\in V_T} G[B(w)] = G$, *and*
2. $B(x)\cap B(y) \subseteq B(w)$ *for all $w \in V_T$ on the path from $x \in V_T$ to $y \in V_T$ in T.*

The width *of (T,B) is $\max_{w\in V_T}\{|B(w)|-1\}$. The* treewidth of a graph G, *denoted by* $\mathrm{tw}(G)$, *is the width of a tree decomposition for the graph having smallest width.*

Seymour and Thomas [10] observed a connection between the treewidth of a graph G and the following *k-cops-and-robber game* on G:

- In a first round k cops place themselves on a set X_1 of at most k vertices of G and afterwards a robber chooses a vertex in $G - X_0$.
- In each of the following rounds $i = 2, \ldots$ the cops move from a set X_{i-1} of at most k vertices to a set X_i of at most k vertices. The robber tries to escape which means that he moves along a path in $G - (X_{i-1} \cap X_i)$ to a vertex in $G - X_i$. For an explanation of this rule, think of the cops as moving in helicopters whereas the robber has to use streets modeled by the edges of the graph. Each vertex in $X_{i-1} \cap X_i$ is occupied by at least one cop who does not want to move away using his helicopter, so that the robber cannot pass a vertex in $X_{i-1} \cap X_i$. However, during the time in which the other cops are in the helicopters, he may pass through all vertices that he can reach by a path in $G - (X_{i-1} \cap X_i)$. For not being catched, the robber must stop in a cop-free vertex of $G - X_i$.

The cops win if the robber cannot escape anymore, and the robber wins if he always can escape. Seymour and Thomas [10] have shown:

Lemma 2. *A graph G has treewidth $\geq k$ if the robber wins the k-cops-and-robber game on G.*

3 Lower Bound

The $2k \times 2k$ grid is a simple example of a k-outerplanar graph having treewidth $2k$. We next show that there are k-outerplanar graphs of larger treewidth.

Lemma 3. *For all $k, n \in \mathbb{N}$ with $n \geq 2k^3 + 6k^2$, there exists a k-outerplanar graph G with n vertices and $\mathrm{tw}(G) \geq 3k - 1$.*

Proof. Let G be the graph obtained from a $2k \times 2k$-grid G_1 and a $2k(k+1) \times k$-grid G_2 by connecting, for all $i \in \{0, \ldots, 2k-1\}$, the rightmost vertex of the $(i+1)$-th row of G_1 by $k+1$ edges with the leftmost vertices of the $(i \cdot (k+1)+1)$th, $(i \cdot (k+1)+2)$th, ..., $((i+1) \cdot (k+1))$th row of G_2 (see Fig. 1). Since G_2 has only

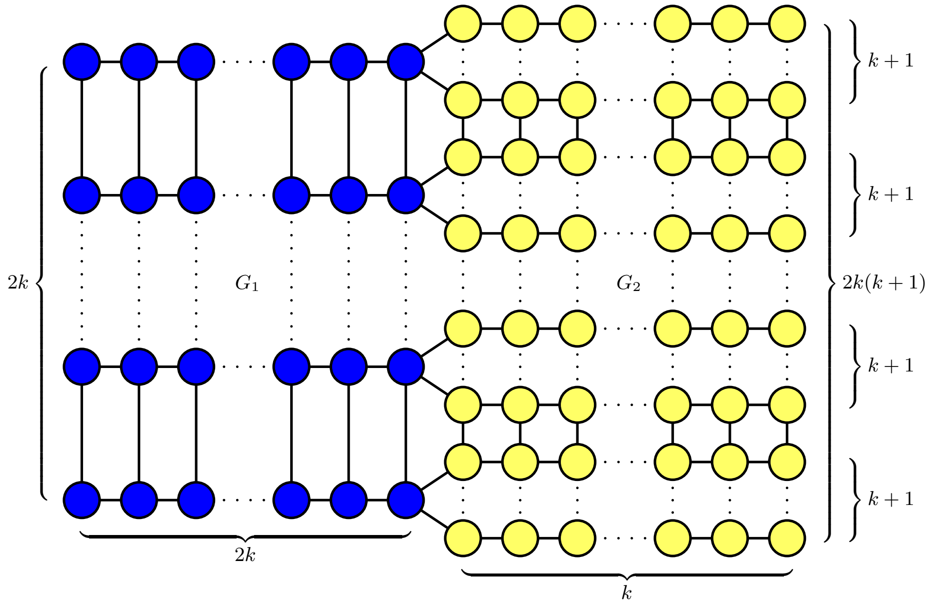

Fig. 1. The graph G consisting of two grids G_1 and G_2.

k columns and G_1 only 2k rows, G is k-outerplanar. For proving $\mathrm{tw}(G) \geq 3k-1$, we consider the robber-cop-game on G with $3k-1$ cops. Let us define an *extended row* as a tuple (r, e, s) of a row r of G_1 connected by edge e to a row s of G_2. Note that for each row r of G_1 there are $k+1$ extended rows containing r. The robber wins the game since he can move such that, before and after each round, one of the two following invariants holds:

(I1) At most $2k-1$ cops are in G_1 and the robber is in a cop-free extended row.
(I2) At most $k-1$ cops are in G_2 and the robber is in a cop-free column of G_2.

For the next conclusions, always keep in mind that there are in total $3k-1$ cops. If there are at most $2k-1$ cops in G_1, there is at least one cop-free row r_1

in G_1. If there is no cop-free extended row containing r_1, then $k+1$ cops must be in G_2 and there is another cop-free row r_2 in G_1. If this row again is not part of a cop-free extended row, there is third row r_3 in G_1 that now definitively must be part of a cop-free extended row. This means that if there are at most $2k-1$ cops in G_1, there is an extended cop-free row. Moreover, at least one column of G_2 is cop-free if there are at least $2k$ cops in G_1. As a consequence, the robber can find an initial position such that (I1) or (I2) holds before the first round. Let us now analyze a fixed round of the game.

Assume that (I1) holds before the round. If the cops want to move such that after the round again there will be at most $2k-1$ cops in G_1, the robber moves along his cop-free extended row to a current cop-free column of G_1 and then along this column to an extended row being cop-free after the movement of the cops. Otherwise, after the round at least $2k$ cops are in G_1. Then the robber moves along his extended row to a column of G_2 being cop-free with respect to the next positions of the cops.

Let us next analyze what happens if (I2) holds before the round. Then the robber stays in a cop-free column of G_2. If after the next movement of the cops there are still at most $k-1$ cops in G_2, then there will be a new cop-free column c. Then the robber can move along his current column to a (non-extended) cop-free row of G_2 and then along this row to column c. Otherwise, after the next movement of the cops, there will be a cop-free extended row and the robber can move along his current column to this row.

Hence the robber has a winning strategy and this proves the lemma for $n = 2k^3 + 6k^2$. For $n > 2k^3 + 6k^2$ just connect the endpoint of a simple path of length $n - (2k^3 + 6k^2 + 1)$ by an edge to an arbitrary vertex of G_1.

References

1. S. Arnborg, D. G. Corneil, and A. Proskurowski, Complexity of finding embeddings in a k-tree, *SIAM J. Alg. Disc. Meth.* **8** (1987), pp. 277-284.
2. S. Arnborg, J. Lagergren, and D. Seese, Easy problems for tree-decomposabel graphs, J. Algorithms **12** (1991), pp. 308–340.
3. B. S. Baker, Approximation algorithms for NP-complete problems on planar graphs, J. Assoc. Comput. Mach. **41** (1994), pp. 153–180.
4. H. L. Bodlaender, A linear-time algorithm for finding tree-decompositions of small treewidth, *SIAM J. Comput.* **25** (1996), 1305–1317.
5. H. L. Bodlaender, A partial k-arboretum of graphs with bounded treewidth, *Theoret. Comput. Sci.* **209** (1998), pp. 1–45.
6. N. Robertson and P. D. Seymour, Graph minors. I. Excluding a forest, J. Comb. Theory Ser. B **35** (1983), pp. 39-61.
7. B. Courcelle, The monadic second-order logic of graphs. I. Recognizable sets of finite graphs, *Inform. and Comput.* **85** (1990), 12–75.
8. T. Hagerup, Algorithms for NP-hard problems, Course notes, SS 07, Institut für Informatik, Universität Augsburg.

9. B. A. Reed, Finding approximate separators and computing tree widths quickly, Proc. 24th Annual ACM Symposium on Theory of Computing (STOC 1992), pp. 221–228.
10. P. D. Seymour and R. Thomas, Graph searching and a min-max theorem for tree-width, *J. Comb. Theory, Ser. B*, **58** (1993), pp. 22–23.