applied sciences

MDPI

Article

Lower Bound for Sculpture Garden Problem: Localization of IoT Devices

Marzieh Eskandari [1,*,†], Bahram Sadeghi Bigham [2,3,*,†] and Mazyar Zahedi-Seresht [4,*,†]

1 Department of Computer Science, New Jersey Institute of Technology, Newark, NJ 07102, USA
2 Department of Computer Science, Faculty of Mathematical Sciences, Alzahra University, Tehran 1993893973, Iran
3 Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan 45137-66731, Iran
4 Department of Quantitative Studies, University Canada West, Vancouver, BC V6Z 0E5, Canada
* Correspondence: Marzieh.eskandari@njit.edu (M.E.); b_sadeghi_b@alzahra.ac.ir or b_sadeghi_b@iasbs.ac.ir (B.S.B.); mazyar.zahedi@ucanwest.ca (M.Z.-S.)
† Authors contributed equally to this work and the Authors' name are in alphabetical order.

Abstract: The purpose of the current study is to investigate a special case of art gallery problem, namely a sculpture garden problem. In this problem, for a given polygon P, the ultimate goal is to place the minimum number of guards (landmarks) to define the interior polygon P by applying a monotone Boolean formula composed of the guards. Using this problem, it can replace the operation-based method with time-consuming, pixel-based algorithms. So, the processing time of some problems in the fields of machine vision, image processing and gamification can be strongly reduced. The problem has also many applications in mobile device localization in the Internet of Things (IoT). An open problem in this regard is the proof of Eppstein's conjecture, which has remained an open problem since 2007. According to his conjecture, in the worst case, $n-2$ vertex guards are required to describe any n-gon. In this paper, a lower bound is introduced for the special case of this problem (natural vertex guard), which shows that if a polygon can be defined with natural vertex guards, then $n-2$ is a lower bound.

Keywords: art galley; Boolean formula; computational geometry; prison yard problem; sculpture garden problem; IoT

MSC: 68Q25; 68U05; 52Cxx

check for updates

Citation: Eskandari, M.; Sadeghi Bigham, B.; Zahedi-Seresht, M. Lower Bound for Sculpture Garden Problem: Localization of IoT Devices. *Appl. Sci.* **2023**, *13*, 2597. https://doi.org/10.3390/app13042597

Academic Editors: Liang-Bi Chen and Tian-Hsiang Huang

Received: 25 January 2023
Revised: 15 February 2023
Accepted: 15 February 2023
Published: 17 February 2023

1. Introduction

A large and growing body of literature about computational geometry has explored the art gallery problem. The main goal in this problem is to find the minimum number of point guards inside a polygon (P) such that the set of guards can see the whole P. Each guard can see its surroundings in 360 degrees and up to infinity (if there are no obstacles). The number of guards that suffices and is sometimes necessary for any arbitrary polygon with n vertices is $\lfloor n/3 \rfloor$ [1]. The main goal in our study is to find the minimum set of angle guards by which the geometry of the polygon can be defined through two operations, $AND(.)$ and $OR(+)$. An *angle guard* g views an infinite wedge of the plane (by going through the involved obstacles) and can be defined as a Boolean predicate, $B_g(p)$, which is *True* for a given point, $p \in P$, if p is inside the view region of g, otherwise it is False. Given a polygon P, the aim is to place a set of angle guards on P in such a way that the monotone Boolean formula $F_P(p)$ is *True*, if and only if p is inside P or on the boundary of polygon P ($p \in \partial P$), otherwise it is *False*:

$$F_P(p) = \begin{cases} True & \text{if } p \in P \text{ or } p \in \partial P \\ False & \text{otherwise} \end{cases}$$

There are several applications in machine vision, image processing and gamification which require the range of a shape to be determined. Various algorithms for these problems have been presented that are based on pixels, area, vertices and so on. These methods require a lot of processing time in problems that require fast processing (such as computer games and graphics). Even these methods force users to provide more powerful hardware to run a machine vision system or computer game. The method discussed in this paper is related to a problem that uses only a few points, angles and two logical operators to determine the range of a shape. Using this method, various applications in machine vision and computer games are performed much faster and with a lower number of operations. An angle guard vertex placement is considered as *natural* if all the guards of P have the same view of their corresponding vertices [2]. As Eppstein et al. stated, a polygon P can be demonstrated in a way that a natural angle guard vertex placement cannot fully distinguish between points on the inside and outside of P which implies that *Steiner*-point guards are sometimes necessary [2]. According to Figure 1a, even the placement of a natural angle at every vertex of the pentagon is not able to distinguish between the points x and y and at least one unnatural guard is needed to define the polygon (Figure 1b). Consequently, the polygon is defined by $F = A.B.D$.

A variety of cases of the problem is present in which the location and angle of view of the guards are different. We focus on a type of the problem that all the guards are placed on the vertices of P. It was a conjecture [3] that in the worst case, $n - 2$ guards are needed to describe an n-gon. In this paper, we present an algorithm to generate a polygon for a given n which needs at least $n - 2$ natural vertex guards. We do not prove Eppstein's conjecture and this problem is still open. However, by using our algorithm, we create an n-gon for any given n, which needs exactly $n - 2$ natural vertex guards.

In the next section, the sculpture garden problem is introduced and some applications are mentioned. Section 3 provides the main problem and presents an algorithm for generating the n-gon which needs exactly $n - 2$ natural vertex guards to be defined. Finally, Section 4 presents the findings of the study and also some suggestions for further research.

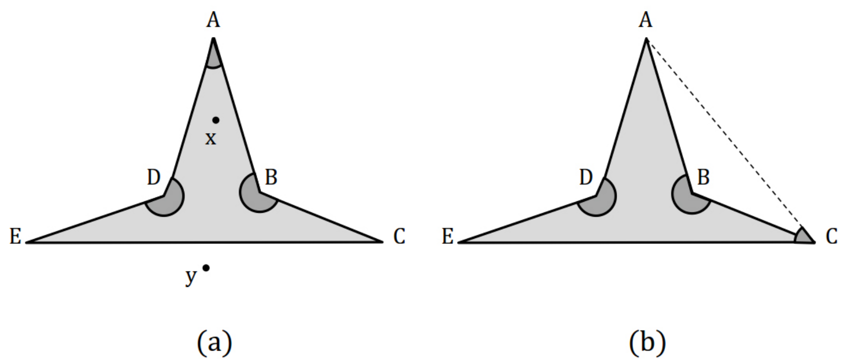


Figure 1. (**a**) Natural angle guards do not suffice to define the polygon; (**b**) coverage by three guards (formula B.C.D define the polygon) [3].

2. Sculpture Garden Problem and Applications

As mentioned previously, the sculpture garden problem can be considered as a special case of an art gallery problem. There are various problems with similarities and differences with the sculpture garden problem. As the sculpture garden problem comes up from localization problems in wireless mobile computing, we wish to determine the position of some devices in a geometric environment.

The sculpture garden problem could be used in localization problems in which a wireless device is used to prove that it belongs to a given polygonal environment. In this case, the locators would be simple and can broadcast information inside a certain angle. In this context, the Boolean predicates could be associated with secret keys. Therefore, each angle guard g could periodically broadcast a secret key K in its transmission angle and consequently the devices in range would have knowledge of this key. The wireless localization problem with natural vertex guards is an NP-hard problem [4,5].

In another application, namely constructive solid geometry (CSG), we wish to construct a geometric shape from simple combinations of primitive shapes [6]. Solutions to the sculpture garden problem can be used to construct an efficient CSG formula that defines a given polygon P. The prison yard problem seeks a set of guards that can simultaneously see both the interior and exterior of a simple polygon, in which $\lfloor n/2 \rfloor$ guards are sufficient and sometimes necessary (tight bound) [7]. Another related problem is the floodlight illumination problem, in which the vertex angle guards (called floodlights) with angles of 180 should see a simple polygon [8]. Likewise, studies have been conducted on the complexity of illuminating wedges with angle-restricted floodlights placed at a fixed set of points [9]. There is another study on a generalization of the classical problem of the illumination of polygons. Aichholzer et al. [10] modeled a wireless device whose radio signal can penetrate a given number k of walls (k-modems) and they studied the minimum number of k-modems sufficient and sometimes necessary to illuminate monotone and monotone orthogonal polygons. They showed that every monotone polygon with n vertices can be illuminated with $\lceil \frac{n-2}{2k+3} \rceil$ k-modems. Ballinger et al. [11] developed lower and upper bounds for the number of k-transmitters that are necessary and sufficient to cover a given collection of line segments, polygonal chains and polygons.

In one of the applications of the problem, the wireless localization problem is considered to deal with the placement of the smallest number of broadcasting antennas required to satisfy some property within a given polygon. The antennas propagate a unique key within a certain antenna-specific angle of broadcast, so that the set of keys received at any given point is sufficient to determine whether that point is inside or outside the polygon. To ascertain this localization property, a Boolean formula must be produced along with the placement of the antennas. Crepaldi et al. [12,13] presented an exact algorithm based on integer linear programming for solving the NP-hard natural wireless localization problem. Cano et al. [14] show that $\lceil n/2 \rceil$ point guards are always sufficient and sometimes necessary to guard a piecewise convex art gallery with n vertices. Karavelas et al. [15,16] showed that for monitoring a piecewise-convex polygon with $n \geq 2$ vertices, $\lfloor \frac{2n}{3} \rfloor$ vertex guards are always sufficient and sometimes necessary. They also presented a polynomial algorithm for computing the location of guards.

Since we are interested in more than simply observing the inside and outside of a polygon, solutions to the art gallery or prison yard problems would not change into solutions to the sculpture garden problem. In other words, we intend to establish the time when a point is inside a polygon using only the guards as witnesses.

Indeed, any polygon P can be triangulated and two angle guards can be used to define each of the resulting $n + 2(h - 1)$ triangles, where h is the number of holes in P. This would give rise to a concise formula F defining P. However, it uses at least $2n + 4(h - 1)$ angle guards, which is a more constant-depth formula.

Upper and Lower Bounds

The sculpture garden problem has different types due to the different restrictions as guards could be observed in varied forms including vertices, edges, interior or exterior of the polygon, and the SGP can be manifested in different types as well. However, in each case, finding the upper and lower bound is a problem that has already been investigated.

An angle guard g with angle $\alpha \in (0, 360)$ is a pair (a, ω_α) of a point a and an infinitive wedge ω_α of aperture α at apex a and views ω_α. It can be shown as a Boolean predicate $B_g(p)$, in which for a point p in the plane, $B_g(p)$ is *True* if p is inside the angle associated with g, otherwise it is *False*. Given a polygon P with n vertices, we intend to allocate the minimum number of angle guards with arbitrary angles at vertices of P. Thus, a monotone Boolean formula, $F_P(.)$, based on the angle guard predicates, $B_g(.)$, is obtained as follows:

$$F_P(p) = \begin{cases} True & \forall p \in int(P) \\ False & \text{otherwise} \end{cases} \quad (1)$$

It is worth mentioning that $int(P)$ is the interior of polygon P. If $F_P(p)$ is a solution of the sculpture garden problem for a given P, P is defined by F_P. The complement of an angle guard $g = (a, \omega_\alpha)$ is an angle guard g' at the point a with angle $2\pi - \alpha$. Hence, the wedge associated with g' is the *complement* of ω_α in the plane. If formula F is a solution for the sculpture garden problem for polygon P, then the complement of F which is denoted by F' defines the exterior of P. To obtain F', initially, we replace every angle guard g by its complement, (i.e., g'), and then swap the operations AND and OR. In addition, we define , a *pocket* of a simple polygon as the areas outside of the polygon and inside of its convex hull.

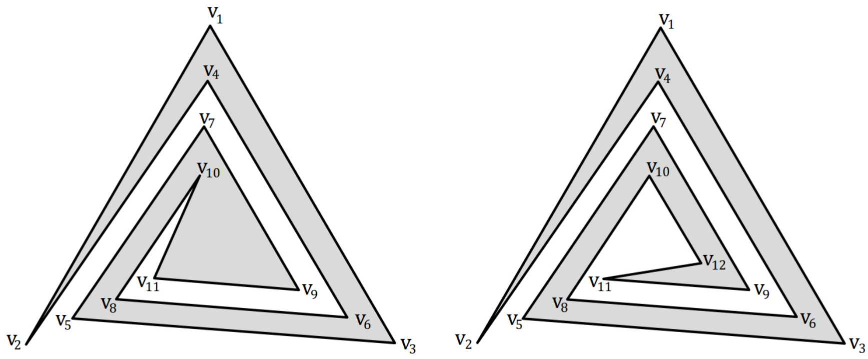
As Eppstein et al. [2,3] reported, for any polygon P, a set of $n + 2(h-1)$ angle guards and an associated concise formula F are present, solving the sculpture garden problem where h is the number of holes in P. So, a simple polygon can be defined with $n-2$ guards. They have conjectured a class of simple n-gons that require at least $n-2$ vertex guards. The main goal of this paper is to solve a special case of this open problem for natural vertex guards. They showed that at least $\lceil n/2 \rceil$ guards are required to solve the sculpture garden problem for any polygon with no two edges lying on the same line. Furthermore, for any convex polygon P, a natural angle guard vertex placement is present whose $\lceil n/2 \rceil$ guards are sufficient. They showed that $\lceil n/2 \rceil + \mathcal{O}(1)$ angle guards suffice to solve the sculpture garden problem for pseudo-triangles. Moreover, for any orthogonal polygon P (which is probably the most likely real-world application), a set of $\lfloor 3(n-2)/4 \rfloor$ angle guards and an associated concise formula F are available to solve the sculpture garden problem using $\lceil n/2 \rceil$ natural angle guards. They gave an example of a class of simple polygons containing sculpture garden solutions that used $\mathcal{O}(\sqrt{n})$ guards. Afterwards, they showed that the bound is optimal. On the contrary, some varied results are obtained for vertex guards. As Damian et al. demonstrated [17], a class of simple n-gons are presented that require at least $\lfloor 2n/3 \rfloor - 1$ guards placed at polygon vertices for localization. Through revealing the point that the maximum number of guards to describe any simple polygon on n vertices is roughly observable at $(3/5n, 4/5n)$, Hoffman et al. enhanced both upper and lower bounds for the general setting [18]. Eskandari et al. [19] improved the large upper bound $n + 2(h-1)$ for an arbitrary n-gon with h holes for placing guards and obtained a tight bound $(n - \lceil c/2 \rceil - h)$, where c is the number of vertices of convex hull of P. So, in simple polygons, this bound is $n - \lceil c/2 \rceil$, which is tight too. To complete the first column of Table 1, a new class of polygons entitled *Helix* is introduced in the next section. In some previous documents, this type of polygon was called spiral polygon.

Table 1. Number of guards needed for a simple polygon with n vertices.

	Natural Vertex	*Natural*	*Vertex*	*General*
UpperBound	*Unknown*	$n-2$ [18]	*Unknown*	$\lfloor \frac{(4n-2)}{5} \rfloor$ [18]
LowerBound	*Unknown*	$n-2$ [18]	$\lfloor \frac{(2n)}{3} \rfloor - 1$ [17]	$\lceil \frac{(3n-4)}{5} \rceil$ [18]

3. Helix Polygon

In this section, we explore the special class of the sculpture garden problem, where the guards are natural. We demonstrate the point that the lower bound for the problem is $n-2$. To do so, we commence with introducing a class of polygons demanding the exact number of $n-2$ natural guards to be defined. In the next section, we introduce this class of polygons named *Helix* (see Figure 2).



(a) (b)

Figure 2. (**a**) A Helix with 11 vertices. (**b**) A Helix with 12 vertices.

3.1. Construction of Helix

An n-gon Helix polygon (i.e., H_n) is constructed by an incremental method using an $n-1$-gon Helix, H_{n-1}. A Helix with three vertices is a triangle. By adding two new edges to H_{n-1} and also removing an edge of H_{n-1} , H_n is constructed on the basis of H_{n-1}, where $n \geq 4$. The details are presented in Algorithm 1 and are illustrated in Figure 3.

Algorithm 1 Constructing Helix Polygon

Input: Integer number $n \geq 3$ as the number of vertices.
Output: The Helix polygon H_n.

Construct $H_3 = \triangle v_1 v_2 v_3$, which is an equilateral triangle, where $v_2 v_3$ is horizontal and the vertices are in counterclockwise order.
Choose a positive real number δ so $0 < \delta < \frac{|v_2 v_3|}{2\lfloor \frac{n-1}{3} \rfloor}$
$p_1 = v_1; p_2 = v_2; p_3 = v_3$.
for $i = 4; i \leq n; i++$ **do**
  $q_1 = p_1$; $p_1 =$ a point on $v_1 v_2$ such that $|p_1 q_1| = \delta$, $a = l_{13}(p_1)$;
  $q_2 = p_2$; $p_2 =$ a point on $v_2 v_3$ such that $|p_2 q_2| = \delta$, $b = l_{12}(p_2)$;
  $q_3 = p_3$; $p_3 =$ a point on $v_1 v_3$ such that $|p_3 q_3| = \delta$, $c = l_{23}(p_3)$;
  if $i == 4$ **then**
    $l = b$;
  else
    $l = l_{24}(v_{i-2})$;
  end if
  if $i == 5$ **then**
    $l' = c$;
  else
    $l' = l_{35}(v_{i-2})$;
  end if
  if $i == 3k$ **then**
    $v_i = a \cap c$;
  end if
  if $i == 3k+1$ **then**
    $v_i = c \cap l$;
  end if
  if $i == 3k+2$ **then**
    $v_i = b \cap l'$;
  end if
  Add edges $v_i v_{i-1}$ and $v_i v_{i-2}$.
  Remove $v_{i-1} v_{i-2}$ to obtain H_i
end for
Return H_n.

It is worth noting that the length of a line segment pq is denoted by $|pq|$, and for an arbitrary point p, $l_{ij}(p)$ denotes a line parallel to $v_i v_j$ which passes through p.

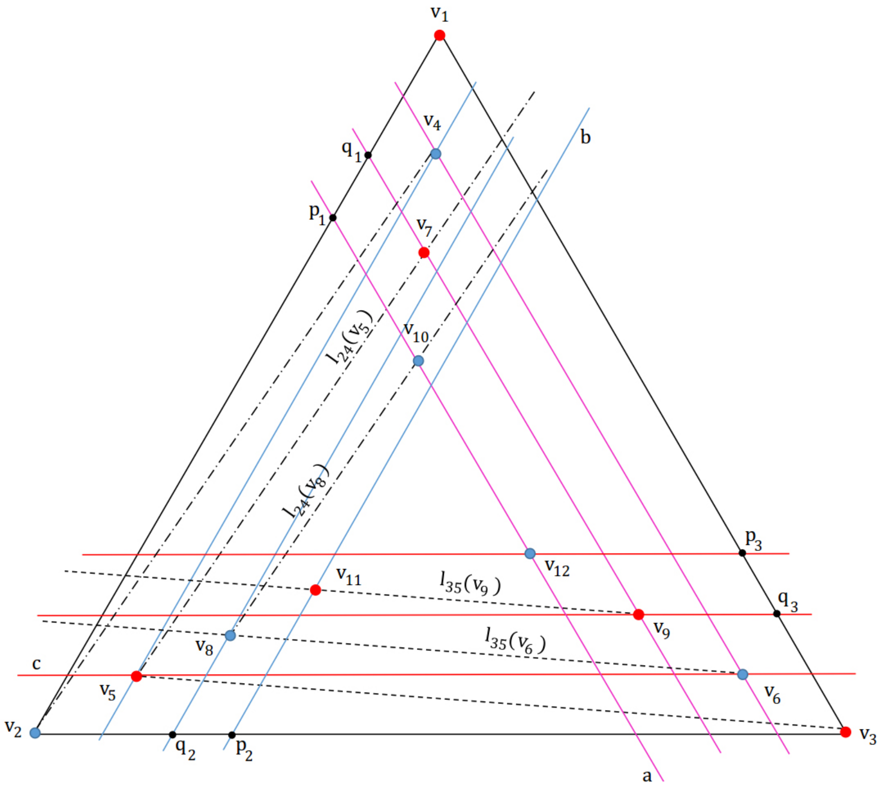


Figure 3. Final step of Algorithm 1 for generating H_{12}.

3.2. Properties of Helix

Constructing the Helix polygon sheds light on the fact that for $i > 2$, the angles made by vertices v_{2i} are concave and for $i > 0$, vertices v_{2i+1} and v_2 are convex (Figure 2).

In fact, the pocket of a polygon P is defined as $CH(P) - P$ where $CH(P)$ is the convex hull of the vertices of P. The pocket of a Helix polygon with n vertices is a Helix polygon with $n - 1$ vertices (Figure 4). The pocket of a polygon H_n is denoted by $P(H_n)$. For i, $1 \leq i \leq n - 1$, the vertices of $P(H_n)$ are called $v_i^{'}$, located on v_{i+1}. For $n > 4$, the angle $\widehat{v_i^{'}}$ in $P(H_n)$ is obtained as follows:

$$\widehat{v_i^{'}} = \begin{cases} \widehat{v_1 v_2 v_3} - \widehat{v_1 v_2 v_4} & i = 1 \\ \widehat{v_1 v_3 v_2} - \widehat{v_1 v_3 v_5} & i = 2 \\ 2\pi - \text{interior angle of } v_{i+1} \text{in } H_n & i \geq 3. \end{cases} \quad (2)$$

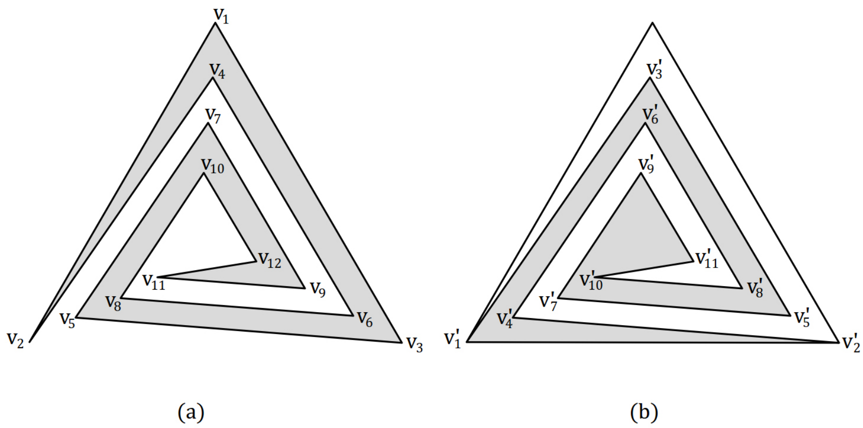


Figure 4. (**a**) Helix H_{12}. (**b**) Pocket of H_{12} which is a Helix with 11 vertices.

We will show that polygon H_n for any given natural number $n[3)$, can be defined by $n-2$ natural vertices guards which are located on $v_1, v_2, \ldots, v_{n-3}, v_n$. The Boolean formula, F_n, is as below:

$$F_n = \sum_{i=1}^{\lceil n/2 \rceil -2} A_i.g_{2i} + A_{\lfloor n/2 \rfloor -1}.g_n \tag{3}$$

where $A_1 = g_1$, $A_{i+1} = A_i.g_{2i+1}$ for all $1 \leq i \leq \lfloor n/2 \rfloor - 2$ and g_i is the natural vertex guard located on the vertex v_i of H_n. To clarify this point, F_8 can be written as follows:

$$F_8 = \sum_{i=1}^{2} A_i.g_{2i} + A_5.g_8 \tag{4}$$

where $A_1 = g_1, A_2 = g_1.g_3$ and $A_3 = g_1.g_3.g_5$. Thus, we have

$$F_8 = g_1.g_2 + g_1.g_3.g_4 + g_1.g_3.g_5.g_8$$

Generally, we expand F_n as follows:

$$F_n = \begin{cases} \sum_{i=1}^{k-1} (\prod_{j=0}^{i-1} g_{2j+1}).g_{2i} + (\prod_{j=0}^{k-2} g_{2j+1}).g_n & n = 2k+1, k \in \mathbb{N} \\ \sum_{i=1}^{k-1} (\prod_{j=0}^{i-1} g_{2j+1}).g_{2i} + (\prod_{j=0}^{k-1} g_{2j+1}).g_n & n = 2k+2, k \in \mathbb{N}. \end{cases} \tag{5}$$

According to Lemma 1, F_n defined by Equation (5) describes H_n.

Lemma 1. *Helix polygon H_n can be defined by $n-2$ natural vertex guards g_i ($1 \leq i \leq n; i \neq n-1, n-2$) located on $v_1, v_2, \ldots v_{n-3}, v_n$ and the correspondent Boolean formula is Formula 5.*

Proof. We will prove the lemma by induction. When $n = 3, k = 1$ and $F_3 = g_1.g_3$ clearly define triangle H_3. For $n = 4, k = 1$ and $F_4 = g_1.g_4$ define H_4 and it implies that Lemma 1 holds for $n = 3, 4$. Now, for $n \geq 5$, without loss of generality, assume that $n = 2k+2, k \in \mathbb{N}$. According to Property 2, $P(H_n)$ is a Helix polygon with $2k+1$ vertices. By induction hypothesis, $P(H_n)$ can be defined as follows:

$$F' = \sum_{i=1}^{k-1} (\prod_{j=0}^{i-1} g'_{2j+1}).g'_{2i} + (\prod_{j=0}^{k-2} g'_{2j+1}).g'_{2k+1} \tag{6}$$

where g'_i is a natural guard on the vertex v'_i of $P(H_n)$. According to correspondence between the vertices of H_n and $P(H_n)$, we have

$$g'_i = \begin{cases} g^c_{i+1} & i \geq 3 \\ G_{i+1}.g^c_{i+1} & i = 1, 2. \end{cases} \tag{7}$$

in which g^c is the complement of guard g and G_2 and G_3 are the guards located on v_2 and v_3 with the angles $\widehat{v_1 v_2 v_3}$ and $\widehat{v_1 v_3 v_2}$, respectively. So, we have

$$F' = g'_1.g'_2 + \sum_{i=2}^{k-1} (\prod_{j=0}^{i-1} g'_{2j+1}).g'_{2i} + (\prod_{j=0}^{k-2} g'_{2j+1}).g'_{2k+1}$$

$$= g'_1.g'_2 + \sum_{i=2}^{k-1} g'_1.(\prod_{j=1}^{i-1} g'_{2j+1}).g'_{2i} + g'_1.(\prod_{j=1}^{k-2} g'_{2j+1}).g'_{2k+1}$$

$$= G_2.G_3.g^c_2.g^c_3 + \sum_{i=2}^{k-1} G_2.g^c_2.(\prod_{j=1}^{i-1} g^c_{2j+2}).g^c_{2i+1} + G_2.g^c_2.(\prod_{j=1}^{k-2} g^c_{2j+2}).g^c_{2k+2}$$

$$= G_2.[G_3.g_2^c.g_3^c + \sum_{i=2}^{k-1}(\prod_{j=0}^{i-1} g_{2j+2}^c).g_{2i+1}^c + (\prod_{j=0}^{k-2} g_{2j+2}^c).g_{2k+2}^c]$$

$$\implies (F')^c = G_2^c + (G_3^c + g_2 + g_3).(\prod_{i=2}^{k-1}(\sum_{j=0}^{i-1} g_{2j+2}) + g_{2i+1}).(\sum_{j=0}^{k-2} g_{2j+2} + g_{2k+2})$$

Consider the point that $F = (g_1.g_2).(F')^c + (g_1.g_3).(F')^c$. By replacing $(F')^c$ from the above relation, we obtain

$$F = g_1.(g_2 + g_2.g_3).(\prod_{i=2}^{k-1}(\sum_{j=0}^{i-1} g_{2j+2}) + g_{2i+1}).(\sum_{j=0}^{k-2} g_{2j+2} + g_{2k+2})$$

$$+ g_1.(g_2.g_3 + g_3).(\prod_{i=2}^{k-1}(\sum_{j=0}^{i-1} g_{2j+2}) + g_{2i+1}).(\sum_{j=0}^{k-2} g_{2j+2} + g_{2k+2})$$

Note that $g_1.g_2.G_2^c = g_1.g_3.G_2^c = g_1.g_2.G_3^c = g_1.g_3.G_3^c = \varnothing$. So, we have

$$F = (\prod_{i=2}^{k-1}(\sum_{j=0}^{i-1} g_{2j+2}) + g_{2i+1}).(\sum_{j=0}^{k-2}(g_{2j+2} + g_{2k+2})).(g_2 + g_3).g_1$$

Now, we show that F can define H_n which contains exactly the natural guards $g_1, g_2, \ldots, g_{n-3}, g_n$ and can be written in the form of F_n.

First, consider the definition of F which contains only natural guards. To prove that H_n can be defined by F, let x be an arbitrary point inside H_n. We have $g_1(x) = True$ and $(F')^c(x) = True$ ($x \in H_n \implies x \notin P(H_n) \implies F'(x) = False \implies (F')^c(x) = True$). There are two cases:

- $g_2(x) = True \implies F(x) = (g_1(x).g_2(x)).(F')^c(x) + (g_1(x).g_3(x)).(F')^c(x) = True$
- $g_2(x) = False \implies g_3(x) = True \implies (g_1(x).g_3(x)).(F')^c(x) = True \implies F(x) = True$.

Thus, F can distinguish the interior of H_n. Now, let $x \notin H_n$. There are two cases:

- $x \in P(H_n) \implies F'(x) = True, (F')^c(x) = False \implies F(x) = False$
- $x \notin P(H_n) \implies x \in Ext(\triangle v_1 v_2 v_3) \implies g_1(x) = False \implies F(x) = False$.

So, F can distinguish the exterior of H_n as well. Now, we show that F can be written in the form of F_n. Let

$$S = (g_2 + g_3).(\prod_{i=2}^{k-1}(\sum_{j=0}^{i-1} g_{2j+2}) + g_{2i+1}).(\sum_{j=0}^{k-2} g_{2j+2} + g_{2k+2})$$

and

$$T = \sum_{i=1}^{k-1}(\prod_{j=1}^{i-1}(g_{2j+1}).g_{2i}) + (\prod_{j=1}^{k-1}(g_{2j+1}).g_{2k+2})$$

Note that $F_n = g_1.T$ and $F = g_1.S$. To prove $F = F_n$, it is sufficient to show that $T = S$. For all integers r where $1 \le r \le k-1$, we define $S_i^{(r)}$ as follows:

$$S_i^{(r)} = \begin{cases} g_{2i+1} + \sum_{j=r}^{i-1} g_{2j+2} & r \le i \le k-1 \\ g_{2k+2} + \sum_{j=r}^{k-2} g_{2j+2} & i = k. \end{cases} \quad (8)$$

So, we have

$$S_i^{(1)} = \begin{cases} g_{2i+1} + \sum_{j=1}^{i-1} g_{2j+2} & 1 \le i \le k-1 \\ g_{2k+2} + \sum_{j=1}^{k-2} g_{2j+2} & i = k. \end{cases} \quad (9)$$

By definition of $S_i^{(r)}$, it is clear that

$$S = \prod_{i=1}^{k}(g_2 + S_i^{(1)}) = g_2 + \prod_{i=1}^{k} S_i^{(1)}$$

On the other hand, $S_i^{(r)} - S_i^{(r+1)} = g_{2r+2}$. Therefore we have

$$\prod_{i=r}^{k} S_i^{(r)} = S_r^{(r)}. \prod_{i=r+1}^{k} S_i^{(r)}$$

$$= g_{2r+1}. \prod_{i=r+1}^{k} S_i^{(r)} = g_{2r+1}. \prod_{i=r+1}^{k} (g_{2r+2} + S_i^{(r+1)}) = g_{2r+1}.(g_{2r+2} + \prod_{i=r+1}^{k} S_i^{(r+1)})$$

By applying obtained recursive relation, $k-2$ times on $S = g_2 + \prod_{i=1}^{k} S_i^{(1)}$, $S = T$. In this respect, we have

$$S = g_2 + \prod_{i=1}^{k} S_i^{(1)} = g_2 + g_3.(g_4 + \prod_{i=2}^{k} S_i^{(2)}) = g_2 + g_3.g_4 + g_3. \prod_{i=2}^{k} S_i^{(2)}$$

$$S = g_2 + g_3.g_4 + g_3.g_5.g_6 + g_3.g_5. \prod_{i=3}^{k} S_i^{(3)}$$

After t times, we have

$$S = (\sum_{i=1}^{t+1}(\prod_{j=1}^{i-1} g_{2j+1}).g_{2i}) + \prod_{j=1}^{t} g_{2j+1}. \prod_{i=t+1}^{k} S_i^{(t+1)}$$

So after $k-2$ times, we have

$$S = (\sum_{i=1}^{k-1}(\prod_{j=1}^{i-1} g_{2j+1}).g_{2i}) + \prod_{j=1}^{k-2} g_{2j+1}. \prod_{i=k-1}^{k} S_i^{(k-1)}$$

Note that $\prod_{i=k-1}^{k} S_i^{(k-1)} = S_{k-1}^{(k-1)}.S_k^{(k-1)} = g_{2k-1}.g_{2k+2}$; therefore,

$$S = (\sum_{i=1}^{k-1}(\prod_{j=1}^{i-1} g_{2j+1}).g_{2i}) + \prod_{j=1}^{k-1} g_{2j+1}.g_{2k+2} = T.$$

$S = T$ implies $F = F_n$ which means that F can be written in the form of F_n and could define H_n. □

3.3. Necessity of $n-2$ Natural Vertex Guards for Helix

In this section, we will prove that it is impossible to define a Helix polygon with fewer than $n-2$ natural vertex guards.

Lemma 2. *Every arbitrary set of natural vertex guards G which defines H_n contains g_1, a natural vertex guard on v_1. The final formula is in the form of $F = F_1.g_1$, where F_1 is a Boolean expression of $G - \{g_1\}$.*

Proof. Let G be an arbitrary set of natural guards which defines H_n by Boolean formula F. Suppose for a contradiction that g_1 does not belong to G. Since v_1v_2 and v_1v_3 are edges of H_n, G should contain two natural guards on v_2 and v_3 which are called g_2 and g_3, respectively. So, F can be written in the general form $F = g_2.g_3.T_1 + g_2.T_2 + g_3.T_3$ where T_is are Boolean formulas which do not contain g_1, g_2 and g_3. This will result in a contradiction, mentioned below.

Consider two regions, R_1 and R_2, as shown in Figure 5. Let x be an arbitrary point inside R_1 or R_2. So, we have $g_2(x) = True$ and $g_3(x) = False$. So,

$$F(x) = T_2(x) \tag{10}$$

Furthermore, note that we can expand T_2 in the following general form:

$$T_2 = T_2^{(1)} + T_2^{(2)} + \ldots + T_2^{(l)} \tag{11}$$

where $T_2^{(i)}$s are the multiplication of some natural guards in G. Let $x \in R_1$ be an arbitrary point. So, from Equation (10), it is implied that

$$x \in R_1 \implies T_2(x) = F(x) = True$$

Therefore, at least one of the expressions of T_2 should be True. Without loss of generality, it can be called $T_2^{(1)}$ and is written as $T_2^{(1)} = g_{i_1}.g_{i_2}.\cdots.g_{i_m}$, where g_{i_j} is the natural guards of G and $j = 1, 2, 3$. Since $T_2^{(1)}(x) = True$, we have

$$\forall j : 1 \le j \le m : g_{i_j}(x) = True$$

Regarding the structure of H_n, none of indices i_j can be odd. This is because we know that

$$\forall i \ge 1 : g_{2i+1}(x) = False$$

Now, let $y \in R_2$ be an arbitrary point. Due to the structure of H_n, it is inferred that for all $\forall y \in R_2$, we have

$$\forall i \ge 2 : g_{2i}(y) = True.$$

So, $T_2^{(1)}(x) = True$ and from Equation (11), $T_2(y) = True$ is obtained and consequently $F(y) = T_2(y) = True$ (due to Equation (10). Nevertheless, $y \notin H_n$, which is a contradiction. With regard to the existence of g_1 in G, F can be written in the form of $F = g_1.T_1 + T_2$, where T_2 does not contain g_1. Indeed, $g_1.T_1 + g_1.T_2$ defines H_n as well. Let $x \in H_n$, then $g_1(x) = True$ and $g_1(x).T_1(x) + g_1(x).T_2(x) = g_1(x).T_1(x) + T_2(x) = F(x) = True$. If $y \notin H_n$ and $g_1(y) = True$, then $g_1(y).T_1(y) + g_1(y).T_2(y) = g_1(y).T_1(y) + T_2(y) = F(y) = False$. On the other hand, if $g_1(y) = False$, then $g_1(y).T_1(y) + g_1(y).T_2(y) = False$. So, $F = g_1.(T_1 + T_2)$ defines H_n. In other words, F can be expressed as $F = g_1.F_1$. □

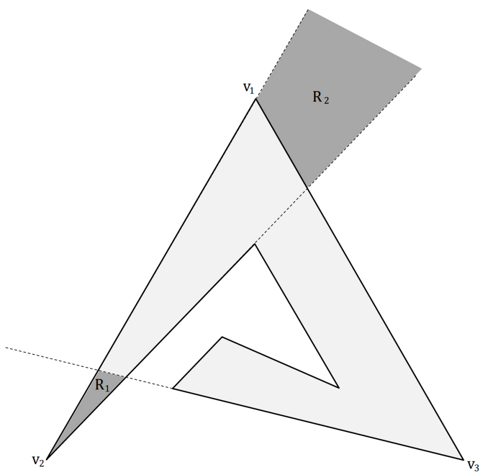


Figure 5. Regions R_1 and R_2 can be distinguishable without the existence of g_1 in the formula.

Lemma 3. *Every arbitrary set of natural vertex guards G defining H_n contains g_2 (i.e., a natural guard on v_2). The final formula is in the form of $F = g_1.(g_2 + F_2)$, where F_2 is a Boolean expression of $G - \{g_1, g_2\}$.*

Proof. Let G be an arbitrary set of natural guards which defines H_n by Boolean formula F, for $n \geq 4$. Suppose a contradiction in which g_2 does not belong to G. From Lemma 2, we can write

$$F = g_1.(T_1 + T_2 + \ldots + T_l)$$

where T_is are Boolean expression of natural guards of G.

Consider two regions, R_1 and R_3, as shown in Figure 6. Let $x \in R_1$ be an arbitrary point. We have $F(x) = g_1(x).(T_1(x) + T_2(x) + \ldots + T_l(x)) = True$. Thus, at least one of the expressions T_is is True. Without loss of generality, it can be named T_1 and is written as $T_1 = g_{i_1}.g_{i_2} \ldots .g_{i_m}$ where g_{i_j}s are some natural guards in G and $i_j \neq 1, 2$. Since $T_1(x) = True$, we have

$$\forall j, 1 \leq j \leq m : g_{i_j}(x) = True$$

Regarding the structure of H_n, i_js are even, because $\forall i \geq 2 : g_{2i}(x) = True$.

Now, let $y \in R_3$ be an arbitrary point. Obviously, $\forall i \geq 2$, $g_{2i}(y) = True$ which implies $T_1(y) = True$. Then, $F(y) = True$. However, $y \notin H_n$, which denotes a contradiction.

In addition, F can be manifested as below:

$$F = g_1.(g_2.T_1 + T_2)$$

Now note that for all points which are located in the interior (or exterior) of H_n, the above formula has the same value with the formula $g_1.(g_2 + T_2)$. This fact can be shown easily by considering all cases. Then, $F = g_1.(g_2 + F_1)$. □

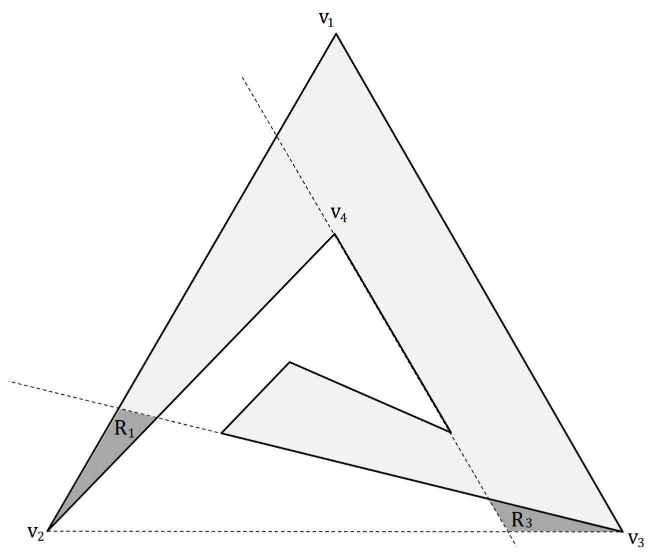


Figure 6. Regions R_1 and R_3 can be distinguishable without the existence of g_2 in the formula.

Lemma 4. *It is not possible to define H_5 with less than 3 natural vertex guards. The formula is $F = g_1.(g_2 + g_5)$.*

Proof. Regarding Lemmas 2 and 3, F can be written as $g_1.(g_2 + F_2)$. The edges v_4v_5 and v_3v_5 should have at least one guard on their endpoints. The optimal possibility is to locate a guard on v_5 as their intersection point. Clearly, $g_1.(g_2.g_5)$ defines H_5. □

Lemma 5. *Every arbitrary set of natural vertex guards G which defines H_n contains g_3 which is a natural guard on v_3. The final formula is in the form of $F = g_1.(g_2 + g_3.F_3)$ where F_3 is a Boolean expression of $G - \{g_1, g_2, g_3\}$.*

Proof. Let G be an arbitrary set of natural guards which defines H_n by Boolean formula F for $n \geq 6$. Suppose a contradiction in which g_3 does not belong to G. By Lemma 3, $F = g_1.(g_2 + F_2)$. Assume that $F_2 = T_1 + T_2 + \ldots + T_l$, where T_is are multiplication of natural guards in G.

Consider two regions, R_3 and R_4, as shown in Figure 7. Let $x \in R_4$ be an arbitrary point. We have $F(x) = F_2(x) = True$. So, at least one of T_is is True. Without loss of generality, we call it T_1, which can be expressed as follows:

$$T_1 = g_{i_1}.g_{i_2}\ldots.g_{i_m}$$

where g_{i_j}s are natural guards in G and $i_j \neq 1, 2, 3$.

Since $T_1(x) = True$, we have

$$\forall j, 1 \leq j \leq m : g_{i_j}(x) = True$$

Therefore, i_js are even. Now, let $y \in R_3$ be an arbitrary point. This point casts light that for all $i \geq 2$, $g_{2i}(y) = True$, which implies $T_1(y) = True$. Then, $F(y) = True$. However, $y \notin H_n$, showing a contradiction. So, $g_3 \in G$.

In addition, F can be written as below:

$$F + g_1.(g_2 + g_3.F_3)$$

It can be easily obtained from equivalency of $g_1.(g_2 + F_2)$ and $g_1.(g_2 + g_3.F_3)$ for all points with respect to H_n. □

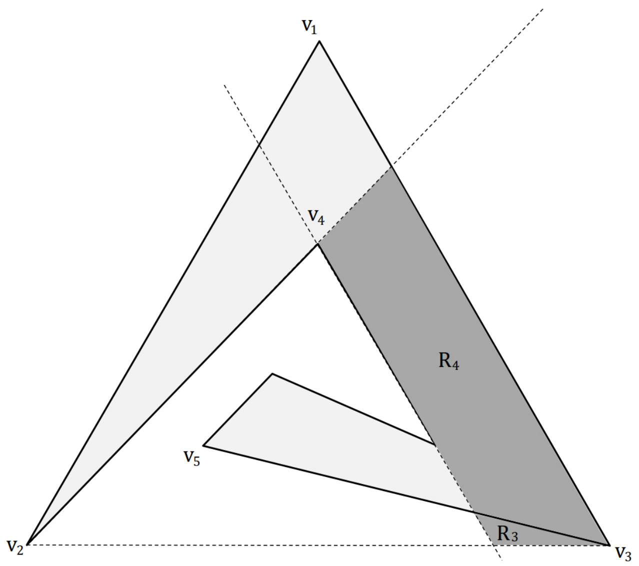


Figure 7. Regions R_3 and R_4 can be distinguishable without the existence of g_3 in the formula.

Lemma 6. *It is not possible to define H_5 with fewer than three natural vertex guards. The formula is $F = g_1.(g_2 + g_3.g_6)$.*

Proof. Considering Lemma 5, H_6 can be defined by $F = g_1.(g_2 + g_3.F_3)$. Since v_4v_6 and v_6v_5 are two edges of H_6, it is required to place at least one guard on one of the endpoints of these two edges. The optimal placement is to place a guard on v_6. Obviously, H_6 can be defined by $F = g_1.(g_2 + g_3.g_6)$. □

Lemma 7. *Every arbitrary set of natural vertex guards G which defines H_n, for all $n \geq 7$, contains g_4, a natural guard on v_4. The final formula is in the form of $F = g_1.(g_2 + g_3.(g_4 + F_4))$, where F_4 is a Boolean expression of $G - \{g_1, g_2, g_3, g_4\}$.*

Proof. Let G be an arbitrary set of natural guards defining H_n by Boolean formula F and $n \geq 7$. Suppose a contradiction in which g_4 does not belong to G. By Lemma 5, $F = g_1.(g_2 + g_3.F_3)$. It is intended to show that $F_3 = g_4 + F_4$. Assume that $F_3 = T_1 + T_2 + \ldots + T_l$, where T_is are the multiplication of natural guards of G. Consider two regions, R_5 and R_6, shown in Figure 8. Let $x \in R_5$ be an arbitrary point, then $F(x) = F_3(x) = True$. So, at least one of T_is is True. Without loss of generality, t is called T_1 which can be expressed as below:

$$T_1 = g_{i_1}.g_{i_2} \ldots .g_{i_m}$$

where g_{i_j}s are natural guards and $i_j \neq 1,2,3,4$. Since $T_1(x) = True$, for all $j : 1 \leq j \leq m$, $g_{i_j}(x) = True$. Therefore, i_js are even. Now, let $y \in R_6$ be an arbitrary point (see Figure 8). It is clear that for all $i \geq 2$, $g_{2i}(y) = True$, which implies $T_1(y) = True$. Then, $F(y) = True$. However, $y \notin H_n$, indicating a contradiction. So, $g_4 \in G$. In addition, F can be written as below:

$$F = g_1.(g_2 + g_3.(g_4 + F_4))$$

It can be easily shown that $g_1.(g_2 + g_3.F_3)$ is equivalent with $g_1.(g_2 + g_3.(g_4 + F_4))$ for all the points inside or outside of H_n. □

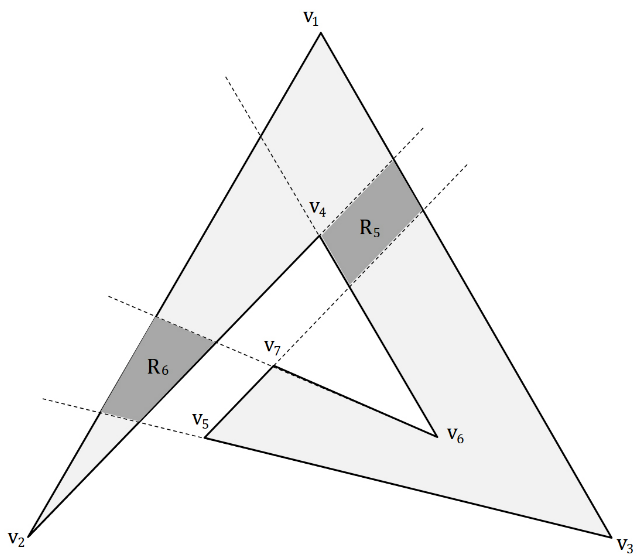


Figure 8. Regions R_5 and R_6 can be distinguishable without the existence of g_4 in the formula.

Lemma 8. *Let H_n be defined by $F = g_1.(g_2 + g_3.(g_4 + F_4))$. Formula $F_P = g_1'.(g_2' + g_3'.F_4^c)$ defines the pocket of H_n, $P(H_n)$, where g_i's are defined in Equation (7).*

Proof. Let $x \in P(H_n)$ and $y \notin P(H_n)$ be two arbitrary points (see Figure 9). Then, it can be demonstrated that $F_P(x) = True$ and $F_P(y) = False$. Additionally, $g_2' = G_3.g_3^c$ and $g_3' = g_4^c$, so $x \in P(H_n) \implies g_1'(x) = True$ and $G_3(x) = True$. So

$$F_P(x) = g_3^c(x) + g_4^c(x).F_4^c(x) \tag{12}$$

On the other hand, $x \in P(H_n)$ implies that $F(x) = False$ and $g_1(x) = True$. Hence,

$$F(x) = g_2(x) + g_3(x).(g_4(x) + F_4(x)) = False \implies g_3(x).(g_4(x) + F_4(x)) = False.$$

So,

$$g_3^c(x) + g_4^c(x).F_4^c(x) = True \quad (13)$$

From Equations (12) and (13), $F_P(x) = True$ can be obtained.

If $g^{'}(y) = False$, for $y \notin P(H_n)$, $F_P(y) = False$ and the process of our calculations has been completed. Assume that $g^{'}(y) = True$, then as $y \notin P(H_n)$, consequently $g_2^{'}(y) = False$ (see Figure 9). If $g_3^{'}(y) = False$, $F_P(y) = False$. Now, suppose that $g_3^{'}(y) = True$ (i.e., $g_4(y) = False$). This assumption implies that $y \in H_n$ and we have $g_1(y) = True$, $g_2(y) = False$, $g_31(y) = True$ and $g_4(y) = False$. Since $y \in H_n$, $F_(y) = F_4(y)$ and consequently, $F_4(y) = True$. So, we have $F_P(y) = False$. This means that F_P defines $P(H_n)$. □

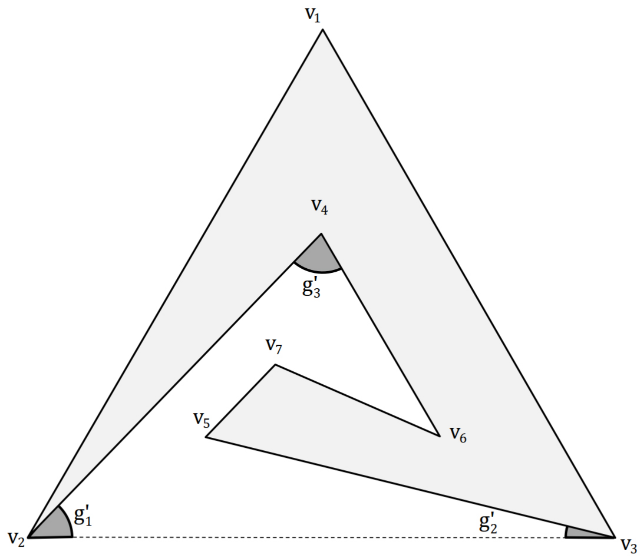


Figure 9. Natural vertex guards for packet of Helix.

Theorem 1. *H_n requires at least $n-2$ natural vertex guards.*

Proof. We prove this theorem by induction. It is clear that for $n = 4$, H_4 is a tetragon and cannot be defined by fewer than two guards. We proved this in corollaries 4 and 6 for $n - 5$. Now, assume that this holds for $n-1$, and we have to prove it for n where $n \geq 7$. Let F be a Boolean formula which defines H_n. With regard to Lemma 7, $F = g_1.(g_2 + (g_3.(g_4 + F_4))$. Let m be the number of natural guards used in F. From Lemma 8, $P(H_n)$, a Helix with $n-1$ vertices, can be defined by $F_P = g_1^{'}.(g_2^{'} + g_3^{'}.F_4^c)$ which contains $m-1$ guards. By induction hypothesis, $P(H_n)$ cannot be defined by fewer than $(n-1)-2$ natural guards. So $m-1$ cannot be less than $n-3$ and hence m cannot be fewer than $n-2$. □

Theorem 2. *H_n requires exactly $n-2$ natural vertex guards.*

Proof. From Lemma 1 and Theorem 9, it is obviously implied that H_n requires exactly $n-2$ natural vertex guards. □

As we proved, there is an n-gon which needs exactly $n-2$ natural vertex guards to be defined. This implies that $n-2$ is the lower bound.

4. Conclusions

Eppstein et al. [3] in 2007 conjectured that for a given number n, at least one simple polygon is present that requires $n-2$ vertex guards to describe the polygon. This is the worst case of the sculptural garden problem which can be applied to speed up the localization approaches of mobile devices in some applications. We proved a special case of the conjecture in which the guards are natural vertex ones, by introducing a new class of polygons named Helix polygon. In further research, one can prove the general case of the conjecture. Additionally, one can investigate the bounds for special cases of polygons (e.g., orthogonal polygons).

As far as the authors know, no algorithm (deterministic or non-deterministic) has been presented to solve the sculpture garden problem. It can have many applications in security, smart cities and the Internet of Things, and it is suggested that researchers try to develop approximate or heuristic algorithms to solve the problem. To be more precise, it is expected that in the future research, an algorithm will be presented to solve the problem, in which a polygon can be described and its range defined with the least number of guards.

Author Contributions: Conceptualization and supervision, M.E.; methodology and validation, M.E. and B.S.B.; validation and writing—original draft preparation, B.S.B. and M.Z.-S.; resources, M.Z.-S.; writing—review and editing, B.S.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by all the authors equally.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. De Berg, M.; Van Kreveld, M.; Overmars, M.; Schwarzkopf, O.C. *Computational Geometry*; Springer: Berlin/Heidelberg, Germany, 2000.
2. Eppstein, D.; Goodrich, M.T.; Sitchinava, N. Guard placement for efficient point-in-polygon proofs. In *Proceedings of the Twenty-Third Annual Symposium on Computational Geometry*; ACM: Gyeongju, Republic of Korea, 2007; pp. 27–36.
3. Eppstein, D.; Goodrich, M.T.; Sitchinava, N. Guard placement for wireless localization. *arXiv* **2006**, arXiv:cs/0603057.
4. Christ, T.; Hoffmann, M.; Okamoto, Y. Natural wireless localization is NP-hard. In *Abstracts 25th European Workshop Computational Geometry*; Citeseer: Brussels, Belgium, 2009; pp. 175–178.
5. Christ, T.; Hoffmann, M. Wireless Localization with Vertex Guards is NP-hard. In *CCCG*; Citeseer: Vancouver, BC, Canada, 2009; pp. 149–152.
6. Dobkin, D.; Guibas, L.; Hershberger, J.; Snoeyink, J. *An Efficient Algorithm for Finding the CSG Representation of a Simple Polygon*; ACM: Atlanta, GA, USA, 1988; Volume 22, Number 4.
7. Füredi, Z.; Kleitman, D.J. The prison yard problem. *Combinatorica* **1994**, *14*, 287–300. [CrossRef]
8. Estivill-Castro, V.; O'Rourke, J.; Urrutia, J.; Xu, D. Illumination of polygons with vertex lights. *Inf. Process. Lett.* **1995**, *56*, 9–13. [CrossRef]
9. Steiger, W.; Streinu, I. Illumination by floodlights. *Comput. Geom.* **1998**, *10*, 57–70. [CrossRef]
10. Aichholzer, O.; Fabila-Monroy, R.; Flores-Penaloza, D.; Hackl, T.; Urrutia, J.; Vogtenhuber, B. Modem illumination of monotone polygons. *Comput. Geom.* **2018**, *68*, 101–118. [CrossRef]
11. Ballinger, B.; Benbernou, N.; Bose, P.; Damian, M.; Demaine, E.D.; Dujmović, V.; Flatland, R.; Hurtado, F.; Iacono, J.; Lubiw, A. Coverage with k-transmitters in the presence of obstacles. *J. Comb. Optim.* **2013**, *25*, 208–233. [CrossRef]
12. Crepaldi, B.E.; de Rezende, P.J.; de Souza, C.C. An Efficient Exact Algorithm for the Natural Wireless Localization Problem. In *CCCG*; Carleton University: Waterloo, ON, Canada, 2013.
13. Crepaldi, B.E.; de Rezende, P.J.; de Souza, C.C. Solving the natural wireless localization problem to optimality efficiently. *Comput. Geom.* **2015**, *48*, 370–379. [CrossRef]
14. Cano, J.; Tóth, C.D.; Urrutia, J. A tight bound for point guards in piecewise convex art galleries. *Comput. Geom.* **2013**, *46*, 945–958. [CrossRef]
15. Karavelas, M.I.; Tóth, C.D.; Tsigaridas, E.P. Guarding curvilinear art galleries with vertex or point guards. *Comput. Geom.* **2009**, *42*, 522–535. [CrossRef]
16. Karavelas, M.I. Guarding curvilinear art galleries with edge or mobile guards via 2-dominance of triangulation graphs. *Comput. Geom.* **2011**, *44*, 20–51. [CrossRef]
17. Damian, M.; Flatland, R.; O'Rourke, J.; Ramaswami, S. A new lower bound on guard placement for wireless localization. *arXiv* **2007**, arXiv:0709.3554.

18. Christ, T.; Hoffmann, M.; Okamoto, Y.; Uno, T. Improved bounds for wireless localization. In *Algorithm Theory-SWAT*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 77–89.
19. Eskandari, M.; Mohades, A.; Bigham, B.S.B. On the Number of Guards in sculpture garden problem. *World Appl. Sci. J.* **2010**, *10*, 1255–1263.