Representing a Graph Using Few Obstacles

Matthew P. Johnson
City University of New York

Deniz Sarıöz[2*]
Google, Inc.

1. INTRODUCTION

Let G be a plane graph with straight edges and vertices in general position; that is, a straight-line drawing of a planar graph with no edge crossings and no three vertices on a line in which the vertices are identified with their positions. We refer to the open line segment between a pair of non-adjacent graph vertices as a *non-edge* of D. An *obstacle representation* of G is a pair $(V(G), \mathcal{O})$ where $\mathcal{O}$ is a set of polygons (not necessarily convex) called *obstacles*, such that:

1. G does not meet any obstacle, and
2. every non-edge of G meets at least one obstacle.

Equivalently, G is the visibility graph on $V(G)$ determined by the obstacles in $\mathcal{O}$. The size of an obstacle representation is the cardinality of $\mathcal{O}$. Denote by *ORPG* the problem of computing a minimum-size obstacle representation of G (the optimum of which is called the *obstacle number* of G). Alpert, Koch, and Laison introduced the notions *obstacle representation* and *obstacle number* for abstract graphs [1] and noted that in any minim*al* obstacle representation, each obstacle can be identified with the face it lies in. Hence, we will use the terms *face* and *obstacle* interchangeably. If the faces have weights then we can seek a minimum-weight obstacle representation.

Finding a minimum-size obstacle representation of a straight-line graph drawing was treated as a computational problem in the more general setting in which D and G need not be planar [4]. This problem was reduced to hypergraph transversal (hitting set), with $O(n^4)$ faces available to pierce $O(n^2)$ non-edges ($O(n)$ faces and $\Theta(n^2)$ non-edges in the ORPG special case). A randomized $O(\log OPT)$-approximation algorithm based on bounding the Vapnik-Chervonenkis dimension of the corresponding hypergraph family was given in [4]. Left open was the question of whether better approximations or perhaps optimal algorithms were feasible.

In this short paper we give partial answers to that question. We show that computing the obstacle number is NP-hard already in the special case of plane graphs (i.e., ORPG); nonetheless, we show that that problem admits a polynomial-time approximation scheme (PTAS) and is fixed-parameter tractable (FPT). We show hardness by a reduction from planar vertex cover; the positive results are consequences of a solution value-preserving reduction to maximum degree 3 planar vertex cover (details omitted).

2. REDUCTION FROM PLANAR VC

THEOREM 1. *ORPG is NP-hard.*

*Research supported by grants from NSA (47149-0001) and PSC-CUNY (63427-0041). Research performed while the author was at the City University of New York Graduate Center.

PROOF. We reduce from planar vertex cover. Recall that in the decision version of planar vertex cover, we are given an abstract planar graph G having (without loss of generality) no isolated vertex, and a number k. Let $n = |V(G)|$, $m = |E(G)|$, and $f = n - m + 2$ (the number of faces in any crossing-free planar drawing of G). We will transform G in polynomial time into a plane graph G' in such a way that G has a vertex cover of size k if and only if G' has an obstacle representation of size k' (for k' defined below).

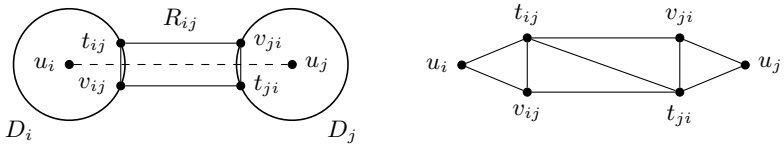
First, we construct from the planar vertex cover instance G a planar vertex cover problem instance G^3 with maximum degree 3, adapting and extending the construction of [3]. The graph G^3 admits a vertex cover of size k' if and only if G admits a vertex cover of size k. Second, we construct an ORPG instance G' in such a way that an obstacle representation of G' will correspond to a vertex cover of G^3 of the same size, and vice versa.

Constructing the maximum degree 3 planar vertex cover instance G^3. The planar graph G^3 is constructed as follows. We transform each vertex v_i of G into a cycle C^i of length $2b_i$, with $b_i \in \deg(v_i) + \{0, 1, 2\}$ (with the exact value decided below). We color the vertices of C^i alternating between blue and red. We then create a single leaf vertex z_i adjacent to some arbitrary red vertex of C^i. We transform each edge (v_i, v_j) of G into a path P_{ij} with *three* edges whose endpoints are *distinct* blue vertices of C^i and C^j. We finally create f copies of the 3-vertex path graph P_3, each constituting a component of G^3.

We claim that G has a vertex cover of size at most k if and only if G^3 has one of size at most $k' = k + f + m + \sum_i b_i$.

Constructing the ORPG instance $\tilde{G}$. In the remainder of the proof, we show how to "implement" the graph G^3 as an equivalent ORPG problem instance. The basic building blocks of the construction are *empty triangles* and *diamonds*. An *empty triangle* is a face of a plane graph that is surrounded by three edges and has no vertex inside. A *diamond* consists of two empty triangles sharing an edge and having their *four* vertices in convex position. Observe that a diamond contains a non-edge between two of its vertices. Hence at least one empty triangle of every diamond must be chosen in an obstacle representation. The f copies of P_3 in G^3 will match the faces of $\tilde{G}$ besides empty triangles, all of which must be chosen. The remaining vertices of G^3 will match the empty triangles of $\tilde{G}$, such that the edges among them match the diamonds of $\tilde{G}$. Hence there is a natural bijection between vertex covers of G^3 and obstacle representations of $\tilde{G}$.

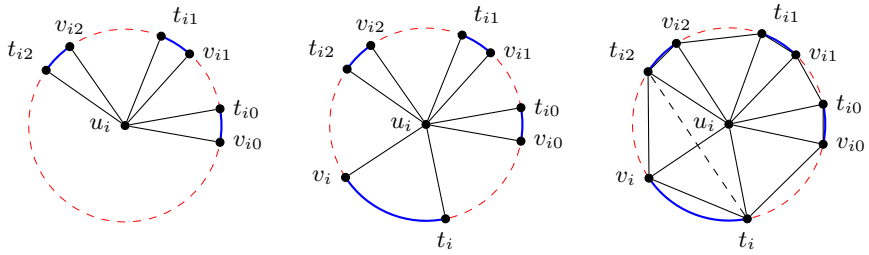
To begin the construction, we use the linear-time algorithm of de Fraysseix, Pach, and Pollack [2] to obtain a planar imbedding of G on a $O(n) \times O(n)$ portion of the integer lattice and then perturb the coordinates to obtain general position. (We do not distinguish



(a) Bold drawing of edge u_iu_j.

(b) Gadget in $\tilde{G}$ for edge u_iu_j of G.

Figure 1: Bold drawing and edge gadget for an edge of G.



(a) Initial circle with blue (solid) arcs of measure α and red (dashed) arcs has a large red arc of measure in $[180^\circ - \alpha, 270^\circ)$.

(b) After subdividing the large red arc into *three*, coloring its middle part blue, and adding dummy v_i and t_i vertices.

(c) In the resulting wheel graph, each pair of triangles sharing an edge induce a diamond. One diamond's non-edge is shown dashed.

Figure 2: Constructing the wheel graph drawing in the case of a large red arc.

between G and this imbedding.) We first visualize $\tilde{G}$ as a bold drawing [5] of G, whose vertices are represented by small disks and edges by solid rectangles: we draw each vertex u_i of G as a disk D_i about u_i (with boundary $\tilde{C}^i$), and every edge u_iu_j as a solid rectangle R_{ij}. See Fig. 1(a). Each R_{ij} has two vertices t_{ij},v_{ij} on $\tilde{C}^i$ and two vertices t_{ji}, v_{ji} on $\tilde{C}^j$ such that the line u_iu_j is a midline of R_{ij}, and $t_{ij}u_iv_{ij}t_{ji}u_jv_{ji}$ is a counterclockwise ordering of the vertices of a convex hexagon.

We draw the disks small enough to ensure that they are well-separated from one another. We set the radius r of every disk to the smaller of 1/4 and half of the minimum distance between a vertex u_i and an edge u_ju_k ($j \neq i \neq k$) of G. To fix a single width for all rectangles (i.e., $||t_{ij} - v_{ij}||$), we set a global angle measure α to the smaller of 45° and half of the smallest angle between two edges of $E(G)$ incident on the same vertex of $V(G)$.

$\tilde{G}$ is modeled on the bold drawing, by implementing each edge of G (path P_{ij} of G^3) with an edge gadget and each vertex of G (cycle C^i of G^3) with a vertex gadget. The edge gadget, consisting of four triangles forming three diamonds, is shown in Fig. 1(b). (Note that each pair $v_{ij}v_{ji}$ defines a non-edge.)

The vertex gadget is a modified wheel graph whose triangles correspond to the vertices of cycles C^i in G^3 (see Fig.). On every circle $\tilde{C}^i$, for every edge u_iu_j in G, we color blue the arc of measure α centered about the intersection of circle $\tilde{C}^i$ with u_iu_j (a non-edge in $\tilde{G}$). We place t_{ij} and v_{ij} at the endpoints of this arc so that $t_{ij}u_iv_{ij}$ is a counterclockwise triple. By the choice of α, all blue arcs are well-separated, and hence the rectangles are well-separated from one another and from other disks, by the choice of r. We color the remaining arcs red to obtain a red-blue striped pattern on each circle $\tilde{C}^i$, corresponding in color to the vertices of the corresponding C^i in G^3.

On every circle $\tilde{C}^i$, we will add the remaining edges between consecutive vertices of $\tilde{C}^i$ to complete the union of the triangles $t_{ij}u_iv_{ij}$, forming a wheel graph on hub u_i, such that every pair of triangles sharing a spoke form a diamond. If a red arc has measure at least $180^\circ - \alpha$, however, we must add additional spokes. By the general position assumption, at most one red arc per wheel can have such great measure. If such a red arc has measure less than 270°, we divide it evenly into *three* parts and color the middle part blue (see Fig.); otherwise, we divide it evenly into *five* parts and color the second and fourth parts blue (figure omitted), maintaining the striped pattern in both cases. We place dummy t_i and v_i verticesat the newly created (*zero*, *two* or *four*) arc endpoints. Finally, we add the requisite edges to complete the wheel graph.

We place a vertex $\tilde{z}_i$ on an arbitrary red arc of $\tilde{C}^i$ and connect it in $\tilde{G}$ to the end vertices (say t_{ij} and v_{ik}) of that arc. Thus an empty triangle $t_{ij}z_iv_{ik}$ is formed in $\tilde{G}$ as part of a diamond with u_i, corresponding to z_i and its incident edge in G^3.

In the unbounded face of $\tilde{G}$ we place two isolated vertices inducing a non-edge inside the unbounded face, thus requiring this face to be chosen in any solution. Every non-triangular face of $\tilde{G}$ must be selected as an obstacle, since every simple polygon with at least 4 vertices has an internal diagonal (i.e., a non-edge). The selection of these faces are forced moves and correspond to the selection, in a vertex cover for G^3, of the central vertex of each P_3.

Since each pair of neighboring triangles in $\tilde{G}$ indeed form a diamond and every non-triangular face is indeed a forced move, the result follows. □

3. REFERENCES

[1] H. Alpert, C. Koch, and J. D. Laison. Obstacle numbers of graphs. *Discrete & Computational Geometry*, 44(1):223–244, 2010.

[2] H. de Fraysseix, J. Pach, and R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10(1):41–51, 1990.

[3] M. R. Garey and D. S. Johnson. The rectilinear Steiner tree problem is NP-complete. *SIAM Journal of Applied Mathematics*, 32:826–834, 1977.

[4] D. Sariöz. Approximating the obstacle number for a graph drawing efficiently. In *CCCG*, 2011.

[5] M. J. van Kreveld. Bold graph drawings. *Comput. Geom.*, 44(9):499–506, 2011.