Telling convex from reflex allows to map a polygon

Jérémie Chalopin Shantanu Das Yann Disser Matúš Mihalák
Peter Widmayer

Abstract

We consider the exploration of a simple polygon $\mathcal{P}$ by a robot that moves from vertex to vertex along edges of the visibility graph of $\mathcal{P}$. The visibility graph has a vertex for every vertex of $\mathcal{P}$ and an edge between two vertices, if they see each other, i.e. if the line segment connecting them lies inside $\mathcal{P}$ entirely.

While located at a vertex, the robot is capable of ordering the vertices it sees in counter-clockwise order as they appear on the boundary, and for every two such vertices, it can distinguish whether the angle between them is convex ($\leq \pi$) or reflex ($> \pi$). Other than that, distant vertices are indistinguishable to the robot.

We assume that an upper bound on the number of vertices is known and show that the robot is always capable of reconstructing the visibility graph of $\mathcal{P}$. We also show that multiple identical, indistinguishable and deterministic such robots can always position themselves such that they mutually see each other, i.e. such that they form a clique in the visibility graph.

1 Introduction

Autonomous mobile robots are used for various tasks like cleaning, guarding, data retrieval, etc. in unknown environments. Many such tasks require the exploration of the environment and the creation of a map. The difficulty of the mapping problem depends on the characteristics of the environment itself and on the sophistication of the robots, i.e. on their sensory and locomotive capabilities. A natural question is how much sophistication a robot needs in order to be able to solve the problem. The ultimate goal is to understand the general difficulty of the mapping problem by finding minimal robot configurations that allow a robot to create a map.

We focus on robots operating inside simple polygons. For many tasks, instead of inferring a detailed map of the geometry of the environment, it is enough to obtain the visibility graph. The visibility graph has a node for every vertex of the polygon, and an edge connecting two nodes, if the corresponding vertices "see each other", i.e. if the straight-line segment between them lies entirely within the polygon. The goal in this context becomes to find minimal robot models that allow a robot inside a polygonal environment to reconstruct the visibility graph of its environment. The information such a robot can gather must be sufficient to *uniquely* infer the visibility graph. A major difficulty when dealing with visibility graphs is that while they have been studied extensively, their characterization remains open [11].

A variety of minimalistic robot models have been studied algorithmically in the past, focusing on different types of environments and objectives [1, 5, 8, 10, 15]. The variant considered in this paper originates from [13]. Roughly speaking, a robot is allowed to move from vertex to vertex along the edges of the visibility graph. While situated at a vertex, the robot sees the vertices visible to its current location and can order them as they appear in counter-clockwise (ccw) order starting with the ccw neighbor along the boundary. Apart from this ordering, the vertices are indistinguishable to the robot. In each move the robot may select one of them and instantaneously move to it. After moving, the robot has no way of "looking back", i.e. it has no immediate way of knowing which vertex it came from among the vertices it sees now. The only piece of global information a robot is assumed to be aware of is an upper bound $\bar{n}$ on the total number of vertices n.

Unless extended with additional capabilities, a robot with the above characteristics cannot reconstruct the visibility graph of a polygon when restricted to moving along the boundary only [3]. If we allow the robot to measure the angles between pairs of visible vertices in addition to ordering them, moving along the boundary was shown to be sufficient to reconstruct the visibility graph however [9]. As soon as a robot starts moving across the polygon (as opposed to along the boundary), the lack of the ability to look back makes it difficult for the robot to relate the information it collected so far to subsequent observations. It thus makes sense to consider *look-back robots* which have the ability to look back and identify the vertex they came from in their last move. This ability empowers a look-back robot to retrace all of its movements. If we add the ability to distinguish convex ($\leq \pi$) and reflex ($> \pi$) angles, it was shown that a look-back robot can reconstruct the visibility graph [3]. Later, it was shown that a look-back robot in fact does not even need to distinguish convex and reflex angles [6]. In the same paper, it was also shown that look-back robots can solve the weak-rendezvous problem in which multiple identical, indistinguishable and deterministic robots need to position themselves such that they mutually see each other. In the following we show that a robot can reconstruct the visibility graph even without looking back, as long as it can distinguish convex and reflex angles. Along the way, we show that such robots can also solve the weak-rendezvous problem.

In the robot model we use, robots move along edges of the visibility graph and can locally access some information about the edges. We can thus view our setting in the context of general robotic exploration of edge-labeled graphs. The edge-labeling is usually restricted to be locally bijective at every vertex (i.e. no two edges incident to the same vertex have the same label). In this more general context, robots are aware of the degree of the vertex they are located at as well as of the labels of the edges incident to it. In every step, a robot can select one edge and move to its other end. In this general setting, it is known that there are graphs which appear mutually indistinguishable to a robot, i.e. the reconstruction problem is not always solvable [2, 4]. The rendezvous problem is not solvable in both cases either [7, 14]. We will see later, that our setting can be transformed to the exploration of a particular class of directed, arc-labeled graphs. We will show that for this class of graphs both the reconstruction problem as well as the weak-rendezvous problem become solvable.

As it is impossible to fully reconstruct a graph in general, it is natural to ask how much information a robot can obtain about a graph. This information is encoded in the unique so-called *minimum base graph* of a graph G, which is the smallest graph among all graphs indistinguishable from G by a robot [4]. For general graphs, the mapping from a graph to its minimum base graph is not one-to-one in the sense that there are graphs which share the same minimum base graph. Our question whether a robot with certain capabilities can reconstruct the visibility graph of a polygon can be translated to the question whether the mapping is one-to-one for the class of visibility graphs with an appropriate labeling. We show that if the number of vertices is bounded, and the labeling locally encodes the convexity information about every angle at a vertex, this mapping becomes one-to-one. In other words, visibility graphs can be reconstructed from their minimum base graph, if a bound $\bar{n}$ on the total number of vertices and the type of every angle (convex or reflex) are known.

2 The visibility graph reconstruction problem

We consider the exploration of a (simple) polygon $\mathcal{P}$ by a robot that moves from vertex to vertex along straight lines inside $\mathcal{P}$. For every pair of vertices u, v that can be connected with a straight line inside $\mathcal{P}$, we say that u and v *see each other*. The visibility graph $G_{\text{vis}} = (V, E)$ of $\mathcal{P}$ is a directed graph where V is the set of vertices of $\mathcal{P}$ and there is a pair of arcs between any two vertices u, v that see each other, i.e. there is an arc from u to v and an arc from v to u. Whenever convenient, we identify G_{vis} with its canonical straight-line embedding in the polygon. For example, we speak of angles between arcs of G_{vis} when we mean the angles between the corresponding line segments of its straight-line embedding.

Depending on what additional capabilities we equip a robot with, it might or might not be able to perform certain tasks. We focus on the *visibility graph reconstruction problem* in which the robot has to uniquely infer G_{vis}. Here and throughout this paper we consider isomorphic graphs to be the "same" graph, as we cannot hope to distinguish graphs further. We also consider the *weak-rendezvous problem* in which multiple identical, indistinguishable and deterministic robots need to position themselves on vertices of the polygon such that they mutually see each other.

Before defining a specific robot model, we introduce some formalism for G_{vis}. We fix a vertex v_0 and denote the vertices of $\mathcal{P}$ in ccw order along the boundary by $v_0, v_1, \ldots, v_{n-1}$. Note that $v_0, v_1, \ldots, v_{n-1}, v_0$ is a Hamiltonian cycle in G_{vis}. By $\text{chain}(v_l, v_r)$ we denote the sequence $(v_l, v_{l+1}, \ldots, v_r)$ and by $\text{chain}_v(v_l, v_r)$ we denote the subsequence of $\text{chain}(v_l, v_r)$ containing only the vertices visible to v. Here and throughout this paper all indices are understood modulo n. Let $v_i \in V$ and $(u_1, \ldots, u_{d_i}) := \text{chain}_{v_i}(v_{i+1}, v_{i-1})$ be the vertices visible to v_i. We say d_i is the degree of v_i and define $\text{vis}_{v_i}(x) := \text{vis}_{v_i}(-(d_i + 1 - x)) := u_x$ to be the x-th vertex visible to v_i in ccw order or equivalently the $(d_i + 1 - x)$-th vertex visible to v_i in clockwise (cw) order for $1 \leq x \leq d_i$. Conversely, we set $O_{v_i}(u_x) := x$ or interchangeably $O_{v_i}(u_x) = -(d_i + 1 - x)$ for $1 \leq x \leq d_i$. For $1 \leq x < y \leq d_i$ we write $A_{v_i}(x, y) = A_{v_i}(y, x)$ to denote the ccw angle between the arcs (v_i, u_x) and (v_i, u_y) in that order. Furthermore, we define the *angle type* $T_{v_i}(\cdot, \cdot)$ as follows: $T_{v_i}(x, y) = T_{v_i}(y, x) = 1$, if $A_{v_i}(x, y) > \pi$ and $T_{v_i}(x, y) = 0$ otherwise. For convenience we set $T_{v_i}(x, x) = 0$. A reflex vertex is a vertex v_i for which $T_{v_i}(1, d_i) = 1$, all other vertices are called convex.

The exploration of G_{vis} can be reduced to the general problem of exploring a strongly connected, directed and arc-labeled graph G (from now on we use the word "graph" to refer to such graphs). In this setting the arcs of the graph are labeled and we write $\lambda(e)$ to denote the label of an arc e. A robot exploring a general graph is assumed to be aware of the labels of all the arcs emanating from its current location. In every move, the robot may choose one of those arcs and follow it to its target. In the following we distinguish between (directed) paths that visit every vertex at most once and (directed) walks that do not have this restriction. Every walk p in the graph uniquely induces a label-sequence $\lambda(p)$. Conversely, any label-sequence Λ induces a set of walks $\Lambda(G)$ such that $\lambda(p) = \Lambda$ for all $p \in \Lambda(G)$. By $\Lambda(v)$ we denote the set of walks in $\Lambda(G)$ that start at v. If no two outgoing arcs of any vertex share the same label, we say the graph has a *local orientation* or is *locally oriented*. Then for every label-sequence Λ and vertex v we have $\Lambda(v) = \emptyset$ or $|\Lambda(v)| = 1$, in the latter case for convenience we write $\Lambda(v)$ to denote this unique walk.

We now introduce in more detail the robot model we will consider throughout this work. As described above, we allow a robot to move along arcs of the visibility graph. In addition, while situated at a vertex v of degree d, the robot can order all outgoing arcs in ccw order starting with the arc to its ccw neighbor along the boundary, and is aware of $T_v(x, y)$ for all $1 \leq x, y \leq d$. We assume the robot to be aware of an upper bound $\bar{n} \geq n$ on the total number of vertices n. From now on, when we talk about a robot in a polygon, we refer to the robot model described above.

The exploration of $\mathcal{P}$ by a robot is in fact equivalent to the exploration of an arc-labeled version G_{vis}, if we define an appropriate labeling that encodes all the information available to a robot into the labels. For every vertex of the polygon we need to encode the local orientation and the angle type information into an arc-labeling of the outgoing arcs of the corresponding vertex in G_{vis}. We introduce a labeling in which each label is a sequence of integers. Let u be a vertex of the visibility graph with degree d and (u, v) be an arc. We label (u, v) with the label $(x_0, x_1, \ldots, x_d)$, where $x_0 := O_u(v)$ and $x_i := T_u(x_0, i)$. Note that by the definition of O_u our labeling makes G_{vis} locally oriented. Further note that in general the arcs (u, v) and (v, u) can be labeled differently. It is immediate to check that a simple robot exploring G_{vis} encounters the exact same information as a robot inside the polygon (that is aware of T_v), if both start at corresponding vertices. It is thus sufficient to show that our labeled graph G_{vis} can be reconstructed in the framework of exploring general graphs in order to show that a robot can indeed solve the *visibility graph reconstruction problem*.

3 Overview of the algorithm

The visibility graph reconstruction algorithm for robots that we design in this paper combines several old and new graph-theoretical and geometrical properties of visibility graphs as well as techniques developed in earlier studies. Rather than formally introducing all relevant concepts right away, this section aims to give an intuitive outline of the algorithm. We informally describe the underlying techniques and defer their formal discussion to later sections. Note that we are primarily interested in showing that a robot is at all capable of uniquely reconstructing the visibility graph of any simple polygon. Hence, the algorithm we provide as a proof does not need to be particularly efficient as long as it is guaranteed to terminate in finite time. An algorithm that solves the weak-rendezvous problem is obtained as a byproduct.

In Section 2 we argued that the exploration of $\mathcal{P}$ by a robot is equivalent to the exploration of (the labeled version of) G_{vis} in the context of general graph exploration. In general and without any prior knowledge of the graph, there can be infinitely many graphs that are compatible with the observations of the robot, no matter how far it moves, i.e. all these graphs are indistinguishable to the robot. However it is known [4] that for every graph G, there is always a unique *minimum base* graph $G^\star$ that is indistinguishable from G and has minimum size. Using the fact that G_{vis} is locally oriented and that an upper bound $\bar{n}$ on n is known a priori, we are able to show the following result.

Theorem 1. *A robot in $\mathcal{P}$ can determine $G^\star_{\text{vis}}$.*

Roughly speaking, the main ingredient for this theorem is that given two candidate graphs for $G^\star_{\text{vis}}$, the robot can eliminate one of the two in finite time by following an appropriate sequence of arc labels. It is then sufficient to iterate over pairs of graphs with size at most $\bar{n}$, discarding one of the two in every step. Once the robot determined $G^\star_{\text{vis}}$, it essentially has all the information it can possibly gather. Subsequent steps of the algorithm can thus operate on $G^\star_{\text{vis}}$ directly and in fact, the robot does not need to move any further since $G^\star_{\text{vis}}$ already contains all the information it can hope to obtain.

We associate each vertex of G_{vis} with a vertex of $G^\star_{\text{vis}}$ such that each vertex of $G^\star_{\text{vis}}$ represents a *class* of vertices of G_{vis}. For two vertices u, v of G_{vis} in the same class we have $\Lambda(u) = \emptyset \Leftrightarrow \Lambda(v) = \emptyset$ for all label-sequences Λ. Furthermore, the classes with which the vertices are associated repeat periodically along the boundary and in particular all classes have the same size. We define a unique order between the classes and use a procedure similar to the one in [6] to show that at least one of them forms a clique in G_{vis}. The idea is to repeatedly "cut off" *ears* of the polygon, i.e. vertices whose neighbors on the boundary see each other. Cutting off such an ear yields a subpolygon of $\mathcal{P}$ and we can repeat the process on the subpolygon. However, the robot cannot operate on G_{vis} directly as it only has access to $G^\star_{\text{vis}}$. The following lemma allows the robot to cut off an entire class of vertices at a time, an operation that can be performed in $G^\star_{\text{vis}}$ simply by deleting the corresponding vertex (and adjusting the arc labels of its neighboring vertices).

Lemma 2. *Let v be an ear of $\mathcal{P}$. Then every vertex in the same class as v is an ear of $\mathcal{P}$.*

As every polygon has at least one ear, the robot can thus "cut off" an entire class of $\mathcal{P}$ in order to obtain a new and smaller polygon $\mathcal{P}'$ (cf. Figure 1). By removing the corresponding vertex of $G^\star_{\text{vis}}$ and updating the arc labels, it obtains a graph $G'^\star_{\text{vis}}$ that is indistinguishable from the visibility graph of $\mathcal{P}'$. If this process is repeated, always selecting the smallest class with respect to the order relation for removal, eventually a situation is reached in which only one (uniquely

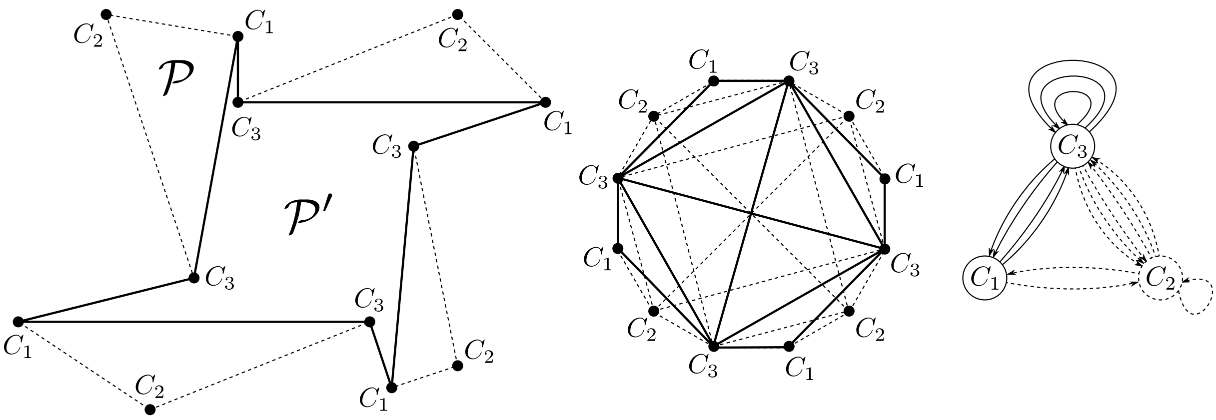

Figure 1: Left: cutting away a class of vertices (ears) from $\mathcal{P}$ to obtain $\mathcal{P}'$. Middle: visibility graph G_{vis} of $\mathcal{P}$. Right: minimum base graph $G^{\star}_{\text{vis}}$ of G_{vis}. Dashed edges are present in $\mathcal{P}$ but not in $\mathcal{P}'$.

defined) class $C^{\star}$ remains. As the corresponding subpolygon must again have at least one ear, by the above lemma the entire class $C^{\star}$ consists of ears and the corresponding subpolygon thus is convex. A convex polygon is a clique in the visibility graph and we may conclude the crucial theorem

Theorem 3. *There is a uniquely defined class $C^{\star}$ in G_{vis} whose vertices form a clique.*

While the robot could explicitly execute the procedure described above, finding the class $C^{\star}$ can be done much more directly. If the number of self-loops of a vertex in $G^{\star}_{\text{vis}}$ equals the size of the corresponding class minus one, this class is a clique. It is thus enough to inspect all classes in turn. Among all classes that form a clique, the largest class with respect to the order relation must be $C^{\star}$. The previous theorem guarantees the existence of such a class. This result also gives a robot the means to infer n from $\bar{n}$: n is equal the size of $C^{\star}$ times the number of classes in G_{vis}. To compute the size of $C^{\star}$, the robot can do the following. Consider a vertex v in G^{*}_{vis} such that the number of self-loops incident to v is greater or equal than the number of self-loops incident to any other vertex of G^{*}_{vis}. Then, the class C corresponding to v is a clique and there are exactly $|C| - 1$ self-loops incident to v.

The above immediately gives an algorithm for multiple robots to weakly meet: As $C^{\star}$ is unique, every robot can determine $C^{\star}$ and then simply position itself on a vertex of $C^{\star}$. We get

Theorem 4. *Any number of robots in $\mathcal{P}$ can solve the weak-rendezvous problem.*

Starting from the clique $C^{\star}$, we show that by sequentially "gluing" ears back to the polygon, a robot can extend the initial clique and reconstruct the entire visibility graph step by step. Every step relies on a recursive counting method that was first introduced in [3]. In order to know how to glue ears back on, the robot explicitly needs to construct $C^{\star}$ by repeatedly cutting away ears and in the process remember in which order the classes are cut off.

Theorem 5. *A robot in $\mathcal{P}$ can solve the visibility graph reconstruction problem.*

4 Finding the minimum base graph $G^{\star}_{\text{vis}}$

This section focuses on the problem of exploring a general, locally oriented directed graph $G = (V, E)$ with a robot. Again, we assume an upper bound $\bar{n}$ on the number of vertices n to be known and we do not impose a limitation on the memory of the robot. We prove a generalization of Theorem 1 to general, locally oriented graphs.

Before we define the notion of the *minimum base graph* $G^\star$ of G we need to introduce a few graph-theoretical concepts. First, given an arc e from vertex u to vertex v, we denote by $s(e)$ the *source of arc* e, i.e. the vertex u, and by $t(e)$ the *target of arc* e, i.e. the vertex v. Note that in the following we allow graphs to have parallel arcs between a pair of vertices. A *morphism* $\mu : G \to G'$ from G to a graph G' is a mapping from G to G' that maps vertices to vertices and arcs to arcs and maintains adjacencies and arc labels. More formally, if e is an arc in G from u to v then $s(\mu(e)) = \mu(u)$, $t(\mu(e)) = \mu(v)$, and $\lambda(e) = \lambda(\mu(e))$. An *opfibration* $\varphi : G \to \bar{G}$ with $\bar{G} = (\bar{V}, \bar{E})$ is a morphism such that for every arc $\bar{e} \in \bar{E}$ with $\bar{u} = s(\bar{e})$ and for every vertex $u \in \varphi^{-1}(\bar{u})$ in the preimage of $\bar{u}$ there is a *unique* arc e with source $s(e) = u$ such that $\varphi(e) = \bar{e}$. We say that $\bar{G}$ is a *base graph* of G and G is a *total graph* of $\bar{G}$. Trivially, G is both its own base graph and total graph. If G has no base graph smaller than itself, we say G is *opfibration prime.* An *out-tree* is a graph that has a *root* vertex r such that there is exactly one directed path from r to every other node.

We give the following properties without proof. For a detailed discussion, refer to [4].

Proposition 6. *Let $\varphi : G \to \bar{G}$ be an opfibration. For every label-sequence Λ and every vertex $v \in V$ we have that $\Lambda(v) \neq \emptyset$ iff $\Lambda(\varphi(v)) \neq \emptyset$.*

Proposition 7. *There is exactly one opfibration prime base graph of G. We call it the* minimum base graph *of G and denote it with $G^\star$.*

Proposition 8. *For every $v \in V$, there is a unique (but not necessarily finite) total graph H_v of G that is an out-tree with root in $\varphi^{-1}(v)$, where φ is the opfibration mapping H_v to G. We call it the* universal total graph *of G at v.*

Proposition 9. *A graph is opfibration prime, iff its universal total graphs are pairwise distinct.*

Proposition 10. *Two different opfibration prime graphs have different sets of universal total graphs.*

We can now show that if we have a local orientation, there is a label sequence of finite length that can be used to distinguish any two opfibration prime graphs.

Lemma 11. *Let $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ be two distinct locally oriented opfibration prime graphs. There is a label-sequence δ of finite length for which $\delta(G_1) = \emptyset$ and $\delta(G_2) \neq \emptyset$ or vice versa.*

Proof. We first show that (without loss of generality) there is a vertex $x \in V_1$, such that for every vertex $v_2 \in V_2$ there is a label-sequence δ_{x,v_2} of finite length with $\delta_{x,v_2}(x) \neq \emptyset$ and $\delta_{x,v_2}(v_2) = \emptyset$ or vice versa: By Proposition 10, without loss of generality, there is a vertex $x \in V_1$ such that the universal total graph H_x of G_1 at x is not a total graph of G_2. Then for every vertex $v_2 \in V_2$, H_x and the universal total graph H_{v_2} of G_2 at v_2 are different. Because G_1 and G_2 are locally oriented, so are H_x and H_{v_2}. Let r_x and r_{v_2} be the roots of H_x and H_{v_2} respectively. Because H_x, H_{v_2} are distinct and locally oriented, there is a finite label-sequence δ_{x,v_2} with $\delta_{x,v_2}(r_x) \neq \emptyset$ and $\delta_{x,v_2}(r_{v_2}) = \emptyset$ or vice versa. By Proposition 6 this implies $\delta_{x,v_2}(x) \neq \emptyset$ and $\delta_{x,v_2}(v_2) = \emptyset$ or vice versa.

We now describe how to use the above to obtain the desired label-sequence δ. We start with the empty label-sequence $\delta^{(0)}$ and iteratively extend it to a longer but still finite label-sequence $\delta^{(i)}$ in step i. Let $A^{(i)} := \{v \in V_1 | \delta^{(i)}(v) \neq \emptyset\}$ and $B^{(i)} := \{v \in V_2 | \delta^{(i)}(v) \neq \emptyset\}$ be the sets of vertices that are “compatible” with $\delta^{(i)}$. As $\delta^{(i+1)}$ extends $\delta^{(i)}$, we have by construction that $A^{(i+1)} \subseteq A^{(i)}$ and $B^{(i+1)} \subseteq B^{(i)}$. We show that our extension satisfies $\left(A^{(i+1)} \cup B^{(i+1)}\right) \subsetneq \left(A^{(i)} \cup B^{(i)}\right)$ in every step and that either $\delta^{(i+1)}(G_1) \neq \emptyset$ or $\delta^{(i+1)}(G_2) \neq \emptyset$. At some point we thus obtain a label-sequence δ for which exactly one graph has no compatible vertices. It remains to show the existence of such an extension.

Let $\delta^{(i)}$ be a finite label-sequence with $\delta^{(i)}(G_1) \neq \emptyset$ or $\delta^{(i)}(G_2) \neq \emptyset$. If $A^{(i)} = \emptyset$ or $B^{(i)} = \emptyset$, we have either $\delta^{(i)}(G_1) = \emptyset$ or $\delta^{(i)}(G_2) = \emptyset$. We can thus set $\delta = \delta^{(i)}$ and are done. So assume $A^{(i)} \neq \emptyset$ and $B^{(i)} \neq \emptyset$. Then, there are two vertices $v_1 \in A, v_2 \in B$. Let $p_1 = \delta^{(i)}(v_1)$, $p_2 = \delta^{(i)}(v_2)$ and v_1' be the target of p_1 (i.e. the vertex at which p_1 ends). As G_1 is strongly connected, there is a path q from v_1' to x, where x is defined as above. Let $\pi = \lambda(q)$ be the associated label-sequence and $\pi^+ = \delta^{(i)} \circ \pi$, where "$\circ$" denotes the concatenation of sequences. We certainly have $\pi^+(v_1) \neq \emptyset$ and thus $\pi^+(G_1) \neq \emptyset$. If $\pi^+(v_2) = \emptyset$, we set $\delta^{(i+1)} = \pi^+$ and have $B^{(i+1)} \subsetneq B^{(i)}$. Otherwise let v_2'' be the target of $\pi^+(v_2)$ (remember that x is the target of $\pi^+(v_1)$). Without loss of generality, we can set $\delta^{(i+1)} = \pi^+ \circ \delta_{x,v_2''}$. By definition of $\delta_{x,v_2''}$, we have $\delta^{(i+1)}(v_1) \neq \emptyset$ and $\delta^{(i+1)}(v_2) = \emptyset$ or vice versa. Thus $A^{(i+1)} \subsetneq A^{(i)}$ or $B^{(i+1)} \subsetneq B^{(i)}$ and hence $\left(A^{(i+1)} \cup B^{(i+1)}\right) \subsetneq \left(A^{(i)} \cup B^{(i)}\right)$. □

Theorem 12. *A robot exploring G can determine $G^\star$, if it knows an upper bound $\bar{n}$ on the size of G.*

Proof. For simplicity we prove the theorem for the case when the robot knows n exactly and show how to generalize the approach to the case when only an upper bound $\bar{n}$ on n is given.

By Proposition 7, $G^\star$ is unique. We will give an algorithm that maintains a finite set C of graphs that is always guaranteed to contain $G^\star$. In every step our algorithm will rule out at least one member of C, until there is only one left. This graph will then be $G^\star$. Throughout the algorithm, we denote by π_{hist} the label-sequence associated with the walk along which the robot has travelled so far and by v_{hist} the target of the walk. As G is locally oriented, π_{hist} together with the initial starting location of the robot uniquely corresponds to this walk in G. The walk however is not explicitly known to the robot as it neither knows G nor its starting location.

We start by setting C to contain all opfibration prime graphs of size at most n. In every step let G_1 be a graph of minimum size in C and G_2 be a graph of minimum size in $C \setminus \{G_1\}$ (if $C \setminus \{G_1\} = \emptyset$, we are done and set $G^\star = G_1$). We now describe how to discard either G_1 or G_2 from C.

By Lemma 11, there is a label-sequence δ for which $\delta(G_1) = \emptyset$ and $\delta(G_2) \neq \emptyset$ or vice versa. The robot can determine the shortest such label-sequence δ simply by enumerating all possible label-sequences in order of increasing lengths and checking for each in turn whether it has the desired property. Without loss of generality assume $\delta(G_1) = \emptyset$ and $\delta(G_2) \neq \emptyset$. The robot does not explicitly know G nor where in G it was initially located. It thus iterates over all candidate graphs $G' = (V', E')$ of size n and all vertices $v' \in V'$ (again there are only finitely many choices). For every choice of G', let $\Pi(G')$ be the set of all label-sequences associated to walks in G' that have the same length as δ. It is easy to see that there is a label-sequence π of finite length in G' for which $\pi(v_{\text{hist}}(v')) \neq \emptyset$ and which contains all label-sequences in $\Pi(G')$ as a subsequence. The robot follows this label-sequence (because of local orientation, every decision is unique) either until is reaches its end, or until it cannot anymore because there is no arc of the required label emanating from its current vertex.

As we iterate over every choice of G' and v', we are sure to reach G and the robot's initial starting location at some point in the process. Therefore we can be sure that in the end π_{hist} contains all label-sequences associated to walks in G with the same length as δ as a substring, and of course conversely all substrings of π_{hist} of that length are label-sequences of walks in G. It thus simply remains to check whether π_{hist} contains δ as a substring. If yes, we discard G_1 from C and otherwise we discard G_2. We can do this because $\delta(G_1) = \emptyset$ and $\delta(G_2) \neq \emptyset$ and because by Proposition 6, any valid choice for $G^\star$ must have the same set of label-sequences as G. We then continue with new choices for G_1 and G_2. After a finite number of steps C will only contain one graph which is a valid choice for $G^\star$. This concludes the proof.

Observe now that if only an upper bound $\bar{n}$ on n is given, the algorithm can easily be adapted to find $G^\star$ in the same way by iterating over all graphs G' of size *at most* $\bar{n}$ for every pair of graphs G_1, G_2. □

We obtain Theorem 1 immediately by applying Theorem 12 to G_{vis}. Note that the results of this section are not restricted to visibility graphs.

5 Identifying the clique $C^\star$

In this section we study structural properties of $G^\star_{\text{vis}} = (V^\star, E^\star)$ which we later use to show Theorem 3.

Let $\varphi : G_{\text{vis}} \to G^\star_{\text{vis}}$ be the opfibration from G_{vis} to $G^\star_{\text{vis}}$. As $G^\star_{\text{vis}}$ is the minimum base of G_{vis}, φ is unique. Every vertex $v^\star$ of $G^\star_{\text{vis}}$ corresponds to a set of vertices of G_{vis}. We write $C_{v^\star} := \varphi^{-1}(v^\star) \subseteq V$ and say $C_{v^\star}$ is the *class* of $v^\star$. For all $v \in \varphi^{-1}(v^\star)$, we set $C_v := C_{v^\star}$. It follows immediately from the definition of opfibrations that every two vertices u, v of the same class C_u have the same degree d and that due to local orientation we have $C_{\text{vis}_u(i)} = C_{\text{vis}_v(i)}$ for all $1 \le i \le d$. We may thus write $C_u(i) := C_{\text{vis}_u(i)}$. Finally, we define $\mathcal{B} := \big(C_{v_0}, C_{v_1}, \ldots, C_{v_{n-1}}\big)$ to be the sequence in which the classes appear along the boundary.

As $G^\star_{\text{vis}}$ is opfibration prime, by Proposition 9 every vertex has its unique universal total graph. We can use this and define a natural order $\mathcal{O}$ on the vertices of $G^\star_{\text{vis}}$ and thus on the classes of G_{vis}.

Lemma 13. *The sequence $\mathcal{B}$ is periodical with period $|V^\star|$ and thus all classes have the same size.*

Proof. The image of the boundary under φ must consist of $\frac{n}{|V^\star|}$ copies of a Hamiltonian cycle in $G^\star_{\text{vis}}$. Therefore $\mathcal{B}$ is periodical with period $|V^\star|$ and all classes have the same size $\frac{n}{|V^\star|}$. □

In the following, we prove that if a vertex from some class is an ear, then every vertex of that class is an ear. Recall that an ear of G_{vis} is a vertex $v_i \in V$ for which v_{i-1} and v_{i+1} see each other. We will need the following property of the shortest curve between two vertices of $\mathcal{P}$.

Theorem 14 ([12])**.** *Let $s, t \in V$. There is a unique shortest curve p from s to t that lies in $\mathcal{P}$. This curve is a chain of straight-line segments connected at reflex vertices of $\mathcal{P}$, and the two line segments at any vertex of p form a reflex angle. We say p is the* (euclidean) shortest path *in $\mathcal{P}$ between s and t.*

Lemma 15. *Let $|V^\star| > 2$ and $v_x, v_y \in V$ such that $C_{v_x}(2) = C_{v_y}$ and $C_{v_y}(-2) = C_{v_x}$. Then, $C_{v_{x+2}} = C_{v_y}$ and every vertex in $C_{v_{x+1}}$ is an ear.*

Proof. We start by observing that for all $v_i \in V$ with $\text{vis}_{\text{vis}_{v_i}(2)}(-2) = v_i$ we have $\text{vis}_{v_i}(2) = v_{i+2}$ and thus v_{i+1} is an ear. For the sake of contradiction assume $\text{vis}_{\text{vis}_{v_i}(2)}(-2) = v_i$ but $\text{vis}_{v_i}(2) \neq v_{i+2}$. Consider the subpolygon induced by $\text{chain}(v_i, \text{vis}_{v_i}(2))$. This subpolygon has at least four vertices as $\text{vis}_{v_i}(2) \notin \{v_{i+1}, v_{i+2}\}$. In the visibility graph of the subpolygon, v_i and $\text{vis}_{v_i}(2)$ are neighbors on the boundary and both have degree two, which is a contradiction to the fact that every polygon must admit a triangulation. Therefore $\text{vis}_{v_i}(2) = v_{i+2}$ and v_{i+1} is an ear as its neighbors on the boundary see each other.

Because of the above observation, it is sufficient to show that for every $v \in C_{v_x}$ we have $\text{vis}_{\text{vis}_v(2)}(-2) = v$. For the sake of contradiction assume in the following that there is a vertex $u^{(0)} \in C_{v_x}$ with $\text{vis}_{\text{vis}_{u^{(0)}}(2)}(-2) \neq u^{(0)}$.

We define an infinite sequence $Z = \big(u^{(0)}, v^{(1)}, u^{(1)}, v^{(2)}, \ldots\big)$ by $v^{(l)} := \text{vis}_{u^{(l-1)}}(2)$ and $u^{(l)} := \text{vis}_{v^{(l)}}(-2)$ for all $l > 0$. Obviously $u^{(l)} \in C_{v_x}, v^{(l)} \in C_{v_y}$ for all $l \ge 0$. Intuitively, Z is the zig-zag line obtained by alternatingly travelling along the first and the last non-boundary arc in ccw order, starting at $u^{(0)}$. It is immediate to see that for any fixed index $l' \ge 0$ we have $u^{(l)}, v^{(l)} \in \text{chain}(u^{(l')}, v^{(l')})$ for all $l \ge l'$. Hence the part of the boundary in which these vertices lie becomes smaller and smaller and from some index $l_0 \ge 0$ on we have $u^{(l)} = u^{(l_0)}$ and $v^{(l)} = v^{(l_0)}$ for all $l \ge l_0$ (we set l_0 to be the smallest such index). Let $0 \le i, j < n$ such that

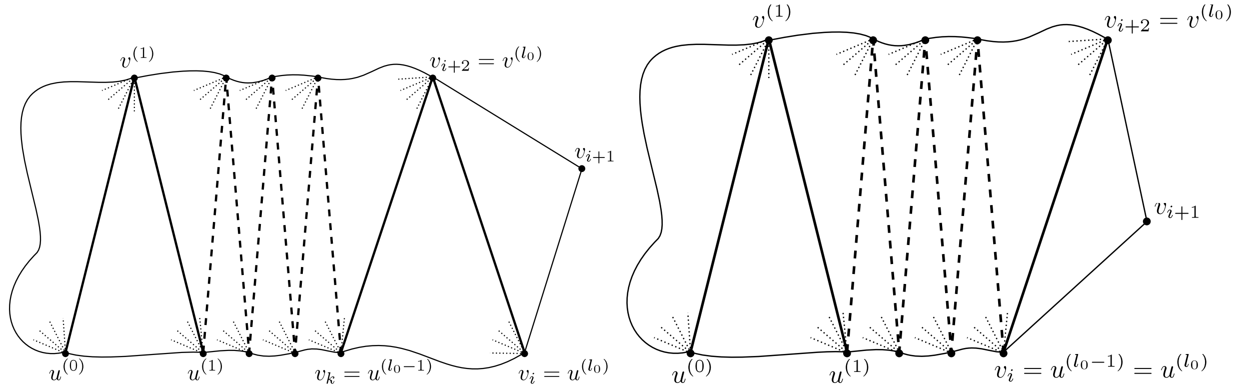

Figure 2: Visualisation of the "zig-zag" sequence Z. As Z does not self-intersect, there is a point l_0 from which on Z's entries do not change anymore. There are two cases how this point is reached: either $u^{(l_0-1)}$ is distinct from $u^{(l_0)}$ (left) or both are the same (right).

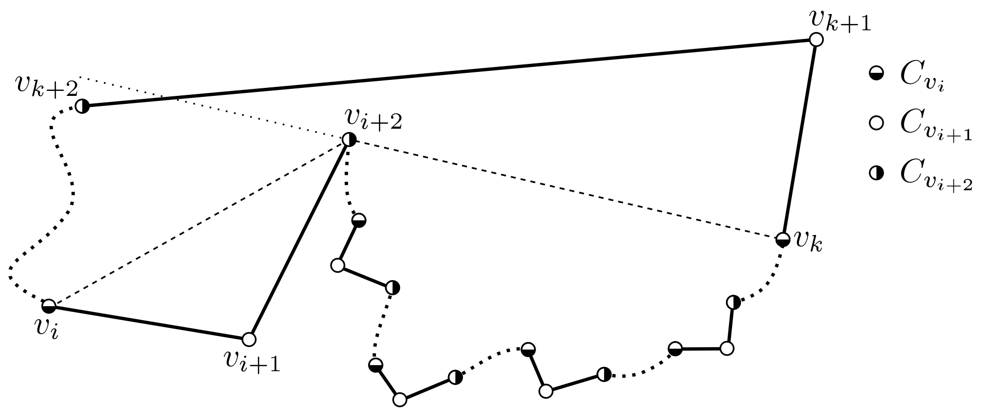

Figure 3: No vertex in chain(v_{i+3}, v_k) can see any vertex in chain(v_{k+2}, v_{i+1}).

$v_i = u^{(l_0)}, v_j = v^{(l_0)}$. We then have $\text{vis}_{v_i}(2) = v_j$ and $\text{vis}_{\text{vis}_{v_i}(2)}(-2) = v_i$. Thus by the above observation, v_{i+1} is an ear and $v_j = v_{i+2}$. As $v_i \in C_{v_x}$ and $v_j \in C_{v_y}$, this implies $C_{v_{x+2}} = C_{v_y}$. It remains to show that every vertex in $C_{v_{x+1}}$ is an ear.

We have to consider two cases. Either $u^{(l_0-1)}$ is distinct from $u^{(l_0)}$ or it is the same vertex (cf. Fig. 2). We assume $u^{(l_0-1)} \neq u^{(l_0)}$ and omit the discussion of the second case which is essentially analogous. Let $0 \leq k < n$ such that $v_k = u^{(l_0-1)}$. As $\text{vis}_{v_k}(2) = v_{i+2}$, we have that v_k does not see any vertex in chain(v_{k+2}, v_{i+1}) (note that this chain is not empty as $v_k \neq v_i$) and thus as $v_{k+1} \in C_{v_{x+1}}$ is in the same class as (the ear) v_{i+1}, the interior angle of the polygon at v_{k+1} is strictly smaller than π. For geometrical reasons (cf. Fig. 3) no vertex in chain(v_{i+3}, v_k) can see any vertex in chain(v_{k+2}, v_{i+1}). Let $X \subset C_{v_x}$ be the set of vertices of C_{v_x} in chain(v_{i+3}, v_k) and let $Y \subset C_{v_y}$ be the set of vertices of C_{v_y} in chain(v_{i+3}, v_k). As $|V^\star| > 2$, $C_{v_x}, C_{v_{x+1}}, C_{v_{x+2}}$ are all different and thus X and Y are disjoint. Note that because $\mathcal{B}$ is periodical with period $|V^\star|$ (Lemma 13) we have $|X| = |Y| + 1$.

We define the (undirected) bipartite graph $B_{xy} = \left(C_{v_x} \cup C_{v_y}, E_{xy}\right)$ with the edge-set $E_{xy} = \left\{\{u, v\} \in C_{v_x} \times C_{v_y} \,|\, (u, v) \in E\right\}$. In B_{xy} all vertices need to have the same degree d as $|C_{v_x}| = \left|C_{v_y}\right|$ and all vertices in either class have the same degree. We have $|X| = |Y| + 1$, we have that vertices in X can only have edges to vertices in $Y \cup \{v_{i+2}\}$ and that vertices in Y can only have edges to vertices in X. For all vertices to have the same degree, v_{i+2} cannot have any edges leading to $C_{v_x} \backslash X$. This is a contradiction to the fact that v_{i+2} sees v_i which is not in chain(v_{i+3}, v_k) and thus not in X. $\square$

We can now consider arbitrary values of $|V^\star|$ and prove Lemma 2.

Proof of Lemma 2. In the following we let $v_i \in V$ be an ear and show that all vertices in C_{v_i} are ears.

First consider the case $|V^\star| > 2$. As $(v_{i-1}, v_{i+1}) \in E$, we have $\text{vis}_{v_{i-1}}(2) = v_{i+1}$ and $\text{vis}_{i+1}(-2) = v_{i-1}$, and thus $C_{v_{i-1}}(2) = C_{v_{i+1}}$ and $C_{v_{i+1}}(-2) = C_{v_{i-1}}$. By Lemma 15 all vertices in C_{v_i} are ears.

Now consider the case $|V^\star| = 1$. In that case as v_i is convex, so are all vertices in C_{v_i}, as convexity is encoded in the arc-labeling. As $|V^\star| = 1$, this means that the polygon is convex and thus all vertices are ears.

It remains to consider the case $|V^\star| = 2$. Let $C_{v_j} \neq C_{v_i}$ be the second class in G_{vis}. Again, v_i being convex implies that all vertices in C_{v_i} are. For the sake of contradiction assume that there is a vertex $v_x \in C_{v_i}$ which is not an ear. Then v_{x-1} and v_{x+1} do not see each other, and by Lemma 13, $v_{x-1}, v_{x+1} \in C_{v_j}$. Let p be the shortest path in $\mathcal{P}$ between v_{x-1} and v_{x+1}. By Theorem 14, all vertices on p are reflex. This means that all vertices on p must be from C_{v_j} and thus all vertices of C_{v_j} must be reflex. Moreover, every vertex u in C_{v_j} has two neighbors u', u'' in C_{v_j} such that the angle between (u, u') and (u, u'') is reflex. If we cut off v_i from $\mathcal{P}$, we do not affect this property (every vertex u in C_{v_j} still has two neighbors from C_{v_j} forming a reflex angle) and we thus obtain a new polygon in which all vertices in C_{v_j} are still reflex. We can continue to obtain smaller and smaller subpolygons by selecting ears and cutting them off, maintaining the property that all vertices in C_{v_j} are reflex. Thus, in this process, we never cut off a vertex of C_{v_j}. This is a contradiction, as every polygon has at least one ear and thus the above process has to cut off all vertices eventually. □

Lemma 2 allows us to employ the following procedure repeatedly until only one class $C^\star$ remains: In step i, select the class $C^{(i)}$ which is smallest w.r.t. the order $\mathcal{O}$ among all classes of ears. We remove $C^{(i)}$ from the polygon by deleting the corresponding vertex from $G^\star_{\text{vis}}$ and updating the arc labels of its neighborhood accordingly. Removing class $C^{(i)}$ in that way produces a (not necessarily minimum) base graph of the visibility graph of the subpolygon obtained by cutting off all ears in $C^{(i)}$. In the next step we effectively consider this new polygon which again has to have at least one ear, and we are guaranteed to again have at least one class that contains only ears. Note that the above procedure does not require the base graph on which it operates in each step to be *minimum*. We start with the *minimum* base graph $G^\star_{\text{vis}}$ because it is the only base graph of G_{vis} the robot can infer at all.

If we repeat our procedure $|V^\star| - 1$ times, we are left with a single class $C^{(|V^\star|)} = C^\star$ and a sequence $\left(C^{(1)}, C^{(2)}, \ldots, C^{(|V^\star|-1)}\right)$ which is fixed by our order relation $\mathcal{O}$. As $C^\star$ again corresponds to a subpolygon and thus must contain at least one ear, every vertex in $C^\star$ must be an ear. Therefore the corresponding subpolygon is convex and $C^\star$ forms a clique in G_{vis}. This proves Theorem 3.

The existence of a clique gives us a way of computing n from $\bar{n}$ using $G^\star_{\text{vis}}$. By Lemma 13 we have $n = |V^\star| \cdot |C|$, where C is any class of G_{vis}. If we inspect the number of self-loops of every vertex of $G^\star_{\text{vis}}$, we are sure to encounter at least one clique, and thus $|C|$ is equal to the maximum number of self-loops plus one.

By Theorem 1, a robot can determine $G^\star$ in finite time. It thus can execute the above procedure and we obtain

Theorem 16. *A robot in $\mathcal{P}$ can determine the sequence $\mathcal{C} = \left(C^{(1)}, C^{(2)}, \ldots, C^{(|V^\star|)}\right)$, where $\mathcal{C}$ is the lexicographically smallest sequence such that for every $1 \leq i \leq |V^\star|$, all vertices in $C^{(i)}$ are ears in the subpolygon obtained by removing all vertices in $\bigcup_{j=1}^{i-1} C^{(j)}$ from $\mathcal{P}$.*

6 Reconstructing the visibility graph

In the following, we assume that the robot has already determined $G^\star_{\text{vis}}$ and the sequence $\mathcal{C} = \left(C^{(1)}, C^{(2)}, \ldots, C^{(|V^\star|-1)}, C^{(|V^\star|)}\right)$ from Theorem 16. For all $1 \leq i \leq |V^\star|$ we denote by $G^{(i)}_{\text{vis}} =$

$(V^{(i)}, E^{(i)})$ the subgraph of G_{vis} induced by $\bigcup_{j=i}^{|V^\star|} C^{(j)}$. By definition of $\mathcal{C}$, $G_{\text{vis}}^{(i)}$ is the visibility graph of a subpolygon $\mathcal{P}^{(i)}$ of $\mathcal{P}$. As $C^{(|V^\star|)} = C^\star$, by Lemma 13 we have that $G_{\text{vis}}^{(|V^\star|)}$ is the complete graph on $\frac{n}{|V^\star|}$ vertices. Together with the following lemma, this suggests a way of reconstructing $G_{\text{vis}} = G_{\text{vis}}^{(1)}$.

Lemma 17. *Let $1 \leq i < |V^\star|$. It is possible to determine $G_{\text{vis}}^{(i)}$ from $G_{\text{vis}}^{(i+1)}$.*

Proof. The set of vertices $V^{(i)}$ of $G_{\text{vis}}^{(i)}$ is given by $V^{(i)} = C^{(i)} \cup V^{(i+1)}$. It remains to show how to construct $E^{(i)}$. Let A be the set of arcs in G_{vis} between vertices of $C^{(i)}$ and $V^{(i+1)}$, and B be the set of arcs between vertices of $C^{(i)}$. We will first show how to construct A using the information contained in $G_{\text{vis}}^{(i+1)}$ and $G_{\text{vis}}^\star$. After having determined A, we can apply the same approach in order to obtain B. This completes the proof as $E^{(i)} = E^{(i+1)} \cup A \cup B$.

Note that every arc in G_{vis} has a counterpart of opposite orientation. In order to construct A it is thus sufficient to consider $e \in V^{(i+1)} \times C^{(i)}$ and show how to decide whether $e \in A$ or $e \notin A$. Deciding which elements of $C^{(i)} \times V^{(i+1)}$ are in A is then immediate. Equivalently, we can consider $v_j \in V^{(i+1)}$ with degree d in $G_{\text{vis}}^{(i)}$ and $1 \leq k \leq d$ such that $\text{vis}_{v_j}(k) \in C^{(i)}$, and show how to "identify" $\text{vis}_{v_j}(k)$, i.e. how to find x with $v_x = \text{vis}_{v_j}(k)$. If $k = 1$, we have $x = j + 1$ and if $k = d$, we have $x = j - 1$ because v_j sees its two neighbors on the boundary. Now assume $1 < k < d$. We will show that $v_y := \text{vis}_{v_j}(k-1) \notin C^{(i)}$. For the sake of contradiction assume that $v_y \in C^{(i)}$. In $\mathcal{P}^{(i)}$ all vertices of $C^{(i)}$ are ears and thus convex. By Lemma 13 and $i < |V^\star|$, there is more than one class and thus there is a vertex $v_z \in \text{chain}(v_{y+1}, v_{x-1})$ which is not visible to v_j. The shortest path in $\mathcal{P}$ from v_j to v_z must visit v_x or v_y, which is a contradiction to both vertices being convex (Theorem 14). We can deduce that $v_y \notin C^{(i)}$ and thus $(v_j, v_y) \in E^{(i+1)}$ is part of $G_{\text{vis}}^{(i+1)}$ and has already been identified, i.e. the index y is known. Because of Lemma 13, it is sufficient to know how many vertices of $C^{(i)}$ are in $\text{chain}(v_{y+1}, v_{x-1})$ in order to find x itself. From the labeling of $G_{\text{vis}}^\star$ we can deduce how many vertices of $C^{(i)}$ are in $\text{chain}_{v_y}(v_{y+1}, v_{x-1})$ (recall that $\text{chain}_{v_y}(v_{y+1}, v_{x-1})$ only contains vertices visible to v_y): As v_x is convex and thus cannot lie on a shortest path in $\mathcal{P}$ from v_j to another vertex of $C^{(i)}$, the first arc in ccw order from v_y to a vertex of $C^{(i)}$ that forms a convex angle with (v_y, v_j) must be (v_y, v_x) as the target of the arc must be visible to v_j. It is thus sufficient to count the number of arcs from v_y to vertices of $C^{(i)}$ before (v_y, v_x) in ccw order. We say the corresponding vertices are *hidden from v_j by v_y*. We still need a way to find the number of vertices of $C^{(i)}$ in $\text{chain}(v_{y+1}, v_{x-1})$ that are *not* visible to v_y. We can find this number by repeating our counting method recursively. For every vertex $v_l \in \text{chain}_{v_y}(v_{y+1}, v_{x-1}) \setminus C^{(i)}$ (in $G_{\text{vis}}^{(i)}$), we count all vertices of $C^{(i)}$ hidden from v_y by v_l. As the vertices in $C^{(i)}$ are convex, they cannot hide any vertices from v_y. The sum of all these counts finally gives the number of vertices of $C^{(i)}$ in $\text{chain}(v_{y+1}, v_{x-1})$. Together with Lemma 13 this number immediately yields the index x. The recursive counting method described above was first introduced in a similar setting where robots are allowed to retrace their movements [3]. Refer to [3] for a detailed proof of its correctness.

Using the fact that the arcs in A have already been identified, we can apply the exact same approach to construct B. □

Theorem 5 follows directly from Theorem 16, Lemma 17 and the fact that $G^{(|V^\star|)}$ is the complete graph on $\frac{n}{|V^\star|}$ vertices.

References

[1] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita. Distributed memoryless point convergence algorithm for mobile robots with limited visibility. *IEEE Transactions on Robotics and Automation*, 15(5):818–828, 1999.

[2] D. Angluin. Local and global properties in networks of processors. In *Proceedings of the twelfth annual ACM symposium on Theory of computing*, pages 82–93, 1980.

[3] D. Bilò, Y. Disser, M. Mihalák, S. Suri, E. Vicari, and P. Widmayer. Reconstructing visibility graphs with simple robots. In *Proceedings of the 16th International Colloquium on Structural Information and Communication Complexity*, pages 87–99, 2009.

[4] P. Boldi and S. Vigna. Fibrations of graphs. *Discrete mathematics*, 243(1-3):21–66, 2002.

[5] J. Brunner, M. Mihalák, S. Suri, E. Vicari, and P. Widmayer. Simple robots in polygonal environments: A hierarchy. In *Proceedings of the Fourth International Workshop on Algorithmic Aspects of Wireless Sensor Networks*, pages 111 – 124, 2008.

[6] J. Chalopin, S. Das, Y. Disser, M. Mihalák, and P. Widmayer. How simple robots benefit from looking back. In *Proceedings of the 7th International Conference on Algorithms and Complexity*, volume 6078 of *Lecture Notes in Computer Science*, pages 229–239. Springer, 2010.

[7] J. Chalopin, E. Godard, Y. Métivier, and R. Ossamy. Mobile agent algorithms versus message passing algorithms. In *Proceedings of the 10th International Conference on the Principles of Distributed Systems*, pages 187–201, 2006.

[8] R. Cohen and D. Peleg. Convergence of autonomous mobile robots with inaccurate sensors and movements. *SIAM Journal on Computing*, 38(1):276–302, 2008.

[9] Y. Disser, M. Mihalák, and P. Widmayer. Reconstructing a simple polygon from its angles. In *Proceedings of the 12th Scandinavian Symposium and Workshops on Algorithm Theory*, volume 6139 of *Lecture Notes in Computer Science*, pages 13–24. Springer, 2010.

[10] A. Ganguli, J. Cortés, and F. Bullo. Distributed deployment of asynchronous guards in art galleries. In *Proceedings of the 2006 American Control Conference*, pages 1416–1421, 2006.

[11] S. K. Ghosh. *Visibility Algorithms in the Plane*. Cambridge University Press, 2007.

[12] D.-T. Lee and F. P. Preparata. Euclidean shortest paths in the presence of rectilinear barriers. *Networks*, 14(3):393–410, 1984.

[13] S. Suri, E. Vicari, and P. Widmayer. Simple robots with minimal sensing: From local visibility to global geometry. *International Journal of Robotics Research*, 27(9):1055–1067, 2008.

[14] M. Yamashita and T. Kameda. Computing on anonymous networks: Part I – characterizing the solvable cases. *IEEE Transactions on Parallel and Distributed Systems*, 7(1):69–89, 1996.

[15] A. Yershova, B. Tovar, R. Ghrist, and S. M. LaValle. Bitbots: Simple robots solving complex tasks. In *Proceedings of the 20th national conference on Artificial intelligence*, volume 3, pages 1336–1341, 2005.