Optimally Sweeping Convex Polygons with Two and Three Guards

by

Zohreh Jabbari

B.Sc., Sharif University of Technology, 2007

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

The Faculty of Graduate Studies

(Computer Science)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

April 2010

Abstract

Given a polygon P, we considered the problem of finding the shortest total paths for two and three mobile guards to cover P. In our definition of the problem, we do not limit the movements of the guards. The guards are allowed to start their paths at any point on the polygon, cross the interior or intersect each other's paths if necessary.

A polygon P is covered by two guards if every point in P is on the line that connects the guards at some point in time. We proved that if the polygon is convex, the optimal sweep limits the paths of the guards to the perimeter of the polygon. In the optimal solution the guards may start their paths at the end-points of the longest edge of P and end their paths at the end-points of the second longest edge in P, visiting the $n-2$ other edges along the way. Finding such a path takes $O(n)$ time.

With three, the guarding problem is defined differently. In the three guard problem a point is covered if it is on the triangle formed by the guards at some point in the sweep. We found that even in convex polygons in some cases crossing the polygon makes the sweep shorter. However, we showed that the optimal sweep is always simple (i.e., the guards paths do not cross one another). By recognizing the conditions where interior paths are beneficial we were able to present an algorithm to find the optimal 3-guard path in $O(n^5)$ time.

Table of Contents

List of Figures

Acknowledgments

This work would not have been possible without the continuous help and support from my supervisors, Dr. David Kirkpatrick and Dr. William Evans. I wish to express my most sincere gratitude to them for their encouragement, patience and guidance that have guided me through the past two years. I am honored to have had the opportunity of working with them and learning from them.

I would like to thank Dr. Patrice Belleville for reviewing my thesis and for his valuable comments. I am also thankful to my colleagues at Beta lab for sharing their knowledge and experiences with me.

would also like to thank my family for their support through my entire life, my mom and dad for believing in me, my younger sister, Farzaneh, for our joyful conversations and time together, my little niece, Viyana, whose laughter makes my day and last but not least my older sister and brother-in-law, Hosna and Majid, without whose advice, encouragement and editing assistance, I would not have finished this thesis.

To my parents.

Chapter 1

Introduction

The original Art Gallery problem was proposed by Victor Klee in 1973. The problem was defined as: "How many guards are necessary and sufficient to guard artistic objects in a given art gallery with n walls?". He posed this question in answer to Vasek Chvátal's request at a conference at Stanford, for an interesting geometric problem. Assuming that the gallery is a polygon, the problem can also be stated as a visibility problem of covering this polygon with minimum star-shaped polygons.

In 1975, Chvátal proved that, $\lfloor n/3 \rfloor$ guards are occasionally necessary and always sufficient to cover a polygon with n vertices. The proof was later simplified by Fisk, via a 3-coloring argument[16]. The problem of determining the minimum number of guards that suffice to cover a given polygon was shown to be NP-hard by Lee and Lin [9] by reducing it from $3SAT$.

Many other variations of the Art Gallery Problems and visibility problems have been since analyzed and published. Many of them use the idea of mobile guarding. In this section we will review some of these problems briefly. More information on them and other related problems can be found in [16], [17], [21] and [22].

1.1 The Watchman Route

Chin and Ntafos[3], defined the Watchman Route Problem (WRP). A watchman route for a simple polygon P is a closed walk in P, such that every point of P is visible from some point of the walk. Two point on polygon P are visible from one another if the line connecting them lies completely in P.

The problem can be divided into "fixed source" and "floating" cases[6]. A fixed source WRP, considers a pre-set point s as the starting point of the route, while the floating case finds the optimal route with any possible starting point.

Chin and Ntafos propose an $O(n^4 \log\log n)$ algorithm to find the shortest fixed source route in a simple polygon [4]. It was shown that the problem can be solved in $O(n)$ for orthogonal polygons. The problem is NP-hard for polygons with holes, even when the holes are convex or orthogonal[2]. The hardness result was achieved by reduction from *the Geometric Traveling Salesman problem*.

Different variations of Watchman Route Problem has been studied since, among these problems are External Watchman route, Several Watchman Routes, and Minimum-Link Watchman Tours.

External watchman route problem was proposed by Ntafos and Gewali, as "Finding a shortest watchman route from which the exterior of the polygon is visible". They presented an $O(n^4 \log\log n)$ algorithm for finding the minimum external watchman routes for simple polygons, and more efficient algorithms for restricted classes of polygons (convex, monotone, star and spiral polygons)[15].

Another class of problems, considers the existence of several watchman routes. They define the problem as "Given an art gallery and an integer $m \geq 1$, compute the routes for m watchmen so that each point in the gallery is seen by at least one watchman from some position on his route, and the sum of the lengths of the routes is minimized" [12]. They propose a $\Theta(n^2)$ to solve the problem. The algorithm extends their $\Theta(n)$ solution for finding the shortest watchman route for a spiral polygon. [1] also considers the problem for m watchmen.

1.1.1 The Robber Route Problem

The Robber Route Problem was introduced by Ntafos [13]. We are given a polygon P and a point x on its boundary, a set S (sights) of edges of P and a set T (threats) of points in P. The goal is to find the shortest possible route (if there is one) from x and back to itself, such that every edge in S is visible from some point on the route, while the route is not visible to any of the points in T [13].

The algorithm they present, solves the problem in $O(n^4 \log\log n)$ for simple polygons and in $O((|S| + |T|)n + n \log\log n)$ in orthogonal polygons.

1.1.2 The ZooKeeper and Safari and Aquarium Problems

Safari Route and *Zookeeper* are both variations of the Watchman Route problem. In these problems a route has to be found such that certain specified areas are visited, and other known areas are avoided. In addition, the length of the route has to be minimal. In other words, given a polygon P with a set of k polygonal sites S within P, find the shortest route in P such that every site in S is visited at some point during the route[21].

In the Safari Route problem, one can enter the sites, but in Zookeeper's it is not allowed. Chin and Ntafos who proposed the problems, showed them to be NP-hard in general. However if all sites are attached to P, then polynomial algorithms can solve the problem. They proposed an $O(n^2)$ algorithm to solve the Zookeeper's problem [21], and an $O(n^3)$ algorithm to find a Safari Route [14].

Another interesting problem in this group is *the Aquarium Problem*, which was proposed by Czyzowicz et al [5]. In this problem the edges of the polygon are the visited areas, i.e., the goal is to find the shortest route that touches all the edges of polygon. They gave an $O(n)$ time algorithm for this problem.

1.2 The Two-Guard Street Problem

The two-guard street problem was first presented by Icking and Klein. Given a simple polygon P, and two distinguished vertices s and g, "is it possible for two guards to move from s to t, each guard walking on one of the boundary chains, such that the line segment connecting the guards is always contained within P?". A movement that allows these constraints is called *a walk*, and a polygon that admits a walk is *walkable*. In a general two guard problem, the goal is to find the minimum of all available walks from s to t [8]. The two-guard problem has also been referred to as *corridor problem* when the s and g are edges rather than vertices.

In a walk, a guard may backtrack on his path, in order to maintain his connection with the other guard. If no backtracking is necessary the walk is called *straight*. A counter walk is a walk where one guards walks from s to g and the other walks from g to s.

Icking and Klein propose an $O(n \log n)$ solution for the decision problem, also possible straight walk and straight counter walk can be found in $O(n \log n + k)$. In a general walk when backtracks are unavoidable, $O(n \log n + k)$ time and linear space is sufficient for finding the walk, where k is the length of the walk and may be $\Theta(n^2)$ [8].

These results were improved by Heffernan [7]. He presented an $O(n)$ algorithm to determine whether a general walk from s to g is possible, although his algorithm did not produce the actual walk.

Tseng, Heffernan and Lee, considered the problem from another perspective, they proposed an algorithm that finds all pairs of points in a simple polygon P, that admit a straight or general walk in $O(n \log n)$ time and all pairs that admit a straight or counter walk in $O(n \log n + m)$[20]. Bhattacharya, Mukhopadhyay and Narasimhan, further improved the results. Their algorithm determines all pairs of points in P which admit walks, straight walks and discrete straight walks in $O(n)$.

1.2.1 The Three-Guard Street Problem

In a recent article, Tan proposes the three-guard problem as an extension of the two-guard problem [19]. As in two-guard case, we are given a simple polygon P and two points s and g on P. One guard walks on the left chain, another on the right chain and the third guard walks inside the polygon; the two guards walking on the perimeter of P should always maintain visibility with the inside guard. The question is if there is a (straight)(counter) walk for three guards from s to g. Their solution, decides the walkability of a given polygon in $O(n \log n)$ time, generates a walk in $O(n \log n + k)$ time, where $k \in O(n^2)$ is the size of the optimal walk [19].

1.3 Searching For an Intruder

There are several problems that fit into this category, they generally assume that there is an intruder or intruders in a polygonal space, and a number of guards have to catch or detect the intruder in finite time. This class of problems is similar to our problem, since in addition to finding the intruder they require a minimal coverage of the polygon. The difficulty of these problems however, is mostly related to polygon visibility, which makes it different from our problem.

1.3.1 The Hunter's Problem

Suzuki and Yamashita, defined *The Hunter's Problem* as searching a polygonal region for a mobile intruder by one searcher. The searcher is either $k-$searcher ($k \geq 1$) or $\infty-$searcher. $k-$searcher carries k flashlights each emanating a ray, that define his sight. The direction of the rays can be changed continuously with a bounded angular rotation speed. $\infty-$searcher can see in 360 degree simultaneously. The goal is to decide if there is a *schedule* for the given searcher, on a polygon P to find the intruder, and if so, to generate the schedule [18].

They show that if there are three points x, y and z in a polygon P, such that the shortest path (within P) between each pair is not visible from the third point, then P is not 1-searchable. Also for hedgehog polygons, 2-searcher is as capable as $\infty-$searcher [18].

1.3.2 Room Search Problem

The Room Search Problem was introduced in [10]. The problem assumes that we are given a simple polygonal room with one door at point d. Assume that there is an intruder in the room, and the question is whether there is a possible path, a searcher who enters the room from the door can follow, and detect the intruder before he slips out of the room through the door. In their problem the polygon is searched with a 1-searcher. They observe that reflex angles are the source of problems and design their algorithm to detect certain conditions that makes a polygon not searchable by a 1-searcher.

Chapter 2

Problem Statement

2.1 Guarding Conditions for k Guards

We define the problem of guarding a polygon P with k mobile guards, as assigning paths to the guards, such that every point on P is covered by the configuration of the guards at some point in time, and also the total length of the paths followed by the guards is minimized.

The guards are free to move in any way they need to, as long as the total path length is minimized. Based on the shape of the polygon, the number of guards and the definition of P, we get different sweeps.

In this problem the convex polygon is a simplification for an art gallery, and its edges are the walls of the gallery. To make the problem definition consistent with the art gallery concept, we limit the view of each guard by the walls surrounding it; thus the guards have to remain inside the polygon in order to cover it, i.e., $G(t) \in P$ for all t, where $G(t)$ refers to the position of guard G at time t.

When studying the problem in convex polygons, the primary focus is on the shape of the sweep. We try to find whether a guard needs to stay on the perimeter of the polygon or crosses the interior, and also if the paths of the guards intersect each other. However when a given polygon is not convex, the mutual visibility of the guards plays a key role in finding the shortest paths for the guards. Therefore we decided to limit our study to convex polygons.

In chapter 3 we study the problem for two guards. Paths are defined in a way that the line segment connecting the guards, sweeps the surface of the polygon. In addition, we make the total distance traveled by the two guards is minimal.

We prove that in any optimal 2-guard sweep of a convex polygon, both guards stay on the boundary, i.e., for every 2-guard sweep that uses interior paths, we offer a new sweep not longer than the original, where the guards do not leave the perimeter of P.

In chapter 4 we study the problem for three guards in a convex polygon. Unlike the two guards case, there are cases where crossing the interior of the polygon makes the sweep shorter. We prove that an optimal 3-guard sweep is *simple*. A sweep is *simple* if none of the paths intersect in the interior of the polygon.

2.2 Coverage Model

In the precious section we used the phrase *covering* a polygon, without defining what exactly coverage means. In this section we explain different coverage models and their relationships.

2-guard coverage is defined as sweeping the polygon with the line that connects the guards. In other words, every point in the polygon has to be on the line that connects the guards at some point in the sweep. We call this model of coverage *pair-wise line coverage*.

Although the definition of coverage for the 2-guard case was simple, there are different possible definitions for coverage when there are more than two guards. It is possible to define 3-guard coverage as pair-wise line coverage. This definition is a direct extension of the two guard coverage, so much so that it can be considered as 3-way 2-Guard Coverage, since every point on P has to be on the line that connects a pair of guards at some point in time to be covered.

However there are other possible definitions; one of them is *area coverage*, where every point inside the triangle region determined by the guards is said to be covered. As the guards move in P, the shape of this covered triangle changes and new points in the polygon are covered. This definition may be clear when P is convex, but when guards are not mutually visible in a non-convex polygon, the definition becomes unclear.

It is clear that a pair-wise line sweep is also an area sweep, the opposite however is not true. Fig. 2.1 shows an area sweep that is not a line sweep,

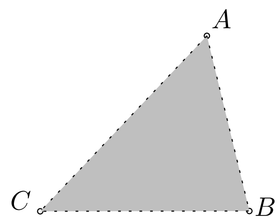

Figure 2.1: Line vs. area coverage, the gray area is not covered by line coverage.

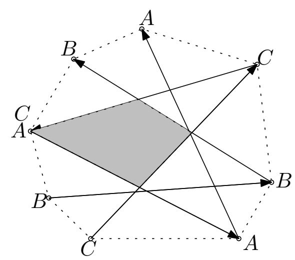

Figure 2.2: The gray area is not covered by area or line coverage.

the gray area in the picture is the area not covered when using line coverage.

Another definition is *boundary coverage*, where a polygon P is covered if every point on the boundary of P has been covered by the guards. Points on the boundary of a polygon are covered when they lie on a closed line segment that connects two of the guards at some point in time, i.e., every point on the boundary of P has to be either on the path of a guard or on an edge, that is visited simultaneously by a pair of guards at some point in time. We refer to such an edge as *free edge*.

Using the definitions of coverage explained so far it is easy to see that a sweep is said to provide boundary coverage if it provides either area coverage or pair-wise line coverage, since in both area and line coverage, the boundary of the polygon has to be covered as well. A boundary coverage on the other hand does not guarantee a line or area coverage for a polygon, even when the polygon is convex. Fig. 2.2 shows a sweep that provides boundary coverage for the polygon but not area or line coverage.

This implies that minimum length of the paths of the guards that provides boundary coverage is not longer than the minimum length of the paths that covers the area of the polygon, i.e., if an optimal boundary coverage provides area or line coverage, then it is also an optimal area or line sweep, respectively.

We define a *boundary cover* as a sweep that provides boundary coverage. A boundary cover may include traversed and untraversed parts of the boundary. A traversed portions of the perimeter of P, contained between two free boundary segments, is in the form of a chain of traversed edges.

A *Chain* is a series of consecutive traversed edges of P between two free boundary segments. The edges are not necessarily visited by the same guard.

The perimeter of a polygon can be divided into chains and free boundary segments, and by definition, every point on a chain has been visited by a guard at some point in time, which leaves us with covering the free boundary segments. In a convex polygon, a free boundary segment can be covered if its end-points are simultaneously visited by a pair of guards at some point during the sweep, therefore we can derive the following condition as the necessary and sufficient condition for a boundary cover:

In a k-guard boundary cover, every boundary segment e of a convex polygon P is

- Traversed by a guard, or
- Its end-points are visited by two guards simultaneously.

Thus, any set of k parameterized curves on a convex polygon P that follow these rules make a boundary cover.

A sweep may include any number of chains separated by free boundary segments. It is easy to see that if the number of chains is less than or equal to the number of guards, then none of the guards need to cross the interior of the polygon. This is true even when there are more than three guards (k guards). We can show that in a k guard sweep with $m < k$ chains the guards can sweep the polygon without crossing the interior.

The definition of a chain implies that the length of a sweep T with m chains is at least

$$\sum_{1 \leq i \leq m} |C_i|$$

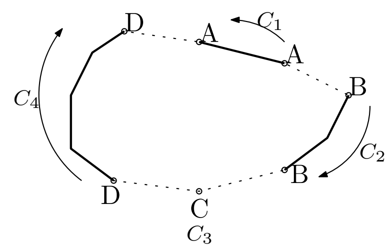

Figure 2.3: 4 guards cover the polygon with a 4-chain sweep, without crossing the interior of the polygon. Guards C is stationary during the sweep, as its chain (C_3) is empty.

where C_i is the i^{th} chain in T, and $C_i \cap C_j = \emptyset$, for $i \neq j$. This lower bound is achievable when $m \leq k$ (Fig. 2.3),

- Assign one guard to each chain and set an arbitrary order of visit for the free boundary segments. The rest of the guards (if there are more guards than chains) are put on different chains, and stay stationary during the sweep.
- Each guard moves on its chain, walking from one free boundary segment to the other.
- Direction of each walk is from the free boundary segment with the lower order towards the free boundary segment with the higher order.

This implies that having fewer than k chains would not be optimal, since a sweep with this number of chains leaves more than one guard in a chain where it is not needed. In fact, we can prove that there should be exactly one guard traversing a chain.

Having too many chains would also make the sweep longer, since a sweep with too many chains implies that the guards are crossing the interior of the polygon many times, and possibly even intersect each other's paths. Although crossing the interior can make the sweep shorter in certain scenarios, this only happens with few interior crossings.

2.3 Minimizing Total Path

Our goal is to find a sweep with minimum total length for the guards, based on the definition on coverage for the polygon. Previous problems limited the paths of the guards to the boundary of the polygon, defined specific start and end points for the guards or limited the movements of the guards by forcing them to stay visible to each other at all times. In our problem we have relaxed all of these conditions, the guards are allowed to start their paths on any point within the polygon, and they are allowed to walk on the interior of the polygon.

For the convex polygon case, we will prove that in a two guard sweep, crossing the interior always makes the paths longer, but with greater than or equal to three guards, then there are certain cases where the guards need to cross the interior in order to realize their minimum paths.

For simple polygons, there are even more possibilities. In addition to crossing the interior, there are cases where the guards do not maintain visibility with one another during their paths, since covering different parts of the polygon with a different small team of guards may be shorter than moving all guards through the polygon. Or in a k-star shaped polygon, k guards may stay stationary at key points, while the rest of the guards walk the boundary.

2.4 Convex Polygons

Finding the unconditional minimum path for the guards that cover a polygon is not trivial even for convex polygons. Although it may appear that crossing the interior of the polygon increases the lengths of the paths, we can easily find examples where this is not true. When the polygon is covered by three or more guards, then there might be a shortcut through the interior that makes the path of a guard shorter. Examples of sweeps with interior crossings will be given in future chapters.

In simple polygons the problem becomes much harder. Fig. 2.4 shows a 2-guard sweep, where the guards cover the polygon by walking along parts of the boundary in opposite directions, while the five long edges of polygon lie on the line that connects the guards at some point in time.

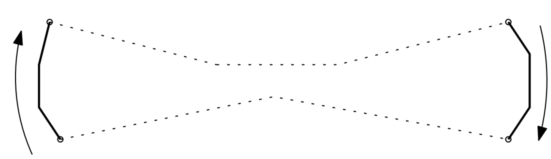

Figure 2.4: An example of a sweep in a simple polygon.

Chapter 3

2-Guard Solution

In the 2-guard case, we sweep a convex polygon with two mobile guards, in a way that the line connecting the guards covers the whole surface of the polygon, and every point in the polygon is on the line segment joining the two guards at some point in time. In addition, we want the total distance traveled by the two guards to be minimal.

Previously studied versions of the problem, have always restrict the paths of the guards to the perimeter of the polygon. In our problem, we allow each guard to start his path anywhere within the convex polygon; and to walk on the interior if needed. We show that allowing the guards this flexibility of movement in the 2-guard case does not improve the result, that is there always exists an optimal solution with the guards walking only on the perimeter of polygon.

Covering points on the boundary of a polygon is clearly a necessary condition for covering the polygon, but it may not always be sufficient. In this section, we will first show that an optimal boundary cover can be achieved while the guards stay on the perimeter of the polygon, following our proposed paths. Then we will show that our proposed paths cover the polygon. It follows that the optimal 2-guard boundary cover provides an optimal 2-guard polygon cover.

For a boundary cover each edge in P is either traversed by a guard or covered. Therefore a perimeter-restricted boundary cover (a sweep restricted to the perimeter) of a convex polygon with n edges using two mobile guards must traverse at least $n-2$ edges; as the guards can start their paths from the endpoints of one edge and finish it at the endpoints of another edge, one guard walking clockwise and the other anti-clockwise (Fig. 3.1).

The total length of such a perimeter-restricted boundary cover is the sum of the lengths of the paths traversed by the guards. The length of shortest 2-guard path, therefore, is the sum of the $n-2$ shortest edges of

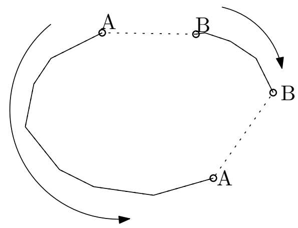

Figure 3.1: 2-Guard perimeter sweep of a convex polygon.

the polygon. This sweep can be obtained in $\theta(n)$. We prove the optimality of this solution, by showing that there always is a minimum length cover that is perimeter-restricted.

3.1 Definitions

In this section, we define the common concepts used in this problem. Many of them are used in the 3-guard case as well.

In this problem, the guards are allowed to walk both on the perimeter and the interior of the polygon; therefore the definition of the polygon P includes the boundary and the interior of the polygon. The boundary of the polygon is denoted ∂P, and the interior of the polygon is denoted $I(P)$.

The guards trace two curves A and B on the polygon. Every point on A or B, is said to be visited by the guards, i.e. a point p in P is *visited* if, $p \in A$ *or* $p \in B$
As the relative position of the guards in the polygon is important in the sweep, the curves walked by the guards has been parameterized by time, such that at $t = 0$ none of the guards have started their walks and the total length of the parameterized curves is 0, and at $t = 1$ both guards have finished their walks.

Since the notion of the *length* plays an important part in this problem, we have defined some the more commonly used lengths formally:

- The length of a line segment e is $|e|$, similarly the length of the line segment ED is $|ED|$.

- The boundary length of the polygon denoted $|\partial P|$ is given by, $|\partial P| = \sum_{1 \leq i \leq n} |P_i P_{i+1}|$, where $P_i P_{i+1}$ is an edge of P.

The notion of *coverage* that has been used so far can be formally defined as follows: A point p is *covered* by a pair of parameterized curves $(A(t), B(t))$, if $p \in A(t)B(t)$ for some $t \in [0, 1]$. Notice that when P is convex, the guards are always visible to each other; it follows that every point visited by a guard is covered. We define a sweep to be "a pair of parameterized curves A and B, that cover P", i.e., $\bigcup_{t\in[0,1]} A(t)B(t) = P$.

In a sweep parts of the perimeter of P may be unvisited, but still covered. We call these parts *free boundary segments.* A *free boundary segment* d of the polygon P in sweep (A, B), is a maximal connected subset of the boundary of P that is unvisited.

3.2 2-Guard Problem

For the 2-guard problem, we defined the coverage of a pair of guard paths (parameterized curves) as the set of all points in the polygon that belong to the line segment joining the two guards at some time $t \in [0, 1]$. In order to find the optimal solution to the 2-guard case, we will first relax the definition of the coverage to boundary coverage, i.e., we find the shortest arrangement of paths for the two guards that is required to cover only the boundary of the polygon. Clearly the length of the shortest such boundary cover is at most the length of the shortest polygon cover. We will then show that since an optimal boundary cover keeps the guards on the perimeter of the polygon, it also covers the interior of the polygon. Thus an optimal 2-guard boundary cover, is also an optimal polygon coverage.

This section starts with proving certain conditions about a free boundary segment. It is clear that a free boundary segment can not include a curve, since the polygon is convex. In this lemma we will show that a free boundary segment can not contain a corner of the polygon.

Lemma 1. *Every free boundary segment in a convex polygon P during a sweep (A, B) is a straight line.*

Note: We assume that the vertices of P are all extreme vertices, that is, no two edges of P are co-linear.

Proof. If a free boundary segment is not a straight line, then it includes at least one extreme vertex v of the polygon, i.e., a vertex on the convex hull of P. No convex combination of two points, not equal to v, in P contains v. Hence, the sweep must visit v. □

There can be any number of free boundary segments in a sweep. We sort these free boundary segments by the time that they are first covered. In a 2-guard sweep all but the first and the last are called the *intermediate free boundary segments*. In a more general case, an intermediate free boundary segment is a free boundary segment which is not next to an exposed end-point of a path.

In the optimum 2-guard sweep, the guards only traverse $n-2$ edges of the polygon, leaving the two longest edges out. They start their paths from the two endpoints of one of the two edges and finish it at the endpoints of the other, walking exclusively on the perimeter. The total length of the sweep would be $|\partial P| - |\ell_1| - |\ell_2|$, where ℓ_1 and ℓ_2 are the two longest edges of P.

Theorem 1. *The length of the optimal 2-guard sweep of a convex polygon P is at least $|\partial P| - |\ell_1| - |\ell_2|$.*

Proof. We want to show that the length of every sweep is at least the length of our proposed optimal solution. First we will show that our solution covers the boundary of the polygon with minimum total length of the paths of the guards. We will then show that the solution also provide a polygon coverage, which leads to the conclusion and the theorem follows.

Suppose a shorter sweep (A, B) exists that covers the boundary of the polygon. Such a sweep must avoid covering more of the boundary of P than the length of $n-2$ shortest edges. It must have more than two free boundary segments, since by Lemma 1 every free boundary segment is a straight line, and therefore a part of an edge. The goal is to show that the length of every such boundary cover is at least the length of our proposed optimal solution. We prove this claim by showing that the length of (A, B) is longer than $|\partial P| - |s| - |t|$, where s and t are the first and last covered free boundary segments in (A, B), respectively. We will then argue that choosing the two longest edges of the polygon as starting and ending free boundary segments gives us the optimal boundary cover.

The solution follows a few steps, in the first step, we will remove any existing intersection of the paths of the guards, by assigning a new path to each guard. The new paths lead the guards through a perimeter path from s to t, one guard on the left side and the other on the right. We then explain that the new paths are shorter than the original sweep.

Since s is the first free boundary segment that is covered by the two guards, the guards do not participate in covering any other free boundary segments prior to s. This implies that if a guard starts its path before walking to s, that part of its path, is entirely contained inside a chain, thus the guard does not have an exposed starting point. Similarly, the guards either end their paths at t, or on chains. These two facts imply that any two guard sweep of a convex polygon, can only contain exposed end-points at the endpoints of its first and last covered free boundary segments (s and t).

In the new sweep, both guards start their paths at the end points of s, and finish them at the end-points of t, one guard walks on the left side of the polygon, and the other on the right side. Their paths are chosen from segments of the original sweep (A, B), we will give the left-most segments (regardless of which guard's path they originally belonged to) to one guard and the right-most segments to the other guard. The rest of this section, describes this method in more detail.

- Guard 1,
  - starts its path at the left end-point of s.
  - takes the leftmost path on $I(P)$ at each intersection.
  - until it reaches the left end-point of t.
- Guard 2,
  - starts its path at the right end-point of s.
  - takes the rightmost path on $I(P)$ at each intersection.
  - until it reaches the right end-point of t.

The new paths of the guards will be easily found if there are no other free boundary segments along the way, since all the edges from the left (right) end-point of s to the left (right) end-point of t would be visited, this implies that the new path of the guards would be restricted to the boundary of the polygon. The length of the new boundary cover is $|\partial P| - |s| - |t|$, and it is less than or equal to the original boundary cover since it does not introduce

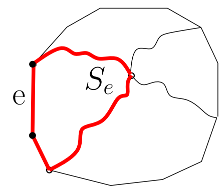

Figure 3.2: Pocket S_e for free boundary segment e.

any new segments to the paths.

If there are any other free boundary segments (not s or t) on the left (right) side, then these free boundary segments are intermediate free boundary segments. When reaching an intermediate free boundary segment e, a guard will choose a path that leaves the perimeter at one end-point of e, walks on a *pocket* made by the interior leftmost (rightmost) paths around e and gets to the other end-point of e. Thus we have to prove that there are unique pockets for all intermediate free boundary segments along the way. We will prove that for any intermediate free boundary segment e there is a pocket S_e made of the paths of the guards, i.e., a set of sub-curves of A and/or B, to each intermediate free boundary segment e, as follows:

Definition. A *pocket* of an intermediate free boundary segment e is the smallest cell, in the *arrangement* of the polygon edges and all guards' paths (both boundary and interior paths), that contains e and lies in P (Fig. 3.2).

A pocket may include parts of the paths of different guards, even parts of the visited boundary of P. If any part of the pocket is traversed multiple times by (A, B), only one of these visits is considered as part of the pocket.

We assumed the left and right do not share any part of the paths. We need this to be true because the argument is:

$$|(A,B)| \geq |(A,B) \cap \partial P| + \sum_{i=1}^{m-1} |S_i| \tag{3.1}$$

$$\geq |(A,B) \cap \partial P| + \sum_{i=1}^{m-1} |e_i| \tag{3.2}$$

$$\geq |\partial P| - |\ell_1| - |\ell_2| \tag{3.3}$$

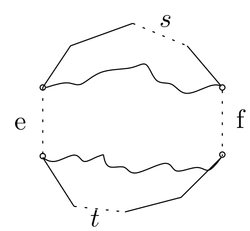

Figure 3.3: Existence of pockets for all intermediate free boundary segments.

Where e_i is an intermediate free boundary segment, and S_i is its pocket.

Line. 3.1 claims that the new boundary cover is not longer than the old one. This is true if the left and right paths use parts of the old paths of the guards and do not use any part twice (left and right paths do not share a segment).

There is a closed pocket around any intermediate free boundary segment, otherwise we can divide (A, B) into two disconnected regions, which is a contradiction (See Fig. 3.3). Since a path is a connected curve from one end-point to another, then a pocket would also be connected and closed around its free boundary segment.

Assume that s is on top of the polygon and t is at the bottom. Any other free boundary segment would therefore, lie between s and t. Since both guards are needed to cover s and t, then there are at least two paths from s to t. In other words, making any cut on the polygon that separates s from t would cross (A, B) in at least at two points. This implies that the left and right paths do not share any segment. Fig. 3.4 and Fig. 3.5 show the two cases where left and right paths share a segment, and it is clear that in both cases we can find a cut that separates s from t and intersects (A, B) at only one point, which is a contradiction.

Line 3.2 follows immediately from line 3.1. Clearly $|S_e| \geq |e|$ for every intermediate free boundary segment e, since e is a straight line segment. Line 3.3 is derived from line 3.2, using the fact that the length of the boundary of the polygon $-$ the length of its first and last covered free boundary segments, is greater than or equal to the length of our proposed solution.

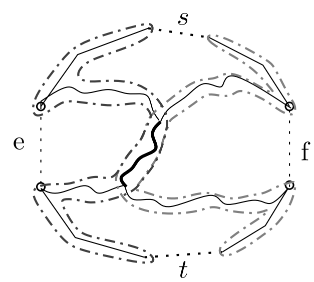

Figure 3.4: Left an right paths share a visited interior segment.

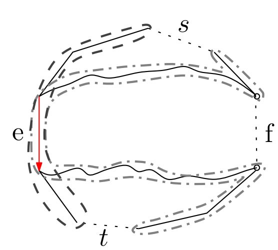

Figure 3.5: Left and right paths share a visited boundary segment.

So far we proved that for any intermediate free boundary segment along our proposed left and right paths, there are pockets which do not share any parts of their paths and can be replaced with their respective free edge, since each pocket is strictly longer than its intermediate free boundary segment. We refer to this replacement of pockets with their free edges as flattening a pocket.

By flattening each pocket, we create boundary paths from s to t for each guard. Thus we create a boundary cover of length $|\partial P| - |s| - |t|$, which is less than or equal to the length of the original boundary cover. It also limits the paths of the guards to the perimeter of P. Thus the shortest boundary cover would have s and t equal to the two longest edges of the polygon.

Having already established that the length of the optimal sweep is greater than or equal to the length of the optimal boundary cover, we can apply this solution as the optimal sweep, as it also provides coverage for the interior of the polygon. $\square$

Chapter 4

3-Guard Solution

In the previous section we proved that the optimum solution to the 2-guard case does not include any interior paths, i.e., the guards stay on the perimeter of the polygon during the sweep.

When we increase the number of guards to 3, it may appear that the same principle is still true, but studying a few simple examples shows that in certain convex polygons, crossing the interior can make a sweep shorter. Fig 4.1 shows a simple example, where crossing the interior is beneficial.

Another such example is depicted in Fig. 4.2. In this picture, two guards stay at opposite sides of the polygon while the third walks part of its first chain, then crosses the polygon to traverse its second chain, walks back to its first chain and covers the rest of it.

We defined 3-guard coverage as *area coverage* or *triangular coverage*, since the area covered by a set of three points in a convex polygon is in the form of a triangle. Having explained that a boundary coverage is a relaxation of the triangular coverage, we will first relax our sweep to boundary cover, since the necessary conditions for a boundary cover are easy to recognize. We will prove that an optimal boundary cover is simple. This limits the optimal 3-guard boundary cover to one of the three forms in Fig. 4.3, Fig. 4.4 or Fig. 4.5. It is easy to see that these cases also provide area coverage, thus an optimal boundary cover gives us an optimal area sweep.

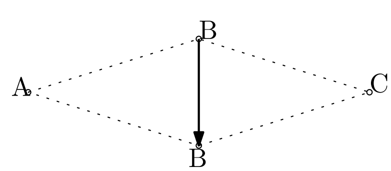

Figure 4.1: An optimal sweep that requires crossing the interior of the polygon.

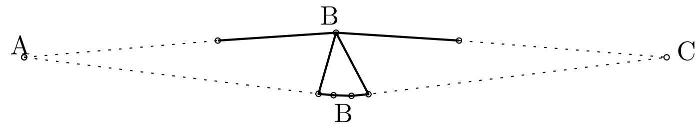

Figure 4.2: Another optimal sweep that requires crossing the interior of the polygon.

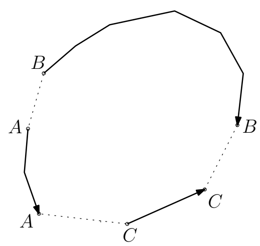

Figure 4.3: A sweep with no interior paths.

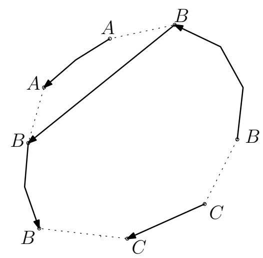

Figure 4.4: A single interior crossing.

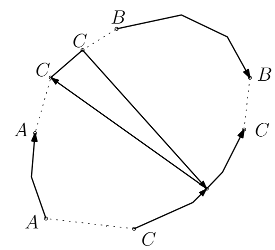

Figure 4.5: One guard crosses the polygon, and back again to its previous chain.

Using these cases we designed algorithm 1 for finding the shortest simple 3-guard boundary cover for convex polygon P. It finds the solution in $O(n^5)$ time, where n is the number of edges in P. Finding further conditions on optimality of a sweep in Fig. 4.3, Fig. 4.4 or Fig. 4.5 may lead to an improved algorithm, but we will not focus on improving the algorithm at this time.

Given two chains as inputs, Distance$(Chain_1, Chain_2, D, P)$ finds the shortest possible path for a guard to visit these chains. A sweep where one guard covers two chains is either as Fig. 4.4 or as Fig. 4.5. Distance finds the shortest interior chords that connect the two chains in a valid format. In an instance of Fig. 4.5 guard C can depart from the boundary of P at either a vertex or even an interior point on an edge.

We want to find a point x on a line l, where the sum of the distances of two points p and q to x is minimized. We draw the orthogonal projections of p and q on l, and call the intersection points p' and q' respectively. Assuming that $|pp'| = a$, $|qq'| = b$, and $|p'q'| = d$, it is easy to see that $|p'x| = \frac{a}{a+b}d$. Since in Fig. 4.6 $|\hat{p}x| + |qx|$ is minimized when $\hat{p}q$ is a straight line.

Algorithm 1 Traversal Re-arrangement(Polygon $\mathcal{P}$)

```
1:  ℓ1, ℓ2, ℓ3 ← three longest edges of P
2:  minimumTraversal ← |∂P| − |ℓ1| − |ℓ2| − |ℓ3|
3:  for every edge e and vertex v in P do
4:      D[v, e] ← distance of v to e
5:      P[v, e] ← orthogonal projection of v on e
6:  end for
7:  for each 4 edges e1, e2, e3 and e4 in P do
8:      C1 ← chain between e1 and e2
9:      C2 ← chain between e2 and e3
10:     C3 ← chain between e3 and e4
11:     C4 ← chain between e4 and e1
12:     minimumTraversal ← min{ minimumTraversal, |C1| + |C2| + |C3| +
        |C4| + min{ Distance(C1, C3, D, P), Distance(C2, C4, D, P)}}
13: end for
14: return minimumTraversal
```

In Algorithm 1 there are two loops, the first one (step 3) calculates D and P for all pairs of edge and vertex thus taking $O(n^2)$ time. The second one (step 7) iterates $\theta(n^4)$ times, since it is run for every set of four vertices in the polygon. In each iteration two calls to Distance$(Chain_1, Chain_2, D, P)$ are made, and each call takes $O(n)$ since $m + k \leq n - 2$. Therefore the algorithm runs in $O(n^5)$ time.

Algorithm 1 assumes that in an optimum sweep the paths are simple.

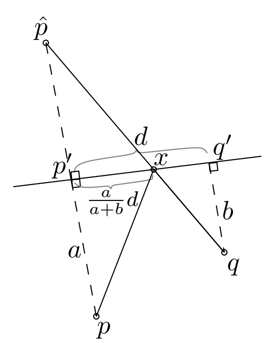

Figure 4.6: $|px| + |qx|$ is minimized when $|p'x| = \frac{a}{a+b}d$.

Algorithm 2 Distance(C_1, C_2, D, P)

$p_1, p_2, ..p_k \leftarrow$ vertices of C_1
$q_1, q_2, ..q_m \leftarrow$ vertices of C_2
$distance \leftarrow \min\{|p_1q_1|, |p_1q_m|, |p_kq_1|, |p_kq_m|\}$
for i from 2 to k **do**
    $a \leftarrow D[q_1, p_{i-1}p_i]$
    $b \leftarrow D[q_m, p_{i-1}p_i]$
    $d \leftarrow |P[q_1, p_{i-1}p_i], P[q_m, p_{i-1}p_i]|$
    $x \leftarrow \dfrac{a}{a+b} d + P[q_1, p_{i-1}p_i]$
    if x is on $p_{i-1}p_i$ **then**
        $distance \leftarrow \min\{distance, |xq_1| + |xq_m|\}$
    end if
    $distance \leftarrow \min\{distance, |p_iq_1| + |p_iq_m|\}$
end for
for j from 2 to m **do**
    $a \leftarrow D[p_1, q_{j-1}q_j]$
    $b \leftarrow D[p_k, q_{j-1}q_j]$
    $d \leftarrow |P[p_1, q_{j-1}q_j], P[p_k, q_{j-1}q_j]|$
    $x \leftarrow \dfrac{a}{a+b} d + P[p_1, q_{j-1}q_j]$
    if x is on $q_{j-1}q_j$ **then**
        $distance \leftarrow \min\{distance, |xp_1| + |xp_k|\}$
    end if
    $distance \leftarrow \min\{distance, |p_1q_j| + |p_kq_j|\}$
end for
return $distance$

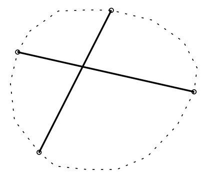

Figure 4.7: Single intersection.

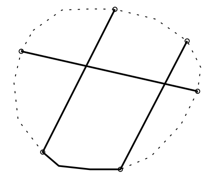

Figure 4.8: Double intersection.

Considering the examples presented for optimal boundary covers that involved interior crossings of the polygon, it is easy to imagine that this assumption is wrong.

If there is an intersection in the paths of the guards, we will show that it is either a single intersection as in Fig. 4.7, where an interior path intersects another interior path once, or a double intersection as in Fig. 4.8, where one path crosses another path twice. We study the single intersection case in section 4.2, and the double intersection case in section 4.3. In both cases we will prove that the sum of the lengths of the involved interior paths is longer than some free boundary segments, i.e., we can find enough free boundary segments to visit in place of the interior paths.

In the rest of this section we will prove that any boundary cover that contains intersections, can be reduced to a boundary cover with no intersection, a boundary cover with a single intersection or a boundary cover with a double intersection. Therefore we can prove that an optimal boundary cover does not contain an intersection of the paths of the guards.

Theorem 2. *In an optimal 3-guard sweep, the paths of the guards never intersect one another.*

Proof. As in the 2-guard case, we will use parts of the existing sweep, without considering the paths of the individual guards, and build a new sweep T' on P, where $|T'| \leq |T|$ and no guards' path intersects the paths of another guard in T'.

We can easily find the first and the last covered free edges of T, calling them s and t respectively. Since every free edge is visited simultaneously by two guards, the two guards that cover s, either start their paths at endpoints of s, or start their paths elsewhere but prior to visiting s their paths are limited to the interior of chains, i.e., they do not visit any free edges

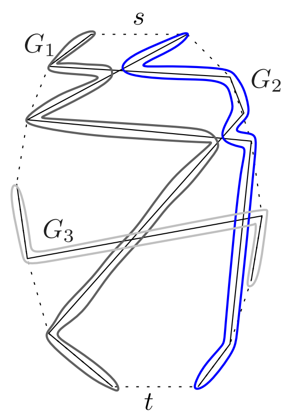

Figure 4.9: The new paths for the three guards, when all paths intersects (Case 3).

prior to s. The same conclusion can be applied to t, i.e., the guards that cover t, do not visit any other free edges after t. This implies that four of the six end-points in the sweep, are either inside chains or at the end-points of s or t. The two remaining end-points can be used to cover additional free edges.

Based on the positions of these end-points sweep T is in one of these three cases:

Case 1 , T is simple, i.e., it does not include any intersections of the paths of the guards.

Case 2 , each guard's path intersects at least another guard's path.

Case 3 , one guard walks its path isolated from the others, i.e., it does not intersect any other path.

Case 1, Clearly if T does not include any intersections of the paths of the guards, then no modifications are necessary, thus we declare $T' = T$. In the rest of the proof we study the other two cases individually, explaining how each case can be changed into a simple sweep.

Case 2, If each path intersects at least one other path, then we will redefine the paths of the three guards to remove the intersections and make T' simpler and shorter than T (see Fig. 4.9).

Assuming that s is on top and t at the bottom of the polygon, we will change the paths of the guards such that one guard covers the right side of the picture, another the left side and the third guard connects the remaining parts, by following these steps:

- The 1st guard walks on the left chain.

  - It starts its path at the left end-point of s.
  - walking the leftmost path in the arrangement of all guards paths, from s to another end-point.
  - while this end-point is not the left end-point t,
    - back track to the last visited intersection and follows the next leftmost path in the arrangement of all guards paths.
  - It should be noted that the guard will eventually reach the left end-point of t, since all paths intersect other paths, implies that there is a path between any two points in T.
  - If any part of the path forms a pocket for a free edges, then change the sweep to include the free edge in place of its pocket.

- The 2nd guard walks on the right chain.
  - It starts its path at the right end-point of s.
  - Walks the rightmost path in the arrangement of all guards paths, from s to the next end-point.
  - while this end-point is not the right end-point t,
    - back track to the last visited intersection and follows the next rightmost path in the arrangement of all guards paths.
  - If any part of the path forms a pocket for a free edges, then change the sweep to include the free edge in place of its pocket.

- There are at most two exposed vertices separate from s and t.

- The 3rd guard takes the path that connects the two remaining end points on P.

The previous algorithm implies that there is always a path between the two remaining end-points that does not include any part of the left and right paths. This is true since even after removing the left and right paths we will be left with these exposed end-points, and since all of the other parts of the

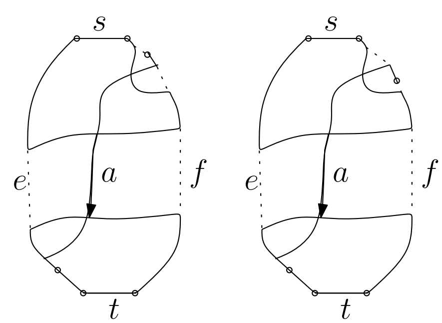

Figure 4.10: Left path and right path share a line segment.

paths have even degrees (since a path is a connected curve), then any path from one exposed end-point has no choice but to end at the other end-point. If there is only one exposed end-point, then the third guard does not cross the interior, since we can consider the chain that includes this end-point as the total path of the guard.

The following lemma proves that when the left and right chain share a line segment then the sweep can be reduced to a single intersection case and we will show that a sweep with single intersection can be reduced to a shorter sweep with only three chains in section 4.2.

Lemma 2. *If the left and right paths in sweep T share a line segment, then there is a sweep T_2 with a single intersection, such that $|T_2| \leq |T|$.*

Proof. In the algorithm we assumed that left and right paths do not share a line segment. Assume that in sweep T, the left and right paths share a line segment a (Fig. 4.10). We will show that this sweep can be transformed into a single intersection case.

In Fig. 4.10, the left and right paths share a. Let t_a be the time that guard 1 walks a in the original sweep T. Since no other guard's path crosses the cut through a that separates s and t and since covering s and t needs two guards then guard 2 is always above a and guard 3 is always below it.

There is at least one free edge that crosses the cut through a (there is exactly one free edge if a is a part of the perimeter of P). Let e be the edge across the cut with the earliest cover time t_e, and f be the one with the

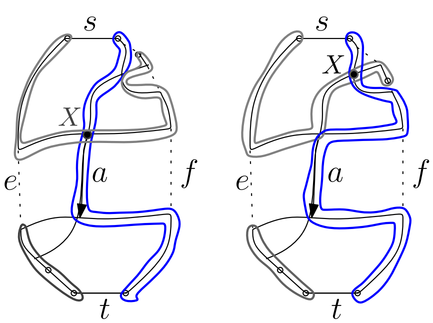

Figure 4.11: Re-arranging of sweep T when left and right path share a line segment.

latest cover time t_f. Note: e might equal f. There two possible cases that define the shape of T,

- $t_f \geq t_a$, implies that guard 2 finishes its path at f (or in a chain), since it cannot participate in covering any more free edge after t_a.
- $t_e \leq t_a$, implies that guard 3 starts its path at e (or in a chain), since it cannot participate in covering any more free edge before t_a.

In either case, one of the top or bottom section cannot contain another exposed end-point. Assume that the top part has a free edge with an exposed end-point and the bottom part contains only intermediate free edges.

We flat the pockets of intermediate free edges using a method similar to the left and right paths, and the outcome would be Fig. 4.10. Based on the direction of the tail that leads to the exposed end-point we can re-arrange the paths to one of the forms in Fig. 4.11. Each has only one intersection, at X. $\square$

If the left and right paths do not share a line segment then guard 1 traversed the left path and guard 2 traverses the right path and the third guard connects the remaining two exposed vertices. Depending on the position of these exposed vertices, there are two possible case: when the exposed vertices are on the same side of the polygon, we have a double intersection, which is addressed in section 4.3, and when they are on different sides, it is a single intersection, addressed in section 4.2. We prove that both cases can be simplified to paths no longer than the original with no intersections.

Case 3, If one guard walks its path isolated from the others, i.e., it does not intersect any other path, then we can create T' from T using a method similar to the left and right path selections.

Assuming that s is on top and t at the bottom of the polygon, we divide the sweep into three regions, isolated, left and right (if they exist); where isolated contains the path that does not intersect other paths. Left and right each are the remaining parts of the sweep on the left and right side of the polygon, respectively. Depending on the position of the isolated region we can divide this case into three parts,

1. It is possible that the isolated chain covers one side of the polygon completely, i.e., there are only two regions (isolated and one of left or right). Without loss of generality we assume that the isolated chain is on the left side. We can consider the isolated chain as the left path, traversed by guard 1. Guard 2, would take the right-most path from s to t. If there are two exposed vertices left (not at s or t), guard 3 would take the path that connects these vertices. Since the left side is an isolated chain, both exposed vertices are on the right side. Connecting these exposed vertices creates a sweep with a double intersections which is addressed in section 4.3.

2. If the isolated region does not contain any end-point of s or t, then by choosing the left and right paths will create T'. Since our isolated region contains two free boundary segments, and the other four are involved in covering s and t, then by flattening the intermediate free edges in left and right paths we will get a sweep that has no intersections.

3. If the isolated region includes one end-point of s, or t, then without loss of generality we can assume that it is the right end-point of s. We have at most one other exposed end-point, since s and t use four of the end-points, and the isolated region uses two, one an end-point of s. If this end-point is not exposed, then we can easily re-assign the paths of the guards in a way that it includes no crossing of the interior (each guard traverses one of the regions).

   Depending on the position of the chain with the exposed vertex and the alignment of its tail, we get four cases. In each case we can transform the sweep into one that includes only a single intersection. Fig. 4.12

shows all four cases and how each case can be transformed.

□

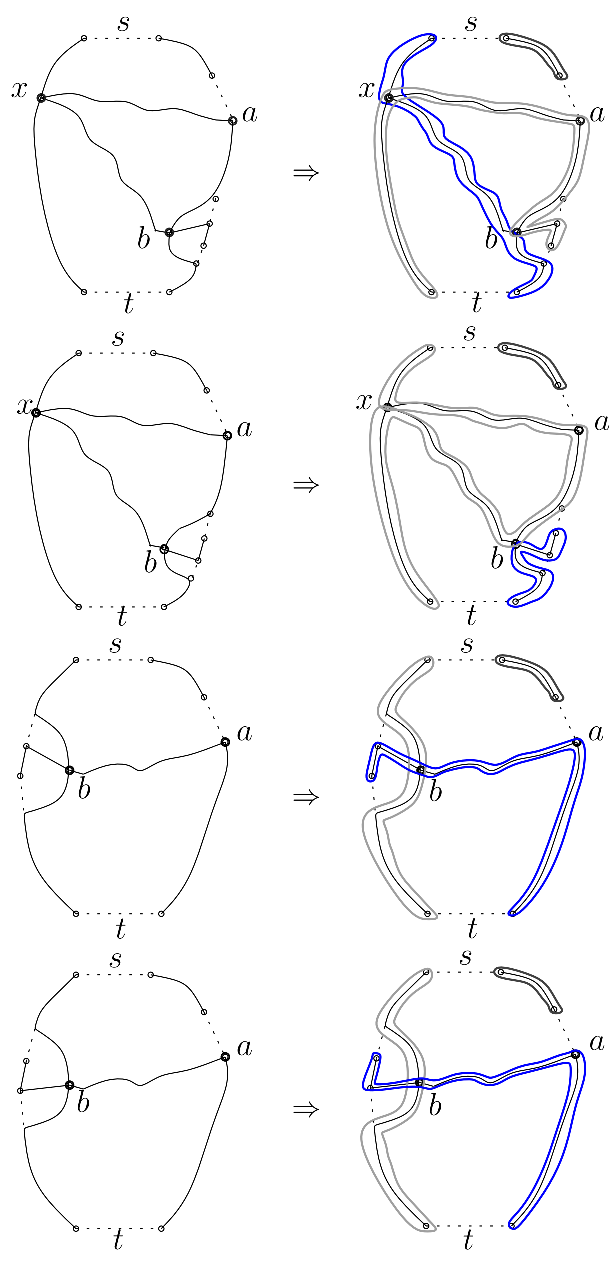

Figure 4.12: Different forms of a Case3 sweep, and its suitable transformation.

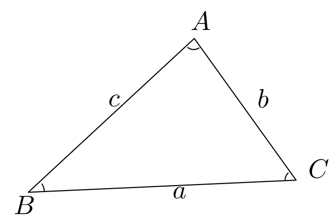

Figure 4.13: An arbitrary triangle ABC.

4.1 Simple Mathematical Lemmas

This section contains multiple geometric lemmas that we commonly use in later chapters in larger lemmas. While each of these proofs are easily explained, we thought that keeping them in another chapter would make the process of understanding larger arguments easier. You may skip this section if you are already familiar with these lemmas.
The following subsections belong to different classes of arguments and they can be read separately, while the lemmas within each subsection might be connected, and it is best to read them in order.

4.1.1 Simple Triangle Lemmas (Edge Length and Angle)

The following two lemmas refer to Figure 4.13

Lemma 3. *In an arbitrary triangle(fig 4.13), $a \geq b \times \sin A$.*

Proof. By the Law of Sines,

$$\frac{a}{\sin A} = \frac{b}{\sin B}$$

The lemma follows since $\sin B \leq 1$. $\square$

Lemma 4. $a \geq (b + c) \sin \dfrac{A}{2}$

Proof. From Lemma 3, we know that $a \geq b \sin A$ and $a \geq c \sin A$, so $2a \geq (b + c) \sin A$. Since sin is concave and $\sin 0 = 0$,

$$\frac{\sin A}{2} = \frac{\sin A + \sin 0}{2} \geq \sin \frac{A + 0}{2} = \sin \frac{A}{2} \qquad \text{(using Lemma 6)}$$

$\square$

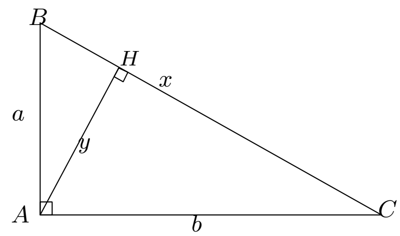

Figure 4.14: When $A = \pi/2$, $a + b < x + y$.

Lemma 5. *In a triangle ABC with $A \geq \pi/2$ (as shown in Fig 4.15), the sum of the lengths of a and b is less than sum of length of x plus the length of any chord y that connects A to x, i.e. $a + b < x + y$.*

Proof. First we will prove our claim for $A = \pi/2$ (Fig 4.14) and then we will expand the results to include $A > \pi/2$. Make y as short as possible, then y would be perpendicular to x, dividing it into x_1 and x_2. Now, triangles ABH and AHC are similar to each other, and also similar to ABC. Using this similarity between ABH and ABC, we get:

$$
\begin{array}{llcll}
 & \text{area}(ABC) & = & \dfrac{ab}{2} = \dfrac{xy}{2} & \\
\text{implies} & ab & = & xy. & \\
\text{Since} & a^2 + b^2 & = & x^2 & \\
\text{we have} & a^2 + b^2 + 2ab & = & x^2 + 2xy & \\
\text{which implies} & (x+y)^2 & > & (a+b)^2 & \text{since } y > 0 \\
\text{thus} & x + y & > & a + b. &
\end{array}
$$

The inequality would hold if y was greater than the altitude, i.e, for any line y connecting A to BC.

To extend the results to $A > \pi/2$ we convert our triangle ABC to a right triangle $AB'C$, see Fig. 4.15.
Draw the line AB' from A to BC in a way that the angle CAB' is a right angle. Now in the right triangle $AB'C$, we know that $a' + b < x' + y$.
Both B and B' are acute angles, and by using triangular inequality in ABB' we have: $a \leq a' + |BB'|$, or $a - a' \leq |BB'|$
Now by adding this inequality to the previous one, we get: $a + b < x' +$

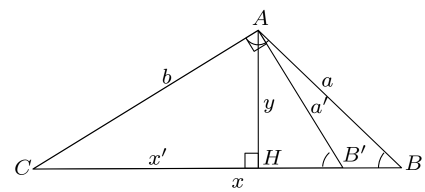

Figure 4.15: Extending the results to an obtuse angle.

$|BB'| + y$ and $x = x' + |BB'|$ therefore:

$$a + b < x + y$$

$\square$

4.1.2 Majorization

For a sequence $x \in \mathbb{R}^n$, let $x_{[i]}$ be the i^{th} largest number in x.

Definition. A vector x is said to be *majorized by y* if:

$$x \prec y \qquad \text{if} \quad \begin{cases} \sum_{i=1}^{k} x_{[i]} \leq \sum_{i=1}^{k} y_{[i]}, & k = 1, ..., n-1 \\ \sum_{i=1}^{n} x_{[i]} = \sum_{i=1}^{n} y_{[i]} \end{cases}$$

This notation was introduced by Hardy, Littlewood, and Pólya(1934, 1952).

This order is very useful in finding inequalities in many branches of mathematics. Many previously proven inequalities have been found to have secondary proofs based on majorization.
A function is Schur-convex, if its value is maximized when the inputs are majorized, i.e.,

$$f : \mathbb{R}^n \to \mathbb{R}^n, \quad f \text{ is Schur-convex, if } f(x) \leq f(y) \text{ for all } \qquad x \prec y$$

For a Schur-concave function, the direction of the inequality is reversed, i.e.,

$$f : \mathbb{R}^n \to \mathbb{R}^n, \quad f \text{ is Schur-concave, if } f(y) \leq f(x) \text{ for all } \qquad x \prec y$$

Lemma 6. *If $0 \leq x_i \leq \pi/2$ for all $i = 1, 2, ...n$, and $\sum_{1 \leq i \leq n} x_i = k\pi/2$ then $\sin x_1 + \sin x_2 + ... + \sin x_n \geq k$ (for fixed constant k).*

Proof. If $0 \leq x_i \leq \pi/2$ then $\sin x_1 + \sin x_2 + ... + \sin x_n$ is Schur-concave. Since, $(x_1, x_2, ..., x_n) \prec (\pi/2, ..., \pi/2, 0, ..., 0)$ the lemma follows.[11, Table 2, Section $3.B.2$] $\square$

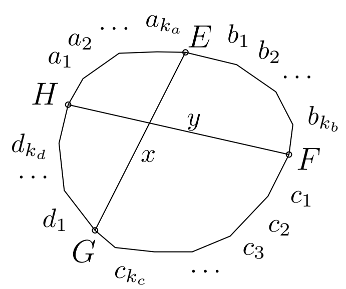

Figure 4.16: A convex polygon with a single intersection.

4.2 Single Intersection Case

In this case, we study a boundary cover which contains a single intersection. Two of the guards cross the polygon in a way that their paths intersect each other, and the third guard remains on it chain and does not participate in the intersection. The intersection divides the polygon into four quadrants. We will show that for every boundary cover T with a single intersection there exists a boundary cover T' with no intersection such that $|T| \geq |T'|$

When there is only one intersection in a boundary cover, the number of chains that can be covered is no more than five. In order to prove our claim we will show that the length of this boundary cover is greater than the length of a boundary cover with three chains, since we already showed that a boundary cover with three chains can be covered by three guards without any interior paths. Thus an alternative non-intersecting boundary cover can be presented with a shorter length, if and only if there are two quadrants, say A and B, each containing a free edge, $a \in A$ and $b \in B$, such that $a + b \leq x + y$, where x and y are the portions of the paths of the guards which lie on the interior of P and shape this intersection (see Fig. 4.16). In other words, we claim $x + y - a - b \geq 0$. By removing these free edges we reduce the number of chains by two and create a boundary cover with no more than three chains.

Definitions:
Let P be a convex polygon with an intersection (see Fig. 4.16).
Let x be segment EG and y be segment FH.

$$\begin{aligned} \text{Let,} \quad & a = \max\{a_1, a_2, \cdots, a_{k_a}\} \\ & b = \max\{b_1, b_2, \cdots, b_{k_b}\} \\ & c = \max\{c_1, c_2, \cdots, c_{k_c}\} \\ \text{and} \quad & d = \max\{d_1, d_2, \cdots, d_{k_d}\} \end{aligned}$$

Let m_1 be the smallest and m_2 the second smallest of a, b, c, and d.

Theorem 1.

$$x + y \geq m_1 + m_2.$$

Proof. If our claim is false, then there is an example where $x+y-m_1-m_2 < 0$. Our strategy is to try and find this example, we start with a general boundary cover and perform a series of transformation on the boundary cover that would make our counter-example stronger i.e., these moves either decrease $x + y - m_1 - m_2$ or keep it unchanged. For this purpose we choose m_1 and m_2 to have their maximum geometrically possible value in their quadrants. m_1 and m_2 are the longest edges of their respective quadrants. Since m_1 and m_2 are the two smallest edges among a, b, c and d, we assume these edges are the longest edges of their quadrants.

Consider the four points on the polygon where the guards leave the boundary to walk on the interior of the polygon or arrive at the perimeter (in Fig. 4.17, they are E, F, G and H). Since the polygon is convex, drawing the tangency lines of the polygon at these four points, creates a quadrilateral which includes the polygon entirely; also the line that connect all adjacent pair of points(EF, FG, GH and HE), lie within the polygon. Fig. 4.17 displays the interior paths, the four quadrants, and the tangency lines.

From these two simple properties of convex polygons we can conclude that the edges of the four quadrants created by this intersection, are contained within their respective triangles shaped by the two tangency lines and the line that connects the points. For example all edges in quadrant A are placed inside $\triangle AHE$, thus $a \leq \max\{|AE|, |AH|, |HE|\}$. Therefore we define,

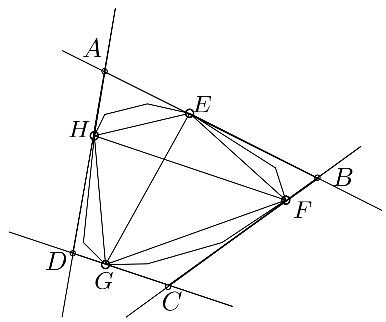

Figure 4.17: The quadrilateral made by drawing tangency lines, fully encompasses polygon P.

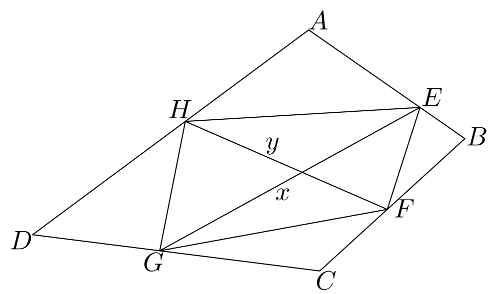

Figure 4.18: An arbitrary quadrilateral $ABCD$ and its two chords x and y.

$$
\begin{aligned}
a &= \max\{|AE|, |AH|, |HE|\}, \\
b &= \max\{|BE|, |BF|, |EF|\}, \\
c &= \max\{|CF|, |CG|, |FG|\}, \\
\text{and } d &= \max\{|DG|, |DH|, |HG|\}.
\end{aligned}
$$

With this definition we can simplify the convex polygon to a quadrilateral shape as in Fig. 4.18. Since $x + y > m_1 + m_2$ in quadrilateral $ABCD$, implies that $x + y > m_1 + m_2$ in polygon P.

We can simplify the quadrilateral further if the interior edge is the longest edge in each quadrant, i.e., if $|EH| \geq \max\{|AH|, |AE|\}$ we flat quadrant A, and make its two adjacent quadrants B and D larger, this transforms the quadrilateral into a triangle. Flattening a quadrant refers to replacing the triangle with the edge that forms the base of the triangle.

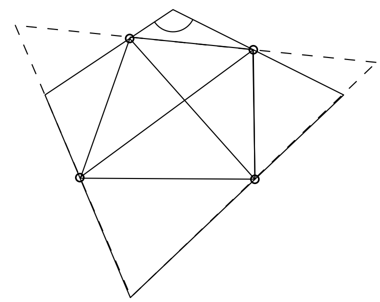

Figure 4.19: The quadrilateral can be transformed into a triangle by flattening one of its corners.

We assume that this condition does not happen in two opposite quadrants, since if $|EH| \geq \max\{|AH|, |AE|\}$ and $|FG| \geq \max\{|FC|, |CG|\}$, then $a+c \leq |EH|+|FG|$. By the triangle inequality we have, $|EH|+|FG| \leq |EG| + |HF| = x + y$, thus $m_1 + m_2 \leq a + c \leq x + y$.

Similarly if $|EF| \geq \max\{|BE|, |BF|\}$ and $|HG| \geq \max\{|DH|, |DG|\}$ then $m_1 + m_2 \leq x + y$.

Every quadrilateral has at least one right or obtuse angle. In a right or obtuse triangle, the longest edge is opposite the right/obtuse angle, thus we can always flat one quadrant. Fig. 4.19 shows the general shape of the resulting triangle.

If there are two adjacent quadrants that can be transformed, the result would be as Fig. 4.20. We claim that in the resulting triangles (either as Fig. 4.19 or as Fig. 4.20) the longest edges of the quadrants are on the boundary of the triangles.

It suffices to show that after the transformation none of the longest edges become shorter.

We prove that our transformation does not shorten a possible longest edge in the adjacent quadrants. We assume that in an adjacent quadrants the longest edge is on the boundary of the quadrilateral, otherwise we would transform the quadrant. This implies that the quadrilateral's angle in that region is acute.

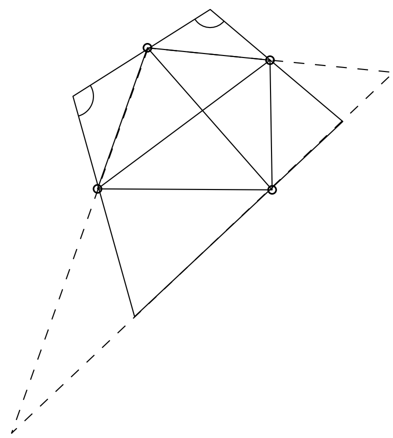

Figure 4.20: An alternative transformation when two adjacent quadrants have angles $\geq \pi/2$.

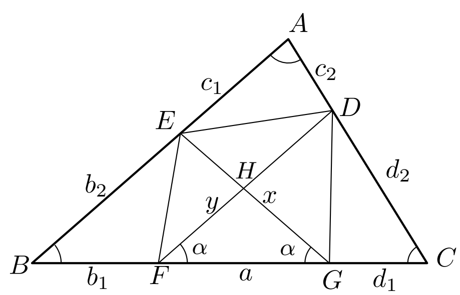

Figure 4.21: Type $\mathcal{A}$, general form of the transformed triangle.

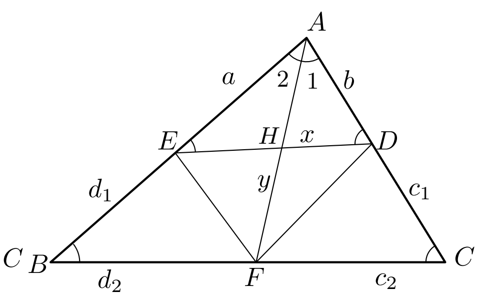

Figure 4.22: Type $\mathcal{B}$, alternate transformed triangle.

When a quadrant is transformed, the angle of its adjacent quadrants (if the quadrant is not transformed) become smaller. Since these angles have been acute prior to the transformation, the boundary edges grow longer.

Going back to our conditions, we can see that $x + y - m_1 - m_2$ has not increased, as x and y are unchanged, and none of the longest edges of the quadrants have become shorter. The quadrilateral $ABCD$ has been transformed into a triangle of type $\mathcal{A}$ in Fig. 4.21 or type $\mathcal{B}$ in Fig. 4.22. Since $x+y-m_1-m_2$ has not increased by the transformation, existence of counter example for the quadrilateral implies a counter example of type $\mathcal{A}$ or type $\mathcal{A}$.

In the rest of this section we will perform a series of transformations on the triangles, that will normalize their shapes, while keeping $x+y-m_1-m_2$ from increasing. Our goal is to turn the triangle into a shape where all (or most) of the long edges are of the same length. During these transformations the triangle may change its form (from type $\mathcal{A}$ to type $\mathcal{B}$ or vice versa), all the while becoming more balanced (the longest edges become closer in length). We will continue with these normalizations, until we either get a shape where the lengths of all longest edges are equal or we arrive at a state where none of our normalizing moves apply.

There are some cases where normalization no longer applies but the longest edges are not all equal either. We call these cases *special cases*, which we will study on a case by case basis.

4.2.1 Normalization Tools

There are three general transformations that we apply to the triangle whenever their conditions are satisfied. After each transformation the resulting shape is more balanced and $x+y-m_1-m_2$ either has decreased or remains unchanged.

We define $b = \max\{b_1, b_2\}$, $c = \max\{c_1, c_2\}$ and $d = \max\{d_1, d_2\}$. A segment (a, b, c or d) is called *critical* if it has length m_1 or m_2. During the normalization phase, we make sure that a critical segment's length is never decreased, since a decrease in a critical length will increase $x+y-m_1-m_2$.

The transformations are as follows:

1. Pivoting: pivoting a triangle edge about an endpoint of segment x or y. This is the only transformation that doesn't change the length of $x+y$ but changes the length of the triangle edge, e.g. we would never pivot AB about E (in triangle type $\mathcal{B}$) since this would change the length of y.

   The purpose of this move is to balance the lengths of the segments with each other such that it increases the lengths of critical edges while decreasing the lengths of the other segments.

2. Moving: Moving an end-point of x or y along an edge of the triangle, in order to decrease $x+y-m_1-m_2$. This move changes the length

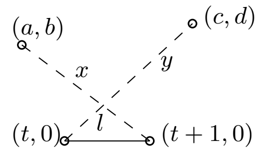

Figure 4.23: Changing t would affect the lengths of x and y.

of x or y, lengths of some segments might also be affected by the move.

3. Sliding: the goal is to minimize $x+y$ by sliding FG (moving F and G by the same amount and in the same direction) on BC in Fig. 4.21. As a result we may assume either,

   (a) $\angle DFG = \angle EGF$ or

   (b) F or G becomes B or C respectively (i.e. we switch to type $\mathcal{B}$)

   Assume we have a line portion l of unit length. Two arbitrary points (a, b) and (c, d) are connected to the vertices of l via two intersecting lines (see Fig. 4.23). The lengths of the two connecting lines (x and y) can be expressed as functions of t, where t shows the position of l.

$$\begin{aligned} x &= \sqrt{(1+t-a)^2+b^2} \\ y &= \sqrt{(t-c)^2+d^2} \end{aligned}$$

   We would like to find the value of $a \leq t \leq c$ that minimizes $x+y$. Differentiating $x+y$ with respect to t, we find $x+y$ is minimized when

$$t = \frac{da - d + bc}{d + b}$$

   Replacing this value for t, we get $b/x = d/y$ which means the angles x and y make with the horizontal line are equal.

4.2.2 Normalization Algorithm

Transforming the convex polygon into a triangle lead to two triangles type$\mathcal{A}$ and type$\mathcal{B}$. We divide these two triangles into further cases based on the position of the longest edge in each of their quadrants, i.e., for type$\mathcal{A}$ we have,

$$
\begin{array}{ll}
\mathcal{A}1 & \begin{cases} d_1 \quad b_1 \quad c_1 \\ d_1 \quad b_1 \quad c_2 \end{cases} \\
\mathcal{A}2 & \begin{cases} d_1 \quad b_2 \quad c_1 \\ d_1 \quad b_2 \quad c_2 \\ d_2 \quad b_1 \quad c_1 \\ d_2 \quad b_1 \quad c_2 \end{cases} \\
\mathcal{A}3 & \begin{cases} d_2 \quad b_2 \quad c_1 \\ d_2 \quad b_2 \quad c_2 \end{cases}
\end{array}
$$

Similarly for type$\mathcal{B}$ we have,

$$
\begin{array}{ll}
\mathcal{B}1 & \begin{cases} \\ d_1 \quad c_1 \end{cases} \\
\mathcal{B}2 & \begin{cases} \\ d_1 \quad c_2 \end{cases} \\
\mathcal{B}3 & \begin{cases} \\ d_2 \quad c_1 \end{cases} \\
\mathcal{B}4 & \begin{cases} \\ d_2 \quad c_2 \end{cases}
\end{array}
$$

In the rest of this section we will explain the algorithm we use in order to normalize the triangle. We will explain how and when each of our normalization methods are used and what is the desired outcome after each step.

Lemma 7. *If $x + y < m_1 + m_2$ then the triangle ABC is acute.*

Proof.

- For type $\mathcal{A}$ triangle:
  - If $A \geq \pi/2$ then $|ED| \geq c$, by the triangle inequality, $x + y \geq a + |ED|$, thus $x + y \geq a + c \geq m_1 + m_2$ (a contradiction).
  - If $B \geq \pi/2$ then $\angle EFG > \pi/2$ and $x + y > a + |EF|$ (by Lemma 5). Also since $B \geq \pi/2$, $|EF| \geq b$ so $x + y > a + b \geq m_1 + m_2$ (a contradiction).
  - If $C \geq \pi/2$ a similar contradiction is obtained.
- For type $\mathcal{B}$ triangle:
  both B and C are acute by an argument similar to the one used to show $A < \pi/2$ for type $\mathcal{A}$ triangles.
  If $A \geq \pi/2$ (in type $\mathcal{B}$) then Lemma 5 implies $x+y > a+b \geq m_1+m_2$ (a contradiction). □

Lemma 8. *If triangle $\triangle ABC$ with $a \neq m_2$ exists where $x + y < m_1 + m_2$, then there exists $\triangle ABC$ with $x + y < m_1 + m_2$ and $a = m_2$.*

Proof.

1. For a type $\mathcal{A}$ triangle:
   (a) If $a > m_2$:
   We will move one of its endpoints towards the other in order to decrease both a and $x + y$ until $a = m_2$.
   (b) If $a = m_1$:
   If either $d = d_2$ or $b = b_2$ (Case$\mathcal{A}$2 or Case$\mathcal{A}$3), then we will increase a by moving G towards C or F towards B respectively. It is easy to see that the increase in a, is more than the possible increase in $x + y$ (triangle inequality), and therefore this transformation will decrease $x + y - m_1 - m_2$.
   We will continue with this transformation until either $a = m_2$ or the triangle becomes of type $\mathcal{B}$, in case of the latter we will continue the transformations as described below.
   The remaining case where $d = d_1$ and $b = b_1$ (Case$\mathcal{A}$1), is one

of the special cases. We can prove that $x + y > m_1 + m_2$ in this case, regardless of any other conditions.
We will not worry about this case here, since a more general case will be analyzed in details later.

2. For a type $\mathcal{B}$ triangle:

   (a) If $a \geq m_2$:
   In this situation we will move the vertex A down until $a = m_2$, and after the transformation the triangle may be of type$\mathcal{A}$.

   (b) If $a = m_1$:
   $b \geq m_2$, in which case we will change our labels, and now we will call the old b, a. The new a is greater than or equal to m_2 and we are in the previous case. □

Lemma 9. *If a triangle ABC of type $\mathcal{B}$ exists where $x + y < m_1 + m_2$, then there exists $\triangle ABC$ with $x + y < m_1 + m_2$ and $b \in \{m_1, m_2\}$.*

Proof. If b is not already one of the minimum pair, then it is greater than m_2. We can shorten b, by moving the endpoint of y down AC, therefore decreasing the length of y while keeping $m_1 + m_2$ unchanged. This transformation may only decrease $x + y - m_1 - m_2$. After this transformation the triangle might become of type $\mathcal{A}$. □

Step 1, Pivot the edges, to balance the lengths of b, c and d in a triangle of type $\mathcal{A}$ and c and d in type $\mathcal{B}$.
This move can be performed as long as the candidate edges of the two involved quadrants have different lengths, and at least one of them is not critical. It is clear that after the two involved segments become the same length we can not use this transformation anymore; we also can not use it if the two involved segments are both critical, even if they are not of the same length.

1. For a triangle of type $\mathcal{A}$:
   We will balance the lengths of b, c and d, by *pivoting* AB on E and AC on D, following the conditions of this transformation. If any of the three edges are not in $\{m_1, m_2\}$, then we can still pivot an edge and

balance its length with one of its adjacent edges. In this way we can propagate the length difference, until either all the lengths are equal, or they are all of critical length, i.e., we can continue the transformation until one of the following outcomes occur.

(a) $b = c = d$.

(b) $c = d = m_2$ and $b = m_1$.

(c) $b = c = m_2$ and $d = m_1$.

(d) $b = d = m_2$ and $c = m_1$.

(e) ABC becomes of type $\mathcal{B}$, where $b > c$.
In this case, we will continue on balancing the lengths of c and d, following the next steps.

2. For a triangle of type $\mathcal{B}$:
By pivoting BC with F as our pivot, we can normalize the lengths of c and d. The following cases are the possible outcomes of this transformation.

(a) $c = d = m_1 = m_2$.
Which implies $a = b = c = d$ (by Lemma 8 and Lemma 9).

(b) $c \neq d$ and $c, d \in \{m_1, m_2\}$.

(c) $c_1 = 0$ or $d_1 = 0$.
This case happens when BC becomes a continuation of EF or DF during the transformation, i.e., B lies on F or C is moved to D. In either case, using the triangular inequality leads to $x+y \geq a+c$ or $x + y \geq b + d$ and we know that $a + c, b + d \geq m_1 + m_2$, thus $x + y - m_1 + m_2 \geq 0$.

(d) $c = d > m_2$.
This case is one of our special cases, we will cover it in two parts, part1, where $c = c_2$ and $d = d_2$ and part2, where $c = c_1$ and $d = d_2$ or vice versa. We will show that assuming c and d are equal to m_2 will not change the structure of the proof.

After this step, if it has not been already proved that $x+y-m_1-m_2 \geq 0$, then we will see one of the following situations:

1. A type $\mathcal{A}$ triangle where the long edges of all four quadrants have a length in $\{m_1, m_2\}$.

2. A triangle of type $\mathcal{B}$ with $a, b \in \{m_1, m_2\}$ (from lemma 8,9 and step1).

Step 2, If ABC is of type $\mathcal{A}$ and $b = b_2$ and $d = d_2$, then *slide* a until either it reaches a vertex (changes to type $\mathcal{B}$) or $x + y$ reaches its minimum value.

If ABC is of type $\mathcal{B}$, then if either $c = c_2$ or $d = d_2$(or both), then slide a or b down their edges to minimize $x + y$, assuming that the other conditions are satisfied (e.g. for sliding a down AB, $A_2 < E$ and $d = d_2$).

After this transformation we will assume a triangle of type $\mathcal{A}$ with $b = b_2$ and $d = d_2$, has $|HF| = |HG|$, or a triangle of type $\mathcal{B}$ with $c = c_2$ or $d = d_2$ has $A_2 > E$ or $A_1 > D$, respectively.

4.2.3 Final and Special Cases

In this section we will study all possible outcomes of the normalization algorithm in detail. In each case we will consider one or a group of possible outcomes and prove that $x + y \geq m_1 + m_2$ in each case.

Case $\mathcal{A}$1 In this case $d = d_1$ and $b = b_1$. We will show that $m_1 + m_2 \leq x + y$, regardless of the exact lengths of b, d and a. First we use Lemma 3 on triangles BEG and FCD.

In $\triangle BEG$,

$$(b_1 + a)\sin(B) \leq x$$

Similarly in $\triangle FCD$,

$$(d_1 + a)\sin(C) \leq y$$

Together these two inequalities imply,

$$min\{(b + a), (d + a)\}(\sin(C) + \sin(B)) \leq x + y$$

Since $0 \leq B, C \leq \pi/2$, Lemma 6 on triangle ABC implies,

$$1 \leq \sin(C) + \sin(B) \leq 2$$

$$\text{Thus} \qquad m_1 + m_2 \leq min\{b + a, d + a\}(\sin(C) + \sin(B)) \leq x + y$$

Case $\mathcal{A}2$ In this case either $b = b_1$ and $d = d_2$, or $b = b_2$ and $d = d_1$. Without loss on generality we will assume $d = d_1$ and $b = b_2$:

$$\begin{array}{lll} & d = d_1 & \geq d_2 \\ \text{and} & a & = m_2. \\ \text{In } \triangle FCD: & |FD| + |DC| & \geq |FC| \\ \text{i.e.,} & y + d_2 & \geq d_1 + m_2, \\ \text{thus} & y & \geq m_2. \end{array}$$

In $\triangle BEG$, if $B \geq \pi/3$,

$$\begin{aligned} |EG| &\geq min\{|BE|, |BG|\} \\ x &\geq min\{b_2, a + b_1\} \geq m_1 \\ x + y &\geq m_1 + m_2 (\text{since } y \geq m_2) \end{aligned}$$

The other possibility is when $B < \pi/3$; we know $B + C > \pi/2$ (otherwise $A \geq \pi/2$). Using Lemma 3 in $\triangle FDC$ and Lemma 4 in $\triangle BGE$ we get:

$$\begin{array}{lrcl} \text{In } \triangle BGE: & \dfrac{x}{\sin(B/2)} & \geq & b_1 + b_2 + a > b_2 + a, \\ \quad \text{and in } \triangle FDC: & \dfrac{y}{\sin(C)} & \geq & d_1 + a, \\ \quad \text{thus} & x + y & \geq & \min\{a + b, a + d\}(\sin(B/2) + \sin(C)). \\ \text{Also} & B + C & > & \pi/2 \\ \quad \text{and} & \dfrac{\pi}{2} \geq & C, B & \geq 0 \\ \quad \text{imply,} & \sin B/2 + \sin C & \geq & \sin B/2 + \sin(\pi/2 - B). \end{array}$$

$\sin B/2 + \sin(\pi/2 - B)$ is minimized when B is at its smallest, i.e., $B = \pi/3$, thus, $\sin B/2 + \sin(\pi/2 - B) \geq 1$. Finally we get:

$$\begin{aligned} x + y &\geq \min\{a + b, a + d\} \\ x + y &\geq m_1 + m_2 \end{aligned}$$

Case $\mathcal{A}3$ In this case $b = b_2$ and $d = d_2$. By symmetry we can assume that $c = c_2$. When $b = b_2$ and $d = d_2$, we slide FG on BC. We get $|FH| = |GH|$, thus in $\triangle FHG$, $F = G$. We call this angle α.

By pivoting, the lengths of a, b, c and d have been normalized. Either they are all equal to $m_2 = m_1$ or one of them is m_1 and the other three are equal to m_2. Comparing B and C with α we get four possible cases:

- $B, C < \alpha$:
  Place point H' so that $DHEH'$ is a parallelogram and so $|DH| + |HE| = |DH'| + |H'E|$. Since $B, C < \alpha$, then A lies inside $\triangle DEH'$. It implies,

  $$|DH| + |HE| = \left|DH'\right| + \left|H'E\right| > |DA| + |AE| > m_1$$

  By the triangular inequality,

  $$|GH| + |HF| > |GF| = m_2.$$

  Thus, $x + y = |DH| + |HE| + |GH| + |HF| > m_1 + m_2$.

- $C < \alpha < B$:
  Similar to the previous case we draw the parallel lines and create parallelogram $HDH'E$, but unlike the previous case A lies outside $HDH'E$. $B > \alpha$ implies that in the current figure EH' would be to the right of AE, and intersects DA before reaching H'. Call the intersection point A'.

  $$\begin{aligned} \text{In } \triangle AEA', \quad \angle AEA' =& B - \alpha, \\ \text{and} \qquad\qquad B <& \quad \pi/2 \\ \text{thus,} \qquad\qquad B <& A + C \\ <& A + \alpha \\ \text{so} \qquad\qquad B - \alpha <& \quad A. \end{aligned}$$

  If $B - \alpha < A$, then AA' would be less than $A'E$.
  Also in $\triangle DA'H'$,

  $$\left|H'A'\right| + \left|H'D\right| > \left|DA'\right|$$

  , and we can conclude

  $$|HD| + |HE| > |DA|$$

thus,

$$x + y > m_1 + m_2$$

.

- $B < \alpha < C$ If $a = b = c = d$ then clearly $C \leq B$, since $|AB| \leq |AC|$. Similarly if $b = m_1$ or $c = m_1$, $C < B$. a can not be m_1, since we set the length of a to m_2 during normalization. The only remaining case has $d = m_1$ and b, c and a are m_2.

  Draw a line FD' from F parallel to AB. Using the similarity of $\triangle CFD'$ and $\triangle ABC$, we have $|CF| \,/\, |FB| = |CD'| \,/\, |D'A| > 1$. We can therefore conclude that D' is positioned on DA, since we know $|CD| = d < c = |AD|$. This implies $\angle B > \alpha$, which is a contradiction.

- $\alpha < B, C$:
  In $\triangle CFD$:
  $$\frac{d}{\sin\alpha} = \frac{y}{\sin C}.$$
  Similarly in $\triangle BEG$:
  $$\frac{b}{\sin\alpha} = \frac{x}{\sin B}.$$
  Thus, $x + y > b + d \geq m_1 + m_2$.

The rest of the section will contain possible outcomes of type $\mathcal{B}$.

Case $\mathcal{B}$1 Since $c = c_1$ and $d = d_1$, we can show that $x + y \geq m_1 + m_2$ regardless of the exact lengths of c, d, a and b.
Using Lemma 3 on $\triangle ABF$, $\triangle AFC$ and $\triangle AED$:

$$
\begin{array}{llcl}
\text{In } \triangle ABF: & y & \geq & (a+d_1)\sin B \\
\text{thus,} & y & \geq & (a+d)\sin B. \\
\text{In } \triangle AFC: & y & \geq & (b+c_1)\sin C \\
\text{thus,} & y & \geq & (b+c)\sin C. \\
\text{In } \triangle AED: & x & \geq & a\sin A, \\
\text{and} & x & \geq & b\sin A \\
\text{thus,} & 2x & \geq & (a+b)\sin A. \\
\text{Therefore} & 2(x+y) & \geq & (a+b)\sin A+(b+c)\sin C+(a+d)\sin B \\
 & & \geq & \min\{a+b,b+c,a+d\}(\sin A+\sin B+\sin C). \\
\text{by Lemma 6} & \sin A+\sin B+\sin C & \geq & 2. \\
\text{Also} & \min\{a+b,b+c,a+d\} & \geq & m_1+m_2 \\
\text{thus,} & x+y & \geq & m_1+m_2.
\end{array}
$$

Case $\mathcal{B}$2 We have $d=d_1$ and $c=c_2$. In triangle AHD, $A_1 \geq D$(otherwise we would have slid AD down AC to minimize $x+y$). Since, $A_1 \geq D$, then in triangle AED we have $A > D$ which implies $|ED| > |AE|$ or $x > a$.
In triangle ABF we have $|BF|+|AF| \geq |AB|$, i.e., $|BF|+y \geq a+d$. We already know that $|BF| = d_2 \leq d$, therefore $y \geq a$, thus $x+y \geq 2a \geq m_1+m_2$ (since $a = m_2$).

Case $\mathcal{B}$3 In this case $d=d_2$ and $c=c_1$. Similar to the previous argument we have $A_2 \geq E$ which leads to $x > b$.

1. If $c_2 \leq b$, then using triangular inequality in $\triangle AFC$ we find out $y+c_2 \geq c_1+b = c+b$, so $y \geq c$. Finally we can conclude that $x+y \geq b+c \geq m_1+m_2$.

2. And if $c_2 > b$ (only occurs if $b = m_1 < c$, since $c \leq b$, implies $c_2 < c < b$), then $|BC| \geq |AC|$, since after pivoting BC on F, either $c = d \geq m_2$, or $c \neq d$ and $c, d \in \{m_1, m_2\}$ ($d = m_1$ or $c = m_1$); the latter is not possible as we already assumed $b = m_1$.

$$|BC| \geq |AC| \text{ implies } A \geq B.$$

$$\begin{array}{ll} \text{In } AED: & (a+b)\sin\dfrac{A}{2} \leq x \quad \text{(Lemma 4).} \\ \text{In } ABF: & (a+d)\sin(\dfrac{B}{2}) \leq (a+d_1+d_2)\sin(\dfrac{B}{2}) \leq y \quad \text{(Lemma 4).} \\ \text{In } AFC: & (b+c)\sin C = (b+c_1)\sin C \leq y \quad \text{(Lemma 3).} \end{array}$$

We also know that $(m_1 + m_2) \leq \min\{a+b, a+d, b+c\}$, thus

$$(m_1+m_2)[2\sin(\frac{A}{2}) + \sin(\frac{B}{2}) + \sin C] \;\; \leq \;\; 2(x+y) \qquad (4.1)$$

Claim. $2\sin(\frac{A}{2}) + \sin(\frac{B}{2}) + \sin C \geq 2$

Proof. The following constraints exist on inequality 4.1: $0 \leq A, B, C < \pi/2$, $A + B + C = \pi$ and $A \geq B$.
We will prove that the minimum value for d, ($d = 2\sin(A/2) + \sin(B/2) + \sin C$) is greater than 2, and then use this result to show that $x + y \geq m_1 + m_2$.
Using $A + C = \pi - B$, we define: $X = \pi - B$. Assuming a fixed value for B, the minimum value for $2\sin(A/2) + \sin(B/2) + \sin C$ would occur when $2\sin(A/2) + \sin C$ is minimized. We can rewrite C as $X - A$. We want to minimize $f(A) = 2\sin(A/2) + \sin(X - A)$. We find the first and second derivatives of f:

$$\begin{aligned} f'(A) &= \cos(\frac{A}{2}) - \cos(X - A) \\ f''(A) &= -\frac{1}{2}\sin(\frac{A}{2}) - \sin(X - A) \end{aligned}$$

$f''(A)$ is negative, thus f has a bell shaped form, and the minimum values happen at the extremes,i.e., comparing different values of d when $A = X$, $A = \pi/2$ or $A = B$,

(a) $A = X$ and $A = \pi/2$

$$\begin{array}{rcl} A = X & \rightarrow & d = 2\sin((\pi - B)/2) + \sin(B/2) + \sin 0 \\ \text{with } X \leq \pi/2 & \rightarrow & B = \pi/2, \quad d = 3\sin(\pi/4) \approx 2.12(A = \pi/2) \end{array}$$

(b) $A = B$
$A = B \rightarrow d = 2\sin(B/2) + \sin(B/2) + \sin(\pi - 2B) = 3\sin(B/2) + \sin(2B)$
We have to find the value for B that minimizes d. Finding the second derivative of d, we realize that d is also minimized when B is at either its maximum or minimum value.

$$\begin{array}{rcl} d'' & = & -\frac{3}{4}\sin(\frac{B}{2}) - 4\sin 2B \leq 0 \\ \pi - 2B & \leq & \pi/2 \rightarrow B \geq \pi/4, \\ B & = & \pi/4 \rightarrow \quad d = 3\sin(\frac{\pi}{8}) + 1 \approx 2.15 \\ B & = & \pi/2 \rightarrow \quad d = 3\sin(\frac{\pi}{4}) + 0 \approx 2.12 \end{array}$$

The over all minimum greater than 2.12 since this value is obtained with $A = B = \pi/2$, which does not show a valid triangle. □

By the claim we know $x + y \geq m_1 + m_2$.

Case $\mathcal{B}4$ We will show that it is not possible to have a normalized type $\mathcal{B}$ triangle where $d = d_2$ and $c = c_2$.
In step 2, of the normalization algorithm, we slide a down AB if $d = d_2$ and angle $A_2 < E$. Similarly we slide b down AC if $c = c_2$ and $A_1 < D$.
We have already assumed that $d = d_2$ and $c = c_2$ in this case, so the only possible explanation is that, $A_1 \geq D$ and $A_2 \geq E$. In triangle ADE we have $A + D + E = \pi$, and $A = A_1 + A_2$. Therefore we have $A \geq E + D$ so $A \geq \pi/2$, which contradicts our first claim that ABC is acute. □

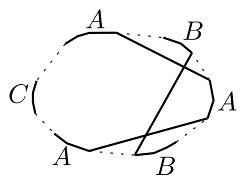

Figure 4.24: Double intersection case1.

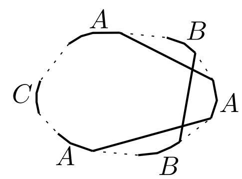

Figure 4.25: Double intersection case2.

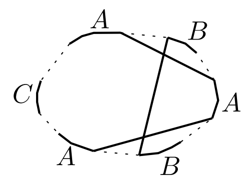

Figure 4.26: Double intersection case3.

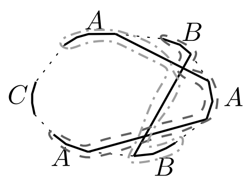

Figure 4.27: Case1 is simplified to a single intersection.

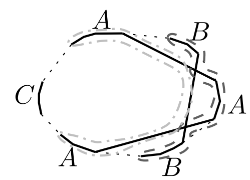

Figure 4.28: Case2 is simplified to a perimeter walk (no interior paths).

4.3 Double Intersection Case

In this case we study the possibility of having an intersection inside a boundary cover where the path of one guard intersects the path of another guard twice. The case consists of six chains, and three interior paths traversed by the guards such that one guard covers three of the chains, another guard traverses two chains and the third guard only walks on one chain.

In the double intersection case, the relative positions of the chains with respect to the interior paths makes for different methods. We define the following three cases based on the the alignment of the tails of the path of guard B in Fig. 4.24 to Fig. 4.26.

The first two cases can be reduced to previous cases,

- In case1, the sweep can be simplified to one intersection case, see Fig. 4.27.
- In case2, the sweep can be simplified to a perimeter walk (with no interior paths), see Fig. 4.28.

In the section 4.2, we proved that the length of a boundary cover with a single intersection, is no shorter than a boundary cover with three chains, which needs no interior paths. Thus the first two cases are both reduced to sweeps with no intersections. Fig. 4.29 shows the remaining case.

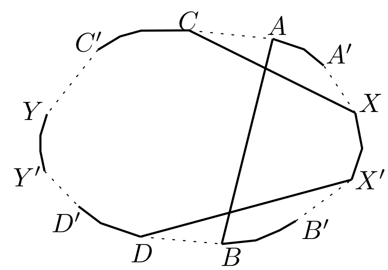

Figure 4.29: The remaining case of double intersection.

In the rest of this section, we will prove that the length of a paths of Fig. 4.29, is greater than or equal to a boundary cover with only three chains.

Theorem 2. *The length of the boundary cover T of polygon P as shown in Fig. 4.29, is greater than or equal to $\geq |P| - |\ell_1| - |\ell_2| - |\ell_3|$, where ℓ_1, ℓ_2 and ℓ_3 are the three longest edges of P.*

Proof. Proving that $T \geq |\partial P| -$ any three edges of P would immediately lead to our claim, and we know that the length of the interior paths is greater than $|CA| + |DB|$, thus we will assume that CA and DB are two of the three edges. We will choose the shortest of the four remaining free boundary segments as the third edge.
It will suffice to show

$$|AB| + |CX| + \left|DX'\right| \geq |CA| + |DB| + \lambda$$

where $\lambda = \min\{|C'Y|, |D'Y'|, |A'X|, |X'B'|\}$. Since then replacing the interior parts of the guards' paths with CA, DB and λ will be a shorter boundary cover.

First we will take our general shape through several steps of transformation, which will normalize the form of the polygon while making sure that $|AB| + |CX| + |DX'| - |CA| - |DB|$ does not increase, and similarly λ never becomes shorter. In other words, at each step we claim that if a counter example to our claim ($|AB| + |CX| + |DX'| > |CA| + |DB| + \lambda$) exists prior to this step, then one would exist with this condition.

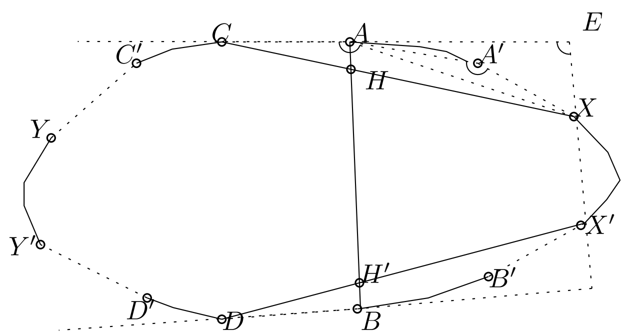

Figure 4.30: A polygon P where the extended lines of AC and DB diverge.

In Fig. 4.29 draw a line XX' and extend it in both directions. Extend CA on both sides, this line intersects XX' on the right side of the picture. Similarly extend DB. The following lemma shows that the extended lines of CA and DB intersect on the left side of the picture.

Lemma 10. *In Fig.4.30 extension of lines CA, DB, and XX' creates a triangle.*

Proof. In other words, we want to prove that CA, DB intersect one another on the left side of the picture, thus we will show that if they do not intersect, $|AB| + |CX| + |DX'| - |AC| - |BD| \geq \lambda$.

Assuming that the lines do not intersect on the left side, then the sum of their intersecting angles with $|XX'|$ is $\geq \pi$. This implies that at least one of them is $\geq \pi/2$. Without loss of generality we will assume that the extension of AC and XX' form an angle at E that is $\geq \pi/2$. Thus both A and A' would also be greater than or equal to $\pi/2$, as they are both greater than E in Fig. 4.30.
Since $A' \geq \pi/2$, $\quad |AX| > |A'X|$.
Since $A \geq \pi/2$, $|CX| + |AH| > |AC| + |AX|$ (Lemma 5). Therefore, $|CX| + |AH| > |AC| + |A'X|$.
Also by triangle inequality, $\quad |DH'| + |H'B| \geq |BD|$.
Thus,

$$|AH| + \left|H'B\right| + |CX| + \left|DH'\right| - |AC| - |BD| \geq \left|A'X\right| \geq \lambda$$

$\square$

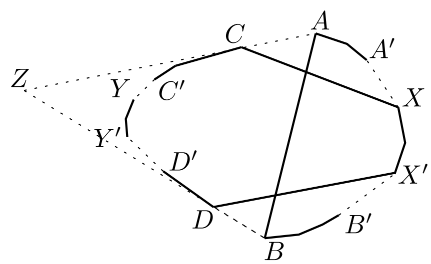

Figure 4.31: The general form of a counter-example.

From Lemma 10, we know that a counter-example must have CA and DB intersect to the left of polygon P, at a point we will call Z. So the counter-example, if one exists, appears as in Fig. 4.31.

1. Make C', C, A and A' co-linear.

   Extend CA and DB on both sides. From Lemma 10 we know that these lines intersect. Call the intersection point Z.
   We know that C', D', Y and Y' lie inside $\triangle ZCD$. Extend YC' until it intersects CZ, move C' to the intersection point. Since the polygon is convex, we know that the extended line would intersect line AC outside of the polygon (after vertex C), thus it will increase the length of YC'. It also might decrease $|CC'|$, but the decrease does not concern us since $|CC'|$ is not contributing in our inequality.
   Similarly by extending XA' we can move A' until it is co-linear with A, C and C'.

2. Make D', D, B and B' co-linear.

   This step is similar to the previous step, since by extending $X'B'$ and $Y'D'$ we can make B', B, D and D' co-linear.

3. Move Y and Y' to Z.

   Both Y and Y' reside within triangle $\triangle ZC'D'$. We claim that we can increase $|YC'| + |Y'D'|$ by moving Y and Y' to Z.
   First extend $C'Y$ and $D'Y'$ until they intersect one another, call the intersection Y_1. We have $|C'Y_1| > |C'Y|$ and $|D'Y_1| > |D'Y'|$.
   Then extend $C'Y_1$ to intersect $D'Z$ at Y_2 (see Fig. 4.32). It is clear that $|D'Y_1|$ is shorter than $|D'Y_2| + |Y_2Y_1|$. Similarly, $|Y_1C'| + |Y_2Y_1| <$

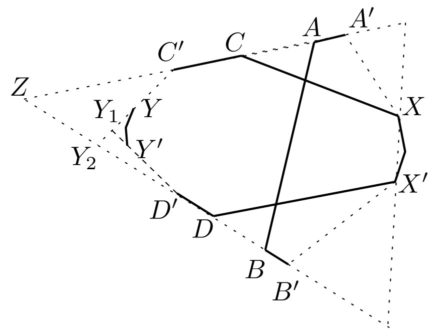

Figure 4.32: The general form of the counter example after the first two steps.

$|Y_2Z| + |C'Z|$, i.e.,

$$\begin{aligned} \left|YC'\right| + \left|Y'D'\right| &< \left|Y_1C'\right| + \left|D'Y_2\right| + |Y_2Y_1| \\ &< \left|D'Y_2\right| + |Y_2Z| + \left|C'Z\right|. \end{aligned}$$

We can conclude that,

$$\left|YC'\right| + \left|Y'D'\right| < \left|D'Z\right| + \left|C'Z\right|.$$

Also

$$\left|D'Z\right| + \left|C'Z\right| \geq 2\lambda.$$

Since $|YC'|$ and $|Y'D'|$ are both $\geq \lambda$. Move Y and Y' to Z. Thus,

$$\left|D'Y\right| + \left|C'Y\right| \geq 2\lambda. \tag{4.2}$$

4. $A = A'$ and $B = B'$.

   Move vertex A until it lies on A'. this will make CA longer, but it also increases AB. CA would get longer by $|AA'|$, but $|BA'| \leq |BA| + |AA'|$ (triangular inequality), thus we will not have an increase in $|AB| + |CX| + |DX'| - |AC| - |BD|$.
   A symmetric transformation will move B to B'.

5. Make A, X, X' and B co-linear.

Extend XX' on both sides until it intersects both YA and YB. We want to move A and B to their respective intersection points.
Call the intersection point of XX' and YA, E. We claim that moving A to E would make both $|CA|$ and $|XA|$ longer, also moving B to the intersection point of XX' and YB would increase both $DB|$ and $|X'B|$. In order to prove this claim we use the fact that $\angle YBX'$ and $\angle YAX$ are acute.

Lemma 11. *$\angle YBX'$ and $\angle YAX$ are both acute.*

Proof. The proof for this claim is very similar to Lemma 10. Assuming that $\angle YAX \geq \pi/2$, we can apply Lemma 5 to $\triangle YAX$. Thus,

$$
\begin{array}{lll}
 & |CX| + |AH| & > |CA| + |AX| \\
\text{and} & |DH'| + |H'B| & \geq |DB| . \\
\text{Also} & |AB| & \geq |AH| + |H'B| \\
\text{and} & |DX'| & \geq |DH'| . \\
\text{Thus} & |AB| + |CX| + |DX'| & \geq |AC| + |BD| + |AX| . \\
 & & \geq |AC| + |BD| + \lambda
\end{array}
$$

Similarly if $\angle yBX' \geq \pi/2$, then $|AB|+|CX|+|DX'| \geq |AC|+|BD|+\lambda$, thus $\angle YAX$ and $\angle YBX'$ are both acute. □

When $\angle YBX'$ and $\angle YAX$ are acute, moving A to E would lengthen both $|CA|$ and $|XA|$, since $|XE| > |XA|$. $|AB|$ would also get longer, but $|BE| < |AB|+|AE|$. Similarly we can argue that $|X'B'|$ and $|DB|$ would both get longer.

6. $C = C'$ and $D = D'$

This move is valid since moving C' to C and D' to D would make YC' and YD' longer.

After these transformations we have a triangle with vertices $A = A'$, $B = B'$ and $Y = Y'$ (see Fig. 4.33).

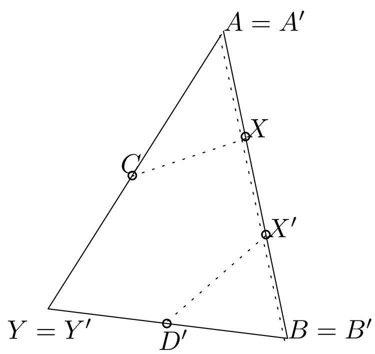

Figure 4.33: The outcome of the first seven steps of transformation on Fig. 4.33.

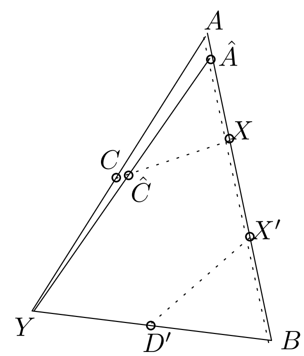

Figure 4.34: Choose $\hat{A}$ on AX such that $\left|\hat{A}X\right| = \lambda$.

7. $|AX| = |BX'| = \lambda$ and $\angle CXA, \angle DX'B \leq \pi/2$

   We claim that either,

   - $\angle CXA = \pi/2$ and $|AX| \geq \lambda$, or
   - $\angle CXA < \pi/2$ and $|AX| = \lambda$

   Since, if $\angle CXA > \pi/2$ then we can decrease it by moving X towards X', and if $\angle CXA < \pi/2$ while $|AX| > \lambda$ then we will decrease $|AX|$ by moving X towards A.

   The same is true for the other side of the triangle:

   - $\angle DX'B = \pi/2$ and $|BX'| \geq \lambda$, or
   - $\angle DX'B < \pi/2$ and $|BX'| = \lambda$

   We conclude that $|AX| > \lambda$ or $|BX'| > \lambda$ only if $\angle CXA = \pi/2$ or $\angle DX'B = \pi/2$ respectively.
   If $|AX| > \lambda$, select $\hat{A}$ on AX such that $\left|\hat{A}X\right| = \lambda$. Connect $\hat{A}$ to Y, intersecting XC at $\hat{C}$.

   - $\left|Y\hat{C}\right| > |YC|$, since $\angle YCX = \angle CXA + \angle CAX > \pi/2$

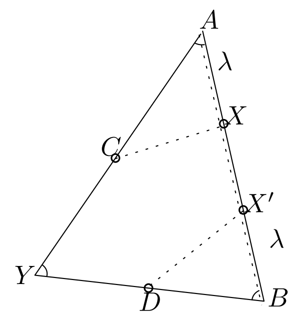

Figure 4.35: Normalization would transform the polygon into this triangle.

- $\left|\hat{C}X\right| = |CX| - \left|C\hat{C}\right|$
- $\left|X\hat{A}\right| = |XA| - \left|A\hat{A}\right|$
- $\left|\hat{C}\hat{A}\right| > |CA| - \left|C\hat{C}\right| - \left|A\hat{A}\right|$

Therefore $\left|\hat{A}B\right| + \left|\hat{C}X\right| + |DX'| - \left|\hat{C}\hat{A}\right| - |BD| < |AB| + |CX| + |DX'| - |AC| - |BD|$.
A similar transformation will be done if $|BX'| > \lambda$ in order to adjust its length, thus we will assume that $|BX'| = |AX| = \lambda$ and $\angle CXA, \angle DX'B \leq \pi/2$. After these seven steps we have a triangle as Fig. 4.35. In the rest of this section we will analyze our triangle and establish more facts regarding its shape which will eventually lead to the proof of our claim.

8. Assume without loss of generality that $A \geq B$.

9. $\angle CXA > A$, otherwise $|CA| \leq |CX|$, and $|XA| \geq \lambda$, thus $|CX| + |DX'| + |AB| - |AC| - |DB| \geq \lambda$.

10. Similarly $\angle DX'B > B$.

11. $A \geq \pi/3$.

Assume $A < \pi/3$. Since $B < A$ then $Y > \pi/3$. We claim that if $Y > \pi/3$, then sum of the projections of CY and YD on AB is

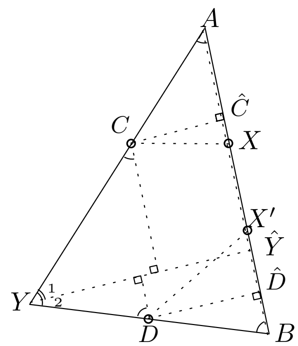

Figure 4.36: If $A < \pi/3$, $\left|\hat{C}\hat{D}\right| > \lambda$.

greater than λ.

We start by drawing perpendiculars from C and D to AB, calling the intersections $\hat{C}$ and $\hat{D}$ respectively. $\hat{C}\hat{D}$ is sum of the projections of CY and YD on AB. If $\left|\hat{C}\hat{D}\right| > \lambda$ then it is easy to see that,

$$\begin{array}{llcl} & \lambda & < & \left|\hat{C}\hat{D}\right| \\ \text{also} & |AC| & < & \left|C\hat{C}\right| + \left|A\hat{C}\right| \\ \text{and} & |DB| & < & \left|D\hat{D}\right| + \left|B\hat{D}\right|. \\ \text{Thus} & |DB| + |AC| + \lambda & < & |AB| + \left|C\hat{C}\right| + \left|D\hat{D}\right| \\ & & \leq & |AB| + |CX| + \left|DX'\right|. \end{array}$$

Lemma 12. *In $\triangle ABY$ of Fig. 4.36, if $A, B < \pi/3$ then* $\left|\hat{C}\hat{D}\right| > \lambda$

Proof. Let $\hat{Y}$ be the projection of Y onto AB. $\hat{C}\hat{Y}$ is the projection of CY on AB, and $\hat{D}\hat{Y}$ is the projection of DY on AB. It is clear that $\left|\hat{C}\hat{Y}\right| = \sin Y_1 |CY|$ and $\left|\hat{D}\hat{Y}\right| = \sin Y_2 |DY|$. Since A and B are both $< \pi/3$, then Y_1 and Y_2 are both $> \pi/6$, thus:

$$\sin Y_1 > \frac{1}{2} \quad \Rightarrow \left|\hat{C}\hat{Y}\right| > |CY|/2$$

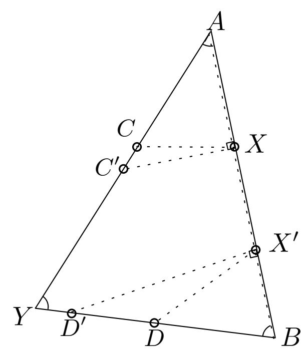

Figure 4.37: XC' and $X'D'$ are drawn perpendicular to AB.

and similarly:

$$\sin Y_2 > 1/2 \quad \Rightarrow \left|\hat{D}\hat{Y}\right| > |DY|/2$$

therefore, $\left|\hat{C}\hat{D}\right| > (|DY| + |CY|)/2 \geq \lambda$, since we already established that $|DY| + |CY| \geq 2\lambda$ (see equation 4.2). $\square$

12. $B < \pi/6$
We will show that if $B \geq \pi/6$ then $|XC| + |XA| - |CA| + |X'D| + |X'B| - |DB| > \lambda$, and therefore $|AB| + |CX| + |DX'| > |CA| + |DB| + \lambda$.
Draw XC' from X perpendicular to AB. $|C'A|$ is longer than $|CA|$ by $|CC'|$ (since $\angle AXC \leq \pi/2$), and $|XC'|$ is longer than $|XC|$ by less than $|CC'|$ (triangular inequality), thus $|XC'| - |C'A| \leq |XC| - |CA|$.

Similarly draw $X'D'$ from X' and we have $|X'D'| - |D'B| \leq |X'D| - |DB|$ (see Fig. 4.37).

We want to find $|XC| + |XA| - |CA| + |X'D| + |X'B| - |DB|$, therefore we will find the value of $|XC| + |XA| - |CA|$ in $\triangle AXC'$ and $|X'D| + |X'B| - |DB|$ in $\triangle BD'X'$.

Considering an arbitrary right triangle ABC, as in Fig. 4.38. We want to find a lower bound for $|AB| + |BC| - |AC|$. Define $f(\theta) = (\lambda(\sin\theta + \cos\theta - 1))/\cos\theta = |AB| + |BC| - |AC|$.

By increasing the angle θ both AC and BC become longer. Using the triangular inequality we can see that the increase in AC is less

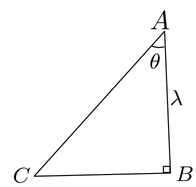

Figure 4.38: The relationship between edge lengths and θ in a right triangle.

than the increase in BC, therefore $f(\theta)$ is minimized when θ is at its minimum.

This leads to the conclusion that when applied to triangles $\triangle AXC'$ and $\triangle BD'X'$, $f(A) \geq f(\pi/3)$ and $f(B) \geq f(\pi/6)$. After calculating $f(\pi/3)$ and $f(\pi/6)$ we get:

$$\begin{array}{rrcl}
& f(\pi/3) & = & \dfrac{\lambda(\sin(\pi/3) + \cos(\pi/3) - 1)}{\cos(\pi/3)} \\
& & = & 2\lambda(\dfrac{1}{2} + \dfrac{\sqrt{3}}{2} - 1) \\
\text{and,} & f(\pi/6) & = & \dfrac{2}{\sqrt{3}}\lambda(\dfrac{1}{2} + \dfrac{\sqrt{3}}{2} - 1). \\
\text{Thus,} & f(\pi/3) + f(\pi/6) & = & \lambda(\dfrac{\sqrt{3} - 1 + 3 - \sqrt{3}}{\sqrt{3}}) \\
& & = & \lambda\dfrac{2}{\sqrt{3}} \\
& & > & \lambda
\end{array}$$

So we have shown that $|AB| + |CX| + |DX'| > |CA| + |DB| + \lambda$.

13. $|CA| > |CX|$

Assume $|CA| \leq |CX|$, then $|CA| + |AX| \leq |CX| + |AX|$ By the triangle inequality $|DB| \leq |DX'| + |X'B|$, thus $|CA| + |DB| + |AX| \leq |CX| + |DX'| + |AX| + |X'B| \leq |CX| + |DX'| + |AB|$. This indicates that,

$$|CX| + \left|DX'\right| + |AB| - |CA| - |DB| \geq |AX| \geq \lambda$$

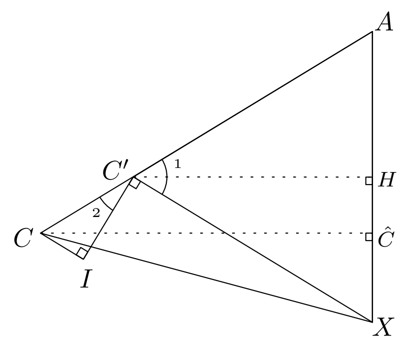

Figure 4.39: $|CX| + \left|A\hat{C}\right| - |CA| \geq \lambda/2$

14. $|CX| + \left|A\hat{C}\right| - |CA| \geq \lambda/2$

We show that when $A \geq \pi/3$ and $|CA| > |CX|$, then $|CX| + \left|A\hat{C}\right| - |CA| \geq \lambda/2$, where $\hat{C}$ is the projection of C on AX.

Choose C' on YA such that $|AC'| = |C'X|$. Clearly C' has to be on CA (Fig.4.39), otherwise $|CX|$ would be greater than $|CA|$ since any move to the right of C' will decrease $C'A$ more than $C'X$.

Define H as the projection of C' on AX. It is clear that $|AH| = \lambda/2$, since $\triangle AC'X$ is an isosceles triangle, and by fact 7, $|AX| = \lambda$. Thus $|C'X| + |AH| - |C'A| = \lambda/2$. Subtracting this equation from $|CX| + \left|A\hat{C}\right| - |CA| \geq \lambda/2$, we obtain the equivalent inequality $|CX| - |C'X| + \left|\hat{C}H\right| - |CC'| \geq 0$, which if shown, will establish fact 14.

Select I such that $\angle CIC' = \pi/2$ and IC' is perpendicular to XC'.

In Fig. 4.39, $|CX| - |C'X| \geq |CI|$ and $|CI| = |CC'|\sin(C'_2) = |CC'|\sin(2A - \pi/2)$. Also $\left|H\hat{C}\right| = |CC'|\sin((\pi - 2A)/2)$.

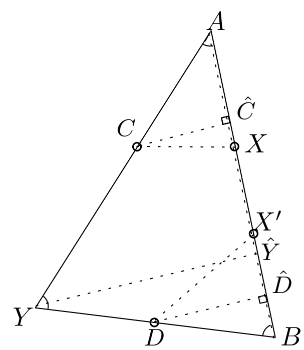

Figure 4.40: $\left|\hat{C}\hat{D}\right| > \lambda/2$.

Adding these equations together,

$$\begin{aligned}|CX| + \left|H\hat{C}\right| - \left|C'X\right| - \left|CC'\right| &\geq \left|CC'\right| (\sin(\pi/2 - A) + \sin(2A - \pi/2) - 1)\\ &= \left|CC'\right| (\cos A - \cos 2A - 1)\\ &= \left|CC'\right| \cos A(1 - 2\cos A)\\ &\geq 0\end{aligned}$$

Since $A > \pi/3$ implies $\cos A < 1/2$.

15. In Fig. 4.40 $\left|\hat{C}\hat{D}\right| > \lambda/2$

If $\left|\hat{C}\hat{D}\right|$ were less than $\lambda/2$, then $\left|\hat{C}B\right| \leq 3\lambda/2$ (since $\left|\hat{D}B\right| \leq \lambda$ by fact 7). Draw $Y\hat{Y}$ perpendicular to AB. $\left|Y\hat{Y}\right|/(3\lambda/2) < \left|Y\hat{Y}\right|/\left|\hat{C}B\right| < \left|Y\hat{Y}\right|/\left|B\hat{Y}\right| = \tan B < 1/\sqrt{3}$ (since $B < \pi/6$ by fact 12). Therefore

$$\left|Y\hat{Y}\right| < \lambda\sqrt{3}/2. \tag{4.3}$$

This upper-bound on $\left|Y\hat{Y}\right|$ poses a problem since $\left|C\hat{C}\right|$ would be longer than $\left|Y\hat{Y}\right|$, which is not possible. We know that $|AB| \geq 2\lambda$, which implies $\left|A\hat{C}\right| \geq \lambda/2$.

$$\left|C\hat{C}\right| = \left|A\hat{C}\right| \tan A > \lambda/2 \tan A > \lambda\sqrt{3}/2 \quad \text{(since } A > \pi/3 \text{ by fact 11)} \tag{4.4}$$

This contradiction (equations 4.3 and 4.4) implies $\left|\hat{C}\hat{D}\right| > \lambda/2$.

After these steps we have all the required information, thus

$$\begin{aligned}
&|AB| + |CX| + \left|DX'\right| - |BD| - |CA| \\
&\qquad = \left|A\hat{C}\right| + \left|\hat{C}\hat{D}\right| + \left|\hat{D}B\right| + |CX| + \left|DX'\right| - |BD| - |CA| \\
&\qquad > \left|A\hat{C}\right| + \lambda/2 + \left|\hat{D}B\right| + |CX| + \left|DX'\right| - |BD| - |CA| \quad \text{(fact 15)} \\
&\qquad > \lambda + \left|\hat{D}B\right| + \left|DX'\right| - |BD| \quad \text{(fact 14)} \\
&\qquad > \lambda + \left|\hat{D}B\right| + \left|D\hat{D}\right| - |BD| \quad \text{(since } \left|D\hat{D}\right| \perp |AB|\text{)} \\
&\qquad > \lambda \quad \text{(by triangle ineq.)}
\end{aligned}$$

□

Chapter 5

Conclusion

In this work, we studied the problem of finding the shortest path to cover a convex polygon with two or three mobile guards.

When there are two guards a polygon P is covered if every point in P lies on the line that connects the two guards, at some point in time. In the 3-guard case the covered area is defined as the triangle that the guards make, thus the guards move in a way that their triangle covers the surface of P.

In both cases we relaxed the problem to a boundary cover, where the objective is to cover the boundary of P. Finding a boundary cover is not harder than finding the optimal sweep, since in an optimal sweep the boundary is also covered.

In the two guards case, we proved that the shortest boundary cover is achieved when the guards stay on the polygon. The shortest boundary cover therefore is when the guards leave out the two longest edges of P and visit the other $n-2$ edges during the sweep. Since the polygon is convex this solution also provides coverage for the interior of the polygon, thus it is also a solution to the more general problem of finding the optimal sweep for two guards. This solution can be found in order $O(n)$ time, where n is the size of the polygon.

In the three guards case, we were able to show that the optimal boundary cover is simple, i.e., the paths of the guards do not intersect on the interior of P. We proved that any boundary cover which included intersection in the paths of the guards, could be simplified to one of two intersection cases, either a single intersection or a double intersection. Each of these cases where shown to be longer than an alternative simple boundary cover.

A simple three guard boundary cover has a maximum of four chains, and this makes the optimal boundary cover one of three certain shapes. All three shapes provide coverage for the interior of the polygon, thus the optimal boundary cover is the optimal three guard sweep.

5.1 Future Work

Our objective in this problem was to find the minimum total length of the paths of the guards. With this idea we might find the best solution consists of one guard that walks along the perimeter while the rest are stationary. This solution is of course not practical if we want to cover the polygon in minimum time.

If we assume that speed of each guard is limited, and we want to cover the polygon with minimum time, then the problem turns into finding the paths of the guards such that the longest path is minimized. This problem is not trivial even when two guards are covering a convex polygon.

Another interesting question arises when there are three or more guards. The purpose of area coverage was to cover a polygon such that every point in the polygon is visible from different directions. One might assume that a narrow triangle does not provide the same quality as an equilateral triangle. This assumption leads to the problem of finding the shortest paths for the guards such that their triangle does not have very large (e.g., $\geq 2\pi/3$) or very small angles (e.g., $\leq \pi/6$). A similar problem can also be defined when the guards have limited visibility, i.e., the edges of the covered triangle are not longer than a certain distance.

Also the problem can be considered when there are more than three guards. When there are more than three guards, the problem of finding minimum length sweeps becomes more complicated. Since the definition of covered area needs further speculation, as the guards are now able to form shapes that are not convex.

Bibliography

[1] Svante Carlsson, Bengt J. Nilsson, and Simeon Ntafos. Optimum guard covers and m-watchmen routes for restricted polygons, 1993.

[2] Wei-Pang Chin. Optimum watchman routes. *Information Processing Letters*, 8:39–44, 1988.

[3] Wei-Pang Chin and Simeon Ntafos. Optimum watchman routes. In *Annual Symposium on Computational Geometry*, pages 24–33, 1986.

[4] Wei-Pang Chin and Simeon Ntafos. Shortest watchman routes in simple polygons. *Discrete Computational Geometry*, 6:9–31, 1991.

[5] Jurek Czyzowicz, Peter Egyed, Hazel Everett, David Rappaport, Thomas Shermer, Diane Souvaine, Godfried Toussaint, and Jorge Urrutia. The aquarium keeper's problem. In *SODA '91: Proceedings of the second annual ACM-SIAM symposium on Discrete algorithms*, pages 459–464, Philadelphia, PA, USA, 1991. Society for Industrial and Applied Mathematics.

[6] Moshe Dror, Alon Efrat, Anna Lubiw, and Joseph S.B. Mitchell. Touring a sequence of polygons. In *STOC'03*, pages 473–482, 2003.

[7] Paul J. Heffernan. An optimal algorithm for the two-guard problem. In *SCG '93: Proceedings of the ninth annual symposium on Computational geometry*, pages 348–358, New York, NY, USA, 1993. ACM.

[8] Christian Icking and Rolf Klein. The two guards problem. In *SCG '91: Proceedings of the seventh annual symposium on Computational geometry*, pages 166–175, New York, NY, USA, 1991. ACM.

[9] D.T. Lee and Arthur K.Lin. Computational complexity of art gallery problems. *IEEE Transaction on Information Theory*, 32:276–282, 1986.

[10] Jae-Ha Lee, Sang-Min Park, and Kyung-Yong Chwa. Searching a polygonal room with one door by a 1-searcher. *International Journal of Computational Geometry and Applications*, 10(2):201–220, 2000.

[11] Albert W. Marshall and Ingram Olkin. *Inequalities: theory of Majorization and Its Applications.* Academic Press INC., New York, first edition, 1979.

[12] Bengt J. Nilsson and Derick Wood. Optimum watchmen routes in spiral polygons. In *Proceeding of 2nd Conference on Computational Geometry*, page 269272, 1990.

[13] S. Ntafos. The robber route problem. *Inf. Process. Lett.*, 34(2):59–63, 1990.

[14] Simeon Ntafos. Watchman routes under limited visibility. *Comput. Geom. Theory Appl.*, 1(3):149–170, 1992.

[15] Simeon Ntafos and Laxmi Gewali. External watchman routes. *The Visual Computer*, 10(8):474–483, 1994.

[16] Joseph O'Rourke. *Art Gallery Theorems and Algorithms.* Oxford University Press, New York, US, first edition, 1987.

[17] Thomas C. Shermer. Recent results in art galleries. In *In Proceedings of the IEEE*, volume 80(9), pages 1384–1399, 1992.

[18] Ichiro Suzuki and Masafumi Yamashita. Searching for a mobile intruder in a polygonal region. *Society for Industrial and Applied Mathmatics*, 20(5):863–888,, 1992.

[19] Xuehou Tan. An efficient algorithm for the three-guard problem. *Discrete Applied Mathmatics*, 156(17):3312–3324, 2008.

[20] L. H. Tseng, Paul J. Heffernan, and D. T. Lee. Two-guard walkability of simple polygons. *International Journal of Computational Geometry and Applications*, 8(1):85–116, 1998.

[21] J Urrutia. *Handbook of Computational Geometry.* North Holland, 2000.

[22] Zhong Zhang. *Applications of Visibility Space in Polygon Search Problems.* PhD thesis, Simon Fraser University, 2005.