# Tight Bounds for Linkages in Planar Graphs

Isolde Adler[1] Stavros G. Kolliopoulos[2] Philipp Klaus Krause[1]
Daniel Lokshtanov[3] Saket Saurabh[4] Dimitrios Thilikos[2]

[1] Goethe-Universität, Frankfurt am Main
[2] National and Kapodistrian University of Athens
[3] University of California, San Diego
[4] Institute of Mathematical Sciences, Chennai

**Abstract.** The Disjoint-Paths Problem asks, given a graph $G$ and a set of pairs of terminals $(s_1, t_1), \ldots, (s_k, t_k)$, whether there is a collection of $k$ pairwise vertex-disjoint paths linking $s_i$ and $t_i$, for $i = 1, \ldots, k$. In their $f(k) \cdot n^3$ algorithm for this problem, Robertson and Seymour introduced the *irrelevant vertex technique* according to which in every instance of treewidth greater than $g(k)$ there is an "irrelevant" vertex whose removal creates an equivalent instance of the problem. This fact is based on the celebrated *Unique Linkage Theorem*, whose – very technical – proof gives a function $g(k)$ that is responsible for an immense parameter dependence in the running time of the algorithm. In this paper we prove this result for planar graphs achieving $g(k) = 2^{O(k)}$. Our bound is radically better than the bounds known for general graphs. Moreover, our proof is new and self-contained, and it strongly exploits the combinatorial properties of planar graphs. We also prove that our result is optimal, in the sense that the function $g(k)$ cannot become better than exponential. Our results suggest that any algorithm for the Disjoint-Paths Problem that runs in time better than $2^{2^{o(k)}} \cdot n^{O(1)}$ will probably require drastically different ideas from those in the irrelevant vertex technique.

## 1 Introduction

One of the most studied problems in graph algorithms is the Disjoint-Paths Problem (DPP): *Given a graph $G$, and a set of $k$ pairs of terminals, $(s_1, t_1), \ldots, (s_k, t_k)$, decide whether $G$ contains $k$ vertex-disjoint paths $P_1, \ldots, P_k$ where $P_i$ has endpoints $s_i$ and $t_i$, $i = 1, \ldots, k$.* In addition to its numerous applications in areas such as network routing and VLSI layout, this problem has been the catalyst for extensive research in algorithms and combinatorics [22]. DPP is NP-complete, along with its edge-disjoint or directed variants, even when the input graph is planar [23,15,12,14]. The celebrated algorithm of Roberson and Seymour solves it however in $f(k) \cdot n^3$ steps, where $f$ is some computable function [17]. This implies that when we parameterize DPP by the number $k$ of terminals, the problem is fixed-parameter tractable. The Robertson-Seymour algorithm is the central algorithmic result of the Graph Minors series of papers, one of the deepest and most influential bodies of work in graph theory.

The basis of the algorithm in [17] is the so called *irrelevant-vertex technique* which can be summarized very roughly as follows. As long as the input graph $G$ violates certain structural conditions, then it is possible to find a vertex $v$ that is *solution-irrelevant:* every collection of paths certifying a solution to the problem can be rerouted to an *equivalent* one, that links the same pairs of terminals, but in which the new paths avoid $v$. One then iteratively removes such irrelevant vertices until the structural conditions are met. By that point the graph has been simplified enough so that the problem can be attacked via dynamic programming.

The following two structural conditions are used by the algorithm in [17]: (i) $G$ excludes a clique, whose size depends on $k$, as a minor and (ii) $G$ has treewidth bounded by some function of $k$. When it comes to enforcing Condition (ii), the aim is to prove that in graphs without big clique-minors and with treewidth at least $g(k)$ there is always a solution-irrelevant vertex. This is the most complicated part of the proof and it was postponed until the later papers in the series [18,19]. The bad news is that the complicated proofs also imply an *immense* dependence, as expressed by the function $f$, of the running time on the parameter $k$. This puts the algorithm outside the realm of feasibility even for elementary values of $k$.

The ideas above were powerful enough to be applicable also to problems outside the context of the Graph Minors series. During the last decade, they have been applied to many other combinatorial problems and now they constitute a basic paradigm in parameterized algorithm design (see, e.g., [3,4,7,8,9,11]). However, in most applications, the need for overcoming the high parameter dependence emerging from the structural theorems of the Graph Minors series, especially those in [18,19], remains imperative. Hence two natural directions of research are: simplify parts of the original proof for the general case or focus on specific graph classes that may admit proofs with better parameter dependence. An important step in the first direction was taken recently by Kawarabayashi and Wollan in [10] who gave an easier and shorter proof of the results in [18,19]. While the parameter dependence of the new proof is certainly much better than the previous, immense, function, it is still huge: a rough estimation from [10] gives a lower bound for $g(k)$ of magnitude $2^{2^{2^{\Omega(k)}}}$ which in turn implies a lower bound for $f(k)$ of magnitude $2^{2^{2^{2^{\Omega(k)}}}}$.

In this paper we offer a solid advance in the second direction, focusing on planar graphs (see also [16,21] for previous results on planar graphs). We prove that, for planar graphs, $g(k)$ is singly exponential. In particular we prove the following result.

**Theorem 1.** *There is a constant $c$ such that every $n$-vertex planar graph $G$ with treewidth at least $c^k$ contains a vertex $v$ such that every solution to* DPP *with input $G$ and $k$ pairs of terminals can be replaced by an equivalent one avoiding $v$.*

Given the above result, our Theorem 6 shows how to reduce, in $O(n^2)$ time, an instance of DPP to an an equivalent one whose graph $G'$ has treewidth $2^{O(k)}$. Then, using dynamic programming, a solution, if one exists, can be found in $k^{O(\text{treewidth}(G'))} \cdot n = 2^{2^{O(k)}} \cdot n$ steps.

The proof of Theorem 1 deviates significantly from those in [18,19,10]. It is self-contained and exploits extensively the combinatorics of planar graphs. Moreover, we give strong evidence that a parameterized algorithm for DPP with singly exponential dependence, if one exists, should require entirely different techniques. Indeed, in that sense, the result in Theorem 1 is tight:

**Theorem 2.** *There exists an instance of the* DPP *on a* $2^{\Omega(k)}$*-treewidth planar graph* $G$ *that has a unique solution spanning all the vertices of* $G$*.*

Notice that, due to the recent lower bounds in [13], the Disjoint-Paths Problem cannot be solved in $2^{o(w \log w)} \cdot n^{O(1)}$ for graphs of treewidth at most $w$, unless the Exponential Time Hypothesis fails. This result, along with Theorem 2, reveals the limitations of the irrelevant vertex technique: any algorithm for the Disjoint-Paths Problem whose parameter dependence that is better than doubly exponential, will probably require drastically different techniques.

## 2 Preliminaries

Graphs are finite, undirected and simple. We denote the vertex set of a graph $G$ by $V(G)$ and the edge set by $E(G)$. Every edge is a two-element subset of $V(G)$. A graph $H$ is a *subgraph* of a graph $G$, denoted by $H \subseteq G$, if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. A *path* in a graph $G$ is a sequence $P = v_1, \dots, v_n$ of pairwise distinct vertices of $G$, such that $v_i v_{i+1} \in E(G)$ for all $1 \leq i \leq n-1$. For a graph $G$ with $e = vw \in E(G)$ let $G/e$ denote the graph obtained from $G$ by *contracting* $e$, i.e. $V[G/e] := (V(G) \setminus \{v, w\}) \cup \{x_e\}$, where $x_e$ is a new vertex, and $E(G/e) := \big(E(G) \setminus \{uu' \mid uu' \cap e \neq \emptyset\}\big) \cup \{ux_e \mid uv \in E(G) \text{ or } uw \in E(G)\}$. A graph $H$ is a *minor* of a graph $G$, if $H$ can be obtained from a subgraph of $G$ by a sequence of edge contractions. We use standard graph terminology as in [5]. The *disjoint paths problem* (DPP) is the following problem.

| DPP |
|---|
| **Input**: A graph $G$, and pairs of terminals $(s_1, t_1), \dots, (s_k, t_k) \in V(G)^{2k}$ |
| **Question**: Are there $k$ pairwise vertex disjoint paths $P_1, \dots, P_k$ in $G$ such that $P_i$ has endpoints $s_i$ and $t_i$? |

We will call such a sequence $P_1, \dots, P_k$ a *solution* of the DPP.

Given an instance $(G, (s_1, t_1), \dots, (s_k, t_k))$ of DPP we say that a non-terminal vertex $v \in V(G)$ is *irrelevant*, if $(G, (s_1, t_1), \dots, (s_k, t_k))$ is a YES-instance if and only if $(G \setminus v, (s_1, t_1), \dots, (s_k, t_k))$ is a YES-instance. From now on $G$ will always be an instance to DPP accompanied by $k$ terminal pairs.

**Definition 1 (Grid).** *Let $m, n \geq 1$. The $(m \times n)$* grid *is the Cartesian product of a path of length $m-1$ and a path of length $n-1$. In case of a square grid where $m = n$, then we say that $n$ is the* size *of the grid.*

A *subdivided grid* is a graph obtained from a grid by replacing some edges of the grid by pairwise internally vertex disjoint paths of length at least one. *Embeddings* of graphs in the plane, *plane graphs*, *planar graphs* and *faces* are defined in the usual way. A graph is *outerplanar*, if it has an embedding in the plane where all vertices are incident to the infinite face.

A *cycle* in a graph $G$ is a subgraph $H \subseteq G$, such that $V(H) = \{x_0, x_1, \ldots, x_{k-1}\}$, $E(H) = \{x_0x_1, x_1x_2, \ldots, x_{k-2}x_{k-1}, x_{k-1}x_0\}$ for some $k \in \mathbb{N}$, $k \neq 1$, and $i \neq j \Rightarrow x_i \neq x_j$. Notice that we allow cycles to consist of a single vertex.

We use the fact that a subdivided grid has a unique embedding in the plane (up to homeomorphism). For $(1 \times 1)$ grids and subdivided $(2 \times 2)$ grids this is clear, and for $(n \times n)$ grids with $n \geq 2$ this follows from Tutte's Theorem stating that 3-connected graphs have unique embeddings in the plane (up to homeomorphism). This implies that subdivisions of $(n \times n)$ grids have unique embeddings as well. The *perimeter* of a subdivided grid $H$ is the cycle in $H$ that is incident to the outer face (in every planar embedding of $H$). We refer the reader to [2] for the definition of *tree-width* of a graph $tw(G)$.

A *directed graph* is a pair $D = (V, E)$ where $V$ is a set and $E \subseteq V \times V$. We call the elements of $E$ *directed edges*. For a directed edge $(u, v) \in E$ we say that $u$ is the *tail* of $(u, v)$, $u = \text{tail}(u, v)$, and $v$ is the *head* of $(u, v)$, $v = \text{head}(u, v)$.

## 3 Upper Bounds

The main result of this section is Theorem 6 stating that there is an $O(n^2)$-step algorithm that, given an instance $G$ of DPP of treewidth $2^{\Omega(k)}$, can find a set of irrelevant vertices whose removal from $G$ creates an equivalent instance of treewidth $2^{O(k)}$.

### 3.1 Basic Definitions

**Observation 1** *Let $G$, $(s_1, t_1, \ldots, s_k, t_k)$ be a planar instance of* DPP *and let $h \in \mathbb{N}$. If $G$ contains a subdivided $((h\sqrt{2k+1}) \times (h\sqrt{2k+1}))$ grid, then $G$ contains a subdivided $h \times h$ grid $H$ such that in every embedding of $H$ all terminals lie outside the open disc bounded by the perimeter of $H$.*

Notice that Observation 1 ensures that once we have a large grid we can also assume that we have a large grid *that does not contain any terminal vertices*. Next we define a specific kind of embedding of cycles that helps us enforce structure in the proof.

**Definition 2 (Tight concentric cycles).** *Let $G$ be a plane graph and let $C_0, \ldots, C_n$ be a sequence of cycles in $G$ such that each cycle bounds a closed disc $D_i$ in the plane. We call $C_0, \ldots, C_n$* concentric*, if for all $i \in \{0, \ldots, n-1\}$, the cycle $C_i$ is contained in the interior of $D_{i+1}$. The concentric cycles $C_0, \ldots, C_n$*

*are* tight*, if, in addition,* $C_0$ *is a single vertex and for every* $i \in \{0, \ldots, n-1\}$*,* $D_{i+1} \setminus D_i$ *does not contain a cycle* $C$ *bounding a disc* $D$ *in the plane with* $D_{i+1} \supsetneq D \supseteq D_i$*.*

*Remark 1.* Let $G$ be a plane graph and let $C_0, \ldots, C_n$ be tight concentric cycles in $G$ bounding closed discs $D_1, \ldots, D_n$, respectively, in the plane. Let $P$ be a path connecting vertices $u$ and $v$ with $u, v \notin D_n$. If a vertex of $P$ is contained in the interior of $D_i$ (i.e. in $D_i \setminus C_i$), then $P$ has a vertex on $C_{i-1}$.

*Remark 2.* If a graph contains a $((2n+1) \times (2n+1))$ grid minor, it contains a sequence $C_0, \ldots, C_n$ of tight concentric cycles.

A *linkage* in a graph $G$ is a family of pairwise disjoint paths in $G$. The *endpoints* of a linkage $L$ are the endpoints of the paths in $L$, and the *pattern* of $L$ is the matching on the endpoints induced by the paths, i.e. the pattern is the set $\big\{\{s,t\} \mid L \text{ contains a path from } s \text{ to } t\big\}$.

**Definition 3 (Segment & Handle).** *Let* $G$ *be a plane graph, let* $C$ *be a cycle in* $G$ *bounding a closed disc* $D$ *in the plane and let* $P$ *be a path in* $G$ *such that its endpoints are outside of* $D$*. We say that a path* $P_0$ *is a* $D$-segment *(resp.* $D$-handle*) of* $P$*, if* $P_0$ *is a non-empty maximal subpath of* $P$ *whose endpoints are on* $C$, *and* $P_0 \subseteq D$ *(resp.* $P_0 \cap D$ *contains only the endpoints of* $P_0$*). For a linkage* $\mathcal{P}$ *in* $G$ *we say that a path* $P_0$ *is a* $D$-segment *(*$D$-handle*) of* $\mathcal{P}$*, if* $P_0$ *is a* $D$-segment *(*$D$-handle*) of some path* $P$ *of* $\mathcal{P}$*.*

*Remark 3.* Let $G$ be a plane graph, let $C$ be a cycle in $G$ bounding a closed disc $D$ in the plane and let $P = su_1 \ldots u_q t$ be a path in $G$ such that $s$ and $t$ are outside $D$. Suppose $x_a, x_b, x_c, \ldots, x_j$ is the order in which the vertices of path $P$ appear on the cycle $C$ when we traverse it from $s$ to $t$. Then the subpath of $P$ between $x_a$ and $x_b$ is a $D$-segment while the subpath of $P$ between $x_b$ and $x_c$ is a $D$-handle, and $D$-segments and $D$-handles alternate.

From now on we will assume that $G$ is a plane graph containing a sequence $C_0, \ldots, C_n$ of concentric cycles bounding closed discs $D_0, \ldots, D_n$, respectively, in the plane. Furthermore there are no terminals contained in $D$ and $G$ is an YES-instance. That is, there are paths between $s_i$ and $t_i$ such that they are mutually disjoint. These paths form a linkage that will be denoted by $\mathcal{P}$. From now onwards whenever we say *linkage* we mean a set of disjoint paths between pairs $(s_i, t_i)$. We will often refer to the $D_n$-segments ($D_n$-handles) of $\mathcal{P}$ simply as the *segments (handles)* of $\mathcal{P}$.

**Definition 4 ($I$(Handle) and $\beta$(Handle)).** *Let* $G$ *be a plane graph containing a sequence* $C_0, \ldots, C_n$ *of concentric cycles bounding closed discs* $D_0, \ldots, D_n$*, respectively, in the plane and* $\mathcal{P}$ *be a linkage. Let* $P$ *be a* $D_n$*-handle and let its endpoints be* $x$ *and* $y$*. Let* $C_n[x,y]$ *denote the path between* $x, y$ *on the cycle* $C_n$ *such that the finite face bounded by* $P \cup C_n[x,y]$ *does not contain the interior of* $D_n$*. By* $I(P)$ *we denote the subgraph of* $G$ *that has boundary* $P \cup C_n[x,y]$*, and we let* $\beta(P) := C_n[x,y]$*.*

**Definition 5 (Cheap solution).** *Let $G$ be a plane graph containing a sequence $C_0, \ldots, C_n$ of concentric cycles bounding closed discs $D_0, \ldots, D_n$, respectively, in the plane. For a linkage $\mathcal{P}$ of $G$, define its* cost *$c(\mathcal{P})$ as the number of edges of $\mathcal{P}$ that do not belong to $\bigcup_{i=0}^{n} C_i$. A linkage $\mathcal{P}$ is called* cheap, *if there is no other linkage $\mathcal{Q}$, such that $c(\mathcal{Q}) < c(\mathcal{P})$.*

Observe that the contribution of a $D_n$-handle of $\mathcal{P}$ to $c(\mathcal{P})$ is always positive. Edges of $D_n$-segments contribute to $c(\mathcal{P})$ whenever they do not belong to a concentric cycle. We assume for the remainder of Section 3 that we are given a cheap solution $\mathcal{P}$ to our input instance and we explore its structure.

### 3.2 Simple properties of a cheap solution $\mathcal{P}$

**Lemma 1.** *If $\mathcal{P}$ is a cheap solution to the input instance then there is no segment $P$ of $\mathcal{P}$ with vertices appearing in the order $\ldots, v_0, \ldots, v_1, \ldots, v_2, \ldots$ where $v_0$ and $v_2$ are vertices of $C_\ell$, and $v_1$ is a vertex of $C_j$, for $n \geq j > \ell \geq 0$.*

**Lemma 2.** *Let $\mathcal{P}$ be a cheap solution to the input instance and $Q$ be a handle. Then there is terminal inside $I(Q)$.*

We remark that Lemma 2 is true in more general setting. We will use the generalized version in a proof later. Let $D$ be a disc with the boundary cycle $C$ and $\mathcal{T}$ be a subpath of a path in $\mathcal{P}$ with endpoints $x$ and $y$ on the disc and no vertices in the interior of the disc. Let $C[x, y]$ denote the path between $x, y$ on cycle $C$ such that the finite face bounded by $\mathcal{T} \cup C[x, y]$ does not contain the interior of $D$. By $I(\mathcal{T})$ we denote the subgraph that has boundary $\mathcal{T} \cup C[x, y]$. A proof similar to the one in Lemma 2 gives us the following.

**Lemma 3.** *Let $\mathcal{P}$ be a cheap solution to the input instance and $\mathcal{T}$ be a subpath of a path in $\mathcal{P}$ with endpoints on the disc and no points in the interior of the disc. Then there is a terminal inside $I(\mathcal{T})$.*

### 3.3 Bounding the number of segment types

In this section we define a notion of segment types and obtain an upper bound on the number of segment types.

**Definition 6 (Segment Type).** *Let $\mathcal{P}$ be a solution to the input instance. Let $R$ and $S$ be two $D_n$-segments. Let $Q$ and $Q'$ be the two paths on $C_n$ connecting an endpoint of $R$ with an endpoint of $S$ and passing through no other endpoint of $R$ or $S$. We say that $R$ and $S$ are* equivalent, *and we write $R \parallel S$, if no $D_n$-segment of $\mathcal{P}$ has both endpoints on $Q$ and no $D_n$-segment has both endpoints on $Q'$. A* type *of $D_n$-segments is an equivalence class of $D_n$-segments under the relation $\parallel$.*

**Definition 7 (Segment graph).** *We start with the subgraph of* $G$ *contained in* $D_n$*. Retain only the edges and vertices of* $\bigcup \mathcal{P} \cup C_n$*. Choose an edge. If it is part of* $C_n$*, contract it unless it connects endpoints of segments of different type. If it is not part of* $C_n$*, contract it unless it connects endpoints of segments. Repeat until there are no contractable edges left. Remove duplicate edges and loops, such that the graph becomes simple again. The resulting graph is the* segment graph *of* $D_n$ *(clearly, segment graphs are outerplanar graphs).*

**Definition 8 (Tongue tip).** *A* $D_n$*-segment type is called* tongue tip*, if it is a single vertex in the segment graph of* $D_n$*.*

**Definition 9 (Segment dual graph).** *We take the dual graph of the segment graph of* $D_n$*. Delete the vertex that represents the infinite face. Add the vertices representing the tongue tips of the segment graph and connect them to the vertices representing neighboring faces in the segment graph. The resulting graph is the* segment dual graph *of* $D_n$*.*

*Remark 4.* Since the segment graph is outerplanar, the segment dual graph is a tree. All inner nodes of the segment dual graph have degree at least 3.

The next lemma is based on Lemmata 2 and 3.

**Lemma 4 (Tongue-taming).** *Let* $\mathcal{P}$ *be a cheap solution to the input instance. Then there are at most* $2k - 1$ *tongue tips.*

**Theorem 3.** *Let* $\mathcal{P}$ *be a cheap solution to the input instance then* $\mathcal{P}$ *has at most* $4k - 3$ *different types of* $D_n$*-segments.*

### 3.4 Bounding the size of segment types

In this section we find a bound on the size of segment types in cheap solutions and we combine it with the bound on the number of segment types obtained in the previous section to find irrelevant vertices. Indeed, we find that cheap solutions only pass through a bounded number of concentric cycles.

We find the bound on the size of segment types by rerouting in the presence of a large segment type. In a first step, we allow ourselves to freely reroute in a disc (making sure that the graph remains planar), and we bound the number of segments of solution paths in the disc. In a second step, we realize our rerouting in a sufficiently large grid.

**Lemma 5.** *Let* $\Sigma$ *be an alphabet of size* $|\Sigma| = k$*. Let* $w \in \Sigma^*$ *be a word over* $\Sigma$*. If* $|w| > 2^k$*, then* $w$ *contains an infix* $y$ *with* $|y| \geq 2$*, such that every letter occurring in* $y$ *occurs an even number of times in* $y$*.*

The following lemma is essentially the main combinatorial result from [1]. The proof is included here for the sake of completeness.

**Lemma 6 (Rerouting in a disc).** *Let $G$ be a plane graph with $k$ pairs of terminals such that the DPP has a solution $\mathcal{P}$. Let $G$ contain a cycle $C$ bounding a closed disc $D$ in the plane, such that no terminal lies in $D$. Assume that every $D$-segment of $\mathcal{P}$ is simply an edge and, except for vertices and edges of $D$-segments, the interior of $D$ contains no other vertices or edges of $G$. Then, if there is a segment type that contains more than $2^k$ segments, then we can replace the outerplanar graph $O$ consisting of all $D$-segments of $\mathcal{P}$ by a new outerplanar graph $O'$ such that in $(G \setminus O) \cup O'$ the DPP (with the original terminals) has a solution and $|E(O')| < |E(O)|$.*

**Definition 10.** *Let $n, m \in \mathbb{N}$. An* untidy $(n \times m)$ grid *is a graph obtained from a set $\mathcal{H}$ of $n$ pairwise vertex-disjoint (*horizontal*) paths and a set $\mathcal{V} = \{V_1, \ldots, V_m\}$ of $m$ pairwise vertex-disjoint (*vertical*) paths as follows: Every path in $\mathcal{V}$ intersects every path in $\mathcal{H}$ in precicely one non-empty path, and each path $H \in \mathcal{H}$ consists of $m$ vertex-disjoint segments such that $V_i$ intersects $H$ only in its $i$th segment (for every $i \in \{1, \ldots, m\}$). A* subdivided untidy $(n \times m)$ grid *is obtained from an untidy $(n \times m)$ grid by subdividing edges.*

Let $\tau$ be a segment type in the plane graph. Recall that all the segments in a type are "parallel" to each other. We say that segments $S_1, \ldots, S_n \in \tau$ are *consecutive*, if they appear in this order (or in the reverse order) in the plane. Segment types that go far into the concentric cycles yield subdivided untidy grids. More precisely, we show the following lemma, that is an easy consequence of Lemma 1.

**Lemma 7.** *Let $l, n, r \in \mathbb{N}$ with $n \geq l - 1$. Let $\mathcal{P}$ be a cheap solution to the input instance. If there is a type $\tau$ of $D_n$-segments of $\mathcal{P}$ with $|\tau| \geq r$ such that $r$ consecutive segments of $\tau$ each contain a vertex of $D_{n-l+1}$, then $G$ contains a subdivided untidy $(2l \times r)$ grid as a subgraph, with the $r$ consecutive segments of $\tau$ as vertical paths, and suitable subpaths of $C_n, \ldots, C_{n-l+1}, C_{n-l+1}, \ldots, C_n$ (in this order) as horizontal paths.* □

The following lemma, whose proof is based on Lemmata 6 and 7, shows that we can reroute a sufficiently large segment type in the case that many segments of the type go far into the concentric cycles.

**Lemma 8 (Rerouting in an untidy grid).** *Let $n, k \in \mathbb{N}$ with $n \geq 2^{k-1} - 1$. Let $\mathcal{P}$ be a cheap solution to the input instance. Then $\mathcal{P}$ has no type $\tau$ of $D_n$-segments with $|\tau| \geq 2^k + 1$, such that each of $2^k + 1$ consecutive segments in $\tau$ contains a vertex in $D_{n-2^{k-1}+1}$.*

The following remark says that if we have a segment type of sufficiently large cardinality, then many segments will go far into the concentric cycles.

**Lemma 9.** *Let $n, l, r \in \mathbb{N}$. Let $\mathcal{P}$ be a cheap solution to the input instance. Let $\tau$ be a type of $D_n$-segments with $|\tau| \geq 2l + r$. Then $n \geq l - 1$ and $\tau$ contains $r$ consecutive segments such that each of them has a vertex in $D_{n-l+1}$.*

The next theorem is based on Lemmata 8 and 9 and Theorem 3.

**Theorem 4.** *Let $\mathcal{P}$ be a cheap solution to the input instance. Then there are at most $(8k-6)\cdot 2^k + 4k - 3$ $D_n$-segments of $\mathcal{P}$.*

Theorem 4 along with Lemma 1 yield the following.

**Theorem 5 (Irrelevant Vertex).** *Let $G$ be a plane graph with $k$ pairs of terminals, $n = (8k-6)\cdot 2^k + 4k - 2$, and let $G$ contain a sequence $C_0, \dots, C_n$ of concentric cycles bounding closed discs $D_0, \dots, D_n$, respectively, in the plane, such that no terminal of the* DPP *lies in $D_n$. Let $C_0 = \{v\}$, and assume that the* DPP *has a solution. Then the* DPP *has a solution that avoids $v$.*

Given a plane graph $G$ and a vertex $v$ we show how to check whether a particular vertex $v$ satisfies the conditions of Theorem 5. We set $C_0 = \{v\}$ and given $C_i$ we construct $C_{i+1}$ by performing a depth first search from a neighbor $u$ of a vertex in $C_i$, always chosing the rightmost edge leaving the vertex we are visiting. This search will either output an innermost cyclic walk (which then can be pruned to a cycle) around $C_i$ or determine that no such walk exists. In the case that a cycle $C_{i+1}$ is output, we check whether the cyclic walk contained a terminal $s_i$ or $t_i$. If it did, it means that this terminal lies on $C_{i+1}$ or in its interior. At this point (or when the search outputs that no cycle around $C_i$ exists), we have determined that there are $i$ tight concentric cycles around $v$ with no terminal in the interior of $C_i$. If $i > (8k-6)\cdot 2^k + 4k - 2$ this implies that $v$ satisfies the conditions of Theorem 5. Clearly this procedure can be implemented to run in linear time. This yields the following theorem.

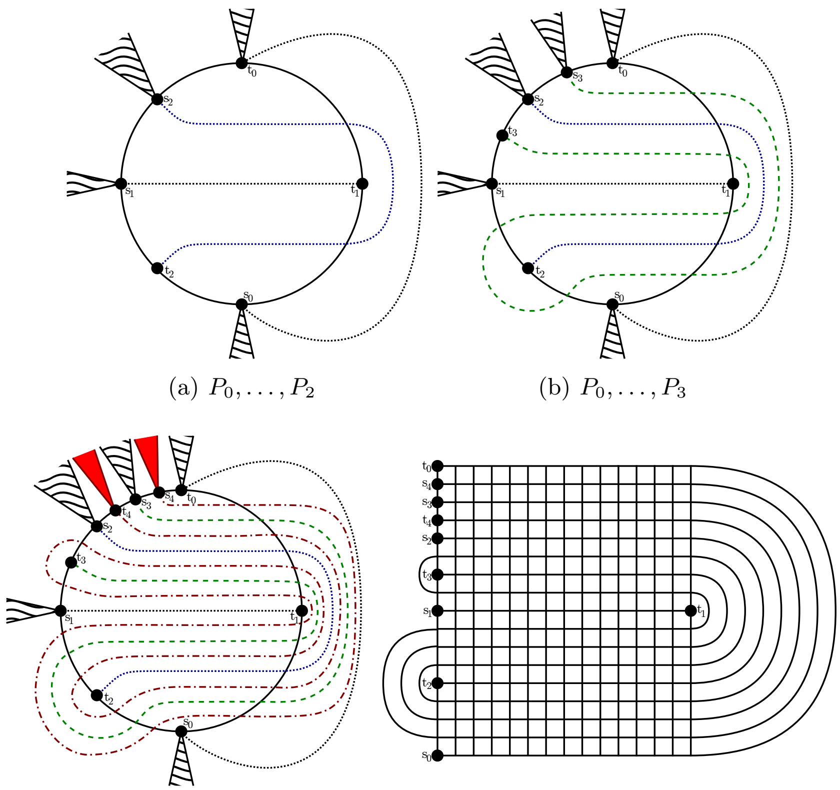
**Theorem 6.** *Let $G$ be a plane graph with $k$ pairs of terminals, there is an $O(|V(G)|^2)$ time algorithm that outputs an induced subgraph $G'$ of $G$ such that $tw(G') \leq 72\sqrt{2}k^{\frac{3}{2}} \cdot 2^k$ and $G$ is a YES-instance for* DPP *if and only if $G'$ is.*

## 4 The Lower Bound

Let $H \subseteq G$ be a subgraph of the plane graph $G$. An *inner vertex* of $H$ is a vertex that is not part of the boundary of $H$.

**Definition 11 (Crossing).** *Let $H \subseteq G$ be a subgraph of the plane graph $G$. We say that a path* crosses *the subgraph $H$ if it contains an inner vertex of $H$ and its endpoints are not inner vertices of $H$. For $k \in \mathbb{N}$ we say that a path $P = p_0, p_1, \dots, p_n$* crosses *$H$ $k$ times, if it can be split into $k$ paths $P_0 = p_0, p_1, \dots, p_{i_1}, P_1 = p_{i_1}, p_{i_1+1}, \dots p_{i_2}, \dots, P_{k-1} = p_{i_{k-1}}, p_{i_{k-1}+1}, \dots, p_n$ with each $P_i, i = 0, \dots k-1$ crossing $H$. The parts of the $P_i$ that do not lie outside of $H$ are called crossings of $H$.*

Intuitively, we construct our example from a grid $H$ of sufficient size. We add endpoints $s_0$ and $t_0$ on the boundary of the grid, mark the areas opposite to the grid as not part of the graph and connect $s_0$ to $t_0$ without crossing the grid. Now we continue to mark vertices by $s_i$ and $t_i$ in such a way that $P_i$ has to cross $H$ as often as possible (in order to avoid crossing $P_j, j < i$). Once $s_i$ and $t_i$ have been added we remove the area opposite to the grid from $s_i$ from the graph. Figure 1(a) shows the situation after doing this for $i$ up to 2. In this construction $P_0$ does not cross the grid at all, while $P_1$ crosses it once and $P_{i+1}$ crosses it twice as often as $P_i$ for $i > 0$: Let $k_i$ be the number of times $P_i$ crosses the grid. $k_0 = 0, k_1 = 1, k_{i+1} = 2k_i, k_i = 2^{i-1}, i > 0$. After the last $P_i$ has been added, the areas opposite to the grid from both $s_i$ and $t_i$ are removed from the graph as seen in Figure 1(c).




(a) $P_0, \ldots, P_2$ (b) $P_0, \ldots, P_3$

(c) $P_0, \ldots, P_4$ (d) Actual graph with grid

**Fig. 1.** Construction of graph and solution

Formally, to construct problem and graph with $k + 1$ terminals, we use a $((2^k + 1) \times (2^k + 1))$ grid. Let the vertices on the left boundary of the grid be $n_0, \ldots, n_{2^k}$. Terminals are assigned as follows: $t_0$ is the topmost vertex on the left boundary on the grid, $t_1$ the middle vertices on the right boundary. For all other terminals: $s_i := n_{2^{k-i}}, t_i := n_{3 \cdot 2^{k-i}}$. Then add edges going around the $t_i$ to the graph: For $i > 1, t_i = n_j$ add $n_{j-1}n_{j+1}, n_{j-2}n_{j+1}, \ldots, n_{j-2^{k-i}-1}n_{j+2^{k-i}-1}$, and on the right boundary of $H$ do the analogue for $t_1$. See Figure 1(d) for a graph constructed this way.

**Theorem 7.** *There is only one solution to the constructed DPP, all vertices of the graph lie on paths of the solution and the grid is crossed $2^k - 1$ times by such paths.*

In particular, $H$ has no *irrelevant* vertex in the sense of [19].

**Corollary 1.** *There is a planar graph $G$ with $k+1$ pairs of terminals such that*

- *$G$ contains a $((2^k+1) \times (2^k+1))$ grid as a subgraph,*
- *the disjoint paths problem on this input has a unique solution,*
- *the solution uses all vertices of $G$; in particular, no vertex of $G$ is irrelevant.*

*Vital linkages and tree-width* We refer the reader to [2] for the definitions of *tree-width* and *path-width.* A linkage $L$ in a graph $G$ is a *vital linkage* in $G$, if $V(\bigcup L) = V(G)$ and there is no other linkage $L' \neq L$ in $G$ with the same pattern as $L$.

**Theorem 8 (Robertson and Seymour [20]).** *There are functions $f$ and $g$ such that if $G$ has a vital linkage with $k$ components then $G$ has tree-width at most $f(k)$ and path-width at most $g(k)$.*

Recall that the $(n \times n)$ grid has path-width $n$ and tree-width $n$. Our example yields a lower bound for $f$ and $g$:

**Corollary 2.** *Let $f$ and $g$ be as in Theorem 8. Then $2^{k-1}+1 \leq f(k)$ and $2^{k-1}+1 \leq g(k)$.*

*Proof.* Looking at the graph $G$ and DPP constructed above the solution to the DPP is, due to its uniqueness, a vital linkage for the graph $G$. $G$ contains a $((2^k+1) \times (2^k+1))$ grid as a minor. The tree-width of such a grid is $2^k+1$ , its path-width $2^k+1$ [6]. Thus we get lower bounds $2^{k-1}+1 \leq f(k), g(k)$ for the functions $f$ and $g$. □

**Acknowledgment.** We thank Ken-ichi Kawarabayashi and Paul Wollan for providing details on the bounds in [10].

## References

1. I. Adler, S. G. Kolliopoulos, and D. Thilikos. Planar disjoint paths completion. Submitted for publication, 2011.
2. H. L. Bodlaender. A tourist guide through treewidth. *Acta Cybernet.*, 11(1-2):1–21, 1993.
3. Anuj Dawar, Martin Grohe, and Stephan Kreutzer. Locally excluding a minor. In *LICS'07*, pages 270–279. IEEE Computer Society, 2007.
4. Anuj Dawar and Stephan Kreutzer. Domination problems in nowhere-dense classes. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2009)*, pages 157–168, 2009.
5. R. Diestel. *Graph Theory.* Springer, 2005.

6. John Ellis and Robert Warren. Lower Bounds on the Pathwidth of some Grid-like Graphs. *Discrete Applied Mathematics*, 156(5):545–555, 2008.
7. Petr A. Golovach, M. Kaminski, D. Paulusma, and D. M. Thilikos. Induced packing of odd cycles in a planar graph. In *Proceedings of the 20th International Symposium on Algorithms and Computation (ISAAC 2009)*, volume 5878 of *Lecture Notes in Comput. Sci.*, pages 514–523. Springer, Berlin, 2009.
8. Ken-ichi Kawarabayashi and Yusuke Kobayashi. The induced disjoint path problem. In *Proceedings of the 13th Conference on Integer Programming and Combinatorial Optimization (IPCO 2008)*, volume 5035 of *Lect. Notes Comp. Sc.*, pages 47–61. Springer, Berlin, 2008.
9. Ken-ichi Kawarabayashi and Bruce Reed. Odd cycle packing. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC 2010)*, pages 695–704, New York, NY, USA, 2010. ACM.
10. Ken-ichi Kawarabayashi and Paul Wollan. A shorter proof of the graph minor algorithm: the unique linkage theorem. In *Proc. of the 42nd annual ACM Symposium on Theory of Computing (STOC 2010)*, pages 687–694, 2010.
11. Yusuke Kobayashi and Ken-ichi Kawarabayashi. Algorithms for finding an induced cycle in planar graphs and bounded genus graphs. In *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2009)*, pages 1146–1155. ACM-SIAM, 2009.
12. M. R. Kramer and J. van Leeuven. The complexity of wire-routing and finding minimum area layouts for arbitrary VLSI circuits. *Advances in Comp. Research*, 2:129–146, 1984.
13. Daniel Lokshtanov, Daniel Marx, and Saket Saurabh. Slightly superexponential parameterized problems. In *22st ACM–SIAM Symposium on Discrete Algorithms (SODA 2011)*, pages 760–776, 2011.
14. J. F. Lynch. The equivalence of theorem proving and the interconnection problem. *ACM SIGDA Newsletter*, 5:31–36, 1975.
15. Matthias Middendorf and Frank Pfeiffer. On the complexity of the disjoint paths problem. *Combinatorica*, 13(1):97–107, 1993.
16. B. Reed, N. Robertson, A. Schrijver, and P.D. Seymour. Finding dsjoint trees in planar graphs in linear time. In N. Robertson and P.D. Seymour, editors, *Graph Structure Theory*, volume 147 of *Contemporary Mathematics*, pages 295–302. American Mathematical Society, 1991.
17. Neil Robertson and P. D. Seymour. Graph minors. XIII. The disjoint paths problem. *J. Combin. Theory Ser. B*, 63(1):65–110, 1995.
18. Neil Robertson and Paul Seymour. Graph minors. XXI. Graphs with unique linkages. *J. Combin. Theory Ser. B*, 99(3):583–616, 2009.
19. Neil Robertson and Paul D. Seymour. Graph Minors. XXII. Irrelevant vertices in linkage problems. *Journal of Combinatorial Theory, Series B*. (to appear).
20. Neil Robertson and Paul D. Seymour. Graph minors. XXI. Graphs with unique linkages. *Journal of Combinatorial Theory, Series B*, 99(3):583–616, 2009.
21. A. Schrijver. Finding k disjoint paths in a directed planar graph. *SIAM J. Comput.*, 23(4):780–788, 1994.
22. Alexander Schrijver. *Combinatorial optimization. Polyhedra and efficiency. Vol. A*. Springer-Verlag, Berlin, 2003.
23. Jens Vygen. NP-completeness of some edge-disjoint paths problems. *Discrete Appl. Math.*, 61(1):83–90, 1995.