Discrete Applied Mathematics 164 (2014) 92–99

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Covering points with orthogonal polygons[☆]

Cem Evrendilek[a], Burkay Genç[b,*], Brahim Hnich[a]

[a] *Izmir University of Economics, Computer Engineering Department, Turkey*
[b] *TED University, Computer Engineering Department, Turkey*

ARTICLE INFO

Article history:
Received 27 September 2010
Received in revised form 15 December 2011
Accepted 18 January 2012
Available online 11 February 2012

Keywords:
Covering
Orthogonal polygon
NP-completeness

ABSTRACT

We address the problem of covering points with orthogonal polygons. Specifically, given a set of n points in the plane, we investigate the existence of an orthogonal polygon such that there is a one-to-one correspondence between the points and the edges of the polygon. In an earlier paper, we have shown that constructing such a covering with an orthogonally *convex* polygon, if any, can be done in $O(n \log n)$ time. In case an orthogonally convex polygon cannot cover the point set, we show in this paper that the problem of deciding whether such a point set can be covered with any orthogonal polygon is NP-complete. The problem remains NP-complete even if the orientations of the edges covering each point are specified in advance as part of the input.

1. Introduction

Constructing orthogonal shapes and objects from limited geometric and topological information is a frequently visited problem in computational geometry. For example Jackson and Wismath [6] use "stab visibilities" for reconstructing an orthogonal polygon. Usually, coordinates of the vertices (corners) of an object are used for reconstructing it. O'Rourke [10] provides an $O(n \log n)$ time algorithm for constructing orthogonal polygons from a given vertex set. Löffler and Mumford [8] show that there is only one orientation for which a provided vertex set can be used to reconstruct a rectilinear graph. Rappaport [12] shows that if straight angles are allowed, the problem becomes NP-hard. Other similar studies can be found in [1–3,9]. In existing works, points are either assumed to be anywhere within the polygon boundary or exactly required to be the vertices of the polygon. In this study, we require them to be exactly on polygon edges. We say that an edge covers a point, if the point can be written as the convex combination of the two endpoints of the edge. A polygon covers a point set if each polygon edge covers exactly one point and each point is covered exactly by one polygon edge.

An *orthogonal polygon* is a polygon whose edges are orthogonal. An orthogonal polygon is *orthogonally convex* if its intersection with any orthogonal line segment is either empty or a single line segment. A well-known problem in computational geometry is computing the orthogonal convex hull of a given point set [11] which treats the points as vertices (endpoints of the edges). An important issue with orthogonally convex hulls is that they may be disconnected [11]. Moreover, they do not even have to be orthogonal polygons. In [5], we proposed an alternative shape: the orthogonally convex polygon cover of the point set which is always an orthogonally convex polygon, and we show that in polynomial time we can compute an orthogonally convex polygon which covers a given point set or report that such a polygon does not exist. Moreover, if more than one such orthogonally convex polygons exist, then we also provide a count of all the solutions in $O(n \log n)$ time and construct all of them in time linearly proportional to this count.

☆ This article is an extended version of Evrendilek et al. (2010) [4] which was presented in ISCO'10.

* Corresponding author. Fax: +90 2324888475.
E-mail addresses: burkay.genc@tedu.edu.tr, burkaygenc@gmail.com (B. Genç).

0166-218X/$ – see front matter
doi:10.1016/j.dam.2012.01.018

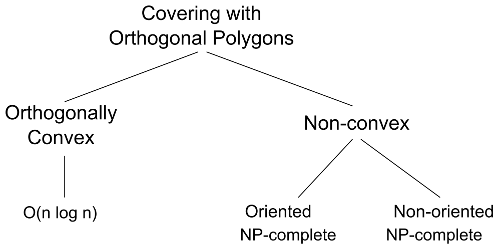

Fig. 1. The complexity of covering points with orthogonal polygons.

In this paper, we provide new research output related to covering points with orthogonal but non-convex polygons and extend our preliminary results in [4]. We show that covering points with orthogonal non-convex polygons is NP-complete. This problem remains intractable even when the orientations of the edges covering the given points are specified as part of the input. Thus, this paper along with the results in [5], closes the problem of covering points in the plane with orthogonal polygons (see Fig. 1).

The rest of the paper is organized as follows: Some background information and required preliminaries are introduced in the next section. Section 3 describes the transformation from a known NP-complete problem to covering points with orthogonal polygons. Next, in Section 4, we provide some theoretical properties to facilitate the remaining steps of the NP-completeness proof given in Section 5. Finally, in Section 6, the paper is concluded by pointing out future research directions.

2. Preliminaries and problem definitions

A *polygonal chain* is an ordered set of $k > 1$ line segments $e_1, e_2, \ldots, e_k$ such that for each $i \in \{1, \ldots, k-1\}$, e_i and e_{i+1} intersect at one of their endpoints and no other intersections occur within the set. A *closed polygonal chain* is a polygonal chain of length at least three for which the first and the last segments are the same. A *polygon* is a region on the plane enclosed by a closed polygonal chain. The *edges* of a polygon are the maximal line segments on the boundary of the polygon. The *vertices* of a polygon are the intersection points of its edges. A line segment is *orthogonal* if it is parallel to one of the coordinate axes.

An orthogonal line segment is *horizontal* (resp. *vertical*) if it is parallel to the *x*-axis (*y*-axis). A line segment *covers* a point if the point is on the line segment and is not one of its vertices. If a horizontal (resp. vertical) line segment covers a point, then we say that the point is a *horizontal point* (resp. *vertical point*) and we define its *orientation* to be horizontal (resp. vertical).

Points with integer coordinates in the plane are called *grid-points*. Two points in the plane that have the same *y*-coordinate (*x*-coordinate) value are said to be *horizontally* (*vertically*) *aligned*.

In this paper, we deal with covering points with orthogonal polygons where we relax the convexity condition as opposed to [5]. We consider, however, an additional aspect of the problem: The issue of whether each point is covered by a horizontal or vertical edge is specified in advance or not as part of the input. We accordingly define two problems *COGOP* and *CGOP* each corresponding to a possible choice below.

Specifically, covering oriented grid-points with an orthogonal polygon (*COGOP*) is the problem of deciding whether a given set *P* of *n* grid-points in the plane, each constrained to a previously specified orientation, can be covered with an orthogonal polygon such that each edge of the polygon covers exactly one point and each point is covered by exactly one edge.

Covering grid-points with orthogonal polygons (*CGOP*) is the problem of deciding whether a given set *P* of *n* grid-points in the plane can be covered with an orthogonal polygon such that each edge of the polygon covers exactly one point and each point is covered by exactly one edge.

It should be noted that *COGOP* and *CGOP* both require that there is a bijection between the points to be covered and the edges of the covering polygon.

3. Transformation to covering points with orthogonal polygons

Any given solution to either *CGOP* or *COGOP* can quickly be verified in polynomial time simply by checking that intersections do not exist between the polygonal edges and that there is a one-to-one mapping between points and edges (and that the orientations match those specified in the instance, in case of *COGOP*). *COGOP* and *CGOP* $\in NP$ is, thus, easily established.

First, the more restricted versions, namely *COGOP* and *CGOP*, where the points are constrained to grid-points are shown to be NP-complete. This helps to simplify the presentation. Finally, the NP-completeness of the more general forms when points have real-valued coordinates easily follows. As the points in a given instance of *CGOP* or *COGOP* are at grid-points, the requirement for a bijection between the points to be covered and the edges of the covering polygon implies that the vertices of any orthogonal polygon corresponding to a covering are also formed at grid-points.

We use a single reduction for *CGOP* and *COGOP*. We start with a problem already known to be NP-complete and then transform it to either *CGOP* or *COGOP* through identical lines.

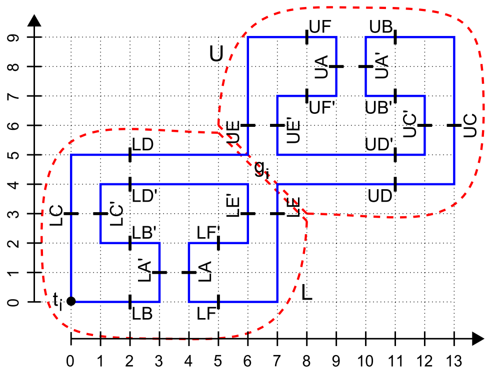

Fig. 2. The gadget used in *CFST* to *CGOP* (respectively *COGOP*) transformation. Each t_i is replaced by 24 new points. The edges of the gadget polygon are purely indicative.

We will make use of the following problem (*CFST*) in our reduction: Given a set P of n grid-points in the plane, deciding whether P admits a crossing-free spanning tree with only axes-parallel edges. This problem is shown to be NP-complete in [7].

We are now in a position to describe the transformation from a given instance of *CFST* to the corresponding instance of either *COGOP* or *CGOP*. First, in linear time we translate all points to the positive quadrant. Next, we scale the coordinates of the points with a certain integer factor α which we soon explain in detail. The scaling is so that when we replace each scaled point with the gadget in Fig. 2, the x (resp. y) range spanned by a gadget overlaps with another gadget if and only if the points corresponding to the gadgets before scaling had the same x (resp. y) coordinate.

After scaling, each obtained point $t_i(x_i, y_i)$ corresponding to the original grid-point p_i of a particular *CFST* instance is replaced by a gadget g_i composed of 24 *gadget points* shown in Fig. 2. These transformed points, by construction, are all at integer coordinates.

The orthogonal polygon (given in solid lines) in Fig. 2 which covers the gadget points is purely indicative. We call this the *gadget polygon*. We use the terms gadget polygon and gadget interchangeably throughout the rest of the paper. In the next section, we prove that the gadget polygon is unique, in that, it is the only orthogonal polygon that can cover the given set of 24 points associated with a single gadget even when the orientations of edges are not specified in the input.

Each gadget spreads a rectangular area of 13 by 9 units. This observation, accordingly, leads to the derivation of the smallest possible value of α as $\max(13, 9) + 1 = 14$ units.

Let $t_i \cdot x_i$ and $t_i \cdot y_i$ denote the x and y coordinates of the point t_i respectively. Gadget points can be regarded as consisting of two groups, namely, those in the upper right region (U) identified by $x \geq t_i \cdot x_i + 6$ and $y \geq t_i \cdot y_i + 4$ and the rest in the lower left region (L) identified by $x \leq t_i \cdot x_i + 7$ and $y \leq t_i \cdot y_i + 5$ (or points to the left and right respectively of the line $y = -x + t_i \cdot x_i + t_i \cdot y_i + 11$). All the points in one of U or L surrounded by dotted lines (in red) as shown in Fig. 2 can be obtained through a rotation of the points in the other region by an angle of π radians around $t_i + (13/2, 9/2)$. There are, hence, a total of 12 points in each of these regions and these are accordingly named to start with either U or L. The points in each region, furthermore, come in pairs of two as: (A, A'), (B, B'), (C, C'), (D, D'), (E, E'), and (F, F'). One point of each such 2-tuple essentially functions as an enforcer used to dictate an orientation. This is actually what makes it possible to use a single reduction for both *CGOP* and *COGOP*.

We identify each gadget point by a two-letter *label*. While the first letter is used to specify the region U or L to which the respective gadget point belongs, the second identifies one of the points that comes in pairs. Any one point in region $r \in \{U, L\}$ designated by letter $l \in \{A, A', C, C', E, E'\}$ will thus be denoted by g_i^{rl}.

Gadgets have been designed in such a way that when two gadget polygons are connected somehow, they are to form an orthogonal polygon with a single boundary. To this end, the gadget includes a pair of sentinel points (A, A') in each region U and L to prevent the formation of holes in the orthogonal covering. A hole in this context means an orthogonal polygon inside another.

While the reduction to *CGOP* requires that only the coordinates of the gadget points are specified, *COGOP* demands that their orientations are also given. Indeed, they are all known in advance. The orientation of a point of some gadget g_i is basically the same as that of the corresponding edge of the gadget polygon associated with g_i as will be demonstrated in the next section. The total time spent in reducing an instance of *CFST* to the corresponding instance of either *COGOP* or *CGOP* is $O(n)$.

4. Theoretical properties

Some observations are now made below to facilitate the remaining steps of the NP-completeness proof.

Two gadgets are said to be *horizontally* (*vertically*) *aligned* if the same labeled points in the gadgets have the same y-coordinate (x-coordinate) values. The *x-range* of a gadget g_i is the range defined by the x-coordinates of g_i^{LC} and g_i^{UC}. Similarly, *y-range* is the range defined by the y-coordinates of g_i^{LB} and g_i^{UB} (see Fig. 2).

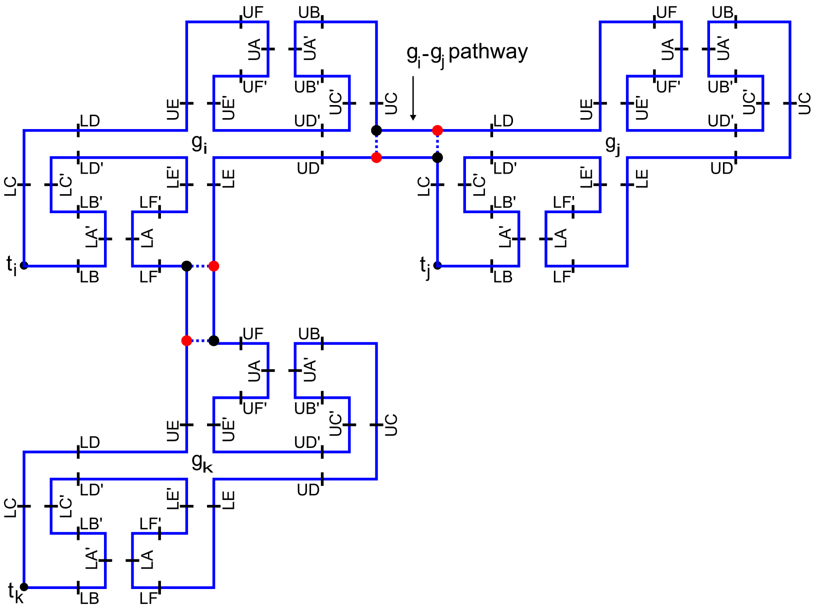

Fig. 3. Possible connections among gadget points.

Lemma 4.1. *After an instance of CFST is transformed as described above to a corresponding instance of either COGOP or CGOP, for any two given gadgets g_i and g_j with $i \neq j$, one and only one of the following three parts holds:*

(i) *they are horizontally aligned and their x-ranges do not overlap,*
(ii) *they are vertically aligned and their y-ranges do not overlap,*
(iii) *they are not aligned and neither their x-ranges nor their y-ranges overlap.*

Proof. If g_i and g_j are not aligned, then the respective points in the gadgets having the same label are at least fourteen units apart in both x and y directions as a result of the α-scaling performed during reduction. Since a gadget can extend in any direction by as far as at most thirteen units, part (iii) in the lemma follows directly. When they are, on the other hand, aligned along the direction of one of the axes, say x (y), they cannot be within the x-range (y-range) of one another as the corresponding gadget polygons cannot intersect again as a result of the α-scaling by construction. Parts (i) and (ii) of the lemma are, thus, established. □

A vertex v of an orthogonal polygon is said to be *implied by a pair of gadget points* (p, q) if v is formed by an intersection of the two edges covering p and q respectively. Each of the vertices of a gadget polygon implied by one of these pairs of gadget points (g_i^{UE}, g_i^{UF}), (g_i^{LE}, g_i^{LF}), (g_i^{UC}, g_i^{UD}), and (g_i^{LC}, g_i^{LD}) (see Fig. 3) is unique in that they are the only vertices that suggest the possibility of alternative vertices to form connections to other gadget polygons. As illustrated in Fig. 3, given two gadgets g_i and g_j which are horizontally aligned, we may have either the two vertices implied by (g_i^{UC}, g_i^{UD}) and (g_j^{LC}, g_j^{LD}) or the two vertices implied respectively by (g_i^{UC}, g_j^{LD}) and (g_i^{UD}, g_j^{LC}) in a covering. Such a structure between aligned gadgets is called a *pathway*. An example of a horizontal g_i–g_j pathway is shown in Fig. 3. Each bidirectional g_i–g_j pathway where g_i and g_j are horizontally (resp. vertically) aligned is said to be *identified by two vertices* implied by the pairs (g_i^{UC}, g_j^{LD}) and (g_i^{UD}, g_j^{LC}) (resp. (g_i^{UE}, g_j^{LF}) and (g_i^{UF}, g_j^{LE})). Two such vertices are alternatively said to be the *identifying vertices* of a pathway.

Some simple observations common to *COGOP* and *CGOP* are now made in the form of two structural lemmas below. They are used repeatedly in the subsequent proofs.

Lemma 4.2. *Given two horizontally (resp. vertically) aligned points $p(x_p, y_p)$ and $q(x_q, y_q)$ where $x_p < x_q$ (resp. $y_p < y_q$) among a set P of n points, they must both have vertical (resp. horizontal) orientations in any possible covering by an orthogonal polygon when it is known that there exists no vertical (resp. horizontal) points to cover in the region designated by $x_p < x < x_q$ (resp. $y_p < y < y_q$).*

Proof. If p and q are horizontally aligned, then $y_p = y_q$ by definition. Neither p nor q in that case can have a horizontal orientation since such a point would be covered by a horizontal edge which, by the given assumption, has no points with a vertical orientation to cover next before hitting the other. On the other hand, hitting the other would be a violation of the requirement for a bijection between the points to be covered and the edges of the covering polygon given in the problem definitions. □

Let us consider, for example, the gadget points g_i^{UC} and $g_i^{UC'}$ which are horizontally aligned as shown in Fig. 2. Both are constrained to have vertical orientations by the above lemma. The points $g_i^{UD'}$ and g_i^{UD} which are vertically aligned, however, are both seen to be vertical by a direct application of Lemma 4.2 as already depicted in Fig. 2.

Lemma 4.3. *Given a horizontal point $p(x_p, y_p)$ and a vertical point $q(x_q, y_q)$ where $x_p \neq x_q$ and $y_p \neq y_q$, the vertex implied by (p, q) is always part of a solution (if any) when the edges covering p and q are both known to extend orthogonally in the direction of the other a distance of no less than $|x_p - x_q|$ and $|y_p - y_q|$ respectively.*

Proof. This leads to a configuration where lines through p and q intersect at the point (x_q, y_p). Consequently, a vertex is always formed at this point as part of a solution. □

Let us now consider the points $g_i^{UC'}$ and $g_i^{UD'}$ already known to have vertical and horizontal orientations respectively. By assumption all the points are at integer coordinates, hence the edges covering both gadget points will extend at least one unit in either possible direction as dictated by their respective orientations. This, in turn, leads to the formation of the vertex implied by $(g_i^{UD'}, g_i^{UC'})$ as suggested by Lemma 4.3.

The lemma below now formalizes the type of connections among gadget points belonging either to the same or to different gadgets. It should be noted at this point that one such lemma concerning only *CGOP* will be needed.

Lemma 4.4. *If S is any orthogonal polygon covering (solution) of the corresponding CGOP instance obtained from a given CFST instance, then the following two hold for all g_i:*

(i) *For any $r \in \{U, L\}$ and $l \in \{A, A', C, C', E, E'\}$ the gadget points g_i^{rl} have all vertical orientations while the rest of the gadget points have all horizontal orientations. In addition to that, all vertices but the four which are implied by the pair of gadget points given as: (g_i^{UE}, g_i^{UF}), (g_i^{UC}, g_i^{UD}), (g_i^{LE}, g_i^{LF}), and (g_i^{LC}, g_i^{LD}) always belong to the vertex set of S.*

(ii) *For every one of the remaining four vertices implied by the pair of gadget points (g_i^{UE}, g_i^{UF}), (g_i^{UC}, g_i^{UD}), (g_i^{LE}, g_i^{LF}), and (g_i^{LC}, g_i^{LD}), either it belongs to the vertex set of S or there exists a gadget g_j aligned with g_i such that they form a pathway whose two identifying vertices belong to the vertex set of S.*

Proof. (i) Let us consider a gadget g_i as depicted in Fig. 3. For a given $r \in \{U, L\}$ and a 2-tuple $(l, l') \in \{(A, A'), (C, C'), (E, E')\}$, points g_i^{rl} and $g_i^{rl'}$ are observed to be horizontally aligned by construction. Besides, by the grid-points assumption there can be no other points in the region in between the vertical lines passing through g_i^{rl} and $g_i^{rl'}$ as they are only 1 unit apart. It is therefore deduced by a direct application of Lemma 4.2 that g_i^{rl} and $g_i^{rl'}$ both have vertical orientations.

Having confirmed all six such pairs, and hence a total of 12 out of 24 gadget points as vertically oriented, we can now proceed to verify the remaining orientations. Points g_i^{rl} and $g_i^{rl'}$ where $r \in \{U, L\}$ and $(l, l') \in \{(B, B'), (D, D'), (F, F')\}$ are seen to be vertically aligned. At this time, either there are no gadget points or the gadget points have already been figured out in the previous step to have vertical orientations in the designated region between the horizontal lines passing through g_i^{rl} and $g_i^{rl'}$. As such, they are all horizontal by Lemma 4.2.

In investigating the vertex set now, we draw on the orientations which have all been already resolved. It is easily observed by the grid-points assumption that an edge covering a horizontal (resp. vertical) point has to extend at least one unit in both directions to the left (resp. top) and to the right (resp. bottom) of that same point. This observation coupled with Lemma 4.3 lets us easily locate the 14 vertices implied by (g_i^{rF}, g_i^{rA}), $(g_i^{rF'}, g_i^{rA})$, $(g_i^{rB}, g_i^{rA'})$, $(g_i^{rB'}, g_i^{rA'})$, $(g_i^{rB'}, g_i^{rC'})$, $(g_i^{rF'}, g_i^{rE'})$, and $(g_i^{rC'}, g_i^{rD'})$ where $r \in \{U, L\}$. Once these vertices are formed, we leave the horizontal edge going through $g_i^{rD'}$ with no option but to extend four units until the vertical line through $g_i^{rE'}$ as all the vertical points along the way have already been covered by forming the vertices discovered so far. It is exactly when, by a direct application of Lemma 4.3, two more vertices, namely, those implied by $(g_i^{rD'}, g_i^{rE'})$ for $r \in \{U, L\}$, are discovered. By the exact same lines, vertices implied by (g_i^{UD}, g_i^{LE}) and (g_i^{LD}, g_i^{UE}) are also seen to be part of a solution.

As the final step of the first part of the lemma, the vertex implied by (g_i^{rB}, g_i^{rC}) for some $r \in \{U, L\}$ is explored. The horizontal and vertical lines through g_i^{rB} and g_i^{rC} respectively are seen to form a vertex by Lemma 4.3 since none of them can take a turn before they intersect at this point. With the addition of these 2 vertices each corresponding to a possible r, all 20 vertices mentioned are shown to belong to S.

(ii) There remains to investigate the remaining 4 pairs of open line segments. Among these (g_i^{UE}, g_i^{UF}) and (g_i^{LE}, g_i^{LF}) are completely symmetrical and fall under a category which we call vertical pathways. (g_i^{UC}, g_i^{UD}) and (g_i^{LC}, g_i^{LD}) can similarly be classified together under the name of horizontal pathways. We do this part of the proof for only horizontal pathways since the vertical pathways are similar.

As the two pairs associated with horizontal pathways are symmetrical, we pick (g_i^{UC}, g_i^{UD}) (see Fig. 3) and do the proof for only that pair. At this state of affairs where the orientations of all the gadget points and 20 out of 24 vertices associated with each g_i have already been determined, the vertical line, after passing all the way down through g_i^{UC}, cannot extend beyond the point implied by this pair. In order to see this, we can simply note that the horizontal line through g_i^{UD} has to extend right at least 2 units till the point implied by (g_i^{UC}, g_i^{UD}). Before this point, it can neither turn up nor turn down to cover any point. It cannot go up as the only vertically oriented point $g_i^{UC'}$ has already been covered forming the vertex implied

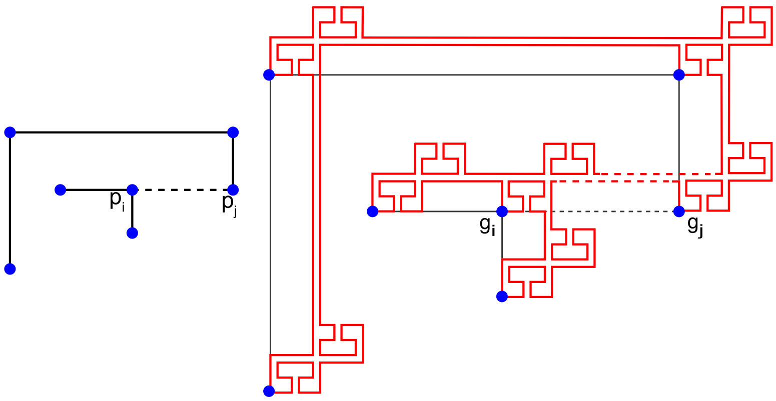

Fig. 4. The correspondence between *CFST* and *COGOP* (or *CGOP*).

by $(g_i^{UD'}, g_i^{UC'})$. It cannot certainly go down if there are no points in this direction which is exactly the case by construction, when there does not exist another gadget aligned vertically with g_i. Even when there is a gadget g_k for some $k \neq i$ vertically aligned with g_i (see Fig. 3), it cannot yet again go down or else it would be intersecting the horizontal edge through g_k^{UB} which has already been observed to form the vertex implied by (g_k^{UB}, g_k^{UC}).

The vertical edge through g_i^{UC} then should make a turn at or before the point implied by (g_i^{UC}, g_i^{UD}). If it extends as far as that point, then from the above discussion, the vertex implied by this pair is necessarily formed. Otherwise when it has to take a turn before the point implied by this pair, there is only one grid-point along the way where it can turn. It is shown by a small circle filled in black right above the point implied by $(g_i^{UD'}, g_i^{UC'})$ in Fig. 3. The only possibility there is to turn right as $g_i^{UD'}$ which is the only horizontal point on the left has already been covered. A right turn, on the other hand, can only be taken when there exists a g_j for some $j \neq i$ horizontally aligned with g_i as shown in Fig. 3. In this case, the gadget point g_j^{LD} with a horizontal orientation is readily seen to be the only one which can be covered next. S must then contain the vertex implied by (g_i^{UC}, g_j^{LD}) as depicted in Fig. 3. An application of Lemma 4.3 subsequently leads to the formation of the additional vertex implied by (g_j^{LC}, g_i^{UD}) as the vertical line through g_j^{LC} and the horizontal line through g_i^{UD} are now known to extend to this point without being able to take any turns along the way. Hence, it is concluded that the two defining vertices of the g_i–g_j pathway are in the solution as the other possibility stated in this part of the lemma. □

An example of the type of connections described by Lemma 4.4 is shown in solid lines in Fig. 3. It is therefore established that points belonging to different gadgets can only be connected via pathways as described by Lemma 4.4. Another consequence of Lemma 4.4 is that the orientations of all the edges in any covering of the transformed instance corresponding to a given instance of *CFST* are uniquely determined and known in advance. This allows for the same reduction used for *CGOP* to be also employed for *COGOP* by simply feeding the known orientations to the latter.

The correspondence between a given instance of *CFST* and the instance of either *COGOP* or *CGOP* obtained by the transformations described is shown in Fig. 4.

After the transformation, all 24 points associated with a single gadget act in unison atomically like the very point in *CFST* instance they replace. Just like a point can be connected to other points through horizontal or vertical edges in *CFST*, a gadget can be connected to other gadgets through horizontal and vertical pathways.

We will now define the operation of *removing a pathway* and make some observations to prepare for the final result. Let us denote by S a solution to a corresponding *CGOP* or *COGOP* instance obtained from a *CFST* instance. S is obviously an orthogonal polygon and contains (possibly zero or more) pathways. If we remove a pathway in S, this will effectively cut the cycle corresponding to S into two. This operation can easily be illustrated by referring to Fig. 3. In case a g_i–g_j pathway shown in this figure is removed, the two identifying vertices of the pathway implied by (g_i^{UC}, g_j^{LD}) and (g_i^{UD}, g_j^{LC}), shown by black dots, are removed from the vertex set of S while the vertices implied by (g_i^{UC}, g_i^{UD}) and (g_j^{LC}, g_j^{LD}), indicated by red dots, are added to the vertex set of S. Thus, removing a pathway in S gives rise to the formation of two orthogonal polygons: S_1 and S_2. One may wonder at that point whether the points associated with a single gadget may somehow, after removing a pathway, be distributed between S_1 and S_2 which we call *gadget splitting*. As gadgets must still be connected only through pathways in either S_1 or S_2, gadget splitting is not possible. This discussion is summarized in the following lemma.

Lemma 4.5. *In any solution S to a corresponding COGOP or CGOP instance obtained from a given CFST instance, when a g_i–g_j pathway is removed, then two non-intersecting orthogonal polygons S_1 and S_2 are created such that no gadgets are split between S_1 and S_2.*

Proof. When a g_i–g_j pathway is removed, we have effectively cut the cycle corresponding to S into two: S_1 and S_2. Since no additional pathways and gadget points are introduced, no intersections are possible between S_1 and S_2. This is because the corresponding *COGOP* or *CGOP* instances have been obtained from a *CFST* instance.

Let us assume by contradiction that a gadget g_k for some k is split between S_1 and S_2. Then, there must be a path in S_1 leading to a point of g_k and another path in S_2 leading to another point of g_k. The existence of these two disjoint paths implies, by Lemma 4.4, the existence of two disjoint sequences of (zero or more) pathways in S before the removal. While the sequence $\pi_{i,k}$ denotes the pathways from g_i to g_k in S, the other from g_k to g_j is represented by $\pi_{k,j}$. The sequences $\pi_{i,k}$ and $\pi_{k,j}$ are disjoint and neither contains in it the g_i–g_j pathway. Let us consider now another sequence π which is constructed by appending $\pi_{k,j}$ to $\pi_{i,k}$ and finally by adding the g_i–g_j pathway as the last element. The sequence π is certainly in S. If we restrict our attention to the gadgets associated with the pathways only in π, we realize that it contains two orthogonal polygons after a connection through the last pathway g_i–g_j in π is performed. The additional orthogonal polygon is a by-product of connecting an orthogonal polygon that was formed before by all the pathways but g_i–g_j in π to itself this time through g_i–g_j. This contradicts the fact that S is an orthogonal polygon. □

Lemma 4.6. *In any solution to a corresponding instance of COGOP (respectively CGOP) with $n > 0$ gadget polygons, there are exactly $n - 1$ pathways.*

Proof. The proof is by induction on the number of gadgets denoted by n. When $n = 1$, the only orthogonal polygon covering all 24 points of a single gadget is, by Lemma 4.4, the one shown in Fig. 2. Assume by the inductive hypothesis that for any instance with n gadgets with $1 \le n \le k$ for some constant k, if it has a solution, then the solution has exactly $n - 1$ pathways. Let S be any covering for $n = k + 1$ gadget polygons. We can now choose any g_i–g_j pathway and remove it. Note that there is always such a pathway in S as we have now at least two gadget polygons that can only be connected through pathways according to Lemma 4.4. Since the removal of one pathway cuts S through the pathway without changing the number of gadgets, it definitely generates two new orthogonal polygons S_1 and S_2 covering all $k + 1$ gadgets. As no new vertices other than the vertices of gadget polygons have been introduced and as no gadget can ever be split by Lemma 4.5 after the removal of the pathway, each of the two disjoint orthogonal polygons clearly still satisfies Lemma 4.4. Thus, we can apply the inductive hypothesis to both S_1 and S_2 since they both contain k or fewer gadget polygons. Let $|S_1|$ and $|S_2|$ denote the number of gadgets in each. The total number of pathways in S can then be given by the sum of the number of pathways in each of S_1 and S_2 plus one pathway already removed which is equal to $|S_1| - 1 + |S_2| - 1 + 1 = k$ by recalling $|S_1| + |S_2| = n$. □

5. Equivalence

We are now just a single step from concluding the proof. The following lemma is to take this last step. Let us use *C(O)GOP* to refer to both *COGOP* and *CGOP* in the remainder of the paper.

Lemma 5.1. *An instance of CFST has a solution if and only if the corresponding C(O)GOP instance has a solution.*

Proof. First, the "only if" part (necessity) is proved. Let us assume that a given instance of *CFST* has a solution. This solution then can be used to construct an orthogonal polygon covering all the (oriented) points in the corresponding *C(O)GOP* instance. The construction scheme is such that while the given solution to a *CFST* instance is rebuilt from scratch by incrementally adding the edges in the solution one at a time, the solution to the corresponding *C(O)GOP* instance is also built incrementally. Initially, the points p_i in the *CFST* instance are all disconnected forming a forest of n subtrees of size one each while the corresponding configuration for *C(O)GOP* has n non-intersecting orthogonal polygons corresponding exactly to the gadget polygons. Each of these polygons is obtained by an independent covering of a set of (oriented) points associated with a single gadget g_i corresponding to p_i which, at the time, constitute the one and only node of some subtree. Therefore, the gadget polygons in the *C(O)GOP* instance are initially in one-to-one correspondence with the single node subtrees of *CFST*. It should be noted that the gadget polygons are guaranteed by Lemma 4.1 not to cross.

Next, the edges in the given solution to *CFST* are examined one at a time in some arbitrary order. When an edge (p_i, p_j) is chosen in some iteration, it is used to modify the existing configurations of both *CFST* and *C(O)GOP* as follows: First, the configuration of *CFST* instance is updated to reflect the addition of the edge (p_i, p_j) by merging two disjoint subtrees P and Q where $p_i \in P$ and $p_j \in Q$ to form a new larger subtree. Last but not least, the configuration of the *C(O)GOP* instance is also modified to form a pathway between the gadgets g_i and g_j as dictated by Lemma 4.4. Linking g_i and g_j with a pathway, in turn, causes the formation of a single orthogonal polygon obtained from the two orthogonal polygons corresponding to the two subtrees P and Q. The resulting orthogonal polygon is, consequently, an independent cover corresponding to the newly formed subtree. Once the iteration is completed, thus, both the number of subtrees in the current configuration of *CFST* and the number of orthogonal polygons in the corresponding configuration of *C(O)GOP* decrease by one. After $n - 1$ iterations, finally, a single orthogonal polygon corresponding to the solution tree is obtained. This concludes the proof for this part, and hence the necessity.

Finally, we must establish the "if" part (sufficiency). Assume that we have a solution to an instance of *C(O)GOP* corresponding to an instance of *CFST*. We know by Lemma 4.4 that a pathway is possible only when the corresponding gadgets are aligned. A solution to *CFST* can then be constructed as follows: Let the solution be represented by the Hamiltonian cycle μ corresponding to a clockwise (or counter clockwise) traversal of the gadget points on the boundary of the orthogonal polygon identified by the solution. The nodes corresponding to the gadget points in μ are visited starting from any node iteratively one node at a time until the cycle is completed. Whenever two gadget points that belong to two separate gadgets

g_i and g_j ($i \neq j$) are consecutively traversed (a g_i–g_j pathway is taken), the corresponding points p_i and p_j are basically joined by an edge if they have not been already joined. Thus, the solution to *CFST* is constructed incrementally with the addition of such edges. Let us prove immediately that the described scheme correctly builds a solution for the corresponding *CFST* instance. We start by noting that all gadget points are included in the Hamiltonian cycle μ. This, in turn, implies that there exists a path between every pair of points in the solution constructed for *CFST*. It can, therefore, be concluded that the obtained solution to *CFST* has, graph theoretically, a single connected component *spanning* all p_i. By Lemma 4.4, points p_i in *CFST* are connected by the described scheme only through axis-parallel edges, hence *orthogonal*. The solution so-constructed can also be easily shown to be *crossing-free*. Lastly, it needs to be shown that the corresponding solution to *CFST* is a *tree*. It is already known that all the points are spanned by the corresponding solution and the number of pathways in a solution with n gadgets is exactly $n - 1$ as specified by Lemma 4.6. This concludes the proof. □

It follows immediately from the previous results that *COGOP* and *CGOP* are both NP-complete.

Theorem 5.2. *COGOP is NP-complete.*

Theorem 5.3. *CGOP is NP-complete.*

Having established Theorems 5.2 and 5.3, we can relax the restriction to grid-points and readily obtain the problems *COROP* and *CROP* corresponding to *COGOP* and *CGOP* respectively. Covering oriented points with an orthogonal polygon (*COROP*) is formally introduced as the problem of deciding whether a given set P of n points in the plane, each constrained to a previously specified orientation, can be covered with an orthogonal polygon such that each edge of the polygon covers exactly one point and each point is covered by exactly one edge. Covering points with orthogonal polygons (*CROP*) is the problem of deciding whether a given set P of n points in the plane can be covered with an orthogonal polygon such that each edge of the polygon covers exactly one point and each point is covered by exactly one edge. It is evident from these definitions that *COGOP* and *CGOP* are special cases of *COROP* and *CROP* respectively.

The following corollaries about the NP-completeness of *COROP* and *CROP* where points may have real-valued coordinates can thus be stated without a proof.

Corollary 5.4. *COROP is NP-complete.*

Corollary 5.5. *CROP is NP-complete.*

6. Conclusions

In this paper, we study the problem of covering points with possibly non-convex orthogonal polygons where the orientations of the edges covering the given points may be specified or not as part of the input. We show that both problems are NP-complete using the same reduction.

In future work, we will study the optimization versions of *CGOP* and *COGOP* where we wish to minimize the number of uncovered points by an orthogonal polygon. We plan to investigate the approximability of these optimization problems.

Acknowledgments

The authors wish to thank the anonymous referees for their constructive criticism and valuable comments.

References

[1] N. Ahuja, Dot pattern processing using Voronoi polygons as neighborhoods, in: International Conference on Pattern Recognition, 1980, pp. 1122–1127.
[2] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, C. Silva, Point set surfaces, in: VIS '01: Proceedings of the Conference on Visualization '01, IEEE Computer Society, 2001, pp. 21–28.
[3] T. Biedl, B. Genç, Reconstructing orthogonal polyhedra from putative vertex sets, Computational Geometry: Theory and Applications 44 (8) (2011) 409–417.
[4] C. Evrendilek, B. Genç, B. Hnich, Covering oriented points in the plane with orthogonal polygons is NP-complete, in: Proceedings of International Symposium on Combinatorial Optimization 2010, in: Electronic Notes in Discrete Mathematics, vol. 36, Elsevier, 2010, pp. 303–310.
[5] B. Genç, C. Evrendilek, B. Hnich, Covering points with orthogonally convex polygons, Computational Geometry: Theory and Applications 44 (5) (2011) 249–264.
[6] L. Jackson, S.K. Wismath, Orthogonal polygon reconstruction, in: Proceedings of the 8th Canadian Conference on Computational Geometry, Carleton University Press, 1996, pp. 44–49.
[7] K. Jansen, G.J. Woeginger, The complexity of detecting crossing free configurations in the plane, BIT Numerical Mathematics 33 (4) (1993) 580–595.
[8] M. Löffler, E. Mumford, Connected rectilinear graphs on point sets, in: I.G. Tollis, M. Patrignani (Eds.), Graph Drawing, in: LNCS, vol. 5417, Springer, 2008, pp. 313–318.
[9] J. O'Callaghan, Computing the perceptual boundaries of dot patterns, Computer Graphics and Image Processing 3 (2) (1974) 141–162.
[10] J. O'Rourke, Uniqueness of orthogonal connect-the-dots, in: G. Toussaint (Ed.), Computational Morphology, Elsevier, 1988, pp. 97–104.
[11] T. Ottmann, E. Soisalon Soininen, D. Wood, On the definition and computation of rectilinear convex hulls, Information Sciences 33 (1984) 157–171.
[12] D. Rappaport, On the complexity of computing orthogonal polygons from a set of points, Tech. Rep. SOCS-86.9, McGill U., Montréal, Canada, 1986.