

Moving Vertices to Make Drawings Plane^{*}

Xavier Goaoc¹, Jan Kratochvíl², Yoshio Okamoto^{3**}, Chan-Su Shin^{4***}, and Alexander Wolff⁵

¹ LORIA – INRIA Lorraine, Nancy, France.

`goaoc@loria.fr`

² Dept. Applied Math. and Inst. Theoret. Comp. Science, Charles Univ., Czech Rep.

`honza@kam.mff.cuni.cz`

³ Dept. Information and Computer Sciences, Toyohashi Univ. of Technology, Japan.

`okamotoy@ics.tut.ac.jp`

⁴ School of Digital Inform. Eng., Hankuk Univ. of Foreign Studies, Yongin, Korea.

`cssin@hufs.ac.kr`

⁵ Faculteit Wiskunde en Informatica, TU Eindhoven, the Netherlands.

`http://www.win.tue.nl/~awolff`

Abstract. In John Tantaló’s on-line game *Planarity* the player is given a non-plane straight-line drawing of a planar graph. The aim is to make the drawing plane as quickly as possible by moving vertices. In this paper we investigate the related problem MINMOVEDVERTICES which asks for the minimum number of vertex moves. First, we show that MINMOVEDVERTICES is NP-hard and hard to approximate. Second, we establish a connection to the graph-drawing problem 1BENDPOINTSEMBEDDABILITY, which yields similar results for that problem. Third, we give bounds for the behavior of MINMOVEDVERTICES on trees and general planar graphs.

1 Introduction

It is somewhat surprising that many people still draw graphs by hand, usually not on a piece of paper but on a computer display. Modern technology gives us the means to edit a drawing by dragging vertices. Even when we use an automatic graph-drawing tool, we often do some manual polishing to obtain nicer drawings.

In this paper, we consider the problem of editing a given drawing to obtain another drawing that fulfills a certain criterion. We restrict ourselves to straight-line drawings of planar graphs. Our edit operation is “moving vertices.” When

* This work was started on the 9th “Korean” Workshop on Computational Geometry and Geometric Networks organized by A. Wolff and X. Goaoc, July 30–August 4, 2006 in Schloß Dagstuhl, Germany. Further contributions were made at the 2nd Workshop on Graph Drawing and Computational Geometry organized by W. Didimo and G. Liotta, March 11–16, 2007 in Bertinoro, Italy.

** Partially supported by Grant-in-Aid for Scientific Research from Ministry of Education, Science and Culture, Japan, and Japan Society for the Promotion of Science.

*** Supported by Research Grant 2007 of the Hankuk University of Foreign Studies.

we move a vertex v to a new position, the incident edges are redrawn so that v is again connected to its adjacent vertices by straight-line segments. Our criterion is planarity. According to the famous theorem of Wagner [15], Fáry [2], and Stein [12] every planar graph has a plane straight-line drawing. We want to obtain such an embedding from a given (usually non-plane) straight-line drawing. Our goal is to minimize the number of vertices to move. This is a natural question because the less vertices we move the better the *mental map* [8] of an observer is preserved when making a given drawing plane, e.g., in a step-by-step fashion. Note that for a given straight-line drawing the minimum number of moves can also be seen as the edit distance from the closest plane drawing.

At the 5th Czech-Slovak Symposium on Combinatorics in Prague in 1998, Mamoru Watanabe asked the following question, which concerns a special case of our problem: Is it true that every polygon P with n vertices can be untangled, i.e., turned into a non-crossing polygon, by moving at most εn of its vertices for some absolute constant $\varepsilon < 1$? Pach and Tardos [9] have answered this question in the negative by showing that there must be polygons where at most $O((n \log n)^{2/3})$ of the vertices can be kept fixed. They also gave a simple algorithm (which can be implemented in $O(n \log n)$ time) that always keeps more than \sqrt{n} vertices. In a longer version of this paper [3] we show that their algorithm is not optimal. Pach and Tardos [9] in turn asked the following question: can any straight-line drawing of any planar graph with n vertices be made plane by vertex moves while keeping $\Omega(n^\gamma)$ vertices fixed for some absolute constant $\gamma > 0$? We still do not know the answer to this question, but we report some progress.

There is a popular on-line game that is related to the problem of Pach and Tardos. In John Tantaló's game *Planarity* [13] the player is given a non-plane straight-line drawing of a planar graph. The player can move vertices, which always keep straight-line connections to their neighbors. The aim is to make the drawing plane as quickly as possible. We study the game from three view points: (a) algorithms, (b) mathematics (upper-bound constructions), and (c) complexity. Our complexity results (detailed below) made us understand why it is so hard to play the game well.

Formalization. In this paper, a *drawing* of a graph $G = (V, E)$ will always mean a straight-line embedding of G in the plane \mathbb{R}^2 . Since such an embedding is completely defined by the position of the vertices, it corresponds to an injective map $\delta: V \rightarrow \mathbb{R}^2$. A drawing is *plane* if no two edges cross, i.e., they are only allowed to intersect in a common endpoint. A graph is planar if it admits a plane drawing; trivially not every drawing of a planar graph is plane.

The *vertex-moving distance* d between two drawings δ and δ' of a graph G is defined as the number of vertices of G whose images under δ and δ' differ:

$$d(\delta, \delta') = |\{v \in V \mid \delta(v) \neq \delta'(v)\}|.$$

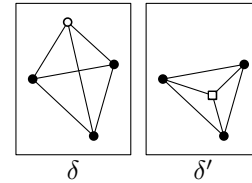


Fig. 1: Two drawings of K_4 : δ is not plane, δ' is plane; $d(\delta, \delta') = 1$.

This distance can easily be computed. Given our edit operation, d represents the edit distance for straight-line drawings of graphs. Figure 1 shows an example. Using d we can express the central question of this paper as follows.

How close is a given drawing of a planar graph to being plane with respect to the vertex-moving distance d ?

For a drawing δ of a planar graph G , denote by $\text{MMV}(G, \delta)$ the minimum number of vertices that need to be moved in order to make δ plane. MMV measures distance from planarity: $\text{MMV}(G, \delta) = \min_{\delta'} d(\delta, \delta')$, where δ' ranges over all plane drawings of G . This gives rise to the following computational problem.

MINMOVEDVERTICES(G, δ): Given a drawing δ of a planar graph G , find a plane drawing δ' of G with $d(\delta, \delta') = \text{MMV}(G, \delta)$.

Sometimes this question is better studied from the symmetric point of view. Given a drawing δ of a graph G , we denote by $\text{MKV}(G, \delta)$ the maximum number of vertices that remain fixed when making δ plane. We refer to such vertices as *fixed* vertices. Obviously it holds that $\text{MKV}(G, \delta) = n - \text{MMV}(G, \delta)$, where n is the number of vertices of G . MKV measures similarity with the closest plane drawing. The corresponding problem is defined as follows.

MAXKEPTVERTICES(G, δ): Given a drawing δ of a planar graph G , find a plane drawing δ' of G with $\text{MKV}(G, \delta)$ fixed vertices.

Let $\text{MKV}(G) = \min_{\delta \text{ drawing of } G} \text{MKV}(G, \delta)$ denote the maximum number of vertices of G that can be kept fixed when starting with the worst-possible drawing of G .

Our results. First, we prove that the decision versions of **MAXKEPTVERTICES** and equivalently **MINMOVEDVERTICES** are NP-hard, see Section 2. We also prove that **MINMOVEDVERTICES** is hard to approximate. Namely, for any $\varepsilon \in (0, 1]$ there is no polynomial-time $n^{1-\varepsilon}$ -approximation algorithm for **MINMOVEDVERTICES** unless $\mathcal{P} = \mathcal{NP}$.

Second, we establish a connection to a well-known graph-drawing problem, namely **1BENDPOINTSEMBEDDABILITY**. Given a planar graph $G = (V, E)$ with n vertices we say that a graph is *k-bend (point-set) embeddable* if for any set S of n points in the plane there is a one-to-one correspondence between V and S such that G can be *k-bend (point-set) embedded on S* , i.e., the edges of G can be drawn as non-crossing simple polygonal chains with at most k bends. Kaufmann and Wiese [5] showed that (a) every 4-connected planar graph is 1-bend embeddable, (b) every planar graph is 2-bend embeddable, and (c) given a planar graph $G = (V, E)$ and set S of n points on a line, it is NP-complete to decide whether there is a correspondence between V and S that makes it possible to 1-bend embed G on S . We strengthen their result by showing that the problem remains hard even if the correspondence is given. We also show that an optimization version of the problem is hard to approximate.

Third, we give bounds on $\text{MKV}(G)$ for trees and general planar graphs, see Sections 3 and 4, respectively. Table 1 summarizes the best known bounds. A lower bound of k means: we can make any drawing of any graph G in the given

graph class	where	lower bound	upper bound
cycles	Pach & Tardos [9]	$\lceil \sqrt{n} \rceil$	$O((n \log n)^{2/3})$
trees	Section 3 [3]	$\lceil \sqrt{n/3} \rceil$	$\lceil n/3 \rceil + 4$
outerplanar graphs	Spillner & Wolff [11]	$\sqrt{n-1}/3$	$2\sqrt{n-1} + 1$
planar graphs	Section 4	3	$\lceil \sqrt{n-2} \rceil + 1$
	Spillner & Wolff [11]	$\frac{1}{3} \sqrt{\frac{2(\log n) - 2}{\log \log n} - 1}$	

Table 1: Best known bounds for $\text{MKV}(G)$, where n is the number of vertices of G .

graph class plane while keeping at least k vertices fixed. An upper bound of k means: there is an arbitrarily large graph G in the given graph class and a drawing δ of G such that at most k vertices can stay fixed when making δ plane.

Independent recent results. In May 2007, Verbitsky [14] considered the function $\text{MMV}(G) = \max_{\delta \text{ plane drawing of } G} \text{MMV}(G, \delta)$, to which he refers as the *shift complexity* of a graph. He, too, observed that $\text{MMV}(G) \leq n-3$ (i.e., $\text{MKV}(G) \geq 3$) for any planar graph G with $n \geq 3$ vertices. Further he gave two linear lower bounds on $\text{MMV}(G)$ depending on the connectivity of G . By reduction from independent set in line-segment intersection graphs he showed that computing $\text{MMV}(G, \delta)$ is NP-hard.

In June 2007, Kang et al. [4] investigated the problem of straightening the edges of a given plane drawing (with curved edges) through vertex moves. They showed that for arbitrary large n , there exist an n -vertex graph G_n and a *plane* (curved-edge) drawing δ_n of G_n with $\text{MKV}(G_n, \delta_n) = O(n^{2/3})$. Our upper bound of $O(\sqrt{n})$ (see Theorem 7) is stronger, but our initial drawings are not plane.

In September 2007, Spillner and Wolff [11] showed that $\text{MKV}(G)$ actually grows with the size of G and gave asymptotically tight bounds for outerplanar graphs, see Table 1.

2 Complexity

In this section, we investigate the complexity of MINMOVEVERTICES and of $\text{1BENDPOINTSETEMBEDDABILITY}$ with given vertex–point correspondence.

Theorem 1. *Given a planar graph G , a drawing δ of G , and an integer $K > 0$, it is NP-hard to decide whether $\text{MMV}(G, \delta) \leq K$.*

Proof. Our proof is by reduction from PLANAR3SAT , which is known to be NP-hard [7]. An instance of PLANAR3SAT is a 3SAT formula φ whose variable-clause graph is planar. Note that that graph can be laid out (in polynomial time) such that variables correspond to points on the x -axis and clauses correspond to non-crossing three-legged “combs” above or below the x -axis [6], see Fig. 2.

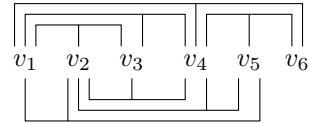


Fig. 2: Embedding of a planar 3-SAT formula.

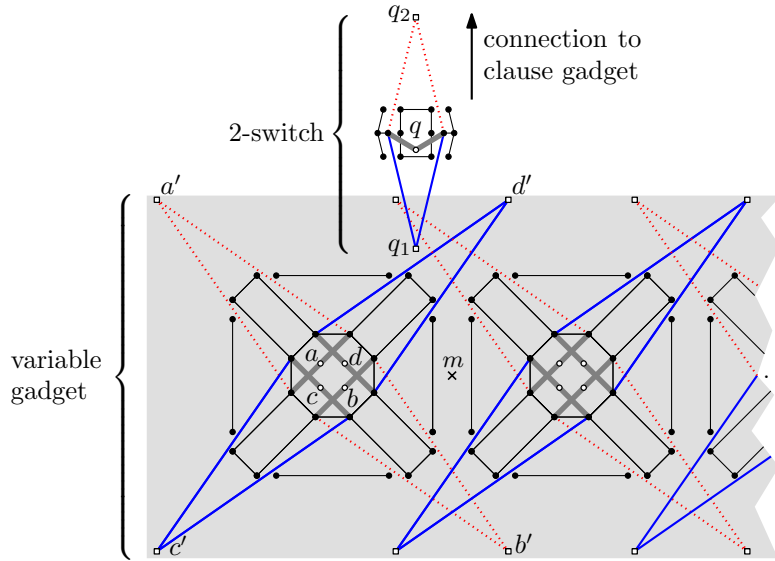


Fig. 3: Edge positions in variable gadget: immobile (thin solid black) and mobile (very thick solid gray). The predestined positions of mobile edges either correspond to *true* (thick solid blue) or to *false* (thick dotted red).

Let v and c be the numbers of variables and clauses of φ , respectively. We now construct a graph G_φ with a straight-line drawing δ_φ such that the following holds: δ_φ can be made plane by moving at most K vertices if and only if φ is satisfiable. We fix K later.

Our graph G_φ consists of two types of substructures (or *gadgets*), modeling the variables and clauses of φ . In our gadgets, see Figs. 3 and 4, there are two types of vertices and edges; those that *may* move and those that are *meant* not to move. We refer to the two types as *mobile* and *immobile*. If φ has a satisfying truth assignment, *all* immobile (and a few mobile) vertices are fixed, otherwise at least one immobile vertex must move. In the figures, immobile vertices are marked by black disks, mobile vertices by circles, and their predestined positions by little squares. Immobile edges are drawn as thin solid black line segments, mobile edges as very thick solid gray line segments, and their predestined positions are drawn as thick colored line segments.

Now consider the gadget for some variable x in φ , see the shaded area in Fig. 3. The gadget consists of a horizontal chain of a certain number of roughly square *blocks*. Each block consists of 28 vertices and 32 edges. Each block has four mobile vertices, each incident to two very thick gray edges. In Fig. 3 the four mobile vertices of the leftmost block are labeled in clockwise order a , d , b , and c . Note that the gray edges incident to a and b intersect those incident to c and d . Thus either both a and b or both c and d must move to make the block plane. Each mobile vertex $w \in \{a, b, c, d\}$ can move into exactly one position

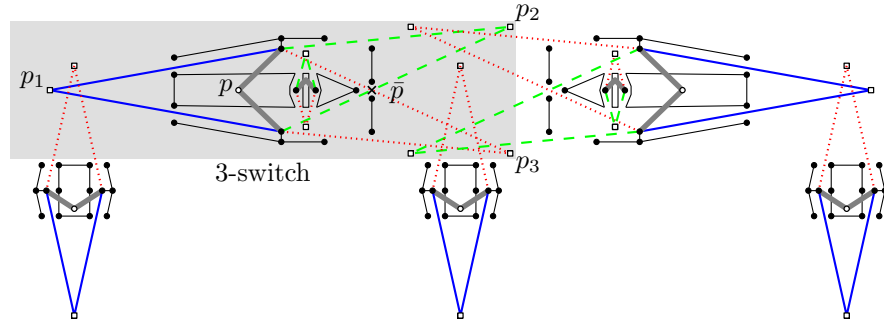


Fig. 4: A clause gadget consists of three big 2-switches and two 3-switches. Each 3-switch contains another small 2-switch. Note that not all immobile vertices are marked.

w' (up to wiggling). The resulting incident edges are drawn by thick dotted red and thick solid blue line segments, respectively. Note that neighboring blocks in the chain are placed such that the only way to make them plane simultaneously is to move *corresponding* pairs of vertices and edges. Thus either all blocks of a variable gadget use the blue line segments or all use the red line segments. These two ways to make a variable gadget plane correspond to the values *true* and *false* of the variable, respectively.

For each of the $3c$ literals in φ we connect the gadget of the corresponding variable to the gadget of the clause that contains the literal. Each block of each variable gadget is connected to a specific clause gadget above or below the variable gadget, thus there are $3c$ blocks in total. Each connection is realized by a part of G_φ that we call a *2-switch*. A 2-switch consists of 15 vertices and 14 edges. The mobile vertex q of the 2-switch in Fig. 3 is incident to two very thick gray edges that intersect two immobile edges of the 2-switch. Thus q must move. There are (up to wiggling) two possible positions, namely q_1 and q_2 , see Fig. 3.

The 2-switch in Fig. 3 corresponds to a positive literal. For negated literals the switch must be mirrored either at the vertical or at the horizontal line that runs through the point m . Note that a switch can be stretched vertically in order to reach the right clause gadget. Further note that if a literal is *false*, the mobile vertex of the corresponding 2-switch must move away from the variable gadget and towards the clause gadget to which the 2-switch belongs. In that case we say that the 2-switch *transmits pressure*.

A clause gadget consists of three vertical 2-switches and two horizontal *3-switches*. A 3-switch consists of 23 vertices and 18 edges plus a small “inner” 2-switch, see the shaded area in Fig. 4. Independently from the other, each of the two 3-switches can be stretched horizontally in order to reach vertically above the variable gadget to which it connects via a 2-switch. The mobile vertex p of the left 3-switch in Fig. 4 is incident to two very thick gray edges that intersect two immobile edges of the 3-switch. Thus p must move. There are (again up to wiggling) three possible positions, namely p_1 , p_2 , and p_3 . Note that we need the inner 2-switch, otherwise there would be a fourth undesired position for moving

p , namely the one labeled \bar{p} in Fig. 4. By construction a clause gadget can be made plane by only moving the mobile vertices of all switches if and only if at most two of the three big 2-switches transmit pressure, i.e., if at least one of the literals in the clause is *true*.

The graph G_φ that we have now constructed has $O(c)$ vertices, $O(c)$ edges, and $X = 26c$ crossings; $4 \cdot 3c$ in blocks and $2 \cdot 7c$ in switches. By moving any mobile vertex to any of its predestined positions, a pair of original crossings disappears. If φ is satisfiable, G_φ can be made plane by moving $K = X/2$ mobile vertices since no new crossings are introduced. If φ is not satisfiable, there is at least one pair of crossings that cannot be eliminated by moving the corresponding mobile vertex alone since all its predestined positions are blocked. Thus at least *two* vertices must be moved to eliminate that pair of crossings—and still all the other $K - 1$ pairs of crossings must be eliminated by moving at least one vertex per pair, totaling in at least $K + 1$ moves. Thus φ is satisfiable if and only if G_φ can be made plane by moving exactly K (mobile) vertices.

Since there is enough slack in our construction, it is possible to place vertices at integer coordinates whose total length is polynomial in the length L of a binary encoding of φ . This and the linear size of G_φ yield that our reduction is polynomial in L . \square

We now consider the approximability of MINMOVEDVERTICES. Since $\text{MMV}(G, \delta) = 0$ for plane drawings, we cannot use the usual definition of an approximation factor unless we slightly modify our objective function. Let $\text{MMV}'(G, \delta) = \text{MMV}(G, \delta) + 1$ and call the resulting decision problem MINMOVEDVERTICES'. Now we can modify the above reduction to get a non-approximability result.

Theorem 2. *For any fixed real $\varepsilon \in (0, 1]$ there is no polynomial-time $n^{1-\varepsilon}$ -approximation algorithm for MINMOVEDVERTICES' unless $\mathcal{P} = \mathcal{NP}$.*

Proof. Let n_φ be the number of vertices of the graph G_φ with drawing δ_φ that we constructed above. We go through all immobile vertices v of G_φ . Let N_v be the neighborhood of v . We replace v by a star with central vertex v adjacent to the vertices in N_v and $n_\varphi^{(3-\varepsilon)/\varepsilon}$ additional new vertices infinitesimally close to v . Let G be the resulting graph, δ its drawing, and $n \leq (n_\varphi^{(3-\varepsilon)/\varepsilon} + 1) \cdot n_\varphi$ the number of vertices of G . Note that φ is satisfiable if and only if $\text{MMV}'(G, \delta) = \text{MMV}'(G_\varphi, \delta_\varphi) = K + 1$. Otherwise, additionally at least one complete star has to be moved, i.e., $\text{MMV}'(G, \delta) \geq K + n_\varphi^{(3-\varepsilon)/\varepsilon} + 2$. Note that G can be constructed in polynomial time since ε is fixed.

Now suppose there was a polynomial-time $n^{1-\varepsilon}$ -approximation algorithm \mathcal{A} for MINMOVEDVERTICES'. We can bound its approximation factor by $n^{1-\varepsilon} \leq ((n_\varphi^{(3-\varepsilon)/\varepsilon} + 1) \cdot n_\varphi)^{1-\varepsilon} \leq (2n_\varphi^{(3-\varepsilon)/\varepsilon} \cdot n_\varphi)^{1-\varepsilon} = 2^{1-\varepsilon} n_\varphi^{(3-3\varepsilon)/\varepsilon} \leq 2n_\varphi^{(3-3\varepsilon)/\varepsilon}$. Now let M be the number of moves that \mathcal{A} needs to make δ plane. If φ is satisfiable, then $M \leq \text{MMV}'(G, \delta) \cdot n^{1-\varepsilon} = (K + 1) \cdot n^{1-\varepsilon} \leq (n_\varphi + 1) \cdot 2n_\varphi^{(3-3\varepsilon)/\varepsilon} = 2n_\varphi^{(3-2\varepsilon)/\varepsilon} + O(n_\varphi^{(3-3\varepsilon)/\varepsilon})$. On the other hand, if φ is unsatisfiable, then $M \geq \text{MMV}'(G, \delta) \geq n_\varphi^{(3-\varepsilon)/\varepsilon}$. Since we can assume that n_φ is sufficiently large, the

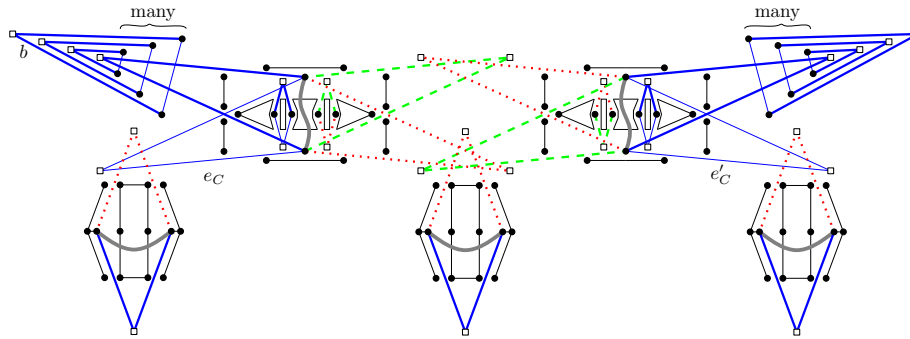


Fig. 5: Gadget of clause C for the non-approximability proof concerning the number of edges with one bend. The edges e_C and e'_C can now be drawn in *four* combinatorially different ways (thin solid blue vs. thick solid blue vs. dotted red vs. dashed green). This makes sure that there always is a drawing with at most one bend per edge. However, if the given planar 3SAT formula φ has no satisfying truth assignment, then for every truth assignment there is a clause that evaluates to *false*, and in the corresponding gadget a large number of edges of type b needs a bend.

result of algorithm \mathcal{A} (i.e., the number M) tells us whether φ is satisfiable. So either our assumption concerning the existence of \mathcal{A} is wrong, or we have shown the NP-hard problem PLANAR3SAT to lie in \mathcal{P} , which in turn would mean that $\mathcal{P} = \mathcal{NP}$. \square

We now state a hardness result that establishes a connection between MINMOVEVERTICES and the well-known graph-drawing problem 1BEND-POINTSEMBEDDABILITY. The proof uses nearly the same gadgets as in the proof of Theorem 1: Set G'_φ to a copy of G_φ where each length-2 path (u, v, w) containing a mobile vertex v is replaced by the edge $\{u, w\}$. The vertices of G'_φ are mapped to the corresponding vertices in δ_φ . Then it is not hard to see that G'_φ has a 1-bend drawing iff the given planar-3SAT formula φ is satisfiable.

Theorem 3. *Given a planar graph $G = (V, E)$ with $V \subset \mathbb{R}^2$, it is NP-hard to decide whether G has a plane drawing with at most one bend per edge.*

Now suppose we already know that G has a plane drawing with at most one bend per edge. Then it is natural to ask for a drawing with as few bends as possible. Let $\beta(G)$ be 1 plus the minimum number of bends over all plane one-bend drawings of G . Then we can show the following hardness-of-approximation result concerning bend minimization.

Corollary 1. *Given a fixed $\varepsilon \in (0, 1]$ and a graph $G = (V, E)$ with $V \subset \mathbb{R}^2$ that has a plane one-bend drawing, it is NP-hard to approximate $\beta(G)$ within a factor of $n^{1-\varepsilon}$.*

For the proof we slightly change the clause gadget in the proof of Theorem 1, see Figure 5. For the calculations, see the proof of Theorem 2.

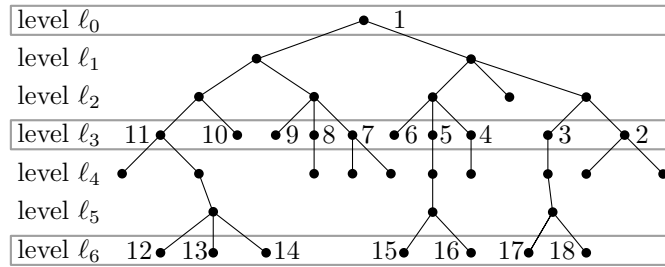


Fig. 6: The ordering of vertices in L_0 .

3 Trees

In this section we give a lower bound on MKV for trees. We use the following well-known theorem.

Theorem 4 (Erdős and Szekeres [1]). *Let $A = (a_1, \dots, a_n)$ be a sequence of n different real numbers. If $n \geq sr + 1$ then A has an increasing subsequence of $s + 1$ terms or a decreasing subsequence of $r + 1$ terms.*

In particular, this theorem implies that a sequence of n distinct integers always contains a monotone subsequence of length at least $\sqrt{n-1} + 1 \geq \lceil \sqrt{n} \rceil$.

Theorem 5. $\text{MKV}(T) \geq \lceil \sqrt{n/3} \rceil$ for any n -vertex tree T .

Proof. Let δ be an arbitrary drawing of T . We pick an arbitrary root r of T . Let $h \geq 0$ be the height of T with respect to r . For $i = 0, \dots, h$ let *level* ℓ_i be the set of vertices of T that are at tree distance i from r . For $j \in \{0, 1, 2\}$ let L_j be the union of all ℓ_i with $i \equiv j \pmod 3$. According to the pigeon-hole principle at least one of the three sets, say L_0 , contains at least $n/3$ vertices. We label the vertices of L_0 with the integers from 1 to $|L_0|$ such that (i) all vertices in the same level are consecutive in alternating directions, i.e., from left to right for every even-numbered level in L_0 and from right to left for every odd-numbered level in L_0 , and (ii) a level closer to the root gets smaller labels, see Fig. 6.

Let ℓ be a line, say the x -axis, such that the projection π orthogonal to ℓ does not map any two vertices of the drawing δ to the same point. The image of π induces an ordering of the vertices in L_0 . By Theorem 4, this ordering contains a monotone subsequence $F_0 \subset L_0$ of at least $\lceil \sqrt{n/3} \rceil$ vertices.

We fix the vertices in F_0 . First we move the vertices in $L_0 \setminus F_0$. Let B be an axis-parallel rectangle such that $F_0 \subset B$ and no point of F_0 lies on the boundary of B . If $1 \notin F_0$ ($|L_0| \notin F_0$), move it to any point on the left (right) edge of B . For any two vertices j, k that are consecutive in $F_0 \cup \{1, |L_0|\}$, move vertices $j + 1, \dots, k - 1$ in an equidistant manner on the line segment \overline{jk} .

We draw nested \sqcap -shaped corridors between vertices in ℓ_i and their respective children in ℓ_{i+3} if i is even. If i is odd, we use \sqcup -shaped corridors, see Fig. 7. Due to our labeling scheme no two such corridors intersect. Finally, the vertices

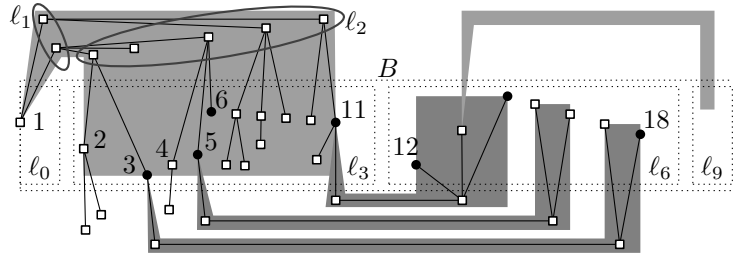


Fig. 7: Corridors connect the vertices in L_0 . Vertices in F_0 are marked by black disks.

in $L_1 \cup L_2$ go to positions near the bends of the corridors (see levels ℓ_1 and ℓ_2 in Fig. 7), which allows us to connect vertices in ℓ_i and ℓ_{i+3} by two-bend edges. \square

Remark 1. The proof of Theorem 5 yields an $O(n \log n)$ -time algorithm for making drawings of trees plane. It uses that the longest monotone subsequence of a sequence of n integers can be found in $O(n \log n)$ time [10].

4 Planar graphs

We now give bounds for the case of general planar graphs. We start with a rather trivial lower bound.

Theorem 6. $MKV(G) \geq 3$ for any planar graph G with $n \geq 3$ vertices.

Proof. Let δ be any drawing of G . Any planar graph admits a plane drawing δ_1 in which no three points are collinear and a plane drawing δ_2 in which some triplet of points is collinear. If there are three vertices v_1, v_2 , and v_3 whose images under δ are not collinear, we can find an affine transform L that maps $\delta_1(v_i)$ to $\delta(v_i)$. Since $L \circ \delta_1$ is a plane drawing of G that agrees with δ on $\{v_1, v_2, v_3\}$ it follows that $MKV(G, \delta) \geq 3$. If the images of all vertices are aligned under δ , we apply the same argument with δ_2 . \square

We now give an upper bound for general planar graphs that is better than the upper bound $O((n \log n)^{2/3})$ of Pach and Tardos [9] for cycles. Our construction uses the sequence $\sigma_q =$

$$(q-1)q, (q-2)q, \dots, 2q, q, \underline{0}, 1+(q-1)q, \dots, 1+q, \underline{1}, \dots, q^2-1, \dots, (q-1)+q, \underline{q-1}.$$

Note that σ_q can be written as $(\sigma_q^0, \sigma_q^1, \dots, \sigma_q^{q-1})$, where $\sigma_q^i = ((q-1)q + i, (q-2)q + i, \dots, 2q + i, q + i, i)$ is a subsequence of length q . Thus σ_q consists of q^2 distinct numbers. Note that the longest monotone subsequence of σ_q has length q .

Theorem 7. For any integer n_0 there exists a planar graph G with $n \geq n_0$ vertices and $MKV(G) \leq \lceil \sqrt{n-2} \rceil + 1$.

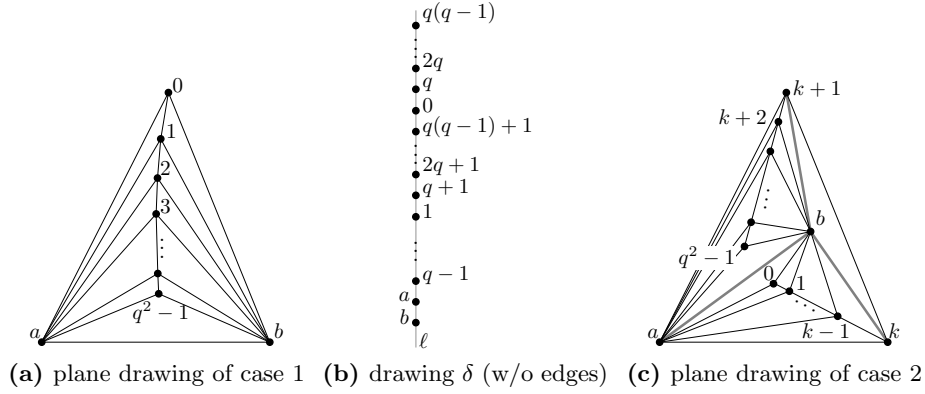


Fig. 8: Drawings of graph G_q (proof of Theorem 7).

Proof. For $q \geq 1$ we define the graph G_q as a chain of q^2 vertices all connected to the two endpoints of an edge $\{a, b\}$, see Fig. 8a. Let δ be the drawing of G_q where the vertices forming the chain are placed on a vertical line ℓ in the order given by σ_q . We place the vertices a and b below the others on ℓ , see Fig. 8b. Let δ' be a plane drawing of G_q with $\text{MMV}(G_q, \delta) = d(\delta, \delta')$. Since all faces of G_q are 3-cycles, the outer face in δ' is a triangle. All faces of G_q contain a or b . This has two consequences. First, a and b must move to new positions in δ' , otherwise all other vertices would have to move. Second, at least one of them, say a , appears on the outer face.

Case 1: Vertex b also lies on the outer face.

Then there are just two possibilities for the embedding of G_q : as in Fig. 8a or with the indices of all vertices reversed, i.e., vertex i becomes $q^2 - i - 1$. Now let $0 \leq i < j < k \leq q^2 - 1$ be three fixed vertices. By symmetry we can assume that j lies in $\Delta(a, b, i)$. Then k also lies in $\Delta(a, b, i)$ since the chain connecting j to k does not intersect the sides of this triangle. Note that k cannot lie between i and j on ℓ as otherwise one of the edges $\{a, k\}$ and $\{b, k\}$ would intersect the polygonal chain connecting i to j . Thus, each triplet of fixed vertices forms a monotone sequence along ℓ . This in turn yields that *all* fixed vertices in $\{0, \dots, q^2 - 1\}$ form a monotone sequence along ℓ . Due to the construction of σ_q such a sequence has length at most $q = \lceil \sqrt{n - 2} \rceil$.

Case 2: Vertex b does not lie on the outer face.

Then the outer face is of the form $\Delta(a, k, k + 1)$ with $0 \leq k \leq q^2 - 2$. The three edges $\{b, a\}$, $\{b, k\}$, and $\{b, k + 1\}$ incident to b split $\Delta(a, k, k + 1)$ into the three triangles $\Delta(a, k, b)$, $\Delta(a, b, k + 1)$, and $\Delta(b, k, k + 1)$, see Fig. 8c. Every vertex of δ' lies in one of them. Since δ' is plane, vertex $k - 1$ must belong to $\Delta(a, k, b)$ and, by induction, so do all vertices $i \leq k$; similarly, all vertices $i \geq k + 1$ lie in $\Delta(a, b, k + 1)$. We can thus apply the argument of case 1 to each of the two subgraphs contained in $\Delta(a, b, k)$ and $\Delta(a, b, k + 1)$. This yields two

non-overlapping monotone sequences of length q each. Note, however, that both must be increasing or both decreasing, since one type forces a to the left and b to the right of ℓ and the other does the opposite. Now Observation 2 of [11] yields that at most $q + 1$ vertices can remain fixed. \square

Remark 2. The drawing δ in the proof of Theorem 7 can be slightly perturbed so that no three vertices are aligned.

5 Conclusion

Inspired by John Tantalo's on-line game *Planarity* we have introduced a new and apparently simple graph-drawing problem, which turned out to be rather difficult. There are many open questions. On the computational side, we showed inapproximability for MINMOVEDVERTICES. However, this does not imply anything for the approximability of MAXKEPTVERTICES, which remains open.

Are the problems in \mathcal{NP} ? Are they hard for cycles? What about parameterized complexity? On the combinatorial side, there are large gaps to be filled and other classes of planar graphs to be studied.

References

1. P. Erdős and G. Szekeres. A combinatorial problem in geometry. *Compos. Math.*, 2:463–470, 1935.
2. I. Fáry. On straight-line representation of planar graphs. *Acta Sci. Math. (Szeged)*, 11:229–233, 1948.
3. X. Goaoc, J. Kratochvíl, Y. Okamoto, C.-S. Shin, and A. Wolff. Moving vertices to make drawings plane, June 2007. Available at <http://arxiv.org/abs/0706.1002>.
4. M. Kang, M. Schacht, and O. Verbitsky. How much work does it take to straighten a plane graph out?, June 2007. Available at <http://arxiv.org/abs/0707.3373>.
5. M. Kaufmann and R. Wiese. Embedding vertices at points: Few bends suffice for planar graphs. *J. Graph Algorithms Appl.*, 6(1):115–129, 2002.
6. D. E. Knuth and A. Raghunathan. The problem of compatible representatives. *SIAM J. Discr. Math.*, 5(3):422–427, 1992.
7. D. Lichtenstein. Planar formulae and their uses. *SIAM J. Comput.*, 11(2):329–343, 1982.
8. K. Misue, P. Eades, W. Lai, and K. Sugiyama. Layout adjustment and the mental map. *J. Visual Languages and Computing*, 6(2):183–210, June 1995.
9. J. Pach and G. Tardos. Untangling a polygon. *Discrete Comput. Geom.*, 28(4):585–592, 2002.
10. C. Schensted. Longest increasing and decreasing subsequences. *Canadian Journal of Mathematics*, 13:179–191, 1961.
11. A. Spillner and A. Wolff. Untangling a planar graph, Sept. 2007. Available at <http://arxiv.org/abs/0709.0170>.
12. S. K. Stein. Convex maps. *Proc. Amer. Math. Soc.*, 2:464–466, 1951.
13. J. Tantalo. Planarity. Web site at <http://planarity.net/>, accessed May 21, 2007.
14. O. Verbitsky. On the obfuscation complexity of planar graphs, May & June 2007. Available at <http://arxiv.org/abs/0705.3748>.
15. K. Wagner. Bemerkungen zum Vierfarbenproblem. *Jahresbericht Deutsch. Math.-Verein.*, 46:26–32, 1936.