

On the Volume and Resolution of 3-Dimensional Convex Graph Drawing

(Extended Abstract)

MAREK CHROBAK*	MICHAEL T. GOODRICH†	ROBERTO TAMASSIA‡
Dept. of Computer Science Univ. of California Riverside, CA 92521 marek@cs.ucr.edu	Dept. of Computer Science Johns Hopkins Univ. Baltimore, MD 21218 goodrich@cs.jhu.edu	Dept. of Computer Science Brown Univ. Providence, RI 02912 rt@cs.brown.edu

Abstract

We address the problem of drawing a 3-connected planar graph as a convex polyhedron in \mathbf{R}^3 . We give an efficient algorithm for producing such a realization using $O(n)$ volume under the vertex-resolution rule. Each vertex in the drawing resulting from this method is guaranteed to need no more than $O(n \log n)$ bits to represent (as a pair of rational numbers). This solves an open problem of Cohen, Eades, Lin, and Ruskey. We also show that under the angular-resolution rule drawing a 3-connected planar graph as a convex polyhedron in \mathbf{R}^3 requires at least exponential volume in the worst case.

1 Introduction

The research area of *Graph Drawing* is concerned with methods for automatically displaying a graph G so as to accent fundamental properties G possesses while also optimizing important aesthetic qualities of the drawing, such as its size. It is a research area that combines computational geometry and graph theory to study interesting theoretical questions concerning algorithms for drawing graphs as well as trade-offs for various geometric optimization criteria, while also being an area with significant practical applications in computer graphics, software engineering, and databases.

Due to the inherent 2-dimensional nature of most display hardware, it should come as no surprise that the vast majority of previous graph drawing research has focused on 2-dimensional drawings (e.g., see the excellent annotated bibliography by Di Battista *et al.* [14]). But recent advances in 3-dimensional visualization hardware have made 3-d drawings technically feasible, and a handful of researchers (and film makers¹) have begun to explore the possibilities of displaying graphs using this new technology [8, 17, 18].

Even more so than in 2-dimensional drawings, however, there is a potential for 3-dimensional drawings to become cluttered. It is not yet clear which 3-dimensional drawing criteria should become the analogues of 2-dimensional properties such as planarity, symmetry, and monotonicity. In this paper we explore a property with easily-motivated aesthetic properties as well as classic underpinnings: convexity.

*This research supported by the NSF under Grant CCR-9503498.

†This research supported by the NSF under Grants IRI-9116843 and CCR-9300079.

‡This research supported by the NSF under Grants CCR-9007851 and CCR-9423847, by ARO under grants DAAL03-91-G-0035 and 34990-MA-MUR, and by ONR and DARPA under contract N00014-91-J-4052, ARPA order 8225.

¹An important plot element in the movie *Jurassic Park* involves a 3-dimensional virtual-reality traversal of a tree representing a Unix file system.

In particular, we are interested in the problem of drawing an n -node graph G as the 1-skeleton of a convex polytope in \mathbf{R}^3 . That is, we wish to construct a set S of n points in \mathbf{R}^3 so that $G = \text{conv}(S)$, where $\text{conv}(S)$ denotes the graph determined by the edges of the convex hull of S . It is well known (by Steinitz’s theorem [36]) that such a drawing exists if and only if G is planar and 3-connected (see also Grünbaum [25]). Thus, let us assume for the remainder of this paper that our given graph G is planar and 3-connected, for these conditions can be tested in linear time [3, 4, 27]. Interestingly, it is a trivial exercise to derive from the published proofs of Steinitz’s theorem an $O(n^3)$ -time method for drawing G as a 3-dimensional simplicial convex polytope in the real-RAM model [33]. Unfortunately, this approach seems to require at least exponential volume and an exponential number of bits to implement. Thus, the goal of the present research is to explore the conditions under which one can perform this embedding using polynomial time, volume, and number of bits.

1.1 Resolution requirements

Of course, specifying a 3-dimensional volume bound (or a 2-dimensional area bound) begs the question of how this is to be measured. We will restrict our attention in this paper to straight-line drawings (where each edge is a straight line segment joining the points associated with its endpoint vertices) that are drawn so as to achieve one of the following *resolution rules*:

- **vertex resolution:** each vertex must be at least unit distance from any other vertex;
- **edge resolution:** each vertex and edge must be at least unit distance from any non-incident (or non-adjacent) edge; and
- **angular resolution:** each vertex must be at least unit distance from any other vertex and the angle defined by any two consecutive edges incident upon the same vertex v must be at least $\alpha(v)$, for some predefined function $\alpha(v)$ (typically, $\alpha(v)$ is defined to be $\Theta(1/d(v))$ where $d(v)$ denotes the degree of v).

These notions are motivated by the respective aesthetic desires that each vertex be distinguished from every other vertex, that each vertex be distinguished from each non-incident edge, and that each edge incident upon the same vertex be distinguished from its neighbors. Note that requiring vertex resolution is strictly weaker than requiring that a drawing of a graph be in an $O(n) \times O(n)$ integer grid, but the edge-resolution and angular-resolution rules can be either more or less restrictive than the $O(n) \times O(n)$ integer grid requirement, depending upon the drawing.

We define the volume of a drawing in \mathbf{R}^3 to be the volume of a smallest axis-oriented bounding box containing the drawing so as to have length at least 1 in each dimension. The area of a drawing in \mathbf{R}^2 is defined similarly.

1.2 Our Results

We give an efficient algorithm for producing a realization of a 3-connected planar graph as a convex polyhedron in \mathbf{R}^3 using $O(n)$ volume under the vertex-resolution rule. Each vertex in the drawing resulting from this method is guaranteed to need no more than $O(n \log n)$ bits to represent (as a pair of rational numbers). This solves an open problem of Cohen, Eades, Lin, and Ruskey [8]. We also show that, under the angular-resolution rule, drawing a 3-connected planar graph as a convex polyhedron in \mathbf{R}^3 requires at least exponential volume in the worst case.

1.3 Related previous work

Before we present our results, however, let us first review some related previous work. There is not a lot of previous work on 3-dimensional graph drawing of an algorithmic nature, however. For example, Smith (see [26]) claims an $O(n^{4.38})$ -time method (acting on real numbers $O(n \log n)$ bits long) for producing an inscribed polytope that realizes a graph known to be inscribable, which can be tested in linear time, for example, for planar triangulations, due to a result of Dillencourt and Smith [16]. Cohen, Eades, Lin, and Ruskey [8] address 3-dimensional graph drawing directly in the integer grid model, showing, for example, that complete graphs can be drawn using straight line segments in an $O(n) \times O(n) \times O(n)$ integer grid without edges crossing. More recently, Eades and Garvan [17] address the problem of drawing 3-connected planar graphs as 3-dimensional convex polyhedra, adapting an approach of Hopcroft and Kahn [28] to produce such drawings in $O(n^{1.5})$ time in the real-RAM model. They do not analyze the bit complexity of their approach, but they do prove that their drawings have at least exponential volume under the vertex resolution rule.

There are also a number of relevant previous results in 2-dimensional graph drawing. Indeed, the straight line drawing of planar graphs is a classic topic in Mathematics, both in the plane [20, 35, 37, 38] and in 3-dimensions [25, 36]. Unfortunately, when translated into algorithms the proofs to these classic theorems produce drawings with poor resolution characteristics. Thus, recent attention has turned to area-efficient schemes for planar straight-line planar graph drawings, with the first breakthrough coming from de Fraysseix, Pach, and Pollack [12, 13], who show that any planar triangulation can be drawn as a straight line embedding in an $O(n) \times O(n)$ integer grid. Incidentally, their method also yields a drawing of area $O(n^2)$ under the edge-resolution rule. Moreover, Chrobak and Payne [7] show that the approach of de Fraysseix *et al.* can be implemented in $O(n)$ time. Using a different and quite elegant approach, Schnyder [34] gives an alternate linear-time scheme for producing an $O(n) \times O(n)$ integer grid drawing of a triangulated planar graph, whose edge resolution is $O(n^3)$. Since then, several researchers have worked on extending and tightening these results in the integer grid model [5, 6, 29].

Several researchers have also considered trade-offs involving the angular resolution (e.g., see [21, 22, 23]). In addition, Di Battista, Tamassia, and Tollis [15] prove an interesting lower bound, which holds under any “reasonable” finite-resolution rule, that producing a straight-line 2-dimensional drawing of a directed acyclic planar graph so that all edges “point up” requires exponential area. Our formulation of the above resolution rules for 3-dimensional graph drawing extends their resolution notions.

2 Convex Equilibrium Stress Graphs and 3-D Convex Drawings

Our method builds upon the classic work of Tutte [37, 38] on the relationship between *equilibrium stress graphs* and planar convex drawings and the recent work of Hopcroft and Kahn [28] relating equilibrium stress graphs and 3-dimensional convex polytopes (see also [17]). In this section we review some important relationships of equilibrium stress graphs [28, 37, 38] and 3-dimensional convex drawings that will form a template for our convex drawing algorithm.

2.1 Convex equilibrium stress graphs

Let G be a 3-connected planar graph embedded in \mathbf{R}^2 . Such an embedding is *convex* if every face of G is convex. Let $(1, 2, \dots, n)$ be a listing of the vertices of G and let $p_i = (x_i, y_i)$ denote the point in the plane corresponding to vertex i . A *stress* function defined on G is an assignment of weights

$w_{i,j}$ so that $w_{i,j} = w_{j,i}$, for all $i \neq j$, and $w_{i,j} = 0$ if (i, j) is not an edge in G . A stress function is *convex* if the weight of each interior edge of G is (strictly) positive while the weight of each exterior edge is (strictly) negative. A stress function is merely *internally convex* if the weight of each interior edge is positive. A stress function w is at *equilibrium* for G if, for all $i \in \{1, 2, \dots, n\}$,

$$\sum_{j=1}^n w_{i,j}(x_i - x_j) = 0 \quad (1)$$

and

$$\sum_{j=1}^n w_{i,j}(y_i - y_j) = 0. \quad (2)$$

A stress function is at *internal equilibrium* if Equations (1) and (2) are guaranteed to hold only for the internal vertices of G . A stress function w' is an *external extension* of a function w if w' agrees with w on each internal edge of G . Tutte establishes an interesting connection between these properties of stress functions and the convexity of the embedding for G :

Theorem 2.1 [38]: *Let G be a 3-connected planar graph embedded in \mathbf{R}^2 to have a convex external face. If there exists an internally-convex stress function at internal equilibrium for G , then G is a convex embedding.*

Tutte shows how to use this theorem to draw G . His approach is to embed the external face of G so as to be convex, define $w_{i,j} = 1$ for each internal edge of G , and then solve the linear system determined by the boundary points and Equations (1) and (2) to determine the locations of all the internal vertices. Unfortunately, for our purposes, this approach does not in general produce nice drawings, for Eades and Garvan [17] show that such drawings can require exponential area under the vertex-resolution rule. Thus, if we are to achieve polynomial area using this approach, we must use a more “adaptive” approach. As a step in this direction we note the following useful result of Hopcroft and Kahn:

Lemma 2.2 [28]: *Let G be an embedded planar graph with triangular external face, and let w be an internally-convex stress at internal equilibrium for G . Then there is an external extension w' of w that is convex and at equilibrium for G .*

By Equations (1) and (2), an external extension w' can be computed from w in linear time simply by solving a linear system defined by the three external vertices (for there are only three undetermined variables).

2.2 3-dimensional convex graph drawings

There is a well-known duality between convex stress graphs and 3-dimensional convex polyhedra, dating back to Maxwell [32] (see also [9, 11, 39]). In this subsection we review the explicit formulation of Hopcroft and Kahn [28] for this mapping.

Let G be a convex embedding of a 3-connected planar graph and let G have a convex equilibrium stress w . With each face r in G associate a linear function

$$f_r(x, y) = a_r x + b_r y + c_r.$$

Let us now view G as being embedded in the plane $z = 1$ and choose an arbitrary reference point $p_* = (x_*, y_*, 1)$ that is not collinear with any edge of G . The set of functions $\mathcal{F} =$

$\{f_r: r \text{ is a face in } G\}$ defines a w -consistent mapping if, for each edge in G between points (p_i, p_j) , incident upon faces r and s ,

$$w_{i,j} = \frac{\delta(r,s)(f_s(x_*, y_*) - f_r(x_*, y_*))}{[p_i, p_j, p_*]}, \quad (3)$$

where $\delta(r,s)$ is an orientation coefficient and $[p_i, p_j, p_*]$ is the usual triple product in \mathbf{R}^3 (if p_i, p_j , and p_* are viewed as column vectors, then $[p_i, p_j, p_*] = \det([p_i \ p_j \ p_*])$). The orientation coefficient, $\delta(r,s)$ is defined to be $+1$ if in a counterclockwise ordering of the vertices around r , v_i precedes v_j , and $\delta(r,s)$ is defined to be -1 otherwise. Hopcroft and Kahn show that w -consistency is independent of the choice of reference point p_* (provided, of course, that it is not collinear with any edge of G).

Of course, Equation (3) may not by itself specify a unique w -consistent mapping \mathcal{F} . We may fix such an \mathcal{F} , however, if necessary, by adding additional constraints implied by the topology, such as $f_r(x, y) = f_s(x, y)$ for any (x, y) on the line segment joining p_i and p_j . Given such a mapping \mathcal{F} , define a convex polyhedron by associating the plane $z = 1$ with the external face and the plane defined by f_r with each internal face r in G . Hopcroft and Kahn [28] show the following:

Theorem 2.3 [28]: *If w is a convex equilibrium stress for a convex embedding G , then the polyhedron defined by a w -consistent mapping is strictly convex.*

Thus, we have a template for producing 3-dimensional strictly-convex drawings of 3-connected planar graphs:

1. Construct an embedding of G with a convex equilibrium stress w ,
2. Find a mapping \mathcal{F} that is w -consistent to define a 3-dimensional convex polyhedron P that has G as its 1-skeleton.

This template forms a very high-level description of our approach, as well as that of Eades and Garvan [17]. Our algorithm differs from theirs significantly in Step 1, however.

Note that, under any of our resolution rules, if G has area A , then we can draw P to have volume A (by scaling the range of z -values to the interval $[0, 1]$). Let us therefore now concentrate on a method for drawing a 3-connected planar graph as a small-area planar convex equilibrium stress graph.

3 Algorithms for Small-Area Convex Equilibrium Stress Graphs

Hopcroft and Kahn [28] show that there are convex planar embeddings that do not admit an equilibrium stress. Nevertheless, they show that embedded graphs that contain x -monotone spanning trees can be weighted to give a stress that satisfies Equation (1), for each $i \in \{1, 2, \dots, n\}$, a condition we call x -equilibrium. Still, their method would not, in general, yield a convex stress. In this section we show that any 3-connected planar graph can be drawn as a small-area convex stress graph under the vertex resolution rule.

3.1 Constructing a convex stress with x -equilibrium

Let G be a 3-connected planar graph. Suppose further that G has a triangular external face (v_1, v_2, v_n) . We make this assumption without loss of generality, for if G does not have a triangular

face, we can use the graph-theoretic planar dual of G . Suppose further that we are given a convex embedding of G in an $O(n) \times O(n)$ integer grid so that there are no vertical edges. This can be achieved by a simple modification of the 2-dimensional convex drawing algorithm of Chrobak and Kant [5], which we explore in the full version of this paper. Vertices v_1, v_2, v_n are mapped into the triangle with coordinates $(0, 0)$, $(4n, 0)$ and $(2n, 2n)$. Define the x -cost, $c_{i,j}$, of an edge (v_i, v_j) to be $|w_{i,j}(x_i - x_j)|$.

Lemma 3.1: *If G is an n -node 3-connected planar graph, with a triangular external face, convexly embedded as above, then one can define a convex x -equilibrium stress on G so that each x -cost, $c_{i,j}$, is a positive integer with magnitude that is $O(n)$, for $i, j \in \{1, 2, \dots, n\}$.*

Proof: Let us orient each edge in G from left to right (which is a well-defined notion, since G contains no vertical edges). Furthermore, let us view the x -cost on each edge as a flow from left to right (with the x -equilibrium equation serving the role of flow conservation at each node). We do not set any capacity constraints on edges, however. We define an initial valid flow by defining the x -cost of each edge in G to be 0. Next, for each edge internal e in G in turn, increase by 1 the flow along some left-to-right path from v_1 to v_n that contains e (maintaining x -equilibrium). Such a path exists for each edge e because G is a 3-connected planar graph with a convex embedding that contains no vertical edges. Since we maintain internal x -equilibrium with each “augmentation,” this procedure will result in an internally-convex stress function that is at internal x -equilibrium. This can be extended to a convex stress at x -equilibrium, then, by Lemma 2.2. Moreover, the flow on any internal edge is increased by 1 at most $O(n)$ times; hence, the x -cost on any internal edge is a positive integer with magnitude that is $O(n)$. By the proof of Lemma 2.2, this implies that all x -costs in G are integers bounded by $O(n)$. ■

The proof of this lemma immediately implies an $O(n^2)$ algorithm for computing such a stress, but we can actually implement this approach in linear time. Intuitively, the approach is to perform flow augmentations for many edges in G simultaneously by a “topological plane sweep” procedure [1, 2, 19].

Lemma 3.2: *If G is an n -node 3-connected planar graph, with a triangular external face, convexly embedded as above, then one can define in $O(n)$ time a convex x -equilibrium stress on G so that each x -cost, $c_{i,j}$, is a positive integer with magnitude that is $O(n)$, for $i, j \in \{1, 2, \dots, n\}$.*

Proof: The proof will be based upon an efficient implementation of the strategy of the previous proof using a monotone spanning tree technique [28]. Define a monotone spanning tree T on G by picking an outgoing (left-to-right oriented) edge for each node in G (other than the rightmost node). Remove the three external edges from G , leaving a spanning forest, T' , of trees rooted at nodes on the external face. Let us initially assign each internal edge (v_i, v_j) in G weight $w_{i,j} = 1/|x_i - x_j|$. This is clearly an internally-convex weight function (and it is well-defined, since G contains no vertical edges). It is not at x -equilibrium, however. Let us therefore define the *residual* weight at each node v_i as

$$r_i = \sum_{j=1}^n w_{i,j}(x_i - x_j).$$

Note that initially each r_i is an integer in the range $(-n, +n)$, for the x -cost of each edge is +1 in this initial weighting. Indeed, for an internal node v_i , r_i corresponds to a signed difference between the number of in-coming and out-going edges incident upon v_i . For each non-root node v_i

in T' , we recursively re-weight the edges coming out of v_i to its children. If we now have $r_i \leq 0$, then we need not re-weight the edge to v_i from its parent. If, on the other hand, $r_i > 0$, then we increase the weight of the edge from v_i to its parent so as to achieve residual stress $r_i = 0$ at v_i . Since we only increase the positively-weighted internal edges, this weight function remains internally-convex. Moreover, by a simple inductive argument, the cost of an edge of T going from v to its parent is at most the sum of the absolute values of the residuals of all the descendants of v . Thus, $c_{i,j} \leq 2|E| \leq 6n$, for each edge (v_i, v_j) in G . We have not yet necessarily achieved a stress that is at internal x -equilibrium, however. Therefore, we reflect the embedding of the graph G by the mapping $(x, y) \rightarrow (-x, y)$ and we repeat the above procedure using the current weighting as the initial weight. Since this transformation forces $r_i \geq 0$, for each internal node v_i , repeating the above procedure on G will yield an internally-convex stress that is at internal x -equilibrium. The lemma follows, then, by an application of Lemma 2.2. ■

Thus, we can take the above convex embedding of a 3-connected planar graph G , which has a triangular outer face, and in linear time produce a convex x -equilibrium stress for G . The stress function we define on G will in general not be at y -equilibrium, however.

3.2 Producing a convex equilibrium stress graph

Nevertheless, we can easily convert such a drawing into a convex equilibrium stress graph. In particular, we let $Ax = b$ denote the linear system defined by the weight function, which achieves x -equilibrium, and Equations (1) and (2) and the boundary conditions fixing the exterior triangle for G . Since all the equations in this system involving x -coordinates are already satisfied, solving the system $Ax = b$ finds the y -coordinates of the vertices of G that produce a convex equilibrium stress graph embedding G' for G , while keeping the x -coordinates unchanged.

This algorithm clearly produces a convex embedding of G in the plane together with a convex equilibrium stress defined on this embedding, by Theorem 2.1. Moreover, if we start with G being embedded in an $O(n) \times O(n)$ integer grid, then G' will be a convex embedding such that each x -coordinate is a positive integer with magnitude at most $O(n)$, and G' will have no vertical edges. In addition, by well-known properties of rational-arithmetic linear system solving, we can guarantee that the number of bits needed to represent any y -coordinate, as a rational number, is $O(n \log n)$. If we scale the y -coordinates to lie in the interval $[0, 1]$, then the drawing will still be a convex equilibrium stress embedding, but will have area $O(n)$ under the vertex-resolution rule. Thus, we have the following:

Theorem 3.3: *Given a 3-connected planar graph G , which has a triangular external face, one can produce a convex equilibrium stress embedding of G that achieves $O(n)$ area under the vertex-resolution rule. The running time needed to achieve this is $O(n + P(n))$, where $P(n)$ is the time needed to solve an $n \times n$ linear system defined by planar constraints.*

Note that this area bound contrasts sharply with the exponential lower bound of Eades and Garvan [17] for the area of Tutte drawings under the vertex-resolution rule.

Incidentally, there are fairly simple separator-based methods [24, 30, 31] for achieving an $O(n^{1.5})$ bound for $P(n)$, while much more sophisticated methods allow one to achieve an $O(M(n^{1/2}))$ bound, where $M(n)$ is the time needed to multiply two $n \times n$ matrices (the current best bound for $M(n)$ is $O(n^{2.375})$ [10]).

Thus, by our template, we have the following:

Theorem 3.4: *Given a 3-connected planar graph G , one can draw G as a convex polyhedron in \mathbf{R}^3 using $O(n)$ volume under the vertex-resolution rule. The running time needed to achieve this is $O(M(n^{1/2}))$.*

Proof: As mentioned above, if G does not have a triangular external face, we embed the graph-theoretic planar dual to G (which must have a triangular face in this case) as a convex polyhedron P in \mathbf{R}^3 . Computing the Poincaré dual to P then gives a realization of G . ■

Thus, under current theoretical definition of $M(n)$ [10], we can achieve a running time of $O(n^{1.19})$, but in practice the $O(n^{1.5})$ bound is probably more realistic.

4 A Lower Bound for Angular Resolution in \mathbf{R}^3

In this section we show that under the angular resolution rule there are 3-connected planar graphs that require exponential volume to draw as 3-dimensional convex polyhedra. Specifically, we will demonstrate a family of fixed-degree 3-connected planar graphs requiring exponential volume to draw as 3-dimensional convex polyhedra with constant angular resolution. We establish this lower bound via a reduction from the problem of drawing a fixed-degree 3-connected planar graph under angular resolution in \mathbf{R}^2 , which was shown to require exponential area by Garg and Tamassia [23].

The main difficulty in extending their proof to convex drawings in \mathbf{R}^3 is that the third dimension allows a tremendous amount of extra drawing freedom. For example, a convex drawing in \mathbf{R}^3 can achieve angular resolution and yet have many 2-dimensional projections that do not achieve angular resolution. The main idea of our lower bound construction is to demonstrate an n -node 3-connected planar graph G_n such that if a convex drawing of G_n achieves angular resolution in \mathbf{R}^3 , then this drawing contains a connected subgraph of size $\Theta(n)$ that projects to a 2-dimensional drawing that also achieves angular resolution. By the lower bound of Garg and Tamassia [23], this would establish an exponential lower bound on the area of this projection, hence the volume of this drawing would also be at least exponential.

We define G_n algorithmically. We begin with a 17-node cycle P_{17} , which will form a face in G_n , hence P_{17} must be drawn in some plane in \mathbf{R}^3 . So, let P'_{17} be a planar drawing of P_{17} as a convex polygon. Orient each edge of P'_{17} in the clockwise direction. For a vertex v on P'_{17} , let $p(v)$ and $s(v)$ respectively denote the predecessor edge and successor edge incident upon v in this orientation. Define the *external angle* $\beta(v)$ at v to be the angle formed at v between an extension of $p(v)$ (as a ray with $p(v)$'s orientation) and an extension of $s(v)$. Also, following Grünbaum [25], let us measure angles as fractions of 1 (so that a right angle is 1/4).

Fact 4.1: *Let P_n be a convex polygon. Then*

$$\sum_{v \in P_n} \beta(v) = 1.$$

This immediately implies the following:

Corollary 4.2: *P'_{17} has two consecutive vertices with external angles less than 1/8.*

Proof: By the previously-mentioned fact, P_{17} can have at most 8 vertices with external angle at least 1/8. Thus, P'_{17} must have at least 9 vertices with external angle less than 1/8. Moreover, by a simple pigeon-hole argument, two of these vertices must be consecutive. ■

Let us continue, then, with our definition of G_n . Our next augmentation is to add a vertex v^* that is adjacent to each vertex on P_{17} (so as to define a pyramid). Let Q denote this new graph. For each edge e of Q incident upon v^* define the *external angle*, $\beta(e)$, at e analogously to the planar external angle at a vertex. Specifically, define $\beta(e)$ to be the fraction of the sphere defined between the two planes incident upon e and oriented in a clockwise direction. Define an edge e to be *shallow* if $\beta(e) \leq 1/8$. By Corollary 4.2, we know that, no matter where v^* is placed, two consecutive edges incident upon v^* must be shallow.

We wish to force there to be a triangle τ in Q with all three of its edges being shallow. This is because any subgraph placed in the interior of τ and drawn to achieve angular resolution would project to the plane containing τ so as to achieve (2-dimensional) angular resolution. This would then allow us to complete the proof by placing the graph, G_k , used in the 2-dimensional lower bound of Garg and Tamassia [23], in the interior of τ . Let us therefore augment Q with additional triangular faces in a fashion that will allow us to argue that there must be at least one triangular face with three shallow edges. If we can accomplish this by adding just a constant number of additional edges to Q , then we can place G_k in the interior of each such face to complete the proof.

Let t be the triangular face of Q with two shallow edges. If t actually has three shallow edges, then we are done, so let us assume that the third edge of t is not shallow. Of course, it must nevertheless have measure less than $1/2$. Define the *stellation* of a triangular face s to be the placement of a new vertex in the interior of s which is then made to be adjacent to the three vertices of s . We start with t and stellate it. This creates two triangular faces t_1 and t_2 that are incident upon v^* and a triangular face that is not incident upon v^* . Moreover, it creates a new edge incident to v^* and this edge must also be shallow. Let us therefore repeat this procedure, likewise stellating t_1 and t_2 . This creates four new triangular faces incident upon v^* and two new shallow edges incident upon v^* as well. Let us continue to iterate this procedure, stellating all the triangular faces incident upon v^* in each iteration. We repeat this procedure for a total of 9 iterations. It is useful to note that the planar dual of the subgraph G_t this procedure creates is a depth-9 complete binary tree B . Moreover, since the third edge of t must have measure less than $1/2$, there can be no more than 4 non-shallow edges dual to edges on any leaf-to-root path in B . But this implies, by a straightforward pigeonhole argument, that there is at least one triangle in G_t with three shallow edges.

To sum up, then, our construction of G_n starts with P_{17} , adds v^* to be adjacent to each vertex of P_{17} , augments each triangle incident to v^* to become the subgraph G_t , and then adds the lower-bound graph G_k of Garg and Tamassia [23] in the interior of each triangle in a G_t to complete the proof. If the resulting graph, G_n , is drawn as a convex polyhedron in \mathbf{R}^3 so as to achieve angular resolution, then, by the above argument, at least one of these G_k 's will project to a plane so as to preserve angular resolution. But by the lower bound of Garg and Tamassia, such a projection must have area at least $\Omega(2^n)$; hence, the drawing in \mathbf{R}^3 must have volume at least $\Omega(2^n)$. Thus, we have proved the following:

Theorem 4.3: *There is a fixed-degree n -node 3-connected planar graph G_n that requires $\Omega(2^n)$ volume to draw as a convex polyhedron in \mathbf{R}^3 under the angular resolution rule, with $\alpha(v) > \alpha_0$ for any fixed constant $\alpha_0 > 0$.*

5 Conclusion

We have performed an investigation into the use of convexity in three-dimensional graph drawing, showing, in particular, that any 3-connected planar graph can be drawn as a convex polyhedron

in polynomial time, volume, and number of bits under the vertex resolution rule. We have also shown that drawing such graphs as convex polyhedra under the angular resolution rule can require exponential volume. We leave as an open problem whether one can achieve polynomial time, volume, and number of bits under the edge resolution rule.

Acknowledgements

We would like to thank Robert Cohen, Peter Eades, and Patrick Garvan for helpful discussions and/or e-mail exchanges regarding topics related to this paper.

References

- [1] E. G. Anagnostou, L. J. Guibas, and V. G. Polimenis. Topological sweeping in three dimensions. In *Proc. 1st Annu. SIGAL Internat. Sympos. Algorithms*, volume 450 of *Lecture Notes in Computer Science*, pages 310–317. Springer-Verlag, 1990.
- [2] T. Asano and T. Tokuyama. Topological walk revisited. In *Proc. 6th Canad. Conf. Comput. Geom.*, pages 1–6, 1994.
- [3] K. Booth and G. Lueker. Testing for the consecutive ones property interval graphs and graph planarity using PQ-tree algorithms. *J. Comput. Syst. Sci.*, 13:335–379, 1976.
- [4] N. Chiba, T. Nishizeki, S. Abe, and T. Ozawa. A linear algorithm for embedding planar graphs using PQ-trees. *J. Comput. Syst. Sci.*, 30(1):54–76, 1985.
- [5] M. Chrobak and G. Kant. Convex grid drawings of 3-connected planar graphs. Technical Report RUU-93-45, Dept. of Computer Sci., Utrecht Univ., 1993.
- [6] M. Chrobak and S. Nakano. Minimum-width grid drawings of plane graphs. Technical Report UCR-CS-94-5, Dept. of Computer Sci., Univ. California, Riverside, 1994.
- [7] M. Chrobak and T. H. Payne. A linear time algorithm for drawing a planar graph on a grid. Technical Report UCR-CS-90-2, Dept. of Math. and Comput. Sci., Univ. California Riverside, 1990.
- [8] R. F. Cohen, P. Eades, T. Lin, and F. Ruskey. Three-dimensional graph drawing. In *Graph Drawing '94*, 1994.
- [9] R. Connelly. Rigidity and energy. *Invent. Math.*, 66:11–33, 1982.
- [10] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. Symbolic Computing*, 9:251–280, 1990.
- [11] H. Crapo and W. Whitely. Statics of frameworks and motions of panel structures, a projective geometric introduction. *Structural Topology*, 6:42–82, 1982.
- [12] H. de Fraysseix, J. Pach, and R. Pollack. Small sets supporting Fary embeddings of planar graphs. In *Proc. 20th Annu. ACM Sympos. Theory Comput.*, pages 426–433, 1988.
- [13] H. de Fraysseix, J. Pach, and R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10:41–51, 1990.
- [14] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. Algorithms for drawing graphs: an annotated bibliography. Preprint, Dept. Comput. Sci., Brown Univ., Providence, RI, November 1993. Preliminary version available via anonymous ftp from `wilma.cs.brown.edu`, `gdbiblio.tex.Z` and `gdbiblio.ps.Z` in `/pub/papers/compgeo`.
- [15] G. Di Battista, R. Tamassia, and I. G. Tollis. Area requirement and symmetry display of planar upward drawings. *Discrete Comput. Geom.*, 7:381–401, 1992.

- [16] M. B. Dillencourt and W. D. Smith. A linear-time algorithm for testing the inscribability of trivalent polyhedra. In *Proc. 8th Annu. ACM Sympos. Comput. Geom.*, pages 177–185, 1992.
- [17] P. Eades and P. Garvan. Drawing stressed planar graphs in three dimensions. In *Graph Drawing '95*, 1995.
- [18] P. Eades, C. Stirk, and S. Whitesides. The techniques of komolgorov and bardzin for three dimensional orthogonal graph drawings. In *Graph Drawing '95*, 1995.
- [19] H. Edelsbrunner and L. J. Guibas. Topologically sweeping an arrangement. *J. Comput. Syst. Sci.*, 38:165–194, 1989. Corrigendum in 42 (1991), 249–251.
- [20] I. Fary. On straight lines representation of planar graphs. *Acta Sci. Math. Szeged.*, 11:229–233, 1948.
- [21] M. Formann, T. Hagerup, J. Haralambides, M. Kaufmann, F. T. Leighton, A. Simvonis, E. Welzl, and G. Woeginger. Drawing graphs in the plane with high resolution. *SIAM J. Comput.*, 22:1035–1052, 1993.
- [22] A. Garg and R. Tamassia. Angular resolution of planar drawings. Technical report, Brown Univ., Dept. of Computer Science, 1993.
- [23] A. Garg and R. Tamassia. Planar drawings and angular resolution: Algorithms and bounds. In *Graph Drawing '95*, 1995.
- [24] J. R. Gilbert and R. E. Tarjan. The analysis of a nested dissection algorithm. *Numer. Math.*, 50:377–404, 1987.
- [25] B. Grünbaum. *Convex Polytopes*. Wiley, New York, NY, 1967.
- [26] C. D. Hodgson, I. Rivin, and W. D. Smith. A characterization of convex hyperbolic polyhedra and of convex polyhedra inscribed in the sphere. *Bull. (New Series) of the AMS*, 27(2):246–251, 1992.
- [27] J. Hopcroft and R. E. Tarjan. Efficient planarity testing. *J. ACM*, 21(4):549–568, 1974.
- [28] J. E. Hopcroft and P. J. Kahn. A paradigm for robust geometric algorithms. *Algorithmica*, 7:339–380, 1992.
- [29] G. Kant. Drawing planar graphs using the *lmc*-ordering. In *Proc. 33th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 101–110, 1992.
- [30] R. J. Lipton, D. J. Rose, and R. E. Tarjan. Generalized nested dissection. *SIAM J. Numer. Anal.*, 16(2):346–358, 1979.
- [31] R. J. Lipton and R. E. Tarjan. Applications of a planar separator theorem. *SIAM J. Comput.*, 9:615–627, 1980.
- [32] J. C. Maxwell. On reciprocal figures and diagrams of forces. *Phil. Mag. Ser.*, 27:250–261, 1864.
- [33] F. P. Preparata and M. I. Shamos. *Computational Geometry: an Introduction*. Springer-Verlag, New York, NY, 1985.
- [34] W. Schnyder. Embedding planar graphs on the grid. In *Proc. 1st ACM-SIAM Sympos. Discrete Algorithms*, pages 138–148, 1990.
- [35] S. K. Stein. Convex maps. *Proc. Amer. Math. Soc.*, 2:464–466, 1951.
- [36] E. Steinitz and H. Rademacher. *Vorlesungen über die Theorie der Polyeder*. Julius Springer, Berlin, Germany, 1934.
- [37] W. T. Tutte. Convex representations of graphs. *Proceedings London Mathematical Society*, 10:304–320, 1960.
- [38] W. T. Tutte. How to draw a graph. *Proceedings London Mathematical Society*, 3(13):743–768, 1963.
- [39] W. Whitney. Motions and stresses of projected polyhedra. *Structural Topology*, 7:13–38, 1982.