Discrete Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

A tight lower bound for Vertex Planarization on graphs of bounded treewidth

Marcin Pilipczuk

Institute of Informatics, University of Warsaw, Poland

ARTICLE INFO

Article history:
Received 26 November 2015
Received in revised form 22 March 2016
Accepted 18 May 2016
Available online xxxx

Keywords:
Vertex planarization
Fixed-parameter tractability
Exponential Time Hypothesis
Treewidth

ABSTRACT

In the Vertex Planarization problem one asks to delete the minimum possible number of vertices from an input graph to obtain a planar graph. The parameterized complexity of this problem, parameterized by the solution size (the number of deleted vertices) has recently attracted significant attention. The state-of-the-art algorithm of Jansen et al. (2014) runs in time $2^{\mathcal{O}(k \log k)} \cdot n$ on an n-vertex graph with a solution of size k. It remains open if one can obtain a single-exponential dependency on k in the running time bound.

One of the core technical contributions of the work of Jansen, Lokshtanov, and Saurabh is an algorithm that solves a weighted variant of Vertex Planarization in time $2^{\mathcal{O}(w \log w)} \cdot n$ on graphs of treewidth w. In this short note we prove that the running time of this routine is tight under the Exponential Time Hypothesis, even in unweighted graphs and when parameterizing by treedepth. Consequently, it is unlikely that a potential single-exponential algorithm for Vertex Planarization parameterized by the solution size can be obtained by merely improving upon the aforementioned bounded treewidth subroutine.

1. Introduction

In the Vertex Planarization problem, given an undirected graph G and an integer k, our goal is to delete at most k vertices from the graph G to obtain a planar graph. If (G, k) is a YES-instance to Vertex Planarization, then we say that G is a *k-apex graph*. Since many algorithms for planar graphs can be easily generalized to near-planar graphs – k-apex graphs for small values of k – this motivates us to look for efficient algorithms to recognize k-apex graphs. In other words, we would like to solve Vertex Planarization for small values of k.

By a classical result of Lewis and Yannakakis [8], Vertex Planarization is NP-hard when k is part of the input. Since one can check if a given graph is planar in linear time [4], Vertex Planarization can be trivially solved in time $\mathcal{O}(n^{k+1})$, where $n = |V(G)|$, that is, in polynomial time for every fixed value of k. However, such an algorithm is impractical even for small values of k; a question for a faster algorithm brings us to the realms of *parameterized complexity*.

In parameterized complexity, every problem comes with a *parameter*, being an additional complexity measure of input instances. The central notion is a *fixed-parameter algorithm*: an algorithm that solves an instance x with parameter k in time $f(k)|x|^{\mathcal{O}(1)}$ for some computable function f. Such a running time bound, while still super-polynomial (the function f is usually exponential), is considered significantly better than say $\mathcal{O}(|x|^k)$, as it promises much faster algorithms for moderate values of k and large instances. We refer to recent textbooks [1,2] for a broader introduction to parameterized complexity.

Due to the aforementioned motivation, it is natural to consider the solution size k as a parameter for Vertex Planarization, and ask for a fixed-parameter algorithm. Since, for a fixed value of k, the class of all k-apex graphs is closed under

E-mail address: malcin@mimuw.edu.pl.

http://dx.doi.org/10.1016/j.dam.2016.05.019
0166-218X/

Please cite this article in press as: M. Pilipczuk, A tight lower bound for Vertex Planarization on graphs of bounded treewidth, Discrete Applied Mathematics (2016), http://dx.doi.org/10.1016/j.dam.2016.05.019

taking minors, the graph minor theory of Robertson and Seymour immediately yields a fixed-parameter algorithm, but with enormous dependency on the parameter in the running time bound.[1] The quest for an explicit and faster fixed-parameter algorithm for Vertex Planarization has attracted significant attention in the parameterized complexity community in the recent years. First, Marx and Schlotter [12] obtained a relatively simple algorithm, with doubly-exponential dependency on the parameter and n^2 dependency on the input size in the running time bound. Later, Kawarabayashi [7] obtained a fixed-parameter algorithm with improved linear dependency on the input size, at the cost of worse dependency on the parameter. Finally, Jansen, Lokshtanov, and Saurabh [6] developed an algorithm with running time bound $2^{\mathcal{O}(k \log k)} \cdot n$, improving upon all previous results.

As noted in [6], a simple reduction shows that Vertex Planarization cannot be solved in time $2^{o(k)} \cdot n^{\mathcal{O}(1)}$ unless the Exponential Time Hypothesis fails.

Conjecture 1 (*Exponential Time Hypothesis [5]*)**.** *Let s_3 be the infimum over all reals $\delta > 0$ for which there exists an algorithm checking satisfiability of 3-CNF formulae in time $\mathcal{O}(2^{\delta n})$, where n is the number of variables of the input formula. Then $s_3 > 0$.*

Informally speaking, the Exponential Time Hypothesis (ETH) [5] asserts that the satisfiability of 3-CNF formulae cannot be verified in time subexponential in the number of variables. In the recent years, a number of tight bounds for fixed-parameter algorithms have been obtained using ETH or the closely related Strong ETH; we refer to [9,11] for an overview. In this light, it is natural to ask for tight bounds for fixed-parameter algorithms for Vertex Planarization. In particular, [6] asks for a single-exponential (i.e., with running time bound $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$) algorithm.

The core subroutine of the algorithm of Jansen, Lokshtanov, and Saurabh, is an algorithm that solves Vertex Planarization in time $2^{\mathcal{O}(w \log w)} \cdot n$ on graphs of treewidth w. A direct way to obtain a single-exponential algorithm for Vertex Planarization parameterized by the solution size would be to improve the running time of this bounded treewidth subroutine to $2^{\mathcal{O}(w)} \cdot n^{\mathcal{O}(1)}$. In this short note we show that such an improvement is unlikely, as it would violate the Exponential Time Hypothesis.

Theorem 2. *Unless the Exponential Time Hypothesis fails, there does not exist an algorithm that solves* Vertex Planarization *on n-vertex graphs of treewidth at most w in time $2^{o(w \log w)} n^{\mathcal{O}(1)}$.*

In fact, our lower bound holds even for a more restrictive parameter of *treedepth*, instead of treewidth.

While Theorem 2 does not exclude the possibility of a $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$-time algorithm for Vertex Planarization, it shows that to obtain such a running time one needs to circumvent the usage of bounded-treewidth subroutine on graphs for which only an $\mathcal{O}(k)$ bound on the treewidth is available, as in the algorithm of Jansen, Lokshtanov, and Saurabh.

The remainder of this paper is devoted to the proof of Theorem 2.

2. Lower bound

We base our reduction on the framework for proving superexponential lower bounds introduced by Lokshtanov, Marx, and Saurabh [10]. For an integer k, by $[k]$ we denote the set $\{1, 2, \ldots, k\}$. Consequently, $[k] \times [k]$ is a $k \times k$ table of elements with *rows* being subsets of the form $\{i\} \times [k]$, and *columns* being subsets of the form $[k] \times \{i\}$. We start from the following auxiliary problem.

$k \times k$ Permutation Clique	**Parameter:** k
Input: An integer k and a graph G with vertex set $[k] \times [k]$.	
Question: Is there a k-clique in G with exactly one element from each row and exactly one element from each column?	

As proven in [10], an $2^{o(k \log k)}$-time algorithm for $k \times k$ Permutation Clique would violate ETH. Hence, to prove Theorem 2, it suffices to prove the following.

Lemma 3. *There exists a polynomial time algorithm that, given an instance (G, k) of $k \times k$* Permutation Clique*, outputs an equivalent instance (H, ℓ) of* Vertex Planarization *where the treedepth of the graph H is bounded by $\mathcal{O}(k)$.*

That is, as announced in the introduction, we in fact prove a stronger variant of Theorem 2, refuting an existence of a $2^{o(w \log w)} n^{\mathcal{O}(1)}$-time algorithm for Vertex Planarization parameterized by the treedepth of the input graph. Recall that the treedepth of a graph G, denoted $\mathrm{td}(G)$, is never smaller than the treewidth of G, and satisfies the following recursive formula.

Lemma 4 (*[13]*)**.** *The treedepth of an empty graph is* 0*, and the treedepth of a one-vertex graph equals* 1*. The treedepth of a disconnected graph G equals the maximum of the treedepth of the connected components of G. The treedepth of a connected graph G is equal to*

$$\mathrm{td}(G) = 1 + \min_{v \in V(G)} \mathrm{td}(G - \{v\}).$$

[1] Formally, this algorithm is non-uniform, that is, it requires an external advice depending on the parameter only. However, we can obtain a uniform algorithm using the techniques of Fellows and Langston [3].

Please cite this article in press as: M. Pilipczuk, A tight lower bound for Vertex Planarization on graphs of bounded treewidth, Discrete Applied Mathematics (2016), http://dx.doi.org/10.1016/j.dam.2016.05.019