



Dynamic programming and planarity: Improved tree-decomposition based algorithms[☆]

Frederic Dorn^{*}

Department of Informatics, University of Bergen, N-5020 Bergen, Norway

ARTICLE INFO

Article history:

Received 19 March 2008
Received in revised form 8 June 2009
Accepted 20 October 2009
Available online 30 October 2009

Keywords:

Tree-decompositions
Dynamic programming
Planar dominating set
Planar Hamiltonian cycle
Planar graph TSP
Branch-decompositions

ABSTRACT

We study some structural properties for tree-decompositions of 2-connected planar graphs that we use to improve upon the runtime of tree-decomposition based dynamic programming approaches for several NP-hard planar graph problems. E.g., we derive the fastest algorithm for PLANAR DOMINATING SET of runtime $3^{tw} \cdot n^{O(1)}$, when we take the width tw of a given tree-decomposition as the measure for the exponential worst case behavior. We also introduce a tree-decomposition based approach to solve non-local problems efficiently, such as PLANAR HAMILTONIAN CYCLE in runtime $6^{tw} \cdot n^{O(1)}$. From any input tree-decomposition of a 2-connected planar graph, one computes in time $O(nm)$ a tree-decomposition with geometric properties, which decomposes the plane into disks, and where the graph separators form Jordan curves in the plane.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Many separator results for topological graphs, especially for planar embedded graphs base on the fact that separators have a structure that cuts the surface into two or more pieces onto which the separated subgraphs are embedded on. The celebrated and widely applied (e.g., in many divide-and-conquer approaches) result of Lipton and Tarjan [24] finds in planar graphs a small sized separator. However, their result says nothing about the structure of the separator; it can be any set of discrete points. Applying the idea of Miller for finding small simple cyclic separators [25] in planar triangulations, one can find small separators whose vertices can be connected by a closed curve in the plane intersecting the graph only in vertices, so-called *Jordan curves* (e.g. see [4]). Tree-decompositions have been historically the choice when solving NP-hard optimization and FPT problems with a dynamic programming approach (see for example [6] for an overview). Although much is known about the combinatorial structure of tree-decompositions (amongst others, [7,32]), only few results are known to the author relating to the topology of tree-decompositions of planar graphs (e.g., [9] studied 3-connected planar graphs, i.e., graphs with a unique plane embedding). A branch-decomposition is another tool, that was introduced by Robertson and Seymour in their proof of the Graph Minors Theorem and the parameters of these similar structures, the *treewidth* $tw(G)$ and *branchwidth* $bw(G)$ of the graph G have the relation $bw(G) \leq tw(G) + 1 \leq 1.5 bw(G)$ [28]. Their proof gives a simple polynomial time algorithm for transforming branch-decompositions and tree-decompositions into one another. Recently, branch-decompositions have become a more popular tool than tree-decompositions, in particular for problems whose input is a topologically embedded graph [10,20,11,17,15,16], mainly for two reasons: the branchwidth of planar graphs can be computed in polynomial time (yet there is no algorithm known for treewidth) with better constants for

[☆] A preliminary version of this paper appeared at WG'07 Dorn (2007) [14].

^{*} Fax: +47 30 2093 3081.

E-mail address: frederic.dorn@ii.uib.no.

Table 1

We state the runtime of dynamic programming given a tree-decomposition of width tw , and a branch-decomposition of width bw , respectively. The worst case runtime $O(n^{O(1)} \cdot f(tw, bw))$ with a function only depending on tw and bw . We state the improvements independently for weighted and unweighted graph problems. Due to the relation $bw(G) \leq tw(G) + 1 \leq 1.5bw(G)$ a polynomial time algorithm for planar branchwidth gives a 1.5-OPT-algorithm for planar treewidth. Hence, for some problems, such as the weighted PLANAR DOMINATING SET, the tree-decomposition based algorithm always gives the best function. However, for the (mostly unweighted) variants of non-local problems from [17] not mentioned here, the branch-decomposition based approach is better.

	Previous results	New results
weighted PLANAR DOM SET	$2^{\min\{2tw, 2.38bw\}}$	$2^{1.58tw}$
unweighted PLANAR DOM SET	$2^{1.89bw}$	$2^{\min\{1.58tw, 1.89bw\}}$
W PLAN INDEPENDENT DOM SET	$2^{\min\{2tw, 2.28bw\}}$	$2^{1.58tw}$
UW PLAN INDEPENDENT DOM SET	$2^{1.89bw}$	$2^{\min\{1.58tw, 1.89bw\}}$
W PLAN TOTAL DOM SET	$2^{\min\{2.58tw, 3bw\}}$	2^{2tw}
UW PLAN TOTAL DOM SET	$2^{2.38bw}$	$2^{\min\{2tw, 2.38bw\}}$
W PLAN PERF TOTAL DOM SET	$2^{\min\{2.58tw, 3.16bw\}}$	$2^{\min\{2.32tw, 3.16bw\}}$
UW PLAN PERF TOTAL DOM SET	$2^{2.53bw}$	$2^{\min\{2.32tw, 2.53bw\}}$
W PLANAR HAMILTONIAN CYCLE	$2^{3.31bw}$	$2^{\min\{2.58tw, 3.31bw\}}$
UW PLANAR HAMILTONIAN CYCLE	$2^{2.66bw}$	$2^{\min\{2.58tw, 2.66bw\}}$
W PLANAR GRAPH TSP	$2^{4.65bw}$	$2^{\min\{4.12tw, 4.65bw\}}$
W PLANAR CONNECTED DOM SET	$2^{4.63bw}$	$2^{\min\{4.09tw, 4.63bw\}}$

the upper bound than treewidth. Secondly, planar branch-decompositions have geometrical properties, i.e. they are assigned with separators that form Jordan curves. Thus, one can exploit planarity in the dynamic programming approach in order to get an exponential speedup, as done by [17,13].

We extend results of [9] for employing planarity obtained by the structure of tree-decompositions so that we get faster algorithms. This enables us to give the first tree-decomposition based algorithms for planar Hamiltonian-like problems with slight runtime improvements compared to [17]. We emphasize our result in terms of the width parameters tw and bw with the example of DOMINATING SET. The graph problem DOMINATING SET asks for a minimum vertex set S in a graph $G = (V, E)$ such that every vertex in V is either in S or has a neighbor in S . Telle and Proskurowski [31] gave a dynamic programming approach based on tree-decompositions with runtime $9^{tw} \cdot n^{O(1)}$, and that was improved to $4^{tw} \cdot n^{O(1)}$ by Alber et al. [1]. Note that in the extended abstract [2], the same authors first stated the runtime wrongly to be $3^{tw} \cdot n^{O(1)}$. Fomin and Thilikos [20] gave a branch-decomposition based approach of runtime $3^{1.5bw} \cdot n^{O(1)}$. In [13], the author combined dynamic programming with fast matrix multiplication to get $4^{bw} \cdot n^{O(1)}$ and for PLANAR DOMINATING SET even $3^{\frac{\omega}{2}bw} \cdot n^{O(1)}$, where ω is the constant in the exponent of fast matrix multiplication (currently, $\omega \leq 2.376$). Exploiting planarity, we improve further upon the existing bounds and give a $3^{tw} \cdot n^{O(1)}$ dynamic programming for PLANAR DOMINATING SET algorithms, representative for a number of improvements on results of [3,13,17,18] as shown in Table 1. This settles an open question in [3] to match the *base value* of the exponential running time function with the *colors* needed to encode the solutions in the dynamic programming. More detailed explanations with an example will be given in Section 6.

Given any tree-decomposition of a 2-connected planar graph as an input, we show how to compute a geometric tree-decomposition that has the same properties as planar branch-decompositions. (If the graph is 1-connected, we compute geometric tree-decompositions of the 2-connected components and combine them to a tree-decomposition.) Employing structural results on minimal graph separators for planar graphs, we create in polynomial time a *parallel* tree-decomposition that is assigned by a set of pairwise parallel separators that form pairwise non-crossing Jordan curves in the plane. In a second step, we show how to obtain a *geometric* tree-decomposition, that has a ternary tree and is assigned Jordan curves that exhaustively decompose the plane into disks (one disk being the infinite disk). In fact, geometric tree-decompositions have all the properties in common with planar branch-decompositions, that are algorithmically exploited in [20,17]. The idea of arguing with Jordan curves for relating minimal separators of planar graphs and maximal cliques of triangulations is already used in the proof of self-duality for planar treewidth by Bouchitté et al. [9]. In fact, their proof *can* be made constructive for computing a geometric tree-decomposition of 3-connected planar graphs in polynomial time when given a minimal triangulation. Our contribution is (a) to show how to algorithmically find this structure of Jordan curves in a given ordinary tree-decomposition for *all* planar graphs and (b) to design faster tree-decomposition based dynamic programming approaches for a variety of planar graph optimization problems. Though, the dynamic programming is similar to that of [18] on semi-nice tree-decompositions in its combinatorial structure, we employ here for the first time topological arguments for dynamic programming on tree-decompositions.

Organization of the paper: after giving some preliminary results and stating the main theorem in Section 2, we introduce in Section 3 an algorithm to compute a parallel tree-decomposition. In Section 4, we describe how Jordan curves and separators in plane graphs influence each other and we get some tools for relating Jordan curves and tree-decompositions of 2-connected planar graphs in Section 5. Finally, we show how to compute geometric tree-decompositions and state in Section 6 their influence on dynamic programming approaches. In Section 7, we argue how our results may lead to faster algorithms when using fast matrix multiplication as in [13].

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات