Doctoral Thesis

Discrete Descriptions of Geometric Objects

Author[s]:
Christ, Tobias

Publication Date:
2011

Permanent Link:
https://doi.org/10.3929/ethz-a-007018092

Rights / License:
In Copyright - Non-Commercial Use Permitted
Discrete Descriptions of Geometric Objects

A dissertation submitted to
ETH ZURICH

for the degree of
DOCTOR OF SCIENCES

presented by
JAKOB ALFRED WERNER TOBIAS CHRIST
Master of Science in Mathematics, University of Basel
born May 8, 1981
citizen of Basel, Switzerland, and Germany

accepted on the recommendation of
Prof. Dr. Emo Welzl, examiner
Dr. Michael Hoffmann, co-examiner
Prof. Dr. Joseph O’Rourke, co-examiner

2011
Contents

Abstract v

Zusammenfassung vii

Acknowledgements x

1 Introduction 1
  1.1 From Art Galleries to Internet Cafes 1
  1.2 From Internet Cafes to Constructive Solid Geometry 4
  1.3 From Solids to Detecting Cancer: Consistent Digital Line Segments 6
  1.4 Definitions and Notations 10

2 CSG-Representations of Polygons and Polyhedra 15
  2.1 Regularized Operations 15
  2.2 Description, Cover and Partition 16
  2.3 Basic Observations 18
  2.4 Results Fitting into the Framework 22
  2.5 Describing a Polygon by Convex Pieces 24

3 The Wireless Localization Problem 29
  3.1 Notation and Basic Properties 31
  3.2 Upper Bounds 32
    3.2.1 Polygonal Halfplanes 32
    3.2.2 Natural Guards 34
    3.2.3 Vertex Guards 36
    3.2.4 Guarding Chains for small n 47
    3.2.5 Combinatorially Different Line Arrangements 53
    3.2.6 General guards 56
  3.3 Lower Bounds 62
    3.3.1 The Blunting Comb 63
    3.3.2 Natural Guards 65
    3.3.3 General Guards 67
Abstract

The core of this thesis is the wireless localization or internet café problem, a relatively new problem which evolved within the field of art gallery problems: An internet café wants to provide wireless internet access for its customers. How can they prevent customers in the competitor’s café next door from using their internet connection? The customers should have the possibility to prove that they are inside while anybody outside should not be able to provide such a proof. We place point stations (usually called guards) that broadcast a unique signal within an angular range. The signals are not blocked by the walls of the café. The goal is to place the guards and adjust their angular range in such a way that a customer in the café can prove to be inside by naming the keys it receives whereas anyone outside of the café cannot provide such a proof.

This can be phrased as a constructive solid geometry (CSG) problem: Describe a simple polygon by a formula over primitive objects—in our case the cones representing the area of broadcast of the guards—using the set operators union and intersection. Instead of asking for a description using cones one can try to use other primitives such as triangles. More generally, we are interested in simple and compact descriptions of polygons and polyhedra using different classes of primitives. Decomposing polygons into primitive objects is an important problem in computational geometry. Besides direct applications, for example in pattern recognition or VLSI design, decomposition is often the first step of geometric algorithms: A polygon is first decomposed into simple pieces, then the problem is solved for each piece individually, and then the partial solutions are combined to give a solution for the original polygon.

How can a complex geometric object be described by simpler objects? We provide a general framework in order to give this question a mathematical meaning. Depending on which objects and which primitives are used, and what operations are allowed to combine the primitives, we get a rich variety of problems that are precise variants of the informal quest for “nice descriptions”. Then, we turn our attention to specific variants. It is well-known that \( n/3 \) guards are sufficient and sometimes necessary to watch an art gallery with \( n \) walls. Similarly, we try to bound the number of guards needed for wireless localization. On one hand, \( 3n/4 \) guards are sufficient to describe any simple polygon with \( n \) non-parallel edges. On the other hand, there is a class of simple
Abstract

polygons that cannot be described by fewer than roughly $2n/3$ guards.

From a complexity theory point of view, most of the problems turn out to be computationally hard when it comes to finding a description using a minimum number of primitives. Beside the wireless localization problem, we investigate the computational complexity of covering a polygon with a minimum number of triangles, which turns out to be NP-hard. It is not clear whether the problem is in NP and hence NP-complete. If we want to cover the boundary of a polygon only, still using triangles that are inside the polygon, the problem is not only NP-hard but, interestingly, it can also shown to be in NP.

Moving on to spatial geometry, we define the wireless localization problem for 3-regular orthogonal polyhedra. It is shown that a result for planar orthogonal polygons generalizes to 3D: Putting a vertex guard onto every other vertex suffices to describe any orthogonal polyhedron. The graph induced by a 3-regular orthogonal polyhedron is always bipartite, regardless of the topology of the polyhedron. Therefore it makes sense to say “every other vertex”. It is easy to construct examples where we need at least $n/4$ guards, $n$ denoting the number of vertices of the polyhedron. One might suspect that this lower bound is tight. However, this question remains open. As a first step we show that $3n/8$ guards always suffice.

The final part of the thesis is devoted to the study of consistent digital line segments. How are line segments drawn on a computer screen? The problem reduces to defining paths on the integer grid—called digital line segments—that come close to the Euclidean line segment between any two grid points. If by “close” we just mean that they should look like their Euclidean counterparts seen from far away, this is an old and well-studied problem. But the way line segments are usually digitized, the resulting digital segments do not behave like line segments: For example, whereas two non-parallel line segments have at most one point in common, two digital line segments can intersect in a very degenerate way. We study the problem of defining digital line segments in such a way that they also interact like line segments in a more mathematical sense, fulfilling a list of axioms that arise naturally from the axioms of Euclidean geometry. In particular, the intersection of two digital line segments should either be empty or a common subsegment. We construct such a system of digital line segments in the plane that are also good approximations of the corresponding Euclidean line segments, thus rediscovering a construction proposed by Michael Luby. Furthermore, we address the analogue problem in higher dimensions. The question whether such a system of digital line segments in 3D exists remains open.
Zusammenfassung

Kern dieser Arbeit ist das Problem der drahtlosen Lokalisierung, auch als Internetcaféproblem bekannt. Es ist ein vergleichsweise neues Problem, das aus dem Bereich der Kunstgalerieprobleme hervorgegangen ist: Ein Internetcafé will seinen Kunden drahtlosen Internetzugang zur Verfügung stellen. Wie jedoch können die Betreiber verhindern, dass die Leute im Café nebenan diese Verbindung missbräuchlich benutzen? Die eigenen Kunden sollen belegen können, dass sie sich innerhalb des Cafés aufhalten, während dies für jedermann ausserhalb des Cafés unmöglich sein soll. Wir plazieren Sendestationen, in Anlehnung an das Kunstgalerieproblem auch Wärter genannt, welche innerhalb eines bestimmten Winkels ein eindeutiges Signal aussenden. Diese Signale dringen durch die Wände des Cafés hindurch. Wir wollen die Wärter so aufstellen und ihren Sendewinkel so einstellen, dass ein Kunde beweisen kann, sich innerhalb des Cafés aufzuhalten, indes er alle Schlüssel aufzählt, die er über die Wärtersignale empfangen hat, während ausserhalb des Cafés sich niemand als Kunde ausgeben kann, da er nicht genügend Schlüssel erhält.


Wie kann man ein geometrisches Objekt durch einfachere primitive Objekte prägnant beschreiben? Wir führen ein allgemeines Begriffsräusserst ein, um dieser Frage einen präzisen mathematischen Sinn zu verleihen. Je nach Art der Objekte, die erlaubten Primitiven und der Möglichkeiten, diese zu kombinieren,
Zusammenfassung

erhalten wir eine Vielfalt von Problemen. Jedes Problem ist eine Instanz des allgemeineren Problems der “prägnanten Beschreibung”.

Wir betrachten spezifische Varianten dieses Problems. Es ist bekannt, dass man eine Kunstgalerie mit \( n \) Wänden immer mit \( n/3 \) Wärtern überwachen kann, und dass man in gewissen Fällen auch mindestens \( n/3 \) Wärter braucht. In diesem Geiste trachten wir danach, für das Internetcaféproblem die Anzahl der benötigten Wärter zu beschränken. Wir zeigen, dass man ein Vieleck mit \( n \) nichtparallelen Kanten durch \( 3n/4 \) Wärter beschreiben kann. Andererseits gibt es Fälle, in denen wir mindestens um die \( 2n/3 \) Wärter brauchen.

Wenn es darum geht, Objekte durch eine kleinmögliche Anzahl von Primitiven zu beschreiben, entpuppen sich die meisten Probleme als algorithmisch schwierig. Neben dem Internetcaféproblem betrachten wir das Problem, ein Vieleck mit möglichst wenigen Dreiecken zu überdecken (nicht notwendigerweise in solche zu zerlegen!). Dies ist NP-schwierig, wie wir sehen werden. Es ist jedoch nicht klar, ob dieses Problem auch in NP enthalten ist. Wenn wir nur den Rand des Vielecks überdecken wollen – mit Dreiecken, die vollständig in dem Vieleck enthalten sind – dieses Problem, das können wir zeigen, ist in NP.


Im letzten Teil dieser Arbeit untersuchen wir **stimmige digitale Geradensegmente**. Ausgangspunkt ist die unschuldige Frage: Wie zeichnen wir Geradensegmente auf einem Rechnerbildschirm? Das läuft auf das Problem hinaus, im ganzzahlichen Gitter Pfade zu definieren, sogenannte digitale Geradensegmente, die annähernd so verlaufen wie Geradensegmente im euklidischen Sinne. Wenn wir nur verlangen, dass die digitalen Geradensegmente von fern betrachtet wie euklidische Geradensegmente aussehen, dann haben wir es mit einem alten und guterforschten Problem zu tun. So wie Geradensegmente jedoch üblicherweise digitalisiert werden, verhalten sich die digitalen Segmente nicht, wie wir es erwarten. Beispielsweise schneiden sich euklidische Segmente in höchstens einem Punkt. Der Schnitt zweier digitaler Segmente jedoch kann sehr entartet ausse-
Acknowledgements

First of all, my gratitude goes to my thesis advisor, Michael Hoffmann, who has been a constant source of guidance, insight and motivation. Not only is he the co-author of several papers this thesis is based on, but in a very deep way his ideas and suggestions permeate the fabric of all my work. I thank him for his reliable and endless support.

I am also in the debt of Emo Welzl, whom I had the honor to work with as a member of his research group. He was there for me whenever I needed advice. I am sure he provides the best working atmosphere ever thinkable.

My sincerest thanks go to Joseph O’Rourke, not only for his intellectual inspiration, but also for the many hours spent in reviewing this thesis. His remarks have been invaluable, and improved my work greatly. In this same vein, I would like to acknowledge Jack Snoeyink for valuable suggestions and comments.

I would also like to recognize the efforts of my co-authors; Aurosish Mishra, Dömötör Pálvölgyi, Miloš Stojaković, Takeaki Uno and Yoshio Okamoto, without whom this work would not have been possible.

My sincerest thanks goes out to all members of Gremo both present; Andrea Francke, Andrea Salow, Anna Gundert, Bernd Gärtner, Emo Welzl, Gabriel Nivasch, Heidi Gebauer, Martin Jaggi, Michael Hoffmann, Sebastian Stich, Robin Moser, Timon Hertli, Uli Wagner, and Yves Brise, and past; Robert Berke, Franziska Hefti, Dieter Mitsche, Andreas Razen, Leo Rüst, Dominik Scheder, Eva Schuberth, Marek Súlovský, Tibor Szabó, Patrick Traxler, Floris Tschurr, and Philipp Zumstein. In particular, I want to thank Andrea, Anna, Dominik, Heidi, Marek, Martin, Timon, and Yves for their very careful proof-reading.

My work has also been enriched by the students who wrote a masters or bachelors thesis under my supervision: Dan Bühler and David Tschirky.

During the course of graduate studies I had the unique pleasure of sharing an office space with my Mitbürner Dominik Alban Scheder. His friendship has been a welcome respite from the daily difficulties of the academic life, and whether it be speaking in German or Chinese, the sound of his voice will be sorely missed. He has provided the German translation of the abstract.

Some of the work was conducted while staying abroad, and being a man of
simple means, I am thankful for the hospitality of my dear friends: William J. Drury III, Roger Alexander Müller, and Enzo Nussio.

Above all, I am grateful to my family; to my girlfriend Nadja Dietliker; my flatmate, and close friend, Cyrill Gaudenz von Planta for constantly driving me to be my best ("what have you published today?"); and to all my other friends, here unnamed but not forgotten, for their support and interest in my work.
1 Introduction

Ich werde es Ihnen herausoperieren.
Wozu habe ich ein Taschenmesser.
Das werden wir gleich haben.
Arbeiten und nicht verzweifeln.
So, das hätten wir.
Aber das ist ja ein Ziegelstein! Ihr Herz ist ein Ziegelstein!
– Aber es schlägt nur für Sie.

Heiner Müller, Herzstück

1.1 From Art Galleries to Internet Cafes

Art gallery problems are a well-studied topic in computational geometry, dating back to 1973 when Victor Klee asked for the number of guards “necessary and sufficient to patrol the paintings and works of art in an art gallery with $n$ walls”. Chvátal [20] answered his question by showing that $\lfloor n/3 \rfloor$ guards are sufficient and may be necessary. Some years later Fisk [33] gave a much simpler proof via a coloring argument. Since the introduction of the first classic problem many variants of the problem have been investigated. For instance, a real guard in an art gallery realized by a camera could maybe only survey a certain angle. This leads to variants of the problem involving angle guards (or floodlights) asking for the number of guards needed with restricted angle of vision. Or maybe one wants to allow vertex guards only, that is, guards that are placed at a corner of the gallery.

There is now a vast literature about these problems [59, 66, 73, 60], ranging from optimization questions (minimizing the number of guards [59, 27, 36, 21] or maximizing the guarded boundary [34]) over special types of guards (mobile guards [58] or vertex pi-guards [67]) to special types of galleries (orthogonal polygons [41] or curvilinear polygons [43, 42]).
Inspired by angle guard problems and wireless localization a new problem was introduced by Eppstein, Goodrich and Sitchinava [29], which we call the wireless localization problem. For illustration, suppose you run a café (modeled as a simple polygon $P$) and you want to provide wireless internet access without giving the customers a password[61]. But you do not want the whole neighborhood to use your infrastructure. Instead, internet access should be limited to those people who are located within the café. To achieve this, you can install a certain number of devices, called guards, each of which broadcasts a unique (secret) key in an arbitrary but fixed angular range. The goal is to place guards and adjust their angles in such a way that everybody inside the café can prove this fact just by naming the keys received, and nobody outside the café can provide such a proof. Formally, this means to ask for a description of $P$ by a monotone Boolean formula over the keys, that is, a formula using only the operators AND and OR, not using any negations, because negations could enable a potential evildoer to cheat by concealing a key he receives. As in other art gallery settings, the primary goal is to minimize the number of guards. A guarding that minimizes the number of guards is called optimal.

For a polygon $P$ with $n$ non-collinear edges it is easy to prove a lower bound of $n/2$ and an upper bound of $n$ for the number of guards needed. Some progress has been made for different variants of the problem, but in the general case a gap still remains to be closed.

Eppstein et al. [29] proved that any simple polygon with $n$ edges and $h$ holes can be guarded using at most $n + 2(h - 1)$ vertex guards. Furthermore, the formula resulting from their algorithm is concise. This means that it is in disjunctive normal form and the size of the terms is bounded by a constant. Additionally, they consider orthogonal (rectilinear) polygons and give a tight upper bound of $n/2$. In Chapter 6 a new proof of the result for orthogonal
polygons is presented, which generalizes to 3-dimensional orthogonal polyhedra. Damian, Flatland, O’Rourke, and Ramaswami [23] describe a family of simple polygons with \(n\) edges that require at least \(\lfloor 2n/3 \rfloor - 1\) vertex guards, but only \(\lfloor n/2 \rfloor\) if general guards are allowed (in this case guards that are outside of \(P\)). In a joint work with Arosish Mishra [16] we improve the upper bound for the number of vertex guards needed to guard polygonal regions to roughly \(8n/9\). In the general setting we are able to give an upper bound of roughly \(3n/4\) if we forbid parallel edges, improving on a previous joint work with Michael Hoffmann, Yoshio Okamoto, and Takeaki Uno [15]. None of these bounds is known to be tight. We are going to improve the lower bound of roughly \(3n/5\) given in [15] to roughly \(2n/3\).

It seems surprisingly hard to prove any general statement about the structure of optimal guardings. For example, one might hope to prove that to find a minimum guarding it is sufficient to consider some special kind of guards, such as guards that are situated on vertices of the line arrangement defined by the edges of \(P\). However, such a proof eludes us. But the question and its relatives keep showing up. For instance, it is even unclear whether the problem of finding a minimum guarding is in NP, because we do not know whether the coordinates of the guards in an optimal solution can be explicitly written down using polynomial space (or computed in polynomial time if the guarding is not given explicitly).

For the classic art gallery problem it is known that finding a minimum guarding is NP-hard [50]. But it turns out that transparency of the edges changes the situation drastically. A basic ingredient in hardness proofs for the classical art gallery problems is a small spike of the polygon which can be guarded from a nearby point only, because the polygon edges shield it away from the rest of the polygon. This argument breaks down if the edges do not block visibility.

We are able to give a reduction from monotone SAT to the wireless localization problem with the restriction that guards have to be inside (or on the boundary of) the polygon, see Chapter 4. The question whether the general problem is NP-hard remains open.

![Figure 1.1: Describing a polygon by a monotone formula over some guards.](image)
1.2 From Internet Cafes to Constructive Solid Geometry

*Constructive solid geometry*, short CSG, is a technique used in the area of solid modeling. Given a physical object in the plane or in the space, there are several ways to describe (model) it mathematically as a subset of the Euclidean plane or space. The object to be described is called the *solid*. One typical way to model the solid, is to describe it implicitly by its boundary. Such a description is called a *boundary representation*. Another typical method to model a solid is to describe it explicitly as a combination of primitive solid objects. Such a description as a Boolean formula over *primitives* is called a *CSG-representation*. What is meant by a primitive depends on the context: For such a description to be useful, the primitives should be considerably easier to handle than the original object. For example, if we are to describe orthogonal polyhedra, it is reasonable to define the class of primitives to be boxes (rectangular parallelepipeds). Or if the objects of interest are all simple polygons, a reasonable class of primitives is the set of all triangles.

The concept of CSG-representations, usually restricted to the area of computer graphics, also appears in a purely theoretical context: As a way to define geometrical concepts. The Swiss mathematician Walter Nef \cite{56} defined polyhedra in the CSG-spirit as anything obtained from Boolean formulas over half-spaces. More recently this concept appears in the Computational Geometry Algorithms Library (CGAL) under the name or *Nef polyhedron* \cite{40}.

There are two basic ways to represent and store geometric objects: Either using a CSG-representation or using a boundary representation. Then immediately the question pops up how to convert one into the other. In the solid modeling world, one is usually concerned with the first direction: how to compute a boundary representation of an object given in a CSG-representation. But what about the other way around? Given an object in boundary representation, how can we find the CSG-representation? Of course, there is no way to answer the question in this generality. It is not even clear what “the CSG-representation” of an object is supposed to mean, as there is no unique representation in general. Do we want to reconstruct the primitives the object originally was built from? It is certainly an interesting question to what extent this is possible, but we are not going to address it. In the following we are either concerned with the question of just finding one possible CSG-representation (maybe having some nice additional properties), or with the more specific question of finding an optimal CSG-representation, where optimal means using a minimum number of primitives.

Now the relation between the wireless localization problem and constructive
solid geometry becomes apparent. Let us model guards as subsets of the plane, namely exactly the area that receives the broadcast from a guard. So guards are cones in the plane and the boundary of a guard is a plane angle. Then the wireless localization problem can be phrased as follows: Describe a simple polygon $P$ as a combination of the operations union and intersection over the guards, see Figure 1.1.

One way of generalizing the wireless localization problem is to ask for a description using other primitives than angle guards, as, for instance, triangles. Of course, a usual triangulation is a description of a polygon as an interior disjoint union of triangles, but certainly not an optimal one in general. See Figure 1.2.

![Figure 1.2: A polygon that can be partitioned into four triangles even though each triangulation has six triangles and a polygon that can be described as the union of two triangles.](image)

In Chapter 5 we show that describing a polygon as a union of a minimum number of triangles is NP-hard. The reduction is reminiscent of the reduction used by Culberson and Reckow [22] to prove the NP-hardness of the related problem of covering a polygon with convex polygons.

If we relax the problem a little bit and only ask for a cover of the boundary of the polygon by triangles that have to be inside, then the problem stays NP-hard, but there is an easy factor-2 approximation algorithm. Interestingly, this variant can also be shown to be in NP.

Moving on to the 3-dimensional space matters become considerably harder. A tetrahedralization is the 3-dimensional analogue of a triangulation: dissect a polyhedron into a set of tetrahedra such that they form a simplicial complex, that is, the intersection of two tetrahedra is either empty, a common vertex, a common edge, or a common face. While in 2D it is possible to triangulate every polygon by cutting along diagonals, it is not possible in general to tetrahedralize non-convex 3-dimensional polyhedra without introducing additional vertices in the interior (Steiner points) [70]. There are, however, positive results concerning tetrahedralization and CSG-representation of (not necessarily
convex) polyhedra, where CSG-representation means a Boolean formula over halfspaces \[25\].

Little to nothing seems to be known about finding CSG-representations using more complex primitives than halfspaces. The natural analogue of the wireless localization problem in \(\mathbb{R}^3\) is to ask for a description of a simple polyhedron as unions and intersections of polyhedral cones. For instance, we can ask for the number of cones needed to describe a 3-dimensional convex polyhedron. If we restrict our choice to “natural guards”, cones that are defined by vertices of the polyhedron, the problem translates to finding a subset of the vertices that cover every face of the polyhedron. The skeleton of the polyhedron is a plane graph. If we look at its dual graph, the corresponding problem is to find a so-called face cover, which is a well-studied problem \[48\].

Considering non-convex polyhedra, we have to realize that our 2D-approach to finding guardings fails. In the special case of orthogonal polyhedra, we can show that the 2D-result about guarding orthogonal polygons carries over to three dimensional orthogonal polyhedra. To guard an orthogonal polyhedron it suffices to put a natural vertex guard onto every other vertex. While in 2D it is obvious what is meant by “every other vertex”, in 3D we have to be careful that this remains meaningful. Indeed, we have to restrict to 3-regular orthogonal polyhedra, which means that at each vertex exactly three edges meet. Doing so, it turns out that the graph of the polyhedron is bipartite. Furthermore, while in 2D this usually yields a minimum guarding, in 3D this approach is certainly not optimal anymore. If we allow general vertex guards, we can bring the bound down to \(3n/8\), where \(n\) denotes the number of vertices of the polyhedron. It remains open whether this bound is tight. It might be possible that every 3-regular orthogonal polyhedron can by guarded with roughly \(n/4\) guards. This is the subject of Chapter 6.

1.3 From Solids to Detecting Cancer: Consistent Digital Line Segments

Sticking to the loose leitmotiv of describing geometric objects, we enter a different field. So far we have thought of our geometric objects to be given in a precise form and we were interested in finding alternative descriptions. In real world application geometric objects often first have to be retrieved from digital images. For example, a medical device might produce a digital 3-dimensional image of a body part. How can we recognize certain substructures in the image, as for instance a tumor? We have entered the field of image processing. We think of the image as 3-dimensional orthogonal orthogonal grid where each grid point
stands for a pixel (in the context of 3D images often called voxel). Every grid point is assigned a gray level. So a monochromatic image is modeled as a mapping from the grid to the unit interval \([0,1]\). The problem of separating a certain, say darker, region from a bright background goes by the name *image segmentation* [76]. To extract a tumor from a medical image, we want to identify three dimensional star-shaped annuli on the grid [13].

Recall that a region is called *star-shaped* if there is a point \(c\), called a *center point*, such that for each point \(p\) in the region the line segment \(cp\) is completely inside the region. A star-shaped annulus is the difference of two star-shaped regions with a common center. But what is it supposed to mean in the “grid geometry”? A natural way to define *star-shaped digital regions* is to define them as subsets of the integer grid that arise from Euclidean star-shaped regions taking all grid points that have distance at most \(1/2\) in the maximum (\(L^\infty\)) metric. This means that a set of grid points is star-shaped if and only if there is a Euclidean star-shaped region that intersects exactly the squares (pixels) of side length 1 centered at the grid points. However, using such a definition, our star-shaped digital regions do not behave as they are supposed to. For instance, taking the intersection of two star shaped regions with a common center point, we expect the result to be star-shaped, too. While this is trivial to prove for Euclidean star-shaped regions, it is not only hard to prove for our digital counterparts—it is also not true, see Figure 1.3 for a 2D-example.

![Figure 1.3: Using the naive definition, the intersection of two star-shaped digital regions with a common centerpoint (cross) does not need to be star-shaped.](image)

In order to define star-shaped digital regions in a better way, we want to introduce a notion of line segments on the grid, so called *digital line segments*. For any pair of grid points \((p,q)\), we want to define a digital segment \(S(p,q)\) from \(p\) to \(q\) as a suitable subset of the grid points. As soon as we have such a notion of digital line segments, all geometric definitions that only involve line segments can be used directly in the digital setting. However, the big
question remains how one should define digital line segments. The problem arises very naturally when a computer is supposed to draw a line segment on the screen. The research field that is concerned with such questions is called digital geometry. For sure, line segments should be somehow “connected”. To make this meaningful, one defines a graph on the integer grid by connecting each grid point to its four neighbors at distance 1. Then we require a digital line segment to be a path in the grid graph. Using the grid graph as explained goes by the name orthogonal grid topology. Another common definition includes the diagonal connections between grid points at Euclidean distance $\sqrt{2}$, resulting in a 8-regular graph. This is called the octagonal grid topology. See the survey by Klette and Rosenfeld [47].

Using the same approach as above, a very natural way of defining digital segments $S(p, q)$ is to take all integer grid points that have distance at most $1/2$ to the Euclidean segment $pq$ in the maximum metric. Thinking again of pixels instead of grid points, this is equivalent to taking all the pixels $pq$ hits. In particular, the resulting segments have the nice property that they induce a path in the grid graph, provided we introduce an appropriate rounding rule for the case that $pq$ passes through points that have distance exactly $1/2$ to four points. Indeed, this comes close to what the computer does drawing segments on the screen.

But this obvious solution is again problematic from a theoretical point of view. One of the basic axioms of geometry, Euclid’s very first axiom phrased in a modern way, is that any two points define a unique line, which immediately implies that any two lines either coincide or intersect in at most one point. It is obvious that we cannot expect this to be true for digital line segments: If two segments have almost the same slope, any reasonable digitalization is bound to have a big intersection. Using digital line segments as explained above, it might even happen that the intersection of two line segments is disconnected. If we are only interested in visualizing line segments, this is tolerable. But using the notion for mathematical purposes causes trouble.

For instance, let us use them to define convex digital sets. We call a set $A$ of grid points convex if and only if for all $p, q \in A$, $S(p, q) \subset A$. To start with, we have to realize that digital line segments are not necessarily convex themselves. If we look at two points $a, b \in S(p, q)$, it might happen that $S(a, b)$ is not contained in $S(p, q)$ but “slightly off”. This causes serious trouble. For example, if we want to take the digital convex hull of $p$ and $q$ (the smallest convex digital region containing $p$ and $q$), it should not only contain $S(p, q)$ but also $S(a, b)$. However, the new grid points that come with $S(a, b)$ might again define subsegments that are not completely contained in what we have so far, and so on. A priori, it is not even clear when this process stops, and
if it does, the result might look rather unpleasant. Similar problems arise if we use this notion of digital line segments to define star-shaped digital regions: it would be desirable that digital lines segments themselves were star-shaped digital regions.

Therefore, we are interested in an alternative definition of digital line segments. We still want them to be close to the corresponding Euclidean segments but at the same time they should “behave better” mathematically. Chun et al. [19] proposed the following four axioms that arise very naturally from properties of line segments in Euclidean geometry.

(S1) **Grid path property:** For all $p, q \in \mathbb{Z}^2$, $S(p, q)$ is the vertex set of a path from $p$ to $q$ in the grid graph.

(S2) **Symmetry property:** For all $p, q \in \mathbb{Z}^2$, we have $S(p, q) = S(q, p)$.

(S3) **Subsegment property:** For all $p, q \in \mathbb{Z}^2$ and every $r \in S(p, q)$, we have $S(p, r) \subseteq S(p, q)$.

(S4) **Prolongation property:** For all $p, q \in \mathbb{Z}^2$, there exists $r \in \mathbb{Z}^2$, such that $r \notin S(p, q)$ and $S(p, q) \subseteq S(p, r)$.

(S5) **Monotonicity property:** If both $p, q \in \mathbb{Z}^2$ lie on a line that is either horizontal or vertical, then the whole segment $S(p, q)$ belongs to this line.
The subsegment property could be equivalently formulated as the property of line segments to be star-shaped with respect to an endpoint (and thus, using (S2), to both endpoints) or to be the convex hull of their two endpoints; formulating it as a variant of Euclid’s first axiom it says: the intersection of any two digital line segments has to be a digital line segment itself. The fourth axiom, the prolongation property, is reminiscent of Euclid’s second axiom: any line segment can be extended to an infinite line.

When Chun et al. proposed these axioms it was not clear whether this is too much to ask for. Is there a definition of digital line segments such that they satisfy all the axioms and still are reasonable approximations of the Euclidean segments? A system of digital line segments that satisfies the axioms is called a consistent digital line segment system or short a CDS.

The good news are that there are indeed consistent digital segments in the plane that are close to the corresponding Euclidean segments. They can be used to define basic geometric notions for digital regions, such as star-shaped digital and convex digital regions, see for example [74]. How to define such a CDS is the main topic of Chapter 7. Furthermore, we try to characterize all CDSes in the plane, which is possible if one introduces an additional axiom. The results are joint work with Dömötör Pálvölgyi and Miloš Stojaković [17, 18]. The same problem has been treated in depth by Michael Luby [52] more than twenty years ago using a different approach and a different notation. Most of our results concerning digital line segments in the plane turn out to be rediscoveries of his results.

The bad news are that we do not know how to define a CDS in higher dimensions that approximates the Euclidean segments well. But there are partial solutions to the problem, see Section 7.6.2.

1.4 Definitions and Notations

Most of the objects considered in the following are subsets of the Euclidean plane or the Euclidean space, denoted by $\mathbb{R}^2$ or $\mathbb{R}^3$ even though we usually think of it as an affine space without fixed origin. We denote the convex hull of a subset $A \subset \mathbb{R}^d$ by $\text{conv}(A)$. The affine hull of $A$ is denoted by $\overline{A}$. This means if $s$ is a line segment, then $\overline{s}$ denotes the line defined by $s$ or if $A$ is a polygon embedded into $\mathbb{R}^3$, then $\overline{A}$ is the underlying affine plane. A subset $C \subset \mathbb{R}^d$ is called a cone if and only if there is a point $a \in C$, called an apex, such that for any other $x \in C$, the ray from $a$ through $x$ is completely contained in $C$. Note that the apex of a cone does not need to be unique; for example, a halfplane is a cone where any point on the bounding line is an apex.
The topological closure of a set $A$ is denoted by $\text{cl}(A)$, the topological interior by $\text{int}(A)$, and the boundary by $\partial A$. For points $v, w \in \mathbb{R}^d$ and $A \subseteq \mathbb{R}^d$, let $d(v, w) = |v - w|$ and $d(v, A) = \inf_{a \in A} d(v, a)$ denote the usual Euclidean distances between two points, and between a point and a set. For two sets $A$ and $B$, we denote their Hausdorff distance by $H(A, B)$,

$$H(A, B) = \max \left\{ \sup_{a \in A} d(a, B), \sup_{b \in B} d(b, A) \right\}.$$

For a point $v$ and a real $\varepsilon > 0$, $B_\varepsilon(v) := \{x \in \mathbb{R}^d \mid d(x, v) < \varepsilon\}$ is the open $\varepsilon$-ball around $p$.

**Simple Polygon**

We think of polygons as closed solid subsets of the plane including the interior, that is, a simple polygon $P$ is a simply connected closed subset of the Euclidean plane the boundary $\partial P$ of which is a simple finite cycle of line segments. Simple means that non-adjacent segments do not intersect and adjacent segments only intersect at their common endpoint.

An edge $e$ of $P$ is a maximum length line segment in $\partial P$. Note that this definition excludes the possibility of adjacent collinear edges. A point $v$ that is the common endpoint of two edges of $P$ is called a vertex of $P$. We say a simple polygon $P$ is in general position if its vertex set $V(P)$ is in general position, meaning that no more than two vertices lie on a common line. We denote the edges of $P$ by $E(P)$ and the number of edges by $n(P) \geq 3$. 
Chapter 1. Introduction

Polygonal Chain and Halfplane

A \emph{polygonal halfplane} is a topological halfplane bounded by a \emph{simple bi-infinite polygonal chain} with edges \((e_1, \ldots, e_n)\), for a positive integer \(n\). For \(n = 1\), the only edge \(e_1\) is a line and the polygonal halfplane is a halfplane. For \(n = 2\), \(e_1\) and \(e_2\) are rays which share a common source but are not collinear. A polygonal halfplane with one or two edges only is a cone.

For \(n \geq 3\), \(e_1\) and \(e_n\) are rays, \(e_i\) is a line segment, for \(1 < i < n\), and \(e_i\) and \(e_j\), for \(1 \leq i < j \leq n\), do not intersect unless \(j = i + 1\) in which case they share an endpoint. Two adjacent edges \(e_i\) and \(e_{i+1}\) may not be collinear. For brevity we use the term \emph{chain} in place of simple bi-infinite polygonal chain.

By the \emph{size} of a polygonal halfplane \(H\) or its bounding chain \(C = \partial H\) we mean its number \(n\) of edges and denote it by \(n(H)\) or \(n(C)\), respectively. Sometimes we use the term \emph{n-chain} to refer to a chain of size \(n\).

Polygonal Region

A \emph{polygonal region} is a closed subset \(P\) of the plane such that its boundary \(\partial P\) is a finite disjoint union of simple polygonal cycles (cycles of line segments) and simple polygonal chains. Note that simple polygons and polygonal halfplanes are polygonal regions. By definition, both the entire Euclidean plane and the empty set are polygonal regions as well.

The edge set \(E(P)\) and the vertex set \(V(P)\) are defined in the same way as for simple polygons, but now edges might be unbounded as well. An edge is an inclusion maximal line segment or ray or line contained in the boundary \(\partial P\) of a polygonal region \(P\); the vertices are the common endpoints of two edges.

If a polygonal region is \emph{bounded}, it consists of several connected components, each of which being a polygon possibly containing holes. A connected bounded polygonal region is called a \emph{polygon with holes}. Note that our definition does not require polygonal regions to be bounded. The unspecified term \emph{polygon} may refer to both polygons with holes or simple polygons depending on the context.

Polyhedron

Intuitively, a polyhedron is the three-dimensional analogue of a polygon. However, properly defining non-convex polyhedra is not an easy task. Very well-explained definitions can be found at several places \cite{5, 24, 35}, but so far no standard definition seems to have emerged. We slightly deviate from the usual definitions in one aspect: in accordance with our definition of polygonal regions, we do allow polyhedral regions to be unbounded and to have unbounded
faces.

**Figure 1.6:** To the left a simple orthogonal polyhedron with a face \( f \) that is not a polygonal region and with two coplanar faces \( g \) and \( h \) where \( h \) has two incident edges that are adjacent and collinear; in the middle a surface that is not a closed 2-manifold without boundary, and to the right an object the boundary of which is not a polyhedral surface.

\[ P \subset \mathbb{R}^3 \text{ is a polygonal region (embedded into } \mathbb{R}^3) \text{ if its affine hull } \overline{P} \text{ is an affine plane and } P \text{ is a polygonal region in } \overline{P}. \]

A polyhedral surface \( S \) is a finite union of polygonal regions embedded into \( \mathbb{R}^3 \) that forms a closed 2-manifold without boundary. (A topological space \( S \) is called a 2-manifold if the neighborhood of any point \( x \in S \) is homeomorphic to a disk. A 2-manifold without boundary closed in \( \mathbb{R}^3 \) cannot “end” anywhere, see Figure 1.6 in the middle.) Unlike other authors, we neither require polygonal surfaces to be bounded (compact) nor to be connected. A set \( P \subset \mathbb{R}^3 \) whose boundary \( \partial P \) is a polyhedral surface is called a polyhedral region. Note that we do not impose any global topological conditions on polyhedral regions, they can be unbounded or disconnected. A connected component can have cavities, thus the boundary of a connected component may again be disconnected. A polyhedron is a bounded and connected polyhedral region. A simple polyhedron is a polyhedron the boundary of which is homeomorphic to the 2-sphere.

Let \( S \) be a polyhedral surface. An inclusion-maximal interior connected subset \( f \subset S \) the affine hull \( \overline{f} \) of which is an affine plane is called a face of \( S \). A face \( f \) of a polyhedral surface \( S \) usually “looks like” a polygonal region. Note, however, that we do not require faces to be simple polygons: they may contain holes and it might even happen that a face is not a polygonal region according to our definition, see face \( f \) in Figure 1.6. Furthermore, faces may be unbounded if \( S \) is unbounded. A point \( p \in S \) is a vertex of \( S \) if it is the intersection of three (or more) faces of \( S \). An edge of \( S \) is an inclusion-maximal line segment (or ray or line) contained in the intersection of two faces.
of $S$ containing no vertices in its relative interior. Note that by this definition a face can have collinear adjacent edges: see face $h$ in Figure 1.6; one of the vertices incident to $h$ is not a vertex of $h$ as a 2D-polygon but it is still a vertex of the polyhedron and hence a vertex of $h$ as a face.

The definitions ensures that edges and vertices of a bounded polyhedral surface $S$ form a geometric graph in $\mathbb{R}^3$ (a 1-dimensional simplicial complex) called the 1-skeleton of $S$. The faces, edges, and vertices of a polyhedral region $P$ are defined as the faces, edges, or vertices, respectively, of $\partial P$. Let $n(P)$ denote the number of faces of $P$. The graph of a polyhedron is the abstract graph defined by its 1-skeleton. A polyhedron $P$ is called 3-regular if its graph is 3-regular, that is, exactly three edges meet at each vertex of $P$. A polyhedral region is orthogonal if each of its faces is orthogonal to one of the coordinate axes. Figure 1.6 shows a simple orthogonal polyhedron that is not 3-regular.
2 CSG-Representations of Polygons and Polyhedra

In this chapter we define the problem of finding CSG-representations formally and provide a common framework for the chapters to follow.

2.1 Regularized Operations

Dealing with CSG-representations, there is a technical difficulty to overcome. The objects of interest as well as the primitives used in a CSG-representation are usually defined as closed subsets of the plane or space including their interior. As long as we only take finite unions and intersections of primitive objects, the result is a closed subset again. But if we also allow set difference, then the result is not necessarily closed. Furthermore, there might be 1-dimensional relics after intersecting two polygons that touch each other at an edge.

Requicha and Tilove \[64, 71\] have proposed to deal with this issue using the concept of \textit{regularization}. Given any set $A$ in a topological space, we define its regularization

$$A^* := \text{cl}(\text{int}(A)),$$

where $\text{cl}(A)$ denotes the topological closure of $A$ and $\text{int}(A)$ the topological interior of $A$. A set $A$ is called \textit{regular} if $A = A^*$. In particular, $(A^*)^* = A^*$, that is, $A^*$ is regular for any $A$ \[64\].

The \textit{regularized} union, intersection, set difference and complement are defined as the usual set operation and then regularizing the result: For sets $A, B$
we define
\[
A \cup^* B := (A \cup B)^*
\]
\[
A \cap^* B := (A \cap B)^*
\]
\[
A \setminus^* B := (A \setminus B)^*
\]

and the regularized complement \(A^c := (A^c)^*\). Whenever we take the union, intersection, difference or complement of solid objects, we are using this regularized operations. We are often going to drop the star \(^*\) in the following. Applying regularized set operations on regular sets, we can be sure to stay inside the family of regular sets. Furthermore, we can calculate with the regularized operations the same way as with the usual set operations. More precisely, the regular sets with the regularized set operators are a Boolean algebra \([64]\).

Note, however, that using regularized Boolean operations and polygonal regions as primitives does not suffice to ensure that we stay within the class of polygonal regions. For instance, if two polygons touch each other at a common vertex, their (regularized) union is not a polygonal region.

### 2.2 Description, Cover and Partition

The general problem of finding a CSG-representation of an object can be differentiated in several ways. First of all, we can vary our objects of interest. For example, do we want to describe all possible polygonal regions? Or are we only interested in simple polygons? Second, we can vary what kind of primitives we allow. For example, we might allow all convex polygons as primitives. Or we allow only convex polygons the vertices of which already appear in the input, i.e., that are a subset of the vertices of the object to describe. Or we might even restrict further and only allow triangles as primitives. Finally, there are several variants which operations to allow over the primitives. For example, we might allow all Boolean set operations or only unions.

Let \(P\) denote a family of regular subsets of the Euclidean space \(\mathbb{R}^d\), which are called the objects of interest.

Then, we have to fix the allowed primitive objects. Let \(Q : P \rightarrow 2^{\mathbb{R}^d}\) be a function that maps an object \(P \in P\) to the set of allowed primitives \(Q(P)\). The elements of \(Q(P)\) are required to be regular. Note that \(Q(P)\) is not necessarily a subset of \(P\). If \(P\) is understood from the context or if \(Q(P)\) does not depend on \(P\), we omit the argument and just write \(Q\) for the set of allowed primitives.

The most general variant of a CSG-representation is what we are going to call a description. Given an object \(P \in P\), a description of \(P\) is a finite subset \(S \subset Q\) such that \(P\) can be expressed by a Boolean formula over \(S\) using the regularized Boolean set operations.
If there is a formula using only the operations union and intersection (and neither set difference nor complement), we call a description monotone. If $P$ is the (regularized) union of the primitives in $S$, that is, $P = \bigcup_{s \in S}^* s$, then we call $S$ a cover of $P$. If $S$ is a cover of $P$ and the primitives in $S$ are pairwise internally disjoint (that is, disjoint in the regularized sense), we call $S$ a partition of $P$.

In many applications, the set of admissible primitives $Q(P)$ is the same for each object $P$. If this is the case, we sometimes speak of general descriptions (monotone descriptions, covers, or partitions) to emphasize the fact that we allow the same set of primitives for each object.

We define $\beta_{P,Q}(P)$ as the minimal integer $k$ such that there is a partition of $P$ using $k$ primitives. Similarly, we define $\gamma_{P,Q}(P)$ to be the smallest $k$ such that $P$ can be covered by $k$ primitives; $\delta_{P,Q}(P)$ is the size of a minimum monotone description; and finally, we define $\delta^c_{P,Q}(P)$ to be the size of the smallest possible description of $P$.

If there exists no (finite) description (monotone description, cover, or partition), we define the corresponding numbers to be infinity. Sometimes we think of a (monotone) description $S$ of $P$ not only as a set of primitives, but also as a formula describing $P$ over the primitives. In Section 2.3 it is shown that given a set of primitives $S$, it is easy to decide whether there is a formula over $S$ describing $P$. With a slight abuse of notation, we might write $s \in S$ even if we think of $S$ as a formula over primitives rather than a set of primitives.

For example, let $\mathcal{P}$ be the set of all simple polygons in the plane and $Q$ the set of all star-shaped polygons. Then, the classical art gallery problem translates to finding a covering of $P \in \mathcal{P}$: $\gamma_{\mathcal{P},Q}(P)$ is the number of guards needed to guard $P$. The problem of finding a description of $P$—using even fewer star-shaped polygons than we need to cover it—is an open problem as far as I know.

In the following, the set of objects $\mathcal{P}$ as well as the allowed primitives $Q(P)$ for a $P \in \mathcal{P}$ are always going to be subsets of the set of polygonal or polyhedral regions (see Section 1.4 for definitions). So for an object $P \in \mathcal{P}$ or for a primitive $s \in Q(P)$, we can use its number of facets $n(P)$ or $n(s)$ as a measure for its complexity. For an integer $n$, we define

$$\delta^c_{\mathcal{P},Q}(n) := \max\{\delta^c_{\mathcal{P},Q}(P) \mid P \in \mathcal{P} \text{ and } n(P) = n\},$$

the number of primitives needed to describe members of $\mathcal{P}$ with $n$ facets in the worst case. Similarly we define

$$\delta_{\mathcal{P},Q}(n) := \max\{\delta_{\mathcal{P},Q}(P) \mid P \in \mathcal{P} \text{ and } n(P) = n\},$$

$$\gamma_{\mathcal{P},Q}(n) := \max\{\gamma_{\mathcal{P},Q}(P) \mid P \in \mathcal{P} \text{ and } n(P) = n\},$$

$$\beta_{\mathcal{P},Q}(n) := \max\{\beta_{\mathcal{P},Q}(P) \mid P \in \mathcal{P} \text{ and } n(P) = n\}.$$
The indices $\mathcal{P}$ and $\mathcal{Q}$ are often dropped if the objects of interest and the primitives are clear from the context.

Beside asking for a partition, covering or description of a given object, there are general questions one can ask given $\mathcal{P}$ and $\mathcal{Q}$: What is the complexity of computing $\beta(P)$, $\gamma(P)$, $\delta(P)$, or $\delta^c(P)$ in general? Can we do it efficiently? Can we give explicit formulas for $\beta(n)$, $\gamma(n)$, $\delta(P)$, $\delta^c(n)$? If not, can we bound them from above and below? Do we know the values for $\beta(n)$, $\gamma(n)$, $\delta(n)$, and $\delta^c(n)$ for small $n$?

### 2.3 Basic Observations

Throughout this section let $\mathcal{P}$ be an arbitrary set of regular polygonal regions or regular polyhedral regions and for any $P \in \mathcal{P}$, let $\mathcal{Q}(P)$ a set of regular polygonal (polyhedral, respectively) regions. By definition a partition $S$ of an object $P$ is also a cover of $P$, while a cover is also a monotone description, and all of them are descriptions of $P$.

**Observation 2.1.** For every $P$, $\delta^c(P) \leq \delta(P) \leq \gamma(P) \leq \beta(P)$. This immediately implies $\delta^c(n) \leq \delta(n) \leq \gamma(n) \leq \beta(n)$.

We call a point $p$ *generic* with respect to a family $\mathcal{S}$ of regular sets if and only if

$$p \notin \bigcup_{s \in \mathcal{S}} \partial s.$$  

Regularized set operations behave like the usual set operations with respect to generic points: Let $p$ be generic with respect to $\mathcal{S}$ and let $s, t \in \mathcal{S}$. Then we observe

$$p \in s \cap^* t \iff p \in s \cap t,$$

$$p \in s \cup^* t \iff p \in s \cup t,$$

$$p \in s \setminus^* t \iff p \in s \setminus t.$$  

Furthermore, note that the set of generic points with respect to a finite family $\mathcal{S}$ of regular sets is dense in the plane or space. This means for every point $p$ and every real $\varepsilon > 0$, we find a generic point $p'$ with $d(p, p') < \varepsilon$.

Let $P \in \mathcal{P}$ and $\mathcal{S}$ be a finite subset of $\mathcal{Q}(P)$. If $\mathcal{S}$ is a description of $P$, then for every point pair $(p, q)$ with $p \in P$ and $q \notin P$ and both $p$ and $q$ generic with respect to $\mathcal{S}$ (sometimes called an inside/outside pair), there must be a primitive $s \in \mathcal{S}$ such that either $p \in s$ and $q \notin s$ or $p \in s^c$ and $q \notin s^c$. We say $s$ distinguishes $p$ and $q$. If there were no such primitive, then the result of every formula composed from $\mathcal{S}$ would either contain both $p$ and $q$ or none of them: If neither $s$ nor $t$ distinguishes $p$ and $q$, then the result of every formula composed from $\mathcal{S}$ would either contain both $p$ and $q$ or none of them.
2.3. Basic Observations

... to \( \{s, t\} \), then neither does \( s \cap^* t, s \cup^* t \), nor \( s \setminus^* t \). Note that this statement would not necessarily be true for non-generic points.

It turns out that this condition is not only necessary but also sufficient. If for every inside/outside pair \((p, q)\) we can find a primitive \( s \in S \) that distinguishes them, then \( S \) is a description of \( P \). We get a describing formula as follows. For every point \( p \in P \), let \( S_p^+ \) be the subset of primitives that contain \( p \), \( S_p^+ := \{ s \in S \mid p \in s \} \). Similarly, let \( S_p^- \) be the set of primitives that do not contain \( p \). Then, we get a describing formula in disjunctive normal form.

**Observation 2.2.** If every inside/outside pair \((p, q)\) can be distinguished by an \( s \in S \), then

\[
P = \bigcup_{p \in P} \left( \bigcap_{s \in S_p^+} s \bigcap_{s \in S_p^-} s^c \right).
\]

Proof. For every generic point \( p \in P \) it holds that \( p \in \bigcap_{s \in S_p^+} s \bigcap_{s \in S_p^-} s^c \), so \( p \) is contained in the right hand side. For every generic point \( q \notin P \), and for every generic \( p \in P \), there is a primitive that distinguishes them, so \( q \notin \bigcap_{s \in S_p^+} s \bigcap_{s \in S_p^-} s^c \), so \( q \) is contained in none of the terms. Therefore, a generic point is in \( P \) if and only if it is in contained in the right hand side.

This implies that equality holds: Let \( p \in P \) be a non-generic point. Because \( P \) is regular and because the generic points are dense, there is a generic point \( p' \in P \) in any neighborhood of \( p \). Consequently, each \( p' \) is also contained in the right hand side. Therefore, we find a sequence of generic points contained in both \( P \) and the right hand side converging to \( p \). Because the right hand side is regular and hence closed, we can conclude that \( p \) is contained in it. The same way it is shown that if \( p \) is a non-generic point in the right hand side, it must be contained in \( P \).

A priori, the above formula from Observation 2.2 is infinite, because we take the union of infinitely many terms. But there are only finitely many different possible terms

\[
\left\{ \bigcap_{s \in T} s \bigcap_{s \in S \setminus T} s^c \mid T \subset S \right\}
\]

(\( 2^{|S|} \), to be precise). So we can easily reduce the formula to a finite one, just taking the union of those terms that appear at least once.
Observation 2.3. A finite set \( S \subset Q(P) \) is a description of \( P \) if and only if for each generic point pair \( p \in P \) and \( q \notin P \), there is a primitive \( s \in S \) that distinguishes \( p \) and \( q \).

If we turn our attention to monotone descriptions, this characterization becomes even more natural. We say a primitive \( s \) distinguishes an inside/outside pair \((p,q)\) monotonically if \( p \in s \) and \( q \notin s \). (In the context of monotone descriptions, we are just saying \( s \) distinguishes an inside/outside pair, without adding “monotonically”.) A set of primitives turns out to be a monotone description if and only if every inside/outside pair can be distinguished monotonically. On one hand, assume there is no \( s \in S \) that distinguishes \( p \) and \( q \) monotonically. Then, if \( p \) is contained in a monotone formula composed from elements of \( S \) so is \( q \). On the other hand, assume for every inside/outside pair, there is a \( s \in S \) distinguishing them. Then we find

\[
P = \bigcup_{p \in P}^* \bigcap_{s \in S_p^+}^* s.
P \text{ generic w.r.t. } S
\]

Observation 2.4. A finite set \( S \subset Q \) is a monotone description of \( P \) if and only if for any two generic points \( p \in P \) and \( q \notin P \), there is a primitive \( s \in S \) that distinguishes \( p \) and \( q \) monotonically.

We get another necessary condition for descriptions by looking at the boundary of an object \( P \in \mathcal{P} \). Let again \( S \subset Q \) be a potential description. Consider a facet \( f \) of \( P \). (If \( P \) is a polygonal region, \( f \) is an edge; if \( P \) is a polyhedral region, \( f \) is a face of \( P \).) A set \( T \subset S \) of primitives covers some point \( r \in f \), if there exists some \( \varepsilon > 0 \) such that for every inside/outside pair \( p,q \in B_{\varepsilon}(r) \) there is a primitive in \( S \) that distinguishes \( p \) and \( q \). We say that a set \( T \subset S \) covers the facet \( f \) if \( T \) covers some point in the relative interior of \( f \). Finally, \( T \) covers \( f \) completely if \( T \) covers all points in the relative interior of \( f \).

Note that if \( \{s\} \) covers \( f \), then the affine hull of one of the facets of \( s \) equals the affine hull of \( f \). So in 2D, where our objects are polygonal regions, this means that there is an edge \( e_s \in E(s) \) such that \( f = \overline{e_s} \), so the two define the same line. In 3D, where the objects and primitives are polyhedral regions, it means that there is a face \( f_s \in F(s) \) such that \( f = \overline{f_s} \), so the two lie in the same plane.

If we want to emphasize that \( s \) covers \( f \) but not completely, we say \( s \) covers \( f \) partly. If a set consisting of just one primitive \( s \) covers a facet \( a \) partly or completely, we simply say \( s \) covers \( a \) partly or completely, respectively.
Observation 2.5. If $S \subset Q(P)$ is a description of $P \in \mathcal{P}$, then each facet $f$ of $P$ gets covered completely by a subset of $S$. Consequently, for every facet $f$ of $P$, there is at least one primitive that covers it completely or there are at least two primitives that cover $f$ partly.

Proof. Let $f$ be a facet and $S$ a description of $P$. Suppose there is no primitive that covers $f$ completely and at most one primitive that covers $f$ partly. Then we can find a point $p \in f$ such that for all $s \in S$, $p \notin \partial s$. Because $p \in \partial P$, there is a generic point $q$ close to $p$ and located outside $P$ such that every primitive contains $p$ if and only if it contains $q$, in contradiction to Observation 2.3.

If the number of facets of all allowed primitives is bounded by a constant, then this observation implies an easy lower bound on $\delta^e(P)$.

Theorem 2.6. Let $\mathcal{P}$ be a set of either polygonal or polyhedral regions and for each $P \in \mathcal{P}$, let $Q(P)$ be the admissible primitives.

(i) Let $P \in \mathcal{P}$ be an object without collinear edges (or coplanar faces, respectively). If there is a constant integer $k$ such that for any $s \in Q(P)$, $n(s) \leq k$, then $\delta^e_{\mathcal{P},Q}(P) \geq \lceil n(P)/k \rceil$.

(ii) Let $n$ be a positive integer. If there is a $P \in \mathcal{P}$ with $n(P) = n$ without collinear edges (or coplanar faces, respectively), then $\delta^e_{\mathcal{P},Q}(n) \geq \lceil n/k \rceil$.

Proof. Let $P$ be a polygonal region without collinear edges (a polyhedral region without coplanar faces, respectively). Then a facet $f$ of a primitive $s$ can contain at most one facet of $P$ in its affine hull $\overline{f}$. Therefore, $s$ can cover at most $k$ different facets of $P$. So it follows from Observation 2.5 that there need to be at least $n/k$ primitives in a valid description of $P$.

In the context of monotone descriptions, we define the notion of covering points or facets exactly the same way, except that we replace “distinguish” by “distinguish monotonically”. If a primitive $s$ covers a facet $a$ in the monotone sense, then there is a facet $a_s$ of $s$ such that $a_s = \overline{a}$ and additionally, the orientations match, that is, both $a$ and $a_s$ have the interior of $P$ or $s$, respectively, on the same side. Observation 2.5 carries over to the monotone setting verbatim.

Observation 2.7. If $S \subset Q(P)$ is a monotone description of $P \in \mathcal{P}$, then each facet $f$ of $P$ is covered (in the monotone sense) by $S$. Consequently, for each facet $f$, there is at least one primitive that covers $f$ completely or there are at least two primitives that cover $f$ partly.
2.4 Results Fitting into the Framework

Finding partitions or covers of polygons or polyhedra are well-known problems in computational geometry, which usually go by the name polygon (polyhedron) decomposition. Pseudotriangles, spiral, convex, and monotone polygons have been studied as possible primitives, see the survey by Keil [45]. Usually, the results are restricted to the problem of finding partitions and finding covers. Monotone or general descriptions are hardly ever addressed in computational geometry papers dealing with decomposition.

An exception are O’Rourke and Supowit [62] mentioning sum/difference decompositions, which is a restricted variant of descriptions in our sense: a sum/difference decomposition is a sequence of unions and set differences of primitives applied one after another: \( P = (((s_1 \cup s_2) \setminus s_3) \cup s_4) \setminus s_5 \ldots \). They state the computational complexity of finding minimum sum/difference decompositions as an open problem.

The problem of finding CSG-representations of orthogonal polygons using rectangles as primitives has been treated in the Master Thesis of Dan Bühler [6]. The notation introduced in this chapter is inspired by his work.

The Wireless Localization Problem

As already mentioned in the introduction, the wireless localization problem can be phrased as a CSG-representation problem. Let \( \mathcal{P} \) be the set of all polygonal regions, and \( \mathcal{Q} \) the set of all planar unbounded cones. Then the monotone description problem is equivalent to the wireless localization problem or internet cafe problem: Given a polygonal region, describe it by a monotone formula over angle guards. Chapter 3 and Chapter 4 are devoted to this problem.

Convex Decomposition

The problem of decomposing polygons or polyhedra into convex pieces has received a lot of attention. It has been known for a long time that covering a polygon \( P \) with a minimal number of convex polygons is NP-hard, as shown in 1983 by O’Rourke and Supowit [62] along with other polygon cover problems. In 1988, Culberson and Reckow [22] gave another NP-hardness proof, which also works for polygons without holes. In 2001 Eidenbenz and Widmayer [28] presented an \( O(\log n) \)-approximation algorithm for the convex cover problem and furthermore, they observed that the reduction of Culberson and Reckow also implies APX-hardness of the problem. In our terminology, setting \( \mathcal{P} \) to be the set of simple polygons and \( \mathcal{C} \) to be the set of convex polygons, this means that computing \( \gamma_{\mathcal{P},\mathcal{C}}(P) \) is NP-hard.
2.4. Results Fitting into the Framework

If we look at the problem of partitioning a polygon into convex pieces, the situation changes. It is possible to partition a simple polygon into convex pieces in time $O(n^3)$ using dynamic programming [8, 46]. This means, if we restrict to simple polygons $P$, $\beta_{P,C}(P)$ can be computed and a minimum partition for $P$ can be constructed efficiently.

If we allow the polygons to contain holes, the problem turns NP-hard as shown by Lingas in 1983 [51]; so computing $\beta_{P,C}(P)$ for general polygonal regions $P$ is NP-hard. The proof by Lingas uses Steiner points. But as he points out, the NP-hardness of the convex partition problem disallowing Steiner points follows using the reduction of O’Rourke and Supowit [62] reducing from planar 3SAT. In other words, for general polygons $P$ allowing holes, the problem of computing $\beta_{P,C}(P)$ is NP-hard even if $C(P)$ is restricted to the set of convex polygons with their vertices in $V(P)$ only. We are going to discuss the problem of describing a polygon with convex pieces in Section 2.5.

Decomposing Polygons into Triangles

A very natural choice for the primitives are triangles. Let $P$ be all simple polygons and $Q$ be the set of all triangles. Then, the problem of finding a partition seems to be the same as finding a triangulation. But this is only true if we disallow Steiner points for the primitives and consider polygons in general position (no collinear edges) only. The problem of partitioning a polygon into a minimum number of triangles is an open question as far as I know. In Chapter 5 we treat the problem of covering polygons with triangles.

There are some results about describing a polygon using triangles spanned by some fixed origin and the edges of the polygon [65]. Then in a formula for the polygon some of the triangles appear negatively. A monotone description of a polygonal region using guards carries over to a description of the region using triangles, see Section 5.7. Concerning a monotone description of a polygon with triangles nothing seems to be known.

Decomposition of Polyhedra

In general, it is impossible to tetrahedralize (non-convex) 3-dimensional polyhedra without introducing additional vertices in the interior (Steiner points) [70]. Dey [25] showed how to compute convex covers of arbitrary polyhedra, which leads to an algorithm tetrahedralizing arbitrary polyhedra using Steiner points. The term CSG-representation is used to denote a description using the halfspaces supporting the facets as primitives. More precisely, for a polyhedron $P$ we define $Q(P)$ to be the set of halfspaces bounded by $\overline{f}$ for an $f \in F(P)$ such that the interior of the halfspace is on the same side as the interior of $P$. 

Table 2.1: Lower and upper bounds (separated by <) or the exact functions \( \beta(n), \gamma(n), \delta(n), \delta^c(n) \) for simple polygons (up to plus/minus one) for different primitives.

<table>
<thead>
<tr>
<th>( Q(P) )</th>
<th>partition</th>
<th>cover</th>
<th>monotone</th>
<th>description</th>
<th>Sec. [Ref.]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Natural guards</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>( n - 2 )</td>
<td>( n - 2 )</td>
<td>15</td>
</tr>
<tr>
<td>Vertex guards</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>( \frac{2n}{3} &lt; \frac{8n}{9} )</td>
<td>?</td>
<td>3.2.3</td>
</tr>
<tr>
<td>General guards</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>( \frac{2n}{3} &lt; \frac{3n}{4} )</td>
<td>?</td>
<td>3.2.6, 3.3.3</td>
</tr>
<tr>
<td>Triangles</td>
<td>( n - 2 )</td>
<td>( n - 2 )</td>
<td>?</td>
<td>( \frac{n}{2} &lt; \frac{3n}{4} )</td>
<td>5.7, 2.5</td>
</tr>
<tr>
<td>Convex</td>
<td>( n - 2 )</td>
<td>( n - 2 )</td>
<td>?</td>
<td>( \frac{n}{2} )</td>
<td>2.5</td>
</tr>
<tr>
<td>Star-shaped</td>
<td>( n/3 )</td>
<td>( n/3 )</td>
<td>?</td>
<td>?</td>
<td>20</td>
</tr>
</tbody>
</table>

In [25] the term Peterson-style formula is used to denote a monotone descriptions using these primitives. It is shown that there are monotone descriptions of size \( O(n^2 \alpha(n)) \) where \( n \) denotes the number of facets of a polyhedron and \( \alpha \) denotes the inverse Ackermann function. In [26] the notion “Peterson-style formula” is used slightly differently: It denotes a monotone description (over the same primitives as above) with the additional property that each primitive appears exactly once. It is shown that Peterson-style formulas in this more restrictive sense do not exist in general. In Chapter 6 we look at the problem of (monotonically) describing orthogonal polyhedra using orthogonal cones.

Summary

The results concerning CSG-representations of polygons are summarized in Tables 2.1 and 2.2. In this thesis we close some gaps, but the general framework has made many more open problems apparent. In particular, it remains open if more of the “NP-hard” entries in the second table can be replaced by “NP-complete”.

2.5 Describing a Polygon by Convex Pieces

As a warm-up for the chapters to follow we consider the problem of describing a polygon by convex pieces. Formally, the objects of interest \( P \) are all simple polygons and for each simple polygon \( P \in P \), the primitives are the set of
2.5. Describing a Polygon by Convex Pieces

<table>
<thead>
<tr>
<th>$Q(P)$</th>
<th>$P$</th>
<th>$\beta(P)$</th>
<th>$\gamma(P)$</th>
<th>$\delta(P)$</th>
<th>Sec.</th>
<th>[Ref.]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Natural guards</td>
<td>G</td>
<td>-</td>
<td>-</td>
<td>NP-C</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>Vertex guards</td>
<td>G</td>
<td>-</td>
<td>-</td>
<td>NP-hard</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>General guards</td>
<td>G</td>
<td>-</td>
<td>-</td>
<td>?</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Non-Steiner Triang.</td>
<td>G</td>
<td>$O(n)$</td>
<td>$O(n)$</td>
<td>?</td>
<td>7</td>
<td>5.5</td>
</tr>
<tr>
<td>Non-Steiner Triang.</td>
<td>S</td>
<td>$O(n^3)$</td>
<td>NP-C</td>
<td>?</td>
<td>10</td>
<td>5.5</td>
</tr>
<tr>
<td>Non-Steiner Triang.</td>
<td>H</td>
<td>NP-C</td>
<td>NP-C</td>
<td>?</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>Triangles</td>
<td>G</td>
<td>?</td>
<td>NP-hard</td>
<td>?</td>
<td>3</td>
<td>5.2</td>
</tr>
<tr>
<td>Non-Steiner Convex</td>
<td>S</td>
<td>$O(n^3)$</td>
<td>NP-C</td>
<td>?</td>
<td>46</td>
<td>5.5</td>
</tr>
<tr>
<td>Non-Steiner Convex</td>
<td>H</td>
<td>NP-C</td>
<td>NP-C</td>
<td>?</td>
<td>62</td>
<td></td>
</tr>
<tr>
<td>Convex</td>
<td>S</td>
<td>$O(n^3)$</td>
<td>NP-hard</td>
<td>?</td>
<td>46</td>
<td>22</td>
</tr>
<tr>
<td>Convex</td>
<td>H</td>
<td>NP-hard</td>
<td>NP-hard</td>
<td>?</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>Star-shaped</td>
<td>G</td>
<td>?</td>
<td>NP-hard</td>
<td>?</td>
<td>44</td>
<td>50</td>
</tr>
</tbody>
</table>

Table 2.2: Time complexity of computing $\beta(P)$, $\gamma(P)$, $\delta(P)$ for a simple polygon in general position (G), for arbitrary simple polygons allowing collinear edges (S), or polygons with holes (H). NP-C stands for NP-complete.

convex polygons $C$.

**Theorem 2.8.** A simple polygon $P$ with $n$ vertices can be described by $\lfloor n/2 \rfloor + 1$ convex pieces, that is, $\delta_p^c(n) \leq \lfloor n/2 \rfloor + 1$.

**Proof.** Let $r$ be the number of reflex vertices of $P$. A polygon with $r$ reflex vertices can be partitioned into $r + 1$ convex pieces. If $r \leq \lfloor n/2 \rfloor$, this yields a description as claimed. If $r > \lfloor n/2 \rfloor$, then describe $P = \text{conv}(P) \setminus (Q_1 \cup \ldots \cup Q_k)$, where $Q_i$ is the polygon we enclose by going from the $i$-th vertex of $	ext{conv}(P)$ to the $(i + 1)$-th vertex of $	ext{conv}(P)$ (in counter-clockwise direction) and then back to the $i$-th vertex along $\partial P$ (in clockwise direction), see Figure 2.1. A convex vertex of $P$ either has become a reflex vertex of one of the $Q_i$’s, or it is a vertex of $	ext{conv}(P)$, too, thus responsible for one of the $Q_i$’s. All in all, after partitioning each $Q_i$ into convex pieces, we get at most one piece per convex vertex of $P$. So in total we have used the number of convex vertices plus one convex piece, which is $n - r + 1 < n - \lfloor n/2 \rfloor + 1 = \lfloor n/2 \rfloor + 1$.

Note that the number of facets of a primitive $C \in C$ cannot be bounded by any constant, therefore Theorem 2.6 is not applicable. But it is still pretty easy
to derive a lower bound on $\delta_{P,C}(n)$ implying that our upper bound is almost tight.

**Theorem 2.9.** For every integer $k \geq 2$, there is a simple polygon with $2k$ edges and one with $2k + 1$ edges that cannot be described by fewer than $k$ convex pieces. This implies $\delta_{P,C}(n) \geq \lfloor n/2 \rfloor$.

**Proof.** Consider polygons as shown in Figure 2.2, which we call combs. For any integer $k \geq 2$, we find an odd comb with $2k + 1$ edges as the one shown in the figure, on the left, and an even comb with $2k$ edges as shown on the right. Except for the base edge $b$ in the odd case, all edges are almost vertical having the interior of $P$ either to the left or to the right. A convex piece $C \in C$ can cover at most two (non-base) edges of a comb, namely at most one having the interior to the left and at most one having the interior to the right. Because in a description of $P$ each edge has to be covered by a primitive (see Observation 2.7), there are at least $k$ convex pieces. \qed

**Figure 2.2:** An odd comb (left) with base edge $b$ and an even comb (right).
Corollary 2.10. For every integer \( n \geq 3 \), there is a simple polygon with \( n \) edges that cannot be described by fewer than \( \lfloor n/2 \rfloor \) triangles.

So we know the values of the function \( \delta_{P,C}(n) \) up to \( \pm 1 \). So far, nothing has been said about the complexity of finding a minimum description. The fact that computing \( \gamma_{P,C}(P) \) is NP-hard \cite{22} suggests that the same might be true for \( \delta_{P,C}(P) \), which would imply that we should not hope for an efficient algorithm that finds a minimum description. But I do not know of any NP-hardness proof for this problem.
3 The Wireless Localization Problem

In the next proof we are going to make use of the principle of exhaustive intimidation.

Loosely based on a question or comment or remark by Prof. Alexander Barvinok

This chapter is devoted to the wireless localization problem. This problem, phrased as an art gallery problem, was introduced by Eppstein, Goodrich, and Sitchinava [29] in 2007, where it is called the sculpture garden problem. Our work on the topic was initiated by a problem posed by Bettina Speckmann at the 5th Gremo Workshop on Open Problems - GWOP 2007. The problem has the flavor of an art gallery problem, but instead of asking for a set of guards that sees the entire polygon, we are asked for a set of guards that describes the polygon.

A guard is modeled as a subset of the plane, namely the area where the broadcast from this guard can be received. This area can be described as an intersection or union of at most two halfplanes. So a guard is an unbounded cone in the plane. Using the terminology introduced in Chapter 2, the objects of interest \( \mathcal{P} \) are polygonal regions and the set of primitives \( \mathcal{Q} \) are polygonal regions with at most one vertex and a connected boundary. Then the wireless localization problem translates to finding a monotone description. But for the rest of the chapter, we are going to stick to the art gallery terminology introduced in [29] and call the primitives guards and a monotone description of a polygonal region \( \mathcal{P} \) a guarding of \( \mathcal{P} \).

**Natural guards.** Natural locations for guards are the vertices and edges of the polygon. A guard that is placed at a vertex of \( \mathcal{P} \) is called a vertex guard. A vertex guard is natural if it covers exactly the interior angle of its vertex. But natural vertex guards alone do not always suffice [29]. If \( \mathcal{P} \) is a convex polygon, placing a natural vertex guard on every other vertex is sufficient to describe \( \mathcal{P} \) as their intersection. A guard placed anywhere on the line given by an edge of
P and broadcasting within an angle of $\pi$ to the inner side of the edge is called a *natural edge guard*. Of course, we can place a natural edge guard on one of the vertices of its incident edge. Hence a natural edge guard can always be realized as a (not necessarily natural) vertex guard. Dobkin, Guibas, Hershberger, and Snoeyink [26] showed that $n$ natural edge guards are sufficient for any simple polygon with $n$ edges.

**Vertex guards.** Eppstein et al. [29] proved that any simple polygon with $n$ edges can be guarded using at most $n - 2$ (general, that is, not necessarily natural) vertex guards. More generally, they show that $n + 2(h - 1)$ vertex guards are sufficient for any simple polygon with $n$ edges and $h$ holes. Damian, Flatland, O’Rourke, and Ramaswami [23] describe a family of simple polygons with $n$ edges that require at least $\lceil 2n/3 \rceil - 1$ vertex guards. In a joint work with Aurosish Mishra we have improved the upper bound to $\lceil (8n - 6)/9 \rceil$, see Section 3.2.3.

**General guards.** In the general setting we do not have any restriction on the placement and the angles of guards. The first upper bound of $n - 2$ given in [29] has been improved to $\lceil \frac{4n-2}{5} \rceil$ in a joint work with Michael Hoffmann, Yoshio Okamoto, and Takeaki Uno [15]. In this thesis, the algorithm is further refined, which leads to the upper bound $\lceil (3n - 2)/4 \rceil$, see Section 3.2.6. If the polygon does not have collinear edges, then at least $\lceil n/2 \rceil$ guards are always necessary [29]. In [15] it is shown that there are polygons that need at least $\lceil \frac{3n-4}{5} \rceil$ general guards. Using a similar class of polygons, but refining the argument, we are able to improve the lower bound to $\lceil (2n - 4)/3 \rceil$. See Section 3.3.3.

There has been some progress since 2006, when O’Rourke [61] wrote that “the considerable gap between the $\lceil n/2 \rceil$ and $n - 2$ bounds remains to be closed.” The gap has narrowed but it has not been closed yet. The different variants of the wireless localization problem and the corresponding results are summarized in Table 3.1.

<table>
<thead>
<tr>
<th>guards</th>
<th>lower bound</th>
<th>upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>vertex</td>
<td>$\lceil 2n/3 \rceil - 1$ [23]</td>
<td>$\lceil (8n - 6)/9 \rceil$ [16]</td>
</tr>
<tr>
<td>general</td>
<td>$\lceil (2n - 4)/3 \rceil$</td>
<td>$\lceil (3n - 2)/4 \rceil$</td>
</tr>
</tbody>
</table>

**Table 3.1:** Number of guards needed for a simple polygon on $n$ vertices.
3.1 Notation and Basic Properties

Recall that a guard $g$ is a closed cone in the plane. Its boundary $\partial g$ is described by a vertex $v_g$ (called the apex of $g$ or simply its position) and two rays emanating from $v$ (see Figure 3.1). The ray $\ell_g$ that has the interior of the guard to its right is called the left ray of $g$, the other ray $r_g$ is called the right ray of $g$. The angle of a guard is the interior angle formed by its bounding rays. For a guard with angle $\pi$ (e.g., a natural edge guard), the apex and thus its position is not unique. Sometimes, we still speak of it as if it had a certain position. In such a case, just imagine it to be placed anywhere on its bounding edge. In particular, a natural edge guard $g_e$ on an edge $e$ can be positioned at one of the endpoints of $e$. Therefore, a natural edge guard is also a (non-natural) vertex guard.

![Figure 3.1: A convex guard $g$ with apex $v_g$, left ray $\ell_g$ and right ray $r_g$ and a reflex guard $g'$ with apex $v_{g'}$ and rays $\ell_{g'}$, $r_{g'}$.](image)

Recall the definitions from Chapter 2. Our objects of interest $\mathcal{P}$ are polygonal regions in the plane. Let $P \in \mathcal{P}$. In the general setting, where all kind of guards are allowed, the set of admissible primitives $Q(P)$ is the set of all closed cones in the plane, regardless of $P$. Then $\delta(P)$ denotes the minimum number of general guards needed to guard $P$. If we only allow vertex guards, then the set of allowed primitives $Q_v(P)$ is the set of guards with apex in $V(P)$ (this includes natural edge guards). We denote the number of vertex guards needed to guard $P$ by $\delta_v(P)$. Finally, let $Q_{\text{nat}}(P)$ denote the set of natural guards defined by $P$ and $\delta_{\text{nat}}(P)$ the number of guards in a minimum natural guarding. A guard $g$ covers an edge $e$ of $P$ completely if $e \subseteq \partial g$ and their orientations match, that is, the inner side of $e$ is on the inner side of $g$; $e$ is covered partly by $g$ if their orientations match and $e \cap \partial g$ is a proper subsegment of $e$ that is not just a single point. We call a guard a $k$-guard if it covers exactly $k$ edges completely. As $P$ is simple, a guard can cover at most one edge partly. If a guard covers an edge partly and $k$ edges completely, we call it a $k'$-guard. See Figure 3.2.

Assuming there are no collinear edges, a guard can cover at most two edges; then a natural vertex guard is a 2-guard and a natural edge guard is a 1-guard.
The wireless localization problem. A guarding $G(P)$ of a polygonal region $P$ is a formula composed of a set of guards and the operators union and intersection that defines $P$. The wireless localization problem is to find a guarding for a given simple polygon with as few guards as possible.

The set of guards that appear in a guarding $G(P)$ of $P$ is a monotone description of $P$. Note that by a guarding $G(P)$ of $P$ we mean a monotone description of $P$ together with a describing formula, as opposed to just a set of primitives. With a slight abuse of notation we write $g \in G(P)$ if a guard $g$ appears in $G(P)$.

3.2 Upper Bounds

In this section we will derive upper bounds for the number of guards needed to guard any simple polygon. Following an idea by Dobkin et al. \cite{26} we use the notion of polygonal halfplanes (see Section 1.4). After giving upper bounds for polygonal halfplanes, we use them to get results for simple polygons and general polygonal regions.

3.2.1 Polygonal Halfplanes

Recall that for a polygonal halfplane $H$, $\delta(H)$ denotes the minimum integer $k$ such that there exists a guarding $G(H)$ for $H$ using $k$ guards. Instead of finding guardings for general polygonal regions, we restrict our attention to polygonal
halfplanes for the rest of this section. So from now on the objects of interest \( P \) are the polygonal halfplanes. This means \( \delta(n) \) denotes the maximum number \( \delta(H) \) over all polygonal halfplanes \( H \) that are bounded by a chain with \( n \) edges. Similarly, \( \delta_v(n) \) denotes the number of vertex guards needed and \( \delta_{nat}(n) \) the number in the natural setting. Obviously, \( \delta_{nat}(1) = \delta_{nat}(2) = 1 \). The results of Dobkin et al. \[26\] imply that \( \delta(n) \leq \delta_v(n) \leq \delta_{nat}(n) \leq n \). Our main goal within this section is to improve this bound.

The (regularized) complement of a polygonal halfplane \( H \), denoted by \( H^c \), is a polygonal halfplane as well. In particular, the (regularized) complement of a guard \( g \), denoted by \( g^c \), is a guard as well.

**Observation 3.1.** Any guarding of any polygonal region \( P \) can be transformed into a guarding of \( P^c \) using the same number of guards.

**Proof.** Use de Morgan’s rules and invert all guards \( g \) to \( g^c \) (keep their location but flip the angle to the complement with respect to \( 2\pi \)), see Figure 3.3. Note that the resulting formula is indeed monotone because only guards complementary to the original ones appear (in SAT terminology: only negated literals); a formula is not monotone only if both a guard \( g \) and its complementary guard \( g^c \) appear in it.  

![Figure 3.3: A polygonal halfplane \( H \) with a guarding \( H = (a \cap b) \cup c \). Using de Morgan’s rules we get a guarding of the complement \( H^c = ((a\cap b)\cup c)^c = (a^c\cup b^c)\cap c^c \).](image)

By Observation 3.1 guarding a polygonal halfplane \( H \) or its complement \( H^c \) is actually the same problem, so we can define the guarding problem for chains and define \( \delta(C) := \delta(H) = \delta(H^c) \).

Our guarding scheme for chains is based on a recursive decomposition in which at each step the current chain is split into two or more subchains. At each split some segments are extended to rays. We have to be careful about the way these rays interact with the new subchains. If the split vertex lies on the convex hull, then the rays resulting from the segment extension do not intersect
the remaining chain. If the two rays $e_1$ and $e_n$ bounding a polygonal halfplane $H$ (look at the plane from a point very high above) form a convex angle ($\leq \pi$), then the convex hull is bounded by a chain that starts and ends with a ray parallel to (or possibly identical to) $e_1$ and $e_n$, see Figure 3.4. The vertices of $\text{conv}(H)$ are also vertices of $H$ and hence candidates for split vertices. However, if the rays $e_1$ and $e_n$ form an angle greater than $\pi$, then $\text{conv}(H)$ is the entire plane. Using Observation 3.1, we can always switch to the complement $H^c$ if necessary. It is not possible that both $\text{conv}(H)$ and $\text{conv}(H^c)$ are the entire plane.

![Figure 3.4: A polygonal halfplane $H$, $\text{conv}(H)$; $H^c$, and $\text{conv}(H^c)$ = $\mathbb{R}^2\mathcal{W}$.

3.2.2 Natural Guards

In the natural setting, it turns out that the bound $\delta_{\text{nat}}(n) \leq n$ cannot be improved by much.

**Theorem 3.2.** For any polygonal halfplane $H$, $\delta_{\text{nat}}(H) \leq n - 1$, that is, $H$ can be guarded using at most $n - 1$ natural guards.

**Proof.** We proceed by induction on $n$. We follow the proof of Dobkin et al. [26] with the only difference in the base case: A polygonal halfplane with 2 edges can be guarded by one natural vertex guard (actually, it is a natural vertex guard itself). Now let $C = \partial H$ be a chain $C$ of size $n \geq 3$. As usual, denote the sequence of edges of $C$ by $(e_1, \ldots, e_n)$ and let $v_i$, for $1 \leq i < n$, denote the vertex of $C$ incident to $e_i$ and $e_{i+1}$. The underlying (oriented) line of $e_i$, for $1 \leq i \leq n$, is denoted by $\overline{e_i}$. For $2 \leq i \leq n - 1$, let $e_i^+$ be the ray obtained from $e_i$ by extending the segment linearly beyond $v_i$. Similarly $e_i^-$ refers to the ray obtained from $e_i$ by extending the segment linearly beyond $v_{i-1}$. Let $e_1^+ = \overline{e_1}$ and $e_n^- = \overline{e_n}$.

Without loss of generality, we may assume that $e_1$ and $e_n$ form a convex angle and hence $\text{conv}(H)$ is a proper subset of the plane (see Observation 3.1).
3.2. Upper Bounds

Let $v_i$ be a vertex on $\partial \text{conv}(H)$. Split $C$ at $v_i$ into two chains $C_1 = (e_1, \ldots, e_i^+)$ and $C_2 = (e_{i+1}^-, \ldots, e_n)$. If $2 \leq i \leq n - 2$, then by induction there is a natural guarding $G(C_1)$ of $C_1$ using at most $i - 1$ natural guards and a natural guarding $G(C_2)$ of $C_2$ with at most $n - i - 2$ guards. So we obtain a natural guarding $G(C) = G(C_1) \cap G(C_2)$ using at most $n - 2$ guards. In the special cases $i = 1$ or $i = n - 1$, that is, if $v_i$ is the first or last vertex of $C$ and one of the chains $C_1$ and $C_2$ is just a line, we still obtain a guarding using $n - 1$ natural guards, because we can guard one chain with $n - 2$ guards and the line with one natural edge guard.

In Section 3.3.2 we are going to show that this bound is tight. There are polygonal halfplanes $H$ with $\delta_{\text{nat}}(H) = n(H) - 1$. A guarding for a simple polygon can easily be derived from guardings of polygonal halfplanes.

**Lemma 3.3.** Any simple polygon $P$ with $n(P) \geq 4$ can be expressed as an intersection of two polygonal halfplanes each of which has at least two edges.

**Proof.** Let $v_-$ and $v_+$ be the vertices of $P$ with minimal and maximal $x$-coordinate, respectively. If they are not adjacent along $P$, split the circular sequence of edges of $P$ at both $v_-$ and $v_+$ to obtain two sequences of at least two segments each. Transform each sequence into a chain by linearly extending the first and the last segment beyond $v_-$ or $v_+$ to obtain rays. As $v_-$ and $v_+$ are opposite extremal vertices of $P$, the two chains intersect exactly at these two points and nowhere else. Thus, the polygon $P$ can be expressed as an intersection of two polygonal halfplanes bounded by these chains.

If $v_-$ and $v_+$ are adjacent along $P$, without loss of generality we may assume that $P$ lies above the edge from $v_-$ to $v_+$. Rotate clockwise until another point $q$ has $x$-coordinate larger than $v_+$. If $q$ and $v_-$ are not adjacent along $P$, then split $P$ at these points as described above. Otherwise the convex hull of $P$ is the triangle $qv_-v_+$. In particular, $q$ and $v_+$ are opposite extremal vertices as well and they cannot be adjacent along $P$ because $P$ has more than three vertices. Therefore we can split at $q$ and $v_+$ as described above.

As a consequence we obtain an upper bound on the number of natural guards needed for a simple polygon. This bound turns out to be tight, as shown in Section 3.3. Observe that the statement is false for triangles, which require two guards even without the restriction to natural guards.

**Corollary 3.4.** Any simple polygon $P$ with $n \geq 4$ edges can be guarded using at most $n - 2$ natural guards.
Chapter 3. The Wireless Localization Problem

Proof. By Lemma 3.3, \( P \) can be described as an intersection of two polygonal halfplanes \( H_1 \) and \( H_2 \) each of which of size at least 2. By Theorem 3.2 we can guard them with \( n(H_1) - 1 + n(H_2) - 1 = n(P) - 2 \) guards.

Corollary 3.5. Let \( P \) be a polygonal region that can be described monotonically by \( k \) polygonal halfplanes, \( t \) of which are halfplanes, having \( n \) edges in total. Then \( \delta_{\text{nat}} P \leq n - k + t \).

This way we can easily treat polygons with holes and obtain a better bound as Eppstein et al. [29] gave for general (not necessarily natural) vertex guards. On the other hand, their result is slightly more general (triangles allowed) and stronger in the sense that the obtained formula is concise (a disjunction of conjunctions of constant size).

Corollary 3.6. Any simple polygon with \( n \geq 4 \) edges and \( h \) non-triangular holes can be guarded using at most \( n - 2(h + 1) \) natural guards.

3.2.3 Vertex Guards

If we allow general vertex guards, we can improve on the \( n - 1 \) upper bound. Treating the cases where we have to split off chains of size 1 separately, one can find guardings using roughly \( 8n/9 \) guards only. However, even to obtain this modest improvement, we have to indulge in a rather elaborate case distinction. There is no reason to believe that this bound is tight.

Theorem 3.7. For any \( n \geq 2 \), \( \delta_v(n) \leq \left\lfloor \frac{8n-3}{9} \right\rfloor \). This means that any polygonal halfplane \( H \) with \( n \geq 2 \) edges can be guarded by at most \( \left\lfloor \frac{8n-3}{9} \right\rfloor \) vertex guards.

Proof. The base cases \( \delta_v(1) = \delta_v(2) = 1 \), \( \delta_v(3) = 2 \), \( \delta_v(4) \leq 3 \), \( \delta_v(5) \leq 4 \) and \( \delta_v(6) \leq 5 \) follow from Theorem 3.2. Now let \( H \) be a polygonal halfplane bounded by an oriented polygonal chain \( C \) with \( n \geq 7 \) edges such that the interior of \( H \) lies to the right of \( C \). Let \( S := V(\text{conv}(V(C))) \) be the vertices of the convex hull of \( V(C) \), that is, the vertices of \( C \) that are extremal.

As in the natural setting, the basic idea is to split \( C \) at a vertex \( v_i \in S \) into two chains \( C_1 = (e_1, \ldots, e_{i-1}, e_i^+) \) and \( C_2 = (e_i^-, e_{i+2}, \ldots, e_n) \). If the “new” rays \( e_i^+ \) and \( e_{i+1}^- \) do not intersect the “old” rays \( e_1 \) and \( e_n \), we can express \( H \) as the intersection or union of the two polygonal halfplanes \( H_1 \) and \( H_2 \) bounded by \( C_1 \) and \( C_2 \) depending on whether \( v_i \) is convex or reflex.

Assume that the angle between \( e_1 \) and \( e_n \) is convex (else, consider \( H^c \) instead of \( H \), see Observation 3.1) and think of \( C \) as going from the left to the right (thus \( H \) being below \( C \)). In other words, \( e_1 \) and \( e_n \) are assumed to go from left
to right, \( e_1 \) having positive slope and \( e_n \) having smaller slope than \( e_1 \) or negative slope. Now look at the convex hull \( \text{conv}(H) \) of \( H \). There must be at least one vertex \( v_i \) in \( S \) which lies on the boundary \( \partial \text{conv}(H) \). Such a vertex is for sure a good splitting vertex in the above sense. If \( 2 \leq i \leq n - 2 \), we split \( C \) at \( v_i \) as explained into two chains \( C_1 := (e_1, \ldots, e_{i-1}, e_i^+) \) and \( C_2 := (e_{i+1}, e_{i+2}, \ldots, e_n) \) and get a guarding \( G(C_1) \cap G(C_2) \), where \( G(C_i) \) denotes the guarding of \( C_i \) (i.e., of the polygonal halfplane \( H_i \) bounded by \( C_i \)) we get by induction, see Figure 3.5. Therefore \( \delta_v(C) \leq \delta_v(i) + \delta_v(n-i) \leq \left\lfloor \frac{8i-3}{9} \right\rfloor + \left\lfloor \frac{8(n-i)-3}{9} \right\rfloor \leq \left\lfloor \frac{8n-6}{9} \right\rfloor \leq \left\lfloor \frac{8n-3}{9} \right\rfloor \).

**Figure 3.5:** Splitting at a convex hull vertex.

**Case 1.** If there is no vertex on \( \partial \text{conv}(H) \) with index \( 2 \leq i \leq n - 2 \), we first consider the case that \( S \cap \partial \text{conv}(H) = \{v_1, v_{n-1}\} \). If there is a vertex \( v_i \in S \) with \( 3 \leq i \leq n - 3 \), split \( C \) at \( v_1, v_i \) and \( v_{n-1} \): Put a natural edge guard \( g_1 \) onto \( e_1 \), a natural edge guard \( g_2 \) onto \( e_n \) and define \( C_1 := (e_2, \ldots, e_i^+) \), \( C_2 := (e_{i+1}, \ldots, e_{n-1}^+) \). (We can place the natural edge guards on the incident vertices, hence the natural edge guards can be realized as (non-natural) vertex guards.) Then, a guarding for \( C \) can be obtained as \( g_1 \cap g_2 \cap (G(C_1) \cup G(C_2)) \), see Figure 3.6. This implies \( \delta_v(C) \leq 2 + \delta_v(i-1) + \delta_v(n-i-1) \leq 2 + \left\lfloor \frac{8i-11}{9} \right\rfloor + \left\lfloor \frac{8(n-i)-11}{9} \right\rfloor \leq \left\lfloor \frac{8n-4}{9} \right\rfloor \).

**Case 1.1.** If \( S = \{v_1, v_2, v_{n-2}, v_{n-1}\} \), consider \( S' := V(\text{conv}(\{v_2, \ldots, v_{n-2}\})) \). Besides \( v_2 \) and \( v_{n-2} \), there must be a third vertex \( v_j \in S' \), without loss of generality \( 4 \leq j \leq n - 3 \) (if \( j = 3 \), reflect \( C \)). If \( e_3^- \) does not intersect \( e_n \), put a natural edge guard \( g_1 \) onto \( e_1 \), a vertex guard \( g_2 \) onto \( v_2 \) with the right ray covering \( e_2 \) and its other ray parallel to \( e_n \) and a natural vertex guard \( g_3 \) onto \( v_{n-1} \) and define \( C_1 := (e_3^-, \ldots, e_j^+) \) and \( C_2 := (e_{j+1}, e_{j+2}, \ldots, e_{n-2}, r) \) where \( r \) is the ray starting at \( v_{n-2} \) in the direction of \( e_n \). See Figure 3.7.
Then we get a guarding as \( g_1 \cap (g_2 \cup (G(C_1) \cap G(C_2)) \cup g_3) \) and conclude \( \delta_v(C) \leq 3 + \delta_v(j - 2) + \delta_v(n - j - 1) \leq 3 + \left[ \frac{8(j - 2) - 3}{9} \right] + \left[ \frac{8(n - j - 1) - 3}{9} \right] \leq \left[ \frac{8n - 3}{9} \right] \).

Figure 3.7: Case 1.1: \( S = \{v_1, v_2, v_{n-2}, v_{n-1}\} \)

If \( e_3^- \) intersects \( e_n \), put a natural vertex guard \( g_1 \) onto \( v_1 \), and an edge guard \( g_2 \) onto \( e_n \) and define \( C' := (e_3^-, \ldots, e_{n-1}^+) \). See Figure 3.7 on the right side. We obtain a guarding \( (g_1 \cup G(C')) \cap g_2 \), which leads to \( \delta_v(C') \leq 2 + \delta_v(n - 3) \leq 2 + \left[ \frac{8n - 24 - 3}{9} \right] = \left[ \frac{8n - 9}{9} \right] \).

Case 1.2. If \( S \) consists of 3 vertices only and there is no \( v_i \in S \) with \( 3 \leq i \leq n - 3 \), assume without loss of generality that \( v_2 \in S \) (if \( v_{n-2} \) is the only vertex in \( S \) besides \( v_1 \) and \( v_{n-1} \), reflect \( C \)). In this case, define \( S' := V(\text{conv}(\{v_2, \ldots, v_{n-1}\})) \). For sure, \( v_2, v_{n-1} \in S' \) but there must be a third vertex \( v_j \in S' \), see Figure 3.8.

If \( 4 \leq j \leq n - 3 \), put edge guards onto \( e_1, e_2 \), and \( e_n \), and split the remaining chain at \( v_j \). This results in \( \delta_v(C) \leq 3 + \delta_v(j - 2) + \delta_v(n - j - 1) \leq 3 + \left[ \frac{8n - 30}{9} \right] = \left[ \frac{8n - 9}{9} \right] \).

Case 1.2.1. If \( v_3 \in S' \), put a natural vertex guard \( g_1 \) onto \( v_1 \), and a non-natural vertex guard \( g_2 \) onto \( v_2 \) covering \( e_3 \) with its left ray and with the right ray parallel to \( e_1 \), and an edge guard \( g_3 \) onto \( e_n \): A guarding can be obtained.
3.2. Upper Bounds

![Figure 3.8](image-url)

**Figure 3.8:** Case 1.2: $v_1$, $v_2$, and $v_{n-1}$ are the only vertices in $S$.

as $g_3 \cap (g_1 \cup (g_2 \cap G(C'))) \cap S$ where $C' = (e_1^-, \ldots, e_{n-1}^+)$. See Figure 3.9. We conclude $\delta_v(C) \leq 3 + \delta_v(n - 4) \leq 3 + \left\lfloor \frac{8n - 32 - 3}{9} \right\rfloor = \left\lfloor \frac{8n - 8}{9} \right\rfloor$.

![Figure 3.9](image-url)

**Figure 3.9:** Case 1.2: $v_3 \in S'$.

**Case 1.2.2.** If $v_{n-2}$ is the only new vertex in $S'$, that is, $S' = \{v_2, v_{n-2}, v_{n-1}\}$, define $C' = (e_3^-, \ldots, e_{n-2}^+)$ and put a natural vertex guard $g_3$ onto $v_{n-1}$. If $e_3^-$ does not intersects $e_n$, put an edge guard $g_1$ onto $e_1$ and a vertex guard $g_2$ onto $v_2$ covering $e_2$ with its right ray and with its left ray parallel to $e_n$. We get a guarding $g_1 \cap (g_2 \cup (G(C') \cap g_3))$. If $e_3^-$ intersects $e_n$, put a natural vertex guard $g_1$ onto $v_1$ and an edge guard $g_2$ onto $e_n$ and observe that $H = g_2 \cap (g_1 \cup (G(C') \cap g_3))$. We conclude $\delta_v(C) \leq 3 + \delta_v(n - 4) \leq 3 + \left\lfloor \frac{8n - 32 - 3}{9} \right\rfloor = \left\lfloor \frac{8n - 8}{9} \right\rfloor$, see Figure 3.10.

![Figure 3.10](image-url)

**Case 2.** Assume there is only one vertex on $\partial \text{conv}(H)$, which is either $v_1$ or $v_{n-1}$. We assume without loss of generality that it is $v_1$. Besides $v_1$, which certainly belongs to $S$, there must be at least two more vertices in $S$. Let $v_i$ be the vertex of $S$ which is extremal orthogonally to the right of $e_n$. If $3 \leq i \leq n - 2$, split $C$ into three parts cutting it at $v_1$ and $v_i$. Then, we get a guarding for $C$ as $g \cap (G(C_1) \cup G(C_2))$, where $g$ is a natural edge guard on $e_1$, $C_1 = (e_2^-, \ldots, e_i^+)$, and $C_2 = (e_{i+1}-, \ldots, e_n)$, see Figure 3.11.
Case 2.1. If \( i = 2 \), define \( S' := V(\text{conv}(\{v_2, \ldots, v_{n-1}\})) \). Let \( v_j \) be the vertex of \( S' \) which is extremal in the opposite direction, that is, to the left of \( e_n \). If \( 4 \leq j \leq n-2 \), we place a natural edge guard \( g_1 \) on \( e_1 \) and a natural edge guard \( g_2 \) on \( e_2 \) and split the rest at \( v_j \) into two chains \( C_1 \) and \( C_2 \), see Figure 3.12. Then \( H = g_1 \cap (g_2 \cup (G(C_1) \cap G(C_2))) \).

Case 2.1.1. If \( j = 3 \), we put a natural vertex guard onto \( v_1 \), a guard onto \( v_2 \) with its left ray covering \( e_3 \) and the right parallel to \( e_1 \). Then, we get
\[\delta_v(C) \leq 2 + \delta_v(n - 3) \leq 2 + \left\lfloor \frac{8n - 27}{9} \right\rfloor \leq \left\lfloor \frac{8n - 9}{9} \right\rfloor.\]

**Case 2.1.2.** Consider the case \(j = n - 1\), that is, there is no vertex above \(e_n\) except \(v_1\). If there is any vertex \(v_s \in S'\) with \(4 \leq s \leq n - 3\), place an edge guard \(g_1\) on \(e_1\), an edge guard \(g_2\) on \(e_2\), and an edge guard \(g_3\) on \(e_n\). Then define \(C_1 = (e_3^-, \ldots, e_s^+)\) and \(C_2 := (e_{s+1}^-, \ldots, e_{n-1})\) to obtain a guarding \((g_1 \cap G(C_1) \cup G(C_2)))\) (or \(g_1 \cap (g_2 \cup (g_3 \cap G(C_1) \cap G(C_2)))\) if \(v_s\) is convex). We conclude \(\delta_v(C) \leq 3 + \delta_v(s - 2) + \delta_v(n - s - 1) \leq 3 + \left\lfloor \frac{8n - 30}{9} \right\rfloor = \left\lfloor \frac{8n - 3}{9} \right\rfloor\), see Figure 3.13

![Figure 3.13: Case 2.1.2: There is no vertex in \(S'\) above \(e_n\).](image)

If \(v_3 \in S'\), put 3 guards explicitly depending on whether \(v_3\) is reflex or convex and cover the remaining chain \(C' = (e_4^-, \ldots, e_{n-1}^+)\) recursively, see Figure 3.14

If \(v_3\) is reflex and \(e_4^-\) intersects \(e_1\), place a natural edge guard \(g_1\) on \(e_1\), a natural vertex guard \(g_2\) on \(e_2\), and a vertex guard \(g_3\) at \(v_{n-1}\) covering \(e_n\) with its left ray, the right ray being parallel to \(e_2^-\). If \(v_{n-2} \in S'\), too, then this guarding still works if we use another natural vertex guard \(g_4\) at \(v_{n-1}\), see the picture in the middle. If \(v_3\) is reflex and \(e_4^-\) does not intersect \(e_1\), use a natural vertex guard \(g_1\) at \(v_1\), a vertex guard \(g_2\) at \(v_2\) covering \(e_3\) with its left ray, the other ray parallel to \(e_1\), and a natural edge guard \(g_3\) covering \(e_n\). This works in both cases if \(S' = \{v_2, v_3, v_{n-1}\}\) as shown in the figure, on the right, or if \(S' = \{v_2, v_3, v_{n-1}\}\).

If \(v_3\) is convex, the direction of \(e_4\) does not matter. We can use a natural vertex guard on \(v_1\), and a non-natural vertex guard on \(v_2\) covering \(e_3\) and the other ray parallel to \(e_1\), see Figure 3.15 in the second row.

If \(S' = \{v_2, v_{n-2}, v_{n-1}\}\) we are in a situation similar to Case 1.1 or 1.2.2. If \(v_{n-2}\) is reflex, define \(S'' = V(\text{conv}(\{v_2, \ldots, v_{n-2}\}))\) and let \(v_s \in S''\), \(4 \leq s \leq n - 3\); let \(C_1 = (e_3^-, e_4^-, \ldots, e_s^+)\) and \(C_2 = (e_{s+1}^-, \ldots, e_{n-2}, r)\) where \(r\) is the ray starting at \(v_{n-2}\) parallel to \(e_n\). See Figure 3.16 on the left. Place natural edge guards \(g_1\) and \(g_2\) on \(e_1\) and \(e_2\) and a natural vertex guard \(g_3\) at \(v_{n-1}\). Then a
guarding of $H$ is given by the formula $g_1 \cap (g_2 \cup G(C_1) \cap G(C_2) \cup g_3)$. If $S'' = \{v_2, v_3, v_{n-2}\}$, we may assume that $e_{n-2}^+ \cap e_n$ (else we can split at $v_{n-2}$ as explained in Case 2). Then we get a guarding as shown in Figure 3.16 in the middle. Else if $v_{n-2}$ is convex, we can simply use a natural vertex guard $g_1$ on $v_1$, another natural vertex guard $g_2$ on $v_{n-1}$ and define $C' = (r', e_3, \ldots, e_{n-2}^+)$ where $r'$ is the ray starting at $v_2$ parallel to $e_1$: $H = g_1 \cup (G(C') \cap g_2)$, see Figure 3.16 on the right.

**Case 2.2.** Finally, consider the case $i = n - 1$, that is, there is no vertex of $S$ below $e_n$. If there is a convex vertex $v_j \in S$ with $3 \leq j \leq n - 3$, place edge guards onto $e_1$ and $e_n$ and split at $v_j$; if there is a reflex vertex $v_j \in S$ with $3 \leq j \leq n - 2$, place an edge guard onto $e_1$ and split at $v_j$, see Figure 3.17.

**Case 2.2.1.** Consider the case $S = \{v_1, v_2, v_{n-2}, v_{n-1}\}$ and $v_{n-2}$ convex. (The case where $v_{n-2}$ is reflex has already been treated under Case 2.2.) If $v_2$ is also convex, we can use a natural vertex guard at $v_1$ a natural vertex guard at $v_{n-1}$ and a non-natural vertex guard at $v_{n-1}$ guarding $e_n$ with its left ray and with
3.2. Upper Bounds

Figure 3.16: Case 2.1.2c: \( S' = \{v_2, v_{n-2}, v_{n-1}\} \).

Figure 3.17: Case 2.2: There is no extremal vertex below \( e_n \).

the right ray parallel to \( e_1 \). If \( v_2 \) is reflex, consider \( S' := V(\text{conv}(\{v_2, ..., v_{n-2}\})) \) and look for a splitting vertex in \( S' \), see Figure 3.18.

Figure 3.18: Case 2.2.1: \( S = \{v_1, v_2, v_{n-2}, v_{n-1}\} \), \( v_2 \) reflex, and \( v_{n-2} \) convex.
Case 2.2.2 If $v_2$ is the only vertex in $S$ besides $v_1$ and $v_{n-1}$ and it is convex, then define $S' := V(\text{conv} \{v_2, \ldots, v_{n-1}\})$. If there is a $v_s \in S'$ with $4 \leq s \leq n-3$, then place natural edge guards on $e_1$, $e_2$, and $e_n$, and split the rest at $v_s$, see Figure 3.19.

Figure 3.19: Case 2.2.2: $S = \{v_1, v_2, v_{n-1}\}$ and $v_2$ is convex.

If $S' = \{v_2, v_3, v_{n-1}\}$, use a natural vertex guard $g_1$ on $v_1$, $g_2$ on $v_{n-1}$, and a vertex guard $g_3$ on $v_{n-1}$ with $\ell_{g_3} = e_n$ and $r_{g_3}$ parallel to $e_1$, see Figure 3.20.

Figure 3.20: Case 2.2.2-a: $S' = \{v_2, v_3, v_{n-1}\}$.

If $S' = \{v_2, v_{n-2}, v_{n-1}\}$, use a natural vertex guard $g_1$ on $v_1$, $g_2$ on $v_{n-1}$, and a vertex guard $g_3$ on $v_{n-1}$ with $\ell_{g_3} = e_n$ and $r_{g_3}$ parallel to $e_1$, see Figure 3.21.

Case 2.2.3 If $S = \{v_1, v_2, v_{n-1}\}$ and $v_2$ is reflex, let $S' = V(\text{conv} \{v_2, \ldots, v_{n-1}\})$ and look for a splitting vertex in $S'$. If there is $v_j \in S'$, $3 \leq j \leq n-3$, place
a natural vertex guard $g_1$ at $v_1$ and a vertex guard $g_2$ on $v_{n-1}$ covering $e_n$. Let $r$ be a ray starting at $v_2$ parallel to $e_1$; define $C_1 := (r, e_3, \ldots, e_j^+)$ and $C_2 := (e_{j+1}^-, \ldots, e_{n-1}^+)$ to obtain a guarding $g_1 \cup (G(C_1) \cap G(C_2)) \cup g_2$, see Figure 3.22.

Figure 3.22: Case 2.2.3: $S = \{v_1, v_2, v_{n-1}\}$ and $v_2$ is reflex.

If $S' = \{v_2, v_{n-2}, v_{n-1}\}$, define $S'' := V(\text{conv}(\{v_2, \ldots, v_{n-2}\}))$ and look for a splitting vertex $v_s \in S''$, $4 \leq s \leq n - 3$, see Figure 3.23 on the left. If there is no such vertex, that is, if $S'' = \{v_2, v_3, v_{n-2}\}$, then distinguish between the case that $e_4^-$ does not intersects $e_1$ (upper right) or intersects $e_1$ (lower right).

Case 2.2.4 If $v_{n-2}$ is the only vertex in $S$ besides $v_1$ and $v_{n-1}$ and it is convex, then proceed as above but now removing $v_1$ or $v_{n-1}$, respectively, defining $S' := V(\text{conv}(\{v_1, \ldots, v_{n-3}\}))$, see Figure 3.24.

The above analysis yields a recursive algorithm to construct a guarding using at most $\left\lfloor \frac{8n-3}{9} \right\rfloor$ vertex guards. Using an appropriate data structure, it can be implemented to run in time $O(n \log n)$: Store the input chain $C$ as an array...
Figure 3.23: Case 2.2.3a: \( S' = \{v_2, v_{n-2}, v_{n-1}\} \).

Figure 3.24: Case 2.2.4: \( S = \{v_1, v_{n-2}, v_{n-1}\} \).

\((e_1, \ldots, e_n)\) of its edges. Each edge \( e_i \) in turn is represented by its direction \( d_i \) and its target vertex \( v_i \) (the latter being undefined for \( e_n \)). A subchain \((e_i, \ldots, e_j)\) of \( C \) is represented by its bounding indices \( i \) and \( j \). Apart from constant time geometric primitives, such as testing whether two given rays intersect, the algorithm needs to find extreme points among a contiguous subsequence \( V_{i,j} := (v_i, \ldots, v_j) \), for some \( 1 \leq i \leq j < n \), of vertices from \( C \). Using a compact interval tree [39] on the vertices of \( C \), we can find extreme points for any \( V_{i,j} \), \( 1 \leq i \leq j < n \), in \( O(\log n) \) time after \( O(n \log n) \) preprocessing. As in each step the current chain is split, the number of steps is linear and the overall run-time is \( O(n \log n) \).
Corollary 3.8. A simple polygon $P$ on $n \geq 4$ edges can be guarded with at most $\left\lfloor \frac{(8n - 6)}{9} \right\rfloor$ vertex guards, that is, $\delta_v(P) \leq \left\lfloor \frac{(8n - 6)}{9} \right\rfloor$. Such a guarding can be obtained in time $O(n \log n)$.

3.2.4 Guarding Chains for small $n$

If we allow general guards, the problem becomes considerably easier. One of the advantages is that it is possible to employ 2-guards that are not natural vertex guards. In other words, two non-adjacent edges $e$ and $f$ can be covered by one guard located at the intersection $e \cap f$ of the lines defined by them provided their orientations match. See Figure 3.25 for an example of a 5-chain that needs 4 vertex guards, but can be guarded by 3 general guards.

Figure 3.25: A polygonal halfplane that needs 4 vertex guards, but can be guarded with 3 general guards (left). If we allow parallel edges, there are 5-chains that cannot be guarded with 3 guards (right).

If $\overline{e}$ and $\overline{f}$ are parallel, a guard cannot cover $e$ and $f$ simultaneously, as shown in Figure 3.25 on the right side. As a warm-up, we can prove that for this 5-chain where $\overline{e_1}$ and $\overline{e_5}$ are parallel, one needs at least 4 general guards.

Lemma 3.9. There are a 5-chains that cannot be guarded by 3 guards.

Proof. Consider the polygonal halfplane $H$ shown in Figure 3.25 on the right. In order to guard it with 3 guards, we have to employ either at least two 2-guards or two 1'-guards and one 2-guard. The second alternative is impossible, because there are no two 1'-guards that jointly cover an edge completely they both cover partly. So we may assume there is a guarding using two 2-guards and one 1- or 1'-guard. Then we find that the two 2-guards have to be natural
vertex guards, because using the 2-guard at \( a \) (which is the only possible 2-guard that is not a natural vertex guard) leaves three edges uncovered, but no two of them can be covered by the same guard. So we have two natural vertex guards covering four of the five edges. By symmetry there remain only three cases to check. In all of them we find that it is impossible to place the third guard anywhere on the line defined by the uncovered edge in order to get a valid guarding.

If we allowed guards to be placed at infinity, in other words, if a stripe bounded by two parallel lines and its complement were defined to be valid guards as well, this problem would not occur anymore. We are going to avoid the trouble by forbidding the polygonal regions to have parallel edges. So from now on, the objects of interest are restricted to polygonal halfplanes \( H \) with the property that for any \( e, f \in E(H) \), \( e \) and \( f \) intersect. But keep in mind that one could drop this condition at the price of a looser definition of guards.

The goal of this section is to prove upper bounds for \( \delta(n) \) for small \( n \); see the last row of Table 3.2. The values in the first and the middle row follow from the preceding sections. Note that usually we prove upper bounds only, but in fact all the bounds in the table are tight, which can be shown with some extra effort using arguments as sketched in the proof of Lemma 3.9.

If \( n(H) = 1 \) or \( n(H) = 2 \), \( H \) is a guard itself, so \( \delta(1) = \delta(2) = 1 \). If \( n(H) = 3 \) or \( n(H) = 4 \), we know that we always find a natural guarding with 2 or 3 guards, respectively. It is obvious that no 3-chain can be guarded by fewer than 2 guards, so \( \delta(3) = 2 \). The case \( n(H) = 4 \) is the first non-trivial case. There are polygonal halfplanes bounded by a 4-chain that can be guarded by two guards and others that need three guards. We call a 4-chain \( C \) \textit{bad}, if \( \delta(C) = 3 \).

**Lemma 3.10.** A 4-chain without parallel edges \( C = (e_1, e_2, e_3, e_4) \) is bad if and only if it is of the following form: \( \overline{e_1} \cap \overline{e_4} \) lies in the interior of \( e_1 \) or \( e_4 \), and \( v_2 \) lies in the region \( R \) bounded by \( e_1, e_4, e'_1 \), and \( e'_4 \), where \( e'_1 \) is the ray starting

<table>
<thead>
<tr>
<th>Number of edges ( n )</th>
<th>( \delta_{\text{nat}}(n) )</th>
<th>( \delta_v(n) )</th>
<th>( \delta(n) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 3.2: Upper bounds on the maximum number of guards needed for an \( n \)-chain (without parallel edges) for small \( n \): in the natural setting; allowing general vertex guards; and in the general setting.
at $v_3$ in the same direction as $e_1$ and $e'_4$ is the ray starting at $v_1$ in the same direction as $e_4$, see Figure 3.26 and Figure 3.28.

![Figure 3.26: The region $R$ bounded by $e_1, e_4, e'_1,$ and $e'_4$.](image)

**Proof.** Let $C$ be a 4-chain. Let $H$ be the polygonal halfplane bounded by $C$ such that $e_1$ and $e_4$ describe a convex angle. If $e_1$ and $e_4$ do not intersect in the interior of $e_1 \cup e_4$, then we find a guarding with two guards in any possible case, see Figure 3.27. If $v_2$ is inside the angle, place a 2-guard at $e_1 \cap e_4$ guarding this angle and a natural vertex guard at $v_2$. If $v_2$ is outside this angle, put a natural vertex guard onto $v_1$ and a natural vertex guard onto $v_3$.

![Figure 3.27: If the lines defined by $e_1$ and $e_4$ meet outside $e_1$ and $e_4$, there is a guarding with 2 guards.](image)

Else, without loss of generality (reflect $H$ if necessary) assume they intersect inside $e_1$. See Figure 3.28. If $v_2$ is outside the region, we can put a natural vertex guard $g_1$ onto $v_1$ and a natural vertex guard $g_2$ onto $v_3$. In the case where $v_2$ is outside $\text{conv}(H)$, we get a guarding $H = g \cap h$. In the case where $v_2$ is below the chain $(e'_1, e_4)$, $H = g \cup h$.

If $v_2$ is inside the described region, we observe that there is no guarding with only two guards, see Figure 3.28 on the right: We have no choice but to use two natural vertex guards as shown. But then the point depicted by a cross gets included even though it is outside $H$.  

\[\square\]
Figure 3.28: If \( v_2 \) is outside the region \( R \) bounded by \((e_1, e_4')\) and \((e_1', e_4)\) (shaded), \( H \) (shaded darker) can be guarded with two guards. If \( v_2 \) is inside, two guards are not sufficient.

Note that if parallel edges were allowed, there would be bad 4-chains of another form, too. For example, if \( e_1 \) and \( e_4 \) are parallel pointing into the same direction, and \( v_2 \) is a reflex vertex, then the chain \((e_1, e_2, e_3, e_4)\) cannot be guarded with two guards only.

The case \( n(H) = 5 \) is more intricate. We have already seen that in order to show \( \delta(5) = 3 \), it is crucial to forbid parallel edges.

**Lemma 3.11.** For any 5-chain \( C \) without parallel edges, \( \delta(C) = 3 \).

**Proof.** \( \delta(C) \geq 3 \) follows directly from Theorem 2.6. Let \( H \) be the polygonal halfplane bounded by \( C \) such that \( e_1 \) and \( e_5 \) describe a convex angle. Consider \( \text{conv}(H) \). If \( v_2 \) or \( v_3 \) lie on \( \partial \text{conv}(H) \), we can split \( C \) into a 2-chain and a 3-chain and we can compose the two guardings to a guarding of \( C \) using 3 guards. If both \( v_1 \) and \( v_4 \) lie on the convex hull, put a 2-guard onto \( e_1 \cap e_5 \) and guard \( C' = (e_1, e_3, e_4, e_5) \) using only two guards. So let \( C' \) be a bad chain. We distinguish two cases. Either \( v_2 \) lies below \( e_5 \) and \( v_3 \) lies inside the region \( R \) as described in Lemma 3.10, see Figure 3.29, on the left; or there is no vertex below \( e_5 \) as shown in Figure 3.29, on the right.

In the first case let \( v' \) be the point where \( e_3 \) and \( e_1 \) intersect, \( e_1^* \) be the ray starting at \( v' \) going into the same direction as \( e_1 \), and \( e_3^* \) be the segment from \( v' \) to \( v_3 \). \( R \) lies completely above the line parallel to \( e_5 \) through \( v_2 \), so \( e_3^* \)
intersects $e_1$ somewhere below $\overline{e_5}$ (i.e., farther away from $v_1$). Therefore, the chain $C'' = (e_1^*, e_3, e_4, e_5)$ is a good 4-chain. So we can guard the halfplane $H$ bounded by $C$ by putting a natural vertex guard onto $v_1$ and taking the union with a guarding of $H''$, which is the halfplane bounded by $C''$, using two guards.

In the second case let $w$ be the intersection of $\overline{e_5}$ and $e_1$. Place a $1'$-guard $g$ at $w$ covering $e_5$ completely and $e_1$ partly. Define $C'' = (e_1, e_2, e_3, e_4^+)$. Note that $C''$ cannot be a bad 4-chain. However, it might happen that $C''$ is not simple but defines a simple polygon $P$ with 4 edges as shown in the picture. But we can guard it with two natural vertex guards anyway, so together with $g$ we get a guarding for $H$ using 3 guards.

Figure 3.29: Guarding 5-chains.

The cases $n(H) = 6$ or $n(H) = 7$ are easier to treat. Split a 6- or 7-chain into two smaller chains at a convex hull vertex, which immediately leads to $\delta(6) \leq 4$ and $\delta(7) \leq 5$. The case $n(H) = 8$, however, poses a challenge: The split argument would only give $\delta(8) \leq 6$, but we want to prove $\delta(8) = 5$.

Given a chain $C = (e_1, e_2, \ldots, e_n)$, we define the line arrangement $L(C)$ of $C$ as the set of lines $\{\overline{e_1}, \overline{e_2}, \ldots, \overline{e_n}\}$. A line arrangement is in general position if no two of its lines are parallel and no three meet in a common point. (Line arrangements in general position are often called simple in the literature.) First we compute a list of all combinatorially different line arrangements with 8 lines in general position. (Two line arrangements are combinatorially equivalent if they have the same dissection type, see the next subsection for an exact definition.)

Then for each line arrangement $L = \{\ell_1, \ldots, \ell_8\}$ in the list, we construct all possible 8-chains $C$ with $L(C) = L$. To do so, one goes through all 8!
permutations of the lines and then tries to construct chains $C = (e_1, \ldots, e_8)$ with $\overline{e_i} = \ell_i$. The positions of the vertices $v_1, \ldots, v_7$ are already given: $v_i = \ell_i \cap \ell_{i+1}$. As a consequence, the edges $e_i$ are given, too, for $2 \leq i \leq 7$. What remains to decide are the direction of the first ray $e_1$ on $\ell_1$ and the direction of the last ray $e_8$ on $\ell_8$. So after checking if the chain is simple for all four possibilities (which most of the time is not the case), we try to find a guarding using at most 5 guards. It turns out that it suffices to look for guardings that are “determined” by the line arrangement $L$: We say a guard $g$ is arrangement-natural with respect to a line arrangement $L$ if its apex $v_g$ lies on a vertex of $L$ (an intersection of two lines of $L$) and its two rays $r_g$ and $\ell_g$ are collinear with lines of the arrangement.

In our case, $L$ consists of 8 lines giving rise to 28 vertices. At each vertex there are 4 possible arrangement-natural convex guards, 4 possible arrangement-natural reflex guards and 4 possible arrangement-natural edge guards, so in total there are $12 \cdot 28 = 336$ possible arrangement-natural guards.

A computer program was used to go through all 5-element subsets until it found a subset that is a guarding. It turns out that there is such a guarding for all possible chains derived from all possible combinatorially different line arrangements.

**Lemma 3.12.** For any 8-chain $C$ without parallel edges, there is a guarding using at most 5 arrangement-natural guards. Consequently, $\delta(8) \leq 5$.

**Proof.** Let $C = (e_1, \ldots, e_8)$ be an 8-chain without parallel edges. The edges of $C$ define a line arrangement $L(C) = \{\overline{e_1}, \ldots, \overline{e_8}\}$.

First we show that we may assume that no three lines of $L$ meet in a common vertex: An 8-chain is defined by the coordinates of its 7 vertices and the directions of the first and the last ray, so we can think of an 8-chain as a point in a 16-dimensional space $V$. Let $C$ be an 8-chain without parallel edges whose line arrangement $L(C)$ is not in general position (not simple). For any $\varepsilon > 0$ we find a chain $C_\varepsilon$ that is $\varepsilon$-close to $C$ (in the space $V$) such that $L(C_\varepsilon)$ is in general position. Furthermore, if all the vertices of $L(C)$ lie within at most distance $M$ from the origin, then we may assume that the vertices of $L(C_\varepsilon)$ are not too far from the origin either, say, within at most $2M$ independent of the choice of $\varepsilon > 0$. (Here we use the assumption that $C$ has no parallel edges. Because there is no way to bring a line arrangement with parallel lines into general position by slightly perturbing the lines without creating new vertices arbitrarily far away from the origin.) Then, if we find an arrangement-natural guarding $G(C_\varepsilon)$ of $C_\varepsilon$ using 5 guards for all $\varepsilon > 0$, this carries over to a guarding of $C$: The coordinates of the guards are bounded by $2M$ and the angular range of a guard is given by two numbers in $[0, 2\pi]$. Therefore, the set of...
arrangement-natural guardings of $C_\varepsilon$ is bounded in the abstract space $W$ of all 5-guardings independently of $\varepsilon$. So the sequence of guardings $(G(C_\varepsilon))_{\varepsilon \to 0}$ has a convergent subsequence with limit $G^*$ in $W$. It remains to show that $G^*$ is a guarding of $C$. Let $H$ be a polygonal halfplane bounded by $C$. Consider a pair of points $(p, q)$ generic with respect to $G^*$, $p$ in the interior of $H$, $q$ outside of $H$ (an inside/outside pair, see Section 2.3). We have to show that there is a guard $g$ in $G^*$ that distinguishes $p$ and $q$. For $\varepsilon$ small enough, $(p, q)$ is also an inside/outside pair with respect to $C_\varepsilon$ and $G(C_\varepsilon)$. So for any $\varepsilon > 0$ small enough, we find a guard $g_\varepsilon \in G(C_\varepsilon)$ distinguishing $p$ and $q$. $(g_\varepsilon)_{\varepsilon \to 0}$ has a convergent subsequence the limit of which is a guard in $G^*$ that distinguishes $p$ and $q$.

There is a line arrangement $L' = \{\ell_1, \ldots, \ell_n\}$ in the list of all combinatorially different line arrangements that is isomorphic to $L(C)$: This means that there is a permutation $\phi$ of the indices $1, 2, \ldots, 8$ and an homeomorphism $\psi$ of the plane to itself that maps $e_i$ to $\ell_{\phi(i)}$. Let $C'$ be the image of $C$ under $\psi$. Observe that $L(C') = L'$. The computer program has checked $C'$ at some point and has found an arrangement-natural guarding $G'$ with respect to $L'$ consisting of at most 5 guards. An arrangement-natural guard $g'$ with respect to $L'$ can be mapped back to an arrangement-natural guard $g = \psi^{-1}(g')$ with respect to $L$. So let $G = \psi^{-1}(G') = \{\psi^{-1}(g) : g \in G'\}$. If $G'$ is a valid guarding of $C'$, then $G$ is a valid guarding of $C$.

The proof of Lemma 3.12 uses the assumption $n = 8$ only to infer that there is always an arrangement-natural 5-guarding (which is what the computer program has checked). Therefore, the lemma could also be formulated more generally: If for all (at most four) $n$-chains giving rise to a (labeled) simple line arrangement of $n$ lines, there is an arrangement-natural guarding using $k$ guards, and this is true for all possible combinatorially different labeled simple line arrangements, then there is a guarding for every $n$-chain using $k$ guards.

3.2.5 Combinatorially Different Line Arrangements

We briefly describe how a list of all combinatorially different simple line arrangements can be obtained for small $n$. We try to keep the terminology as simple as possible, but for a deeper understanding it is inevitable to undertake a journey into the beautiful land of oriented matroids: get a copy of the introduction to oriented matroids by Björner et al. 4 and take a week off, but beware—some people have never returned.

First we have to define what we mean by “combinatorially different”. Let $L = (\ell_1, \ldots, \ell_n)$ be an ordered (labeled) line arrangement. Without loss of generality, we assume that no line passes through the origin. For any line,
fix an arbitrary orientation. Let \( \ell_{n+1} \) denote an imaginary line at infinity circumscribing the whole plane. For an oriented line \( \ell \) that has the origin to its left, let \( \ell^* = (a, b, 1) \in \mathbb{R}^3 \) denote the vector containing the coefficients of the equation describing \( \ell: ax + by + 1 = 0 \). (The point \((a, b)\) is often called the polar dual of \( \ell \).) If \( \ell \) has the origin to the right, let \( \ell^* = -s^* \) where \( s \) is the reorientation of \( \ell \). Consequently, \( \ell^*_{n+1} \) is defined to be \((0, 0, 1)\).

Then we can define the chirotope of \( L \) as follows: for any triple of indices \( 1 \leq i < j < k \leq n + 1 \), define \( \chi(i, j, k) \) to be \(+, -\), or 0, depending on the orientation of the triple \((\ell^*_i, \ell^*_j, \ell^*_k)\). If it is oriented positively (the determinant of the 3-matrix defined by the triple is positive), set \( \chi(i, j, k) := + \); if it is oriented negatively, set \( \chi(i, j, k) := - \); if the determinant is 0, set \( \chi(i, j, k) := 0 \).

Intuitively, \( \chi(i, j, k) \) captures how the three oriented lines \( \ell_i, \ell_j, \) and \( \ell_k \) lie to each other. If \( k = n + 1 \), it tells us how the two oriented lines \( \ell_i \) and \( \ell_j \) lie to each other “with respect to infinity”: \( \chi(i, j, n + 1) \) is positive if and only if the intersection of the two halfplanes defined by \( \ell_i \) and \( \ell_j \) is oriented counterclockwise, meaning that walking around this cone in counter-clockwise order coming from infinity we first walk along \( \ell_i \) until the intersection with \( \ell_j \) and then back to infinity along \( \ell_j \).

Two (unordered and unoriented) line arrangements \( L \) and \( L' \) have the same dissection type if they can be ordered and oriented in such a way that they have the same chirotope. (In the language of oriented matroids, the dissection type is the isomorphy class of an oriented matroid under reorientation and relabeling of the elements fixing an infinity element.) This definition captures what we intuitively expect: Two line arrangements have the same dissection type if and only if they are isomorphic in the sense of the proof of Lemma 3.12.

Two vector configurations \( L^* = \{\ell^*_1, \ldots, \ell^*_{n+1}\} \) and \( S = \{s^*_1, \ldots, s^*_{n+1}\} \) have the same projective order type if they can be ordered and reoriented such that they result in the same chirotope. So if two line arrangements have the same dissection type, the associated vector configurations have the same projective order type. The converse, however, is not true. The difference to the definition of dissection type is that there is no fixed element (the “line at infinity”) anymore. If we are given a set \( P \subset \mathbb{R}^2 \) of \( n + 1 \) points we define its projective order type as the projective order type of the vector configuration in \( \mathbb{R}^3 \) that we get by appending the coordinate 1 to each point. \( P \) can be transformed into an oriented line arrangement with \( n \) lines by fixing one of its elements as the “infinity element”. More precisely, fix an index \( 1 \leq i \leq n + 1 \). First translate the point set such that \( p_i = (0, 0) \). Then take the polar dual of the remaining \( n \) points, that is, for \( j \neq i \), \( p_j = (a, b) \) gets mapped to the line \( p^*_j \) defined by the equation \( ax + by + 1 = 0 \). This results in a line arrangement \( \{p^*_j\}_{j \neq i} \) of \( n \) lines. Now consider an arbitrary line arrangement \( L \) of \( n \) lines. Let \( P \) be a
point configuration of \( n + 1 \) points such that \( P \) has the same projective order type as the vector configuration \( L^* = \{\ell_1^*, \ldots, \ell_{n+1}^*\} \) associated with \( L \). Then there is an \( 1 \leq i \leq n + 1 \) such that the line arrangement resulting from \( P \) by fixing \( p_i \) as the infinity point has the same dissection type as \( L \).

I do not know of any database containing geometric realizations as point sets of all realizable projective order types for small \( n \), but there is the database of affine order types of point sets in general position provided by Aichholzer, Aurenhammer, and Krasser \[1\] up to \( n = 11 \). A point set is said to be in \textit{general position} if no three points lie on a common line. The projective order types can be obtained by going through the affine order type data base and filtering out point sets that have the same projective order type as an other point set already obtained. Of course, we miss all the projective order types that arise from point sets that are not in general position. Being in general position translates to not having any 0-entries in the chirotope. (An oriented matroid the chirotope of which has no 0-entries is called \textit{uniform}.) For each of the uniform projective order types, we go through all \( n + 1 \) possibilities to fix one point as the point at infinity resulting in \( n + 1 \) different line arrangements of \( n \) lines. However, some of these \( n + 1 \) line arrangements can have the same dissection type, so we have to filter them out, keeping only one for each dissection type. This results in a list of combinatorially different line arrangements in general position (simple line arrangements). Consequently, general position means that no three lines intersect in one point and no two are parallel. (If two lines \( \ell_i \) and \( \ell_j \) are parallel, it means that \( \ell_i, \ell_j \) and \( \ell_{n+1} \) “intersect at infinity”: \( \chi(i, j, n + 1) = 0 \).)

Up to \( n = 8 \) and, with some patience, \( n = 9 \) it is possible to check if two chirotopes stand for the same projective order type using brute force: we can go through all possible relabelings and reorientations and compute one unique representative. In accordance with the notation used by Finschi and Fukuda \[32, 31\], we use the relabeling and reorientation that results in the lexicographically maximal chirotope as a unique representative for the whole class. For \( n = 10 \), we used a different kind of representative for each projective class, as it gets to costly to go through all possible relabellings. The same method appears implicitly in Krasser’s work \[49\]: For every possible reorientation of the points we sort them in counter-clockwise order seen from a point on the convex hull. Among the different points on the convex hull we take the one resulting in a lexicographically maximal chirotope. Note that this reorientation and relabeling in general does not result in the lexicographically maximal chirotope, that is, we do not get the oriented matroids as in Finschi’s list. But this method is guaranteed to find a unique member of the class, which is all we need to filter out point sets that have the same projective type as another point set already obtained.
Table 3.3 shows the numbers of combinatorially different line arrangements (in general position) up to 9 lines (in oriented matroid language: uniform realizable dissection types of dimension 2 up to cardinality 9). A database containing the different line arrangement up to \( n = 9 \) lines, and—as a side product—the point configurations of Aichholzer et al. that have different projective order type can be accessed online at [http://www.inf.ethz.ch/personal/christt/line_arrangements.php](http://www.inf.ethz.ch/personal/christt/line_arrangements.php).

### 3.2.6 General guards

**Theorem 3.13.** \( \delta(1) = 1, \delta(4) = 3 \) and \( \delta(n) \leq \left\lfloor \frac{3n-1}{4} \right\rfloor \) for any \( n \geq 2, n \neq 4 \). In words, any polygonal halfplane with \( n \geq 2, n \neq 4 \) pairwise non-parallel edges can be guarded by at most \( \left\lfloor \frac{3n-1}{4} \right\rfloor \) general guards.

**Proof.** We know the exact values for \( \delta(n) \) up to \( n = 8 \). For bigger \( n \) we proceed by induction. Let \( C \) be a chain with \( n \geq 9 \) edges. As usual we denote the sequence of edges defined by \( C \) by \((e_1, \ldots, e_n)\) and the sequence of vertices by \((v_1, \ldots, v_{n-1})\), that is, \( v_i \) is incident to \( e_i \) and \( e_{i+1} \). We use the same notation as in the proof of Theorem 3.2.

Assume that the angle between \( e_1 \) and \( e_n \) is convex (else, consider \( H^c \) instead of \( H \)) and think of \( C \) as going from the left to the right, the interior of \( H \) being below \( C \).

If there is any vertex \( v_i \) on \( \partial \text{conv}(H) \), for some \( 2 \leq i \leq n-2 \), then split \( C \) into two chains \( C_1 = (e_1, \ldots, e_i^+) \) and \( C_2 = (e_{i+1}^-, \ldots, e_n) \). We obtain a guarding for \( C \) as \( G(C_1) \cap G(C_2) \). By assumption, \( i \geq 2 \) and \( n-i \geq 2 \), so if additionally \( i \neq 4 \) and \( n-i \neq 4 \), using induction we can bound \( \delta(C) \leq \delta(C_1) + \delta(C_2) \leq \delta(i) + \delta(n-i) \leq [(3i-1)/4] + [(3(n-i)-1)/4] \leq [(3i-1)/4 + (3n-3i-1)/4] \leq [(3n-2)/4] \leq [(3n-1)/4] \).

In the case \( i = 4 \) we still can bound \( \delta(C) \leq \delta(4) + \delta(n-4) \leq 3 + [(3n-12-1)/4] = [(3n-1)/4] \), and similarly in the case \( n-i = 4 \). (Note that \( n \geq 9 \) by assumption, so not both \( i \) and \( n-i \) can be equal to 4.)
If both \( e_1 \) and \( e_n \) are a subset of \( \partial \text{conv}(H) \), then we place a guard \( g \) that covers both \( e_1 \) and \( e_n \) at the intersection of \( e_1 \) and \( e_n \) and define \( C' = (e_2^-, \ldots, e_{n-1}^+) \) to obtain a guarding \( g \cap \mathcal{G}(C') \) for \( C \). Therefore, in this case \( \delta(C) \leq 1 + \delta(n - 2) \leq 1 + \lceil(3n - 6 - 1)/4\rceil = \lfloor(3n - 3)/4\rfloor \). (Note that at this point we need the property that \( e_1 \) and \( e_n \) are not parallel.)

Otherwise, there is only one vertex on \( \partial \text{conv}(H) \). Assume without loss of generality (reflect \( C \) if necessary) that \( v_i \) is the only vertex on \( \partial \text{conv}(H) \). Let \( i \geq 2 \) be the index of the vertex furthest from \( e_2 \) among all vertices of \( V(C) \) to the right of \( e_2 \) (thinking of \( e_2 \) to be oriented from \( v_1 \) to \( v_2 \)). If there is no vertex to the right of \( e_2 \), set \( i = 2 \). We distinguish four cases.

**Case 1.** \( 3 \leq i \leq n - 2 \) and neither \( i = 5 \) nor \( i = n - 4 \). Place a natural edge guard \( g \) onto \( e_1 \) and split the rest into two chains, \( C_1 = (e_2^-, \ldots, e_i^+) \), and \( C_2 = (e_{i+1}^-, \ldots, e_n) \), see Figure 3.30. By the choice of \( v_i \) there is no intersection between \( C_1 \) and \( C_2 \) other than \( v_i \) and both \( C_1 \) and \( C_2 \) are simple chains. A guarding for \( C \) can be obtained as \( g \cup (\mathcal{G}(C_1) \cap \mathcal{G}(C_2)) \). Therefore, in this case \( \delta(C) \leq 1 + \delta(i - 1) + \delta(n - i) \). We can apply the inductive hypothesis to bound \( \delta(C) \leq 1 + [(3i - 3 - 1)/4] + [(3n - 3i - 1)/4] \leq [(3n - 1)/4] \).

**Case 2.** \( i = 5 \) or \( i = n - 4 \). We distinguish two subcases. If \( e_{i+1}^- \) does not intersect \( e_1 \) we can simply split \( C \) into two chains \( C_1 = (e_1, \ldots, e_i^+) \) and \( C_2 = (e_{i+1}^-, \ldots, e_n) \) and bound \( \delta(C) \leq \delta(C_1) + \delta(C_2) \leq \delta(i) + \delta(n - i) \leq \lfloor(3n - 1)/4\rfloor \) as if \( v_i \) were on \( \partial \text{conv}(H) \).

If \( e_{i+1}^- \) intersects \( e_1 \) (see Figure 3.31), denote the point of intersection by \( v' \). Let \( e_1^* \) be the ray originating from \( v' \) in the same direction as \( e_1 \), and let \( e_{i+1}^* \) be the line segment from \( v' \) to \( v_{i+1} \). Let \( C_1 = (e_1, \ldots, e_i^+) \), which is a simple chain except that the first and the last ray may intersect, in which case it defines a simple polygon \( P \). But because \( n(P) \geq 4 \), \( P \) can be described as the intersection of two polygonal halfplanes bounded by two \( (\geq 2) \)-chains \( C'_1 \) and

![Figure 3.30: v_i is the vertex furthest from the line defined by e_2. Case 1: 3 ≤ i ≤ n-1, i ≠ 5 and i ≠ n-4.](image)
Let $C_2 = (e^*_1, e^*_i+1, e_{i+2}, \ldots, e_n)$. Note that $n(C_1) = i$ and $n(C_2) = n-i+1$, so in both cases one of the numbers equals 5 and the other $n-4$. So we obtain a guarding for $C$ as $\mathcal{G}(C_1) \cup \mathcal{G}(C_2)$ (or $(\mathcal{G}(C'_1) \cap \mathcal{G}(C''_1)) \cup \mathcal{G}(C_2)$, respectively) and conclude $\delta(C) \leq \delta(5) + \delta(n-4) \leq 3 + \left\lfloor \frac{3(n-4)-1}{4} \right\rfloor \leq \left\lfloor \frac{3n-1}{4} \right\rfloor$

**Figure 3.31:** Case 2: $i = 5$ or $i = n-4$.

**Case 3.** $i = n-1$. Let $v_j$ be the vertex furthest from $e_{n-1}$ to the left, see Figure 3.32. If there is no such vertex, let $j = n-2$. Note that $j \neq 1$: If $v_1$ were furthest from $e_{n-1}$, then $v_{n-1}$ would be closer to $e_2$ than $v_{n-2}$. But $v_{n-1}$ is furthest from $e_2$ by assumption.

**Figure 3.32:** Case 3.1: $i = n-1$ and $j \neq 4$, $j \neq n-5$, $j \neq n-2$.

If $j \leq n-3$ and neither $j = 4$ nor $n-j-1 = 4$, split $C$ into two chains, $C_1 = (e_1, \ldots, e_j)$ and $C_2 = (e_{j+1}, \ldots, e_{n-1})$. Both $C_1$ and $C_2$ are simple, except that the first and the last ray of $C_1$ may intersect (in that case split the resulting polygon into two chains as explained under Case 2). Place a guard $g$ at the intersection of $e_n$ with $e_1$ such that $g$ covers $e_n$ completely and $e_1$ partly. A guarding for $C$ can be obtained as $g \cup (\mathcal{G}(C_1) \cap \mathcal{G}(C_2))$. Again this
yields $\delta(C) \leq 1 + \delta(j) + \delta(n - j - 1)$, for some $2 \leq j \leq n - 3$, and thus $\delta(C) \leq \lfloor(4n - 1)/5\rfloor$ as in Case 1.

If $j = 4$ or $j = n - 5$, we proceed in a similar way as in Case 2: If $e_j^+$ does not intersect $e_n$ we can just split at $v_j$ and get $\delta(C) \leq 3 + \lfloor\frac{n-12-1}{4}\rfloor \leq \lfloor\frac{n-1}{4}\rfloor$. Else let $v'$ be the intersection of $e_j^+$ and $e_n$ and $e_j^*$ the ray starting at $v'$ following $e_n$ and $e_j^*$ the edge from $v_{j-1}$ to $v'$, see Figure 3.33. Define $C_1 = (e_1,\ldots,e_{j-1},e_j^*,e_n)$ and $C_2 = (e_{j+1},\ldots,e_n)$, which leads to $\delta(C) \leq \delta(C_1) + \delta(C_2) \leq \delta(j+1) + \delta(n-j) = \delta(5) + \delta(n-4) \leq 3 + \lfloor\frac{n-12-1}{4}\rfloor = \lfloor\frac{n-1}{4}\rfloor$.

![Figure 3.33: Case 3.2: $i = n - 1$ and $j = 4$ or $j = n - 5$.](image)

If $j = n - 2$, put two guards: a first guard $g_1$ at the intersection of $\overline{e_n}$ with $e_1$ such that $g_1$ covers $e_n$ completely and $e_1$ partly and a second guard $g_2$ at the intersection of $\overline{e_{n-1}}$ with $\overline{e_1}$, which might be either somewhere on $e_1$ or outside $e_1$ on $\overline{e_1}$, see Figure 3.34. The guard $g_2$ covers $e_{n-1}$ completely and covers $e_1$ either partly or completely depending on the situation. Anyway, $g_1 \cup (g_2 \cap G(C'))$ provides a guarding for $C$, with $C' = (e_2^-\ldots,e_{n-2}^+)$. In this case we obtain $\delta(C) \leq 2 + \delta(n-3)$ and thus by the inductive hypothesis $\delta(C) \leq 2 + \lfloor(3n - 9 - 1)/4\rfloor = \lfloor(3n - 2)/4\rfloor$.

**Case 4.** Finally we consider the case $i = 2$, where there are no vertices in the wedge defined by $e_1$ and $e_2^+$. If $e_3^-$ does not intersect $e_1$ then put a natural vertex guard $g$ at $v_1$ to obtain a guarding $g \cup G(C')$, where $C' = (e_2^-,\ldots,e_n)$. This yields $\delta(C) \leq 1 + \delta(n-2)$ and thus by the inductive hypothesis $\delta(C) \leq 1 + \lfloor(3n - 6 - 1)/4\rfloor = \lfloor(3n - 3)/4\rfloor$.

Now suppose that $e_3^-$ intersects $e_1$. If $e_3^+$ intersects $\partial\text{conv}(H)$ (i.e., if $\overline{e_3}$ has slope higher than $\overline{e_n}$), let $v_j$ be the vertex of $C$ that is furthest to the left of $e_3^+$, else let $v_j$ be the vertex that is furthest to the right. If there is no vertex to the left or to the right, respectively, let $j = 3$. 


If $j = 3$, then place two guards: a natural vertex guard $g_1$ at $v_1$ and a guard $g_2$ at the intersection of $e_3^-$ with $e_1$ such that $g_1$ covers $e_3$ completely and $e_1$ partially (a natural edge guard $g_1$ at $e_1$ and a natural vertex guard $g_2$ at $v_2$, respectively). See Figure 3.35. A guarding for $C$ is provided by $g_1 \cup (g_2 \cap \mathcal{G}(C'))$ ($g_1 \cap (g_2 \cup \mathcal{G}(C'))$, respectively), with $C' = (e_3^-, \ldots, e_n)$. In this case we obtain $\delta(C) \leq 2 + \delta(n - 3)$ and thus in the same way as shown above $\delta(C) \leq \lceil (3n - 2)/4 \rceil$.

Otherwise, $4 \leq j \leq n - 1$. First suppose $e_{j+1}^-$ does not intersect $e_2$ and $e_1$. Then neither does $e_j^+$ and hence we can split at $v_j$ in the same way as if $v_j$ would be on $\partial \text{conv}(H)$. (If $j = n - 1$, $e_n^-$ must intersect $e_2$; therefore in this situation $j \leq n - 2$.)

Now suppose that $e_{j+1}^-$ intersects $e_2$ and thus $e_1$. Let us first consider the situation where $v_j$ is to the left of $e_3^+$. Let $v' := e_3^- \cap e_1$. Let $e_1^*$ be the ray
emanating from \(v'\) in direction \(e_1\), and let \(e^*_3\) denote the segment from \(v'\) to \(v_3\), see Figure 3.36. Place a natural vertex guard \(g\) at \(v_1\). Regardless of whether or not \(e^+_j\) intersects \(e_2\) and \(e_1\), a guarding for \(C\) is provided by \(g \cup (G(C_1) \cap G(C_2))\), with \(C_1 = (e_1^*, e_3^*, e_4, \ldots, e^+_j)\) and \(C_2 = (e^-_{j+1}, \ldots, e_n)\). Observe that by the choice of \(v_j\) both \(C_1\) and \(C_2\) are simple and \(\delta(C) \leq 1 + \delta(j - 1) + \delta(n - j)\) for some \(4 \leq j \leq n - 1\). As above, this yields \(\delta(C) \leq \lfloor (3n - 1)/4 \rfloor\), provided that neither \(j = 5\) nor \(j = n - 4\) nor \(j = n - 1\). If \(v_j\) is to the right of \(e^+_3\) we put a natural edge guard \(g\) on \(e_1\), set \(C_1 = (e^-_2, e_3, \ldots, e^+_j)\), \(C_2 = (e^-_{j+1}, \ldots, e_n)\) and have a guarding \(g \cap (C_1 \cup C_2)\) and can bound \(\delta(C)\) the same way.

Finally, consider the cases \(j = 5\), \(j = n - 4\), or \(j = n - 1\). First assume \(v_j\) is to the left of \(e^+_3\). Let \(v'\) be the intersection of \(e^-_{j+1}\) and \(e_1\) and \(e^*_1\) the ray starting at \(v'\) following \(e_1\) and \(e^*_j\) the segment from \(v'\) to \(v_{j+1}\). Let \(C_1 = (e^-_3, \ldots, e^+_j)\), \(C_2 = (e^*_1, e^*_j, e^-_{j+1}, e_{j+2}, \ldots, e_n)\), and place a natural vertex guard \(g\) on \(v_1\), see Figure 3.37 on the left. We get a guarding \(g \cup (G(C_1) \cap G(C_2))\). If \(j = 5\), we bound \(\delta(C) \leq 1 + \delta(3) + \delta(n - 4) \leq 3 + \lfloor (3n - 12 - 1)/4 \rfloor = \lfloor (3n - 1)/4 \rfloor\). If \(j = n - 4\), we find \(\delta(C) \leq 1 + \delta(n - 6) + \delta(5) \leq 4 + \lfloor (3n - 18 - 1)/4 \rfloor = \lfloor (3n - 3)/4 \rfloor\). If \(j = n - 1\), then \(\delta(C) \leq 1 + \delta(n - 3) + \delta(2) \leq \lfloor (3n - 2)/4 \rfloor\).

If \(v_j\) is to the right of \(e_3\), distinguish between the case \(j = 5\) or \(j = n - 4\) and the case \(j = n - 1\). In the first case, let \(C_1 = (e_1, \ldots, e^+_j)\) and \(C_2 = (e^*_1, e^*_j, e^-_{j+1}, e^-_{j+2}, \ldots, e_n)\) and we get a guarding \(G(C_1) \cup G(C_2)\), which leads to \(\delta(C) \leq \delta(5) + \delta(n - 4) \leq \lfloor (3n - 1)/4 \rfloor\) for both \(j = 5\) and \(j = n - 4\), see Figure 3.37 in the middle. In the second case \(j = n - 1\), place a natural edge guard \(g_1\) onto \(e_1\), a 2-guard \(g_2\) at the intersection of \(e^-_2 \cap e_{n-1}\), and define \(C' = (e^-_3, \ldots, e^-_{n-1})\), see Figure 3.37 on the right. We get a guarding \(g_1 \cap (g_2 \cup G(C'))\) and conclude \(\delta(C) \leq 2 + \delta(n - 3)\).

**Corollary 3.14.** For any simple polygon \(P\) with \(n \geq 4\), \(n \neq 7\) pairwise non-parallel edges, a guarding using at most \(\lceil (3n - 2)/4 \rceil\) guards can be obtained in
In this section we prove lower bounds on the number of guards needed in the worst case. For natural guards it is possible to show that the upper bound
3.3. Lower Bounds

Derived in Section 3.2.2 is tight and we have $\delta_{nat}(n) = n - 1$ for polygonal halfplanes. For general guards we are able to show that $\delta_v(n) \geq \delta(n) \geq \lceil (2n - 2)/3 \rceil$, improving the previous bound of roughly $3n/5$ given in [15].

### 3.3.1 The Blunting Comb

For any natural number $m$, we construct a polygonal halfplane $H_m$ with $2m$ edges that requires many guards. The polygonal halfplane consists of spikes $S_1, S_2, ..., S_m$ arranged in such a way that the lines defined by both edges of a spike cut into every spike to the left. Denote the convex apex of $S_i$ by $w_i$ and its left reflex vertex by $v_i$. The edge from $v_i$ to $w_i$ is denoted by $e_i$, the edge from $w_i$ to $v_{i+1}$ by $f_i$. From left to right the angle at the apex of each spike becomes more and more obtuse.

More precisely, we construct $H_m$ as follows. For $1 \leq i \leq m$, define $w_i := (5^{2i-2}, -2^{-(2i-2)})$; for $2 \leq i \leq m$, define $v_i := (5^{2i-3}, 2^{-(2i-3)})$. Define the first edge $e_1$ of $H_m$, which is actually a ray, to start at $w_1$ and to pass the origin $(0, 0)$. The $x$-coordinates of the points grow exponentially; the sequence of the $y$-coordinates is alternating and decreasing exponentially. Thus, the absolute value of the slope of the edges decreases exponentially, too. Note that $H_m$ is self-affine. This basically means each spike and its two neighboring spikes look the same after appropriate rescaling. It is very hard to fit more than two or three spikes on one picture, see Figure 3.38. See Figure 3.39 for a picture using logarithmic scales.

We can find a guarding for $H_m$ using roughly $\frac{2}{3}n$ vertex guards, see Figure 3.40. Place a natural vertex guard $g_1$ at $w_1$. Place a non-natural vertex guard $h_2$ onto $v_2$ that guards $e_2$ with its right ray and whose left ray goes up vertically. Continue with a natural vertex guard $h_3$ on $v_3$ and an non-natural vertex guard $g_3$ on $w_3$ that guards $f_3$ with its right ray and with its left ray going vertically down. Then again place a natural vertex guard $g_4$ at $w_4$, a similar non-natural vertex guard $h_5$ on $v_5$ as before, a natural guard $h_6$ on $v_6$, a non-natural guard $g_6$ on $w_6$, a natural vertex guard $g_7$ on $w_7$, and so on. Then, $P_m$ can be described as

$$g_1 \cup (h_2 \cap h_3 \cap (g_3 \cup g_4 \cup (h_5 \cap h_6 \cap (g_6 \cup g_7 \cup (\ldots))))).$$

No two edges of $H_m$ are collinear. Consider the line arrangement $L = \{\overline{e_1, f_1}, \overline{e_2, f_2}, \ldots, \overline{e_m, f_m}\}$. All vertices of $L$ lie on $\partial H_m$. In other words, any pair of lines defined by edges intersects on the boundary of $H_m$. This leads to the following observation:

**Observation 3.15.** $H_m$ has no possible 2-guards except natural vertex guards.
Figure 3.38: The first three spikes of $H_m$, strongly rescaled.

Figure 3.39: A picture of the first few spikes of $H_m$ using logarithmic scales. (Note that the plot has a discontinuity: for the $y$-coordinates we have set the whole interval $[-10^{-6}, 10^{-6}]$ to 0.)
3.3. Lower Bounds

3.3.2 Natural Guards

**Theorem 3.16.** For any even natural number $n$ there exists a polygonal half-plane with $n$ edges that requires at least $n - 1$ natural guards.

We prove the theorem by counting the guards in an optimal natural guarding of $H_m, n = 2m$. We say a guard belongs to a spike $S_i$ if it is a natural edge guard on $e_i$ or $f_i$ or if it is a natural vertex guard on $v_i$ or $w_i$. As only natural guards are allowed, every guard belongs to exactly one spike. The basic idea is that most spikes must have at least two guards belonging to them. Obviously every spike $S_i$ must have at least one guard belonging to it, since the edge $e_i$ must be covered (see Observation 2.7).

**Lemma 3.17.** Consider a guarding $G$ of $H_m$ using natural guards only and let $1 \leq i \leq m - 1$. If only one guard of $G$ belongs to $S_i$, then this guard must be at $v_i$ or on $e_i$. If there is no guard at $w_i$ nor a guard on $f_i$, then both a guard at $v_{i+1}$ and a guard on $e_{i+1}$ are in $G$.

**Proof.** Assume only one guard belongs to $S_i$. It cannot be a natural edge guard on $f_i$, because this would leave $e_i$ uncovered (see Observation 2.7). If we had a natural vertex guard at $w_i$ only, there would be no guard to distinguish (see Observation 2.4) a generic point near $v_i$ outside $H_m$ from a generic point near $v_{i+1}$ located inside $H_m$ and below $f_i$ (see the two circles in Figure 3.41): The
only natural guards that could distinguish this inside/outside pair are a natural edge guard on $e_i$ or a natural vertex guard on $v_i$. This proves the first part of the lemma.

![Figure 3.41: A schematic picture of a spike of $H_m$.](image)

Now assume there are no guards at $w_i$ nor on $f_i$. Then to cover the edge $f_i$ there must be a vertex guard at $v_{i+1}$. Furthermore, the edge guard on $e_{i+1}$ is the only possible natural guard that can distinguish a generic point at the apex of $S_i$ near $w_i$ from a generic point located to the right of the apex of $S_{i+1}$ near $w_{i+1}$ and above $e_{i+1}$ (depicted by two crosses).

This lemma immediately implies Theorem 3.16. Proceed through the spikes from left to right. As long as a spike has at least two guards that belong to it, we are fine. Whenever there appears a spike $S_i$ with only one guard, we know that there must be at least two guards in $S_{i+1}$ namely at $v_{i+1}$ and on $e_{i+1}$. Either there is a third guard that belongs to $S_{i+1}$, and thus both spikes together have at least four guards; or again we know already two guards in $S_{i+2}$. In this way, we can continue until we either find a spike with at least three guards or we have gone through the whole chain. So whenever there is a spike with only one guard either there is a spike with at least three guards that compensates for it, or every spike until the end has two guards. Hence there can be at most one spike with one guard only that is not made up for later. So all in all there are at least $2(m-1)+1 = n-1$ guards.

Instead of looking at a polygonal halfplane, one can add one edge to convert
3.3. Lower Bounds

$H_m$ into a simple polygon and conduct a similar proof \[15\]. Here we just state the result.

**Theorem 3.18.** For any $n \geq 4$ there is a simple polygon with $n$ edges that cannot be described by fewer than $n - 2$ natural guards.

3.3.3 General Guards

Let $G$ be a guarding of $H_m$ with no restrictions on the guards. All $n = 2m$ edges of $H_m$ have to be covered somehow. If no guard covers an edge $e$ completely, then $e$ must be covered by at least two guards partly (Observation \[2.7\]). No guard can cover more than two edges, which immediately implies that a guarding of $H_m$ must have at least $n/2$ guards (see Theorem \[2.6\]). Our goal is to improve the lower bound to roughly $2n/3$.

For any $i \in \{1, ..., m - 1\}$, let $h_i$ be the directed line segment from the intersection of $\overline{e_i}$ and $\overline{e_{i+1}}$ to $v_{i+1}$ (see Figure 3.42). Similarly, let $h'_i$ be the directed line segment from $w_{i+1}$ to the intersection $\overline{f_i}$ and $\overline{f_{i+1}}$.

We say a left ray $\ell_g$ of a guard $g$ crosses an oriented line segment $h$ with **correct orientation** if the following conditions hold:

(i) $\ell_g \cap h$ is a single point and
(ii) $\ell_g$ crosses $h$ from the right side to the left side or it starts on $h$ leaving to the left side.

Similarly, we say a ray $r_g$ crosses an oriented line segment $h$ with **correct orientation** if it intersects $h$ in a single point and $r_g$ leaves $h$ to the right side. We say a guard $g$ crosses $h$ with correct orientation if one of its rays $r_g$ or $\ell_g$ crosses $h$ with correct orientation. (In the special case that $v_g$ lies on $h$ and $\ell_g$ leaves to the left side and $r_g$ to the right side, we count the crossing only once, that is, we say $g$ has one correctly oriented crossing with $h$.)

Consider inside/outside pairs $(p_i, q_i)$ and $(p'_i, q'_i)$, $1 \leq i \leq m - 1$, of points infinitesimally close to the starting point and the endpoint of the corresponding line segment, located as follows: for all $i$, $p_i, p'_i \in H_m$ and $q_i, q'_i \notin H_m$; $p_i$ is outside a potential natural vertex guard at $w_i$, whereas $q_i$ is inside a potential natural vertex guard at $w_{i+1}$, and similarly, $p'_i$ is outside a possible natural vertex guard at $v_{i+2}$ (outside a natural edge guard on $f_m$ if $i = m - 1$), whereas $q'_i$ is inside the natural vertex guard at $v_{i+1}$, see Figure 3.42. There are $n - 2$ such pairs, and they need to be distinguished (Observation \[2.4\]).

**Lemma 3.19.** A guard can distinguish at most three of the pairs $(p_i, q_i)$ and $(p'_i, q'_i)$, $1 \leq i \leq m - 1$. 

Figure 3.42: Imagine the pairs \((p_i, q_i)\) and \((p'_i, q'_i)\) in the interior of the cell of the line arrangement as shown, but infinitesimally close to the starting and end point of \(h_i\) or \(h'_i\), respectively.
Proof. Let \( g \) be an arbitrary guard. Assume \( g \) distinguishes \( p_i \) from \( q_i \). There are three cases: If \( v_g \) is to the left of \( h_i \), then—in order to distinguish \( p_i \) from \( q_i \)—the ray \( r_g \) must intersect \( h_i \) with correct orientation. Symmetrically, if \( v_g \) is to the right of \( h_i \), then \( \ell_g \) must intersect \( h_i \) with correct orientation. Finally, if \( v_g \) is on \( h_i \), then it must be on the line segment \( h_i \) itself. To distinguish \( p_i \) from \( q_i \), the endpoint of \( h_i \) (i.e. \( v_{i+2} \)) must be inside \( g \) (possibly on the boundary of \( g \)), hence \( \ell_g \) must point to the left side of \( h_i \) or into the same direction as \( h_i \), and \( r_g \) must point to the right side of \( h_i \) or into the same direction. See Figure 3.43. Since the claim is trivial for a degenerate guard with angle 0, we can assume without loss of generality that at least one of the two rays is not collinear to \( h_i \). Therefore, \( g \) crosses \( h_i \) with correct orientation. Similarly, if \( g \) distinguishes \( p'_i \) and \( q'_i \), then \( g \) crosses \( h'_i \) with correct orientation.

![Figure 3.43: Different ways \( g \) can distinguish \( p \) and \( q \). In every case \( \ell_g \) intersects \( h \) leaving to the left side or \( r_g \) intersects \( h \) leaving to the right.](image)

The line segments \( h_1, \ldots, h_{m-2} \) lie on an oriented convex curve \( C \), which we obtain by prolonging every line segment until reaching the starting point of the next one. Extend the first and last line segment to infinity. In the same way define a curve \( C' \) for \( h'_1, \ldots, h'_{m-2} \). Any ray can cross a convex curve at most twice. Because of the way \( C \) and \( C' \) are situated with respect to each other (a line that crosses \( C \) twice must have negative slope, a line that crosses \( C' \) twice must have positive slope) a ray can intersect \( C \cup C' \) at most three times.

But we are only interested in crossings with correct orientation. If a ray crosses a curve twice, exactly one of the crossings is oriented correctly. If a ray crosses both \( C \) and \( C' \) once, exactly one of the crossings is oriented correctly. Therefore any ray can have at most two correctly oriented crossings (see Figure 3.44). A guard has at most three correctly oriented crossings: If its left ray has two correctly oriented crossings, it starts in the closed region between the two curves; if its right ray has two correct crossings, it starts either in the closed region above \( C \) or in the closed region below \( C' \). Thus, if both rays have two correctly oriented crossings, \( v_g \) lies on one of the curves. So the first crossing of \( r_g \) and the first crossing of \( \ell_g \) count as one crossing of \( g \) with
the curve.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.png}
\caption{A guard $g$ with three and a guard $g'$ with two correctly oriented crossings (marked with a circle).}
\end{figure}

Lemma 3.20. If a guard covers two edges completely, it distinguishes at most one pair.

Proof. A guard that covers two edges completely must be a natural vertex guard (Observation 3.15). A natural vertex guard distinguishes exactly one pair: If it lies on a vertex $v_i$ it distinguishes $(p_i, q_i)$ and no other pair; else, if it lies on a vertex $w_i$, it distinguishes $(p'_i, q'_i)$ only.

The main idea of the proof is that no guard “does more than three things”, be it covering edges or distinguishing pairs. Unfortunately, this is not true. There might be guards that cover an edge and distinguish three pairs.

We say a ray $r_g$ or $\ell_g$ crosses an oriented line segment $h$ uselessly if the slope of $h$ and the slope of the ray have the same sign and the absolute value of the slope of the ray is smaller than the absolute value of the slope of $h$, in other words, if the ray is less steep than $h$. Otherwise, if the ray is steeper or the slopes have different signs and the crossing is oriented correctly, we say a ray crosses a line segment reasonably.
Observation 3.21. If a ray crosses both $C$ and $C'$ with correct orientation, then one of the two crossings is useless.

Proof. Recall the proof of Lemma 3.19. If a left ray $\ell_g$ has got two correctly oriented crossings, then $v_g$ lies in the region between the curves $C$ and $C'$. Its first correctly oriented crossing (seen from $v_g$) must be useless: In order to cross one curve twice and then the other curve as well, $\ell_g$ must be less steep than the curve it crosses first, see Figure 3.44. Similarly, if a right ray $r_g$ has got two correct crossings, $v_g$ lies above $C$ or below $C'$. In order to produce two correctly oriented crossings, $r_g$ first has to enter the region between the two curves (thus producing a reasonable crossing), then enter the region below $C'$ (above $C$, respectively) and finally leave it again entering the region between the curves, thus producing a correctly oriented crossing that is useless. \[\square\]

Now assume a guard $g$ crosses $h_i$ with correct orientation but only uselessly. Even though it distinguishes $(p_i, q_i)$, it does an “incomplete job” in the following sense. Construct a new inside/outside pair, denoted by $\tilde{p}_i$ and $\tilde{q}_i$: Let $\tilde{e}_i$ be the line segment from $v_i$ to the starting point of $h_i$. Draw a directed line segment $\tilde{h}_i$ from the midpoint of $\tilde{e}_i$ to the midpoint of $\tilde{e}_{i+1}$, however, not going all the way but only until it hits $\tilde{f}_i$. See Figure 3.45.

![Figure 3.45: If no guard crosses $h_i$ reasonably, introduce a new segment $\tilde{h}_i$.](image)
Let $\tilde{q}_i$ be infinitesimally close to the starting point of $\tilde{h}_i$ outside $H_m$ and $\tilde{p}_i$ infinitesimally close to the endpoint of $\tilde{h}_i$ slightly below $f_i$. The ray of $g$ that crosses $h_i$ uselessly does not cross $\tilde{h}_i$: The end point of $\tilde{h}_i$ lies above the starting point of $h_i$. Note that at this point we need the exact geometry of $H_m$: A polygonal chain with the same combinatorial properties with respect to the line arrangement defined by its edges does not necessarily have this property. Therefore, $g$ cannot distinguish $\tilde{p}_i$ and $\tilde{q}_i$ with the same ray; if it does distinguish them, too, it has to “use” its other ray. Similarly, if a guard $g$ distinguishes a pair $p'_i$ and $q'_i$ only uselessly, then it does not distinguish the new pair ($\tilde{p}'_i, \tilde{q}'_i$) defined analogously with the same ray. So there must be another ray $s$, in general from another guard $\tilde{g}$, that distinguishes $\tilde{p}'_j$ and $\tilde{q}'_j$.

For any $i$, prolong $h_i$ and $\tilde{h}_i$ until they meet on $e_i$. Similarly, prolong $h'_i$ and $\tilde{h}'_i$. Keep in mind that from now on we work with these prolonged segments: If a ray distinguishes a point pair, it still has a correctly oriented crossing with the corresponding prolonged segment. But now there also can be guard rays that cross a prolonged segment with correct orientation even though they do not distinguish the corresponding point pair.

Let each of the pairs $\{(p_1, q_1), \ldots, (p_{m-1}, q_{m-1})\} \cup \{(p'_1, q'_1), \ldots, (p'_{m-1}, q'_{m-1})\}$ have initial charge 1. If a point pair is distinguished by a natural vertex guard, then the point pair discharges immediately to this guard. From now on, rays of natural vertex guards are not considered anymore. Else, if there is no natural vertex guard distinguishing the pair, it discharges evenly to the guard rays that cross the corresponding directed line segment $h_i$ (or $h'_i$, respectively) reasonably. (Note that rays might receive charge even though they do not distinguish the inside/outside pair, because the line segment has been prolonged.) If no guard ray intersects $h_i$ (or $h'_i$, respectively) reasonably, we pass the charge over to the new line segment $\tilde{h}_i$ ($\tilde{h}'_i$, respectively) defined as above. Again, if there are guard rays intersecting (the prolonged) $\tilde{h}_i$ ($\tilde{h}'_i$, respectively) reasonably, distribute the charge among them.

Unfortunately, it might happen that neither $h_i$ nor $\tilde{h}_i$ get crossed reasonably by any guard ray. But the following lemma assures that then we find other guard rays we can discharge to.

**Lemma 3.22.** If neither $h_i$ nor $\tilde{h}_i$ are crossed reasonably, then there are two guard rays with the following properties:

1. They intersect $C$ between the starting point of $\tilde{h}_i$ and the starting point of $h_i$,
2. have the interior of the guard above them (i.e., cross $C$ with wrong orientation),
3. are less steep than $\tilde{h}_i$, and
(iv) have received charge at most 1/2 so far.

The analogous holds for $h'_i$, $\tilde{h}'_i$ and $C'$.

Proof. Consider the set of guard rays $R$ that intersect the triangle defined by the prolonged $h_i$ and $\tilde{h}_i$ and $e_i$. (Remember that rays from natural vertex guards are excluded from our consideration, so rays from a potential natural vertex guard on $w_i$ are ignored). Without loss of generality, assume there are no vertical rays. Then $R$ decomposes into two subsets, the set $U$ of rays that have the guard interior above (i.e., either right rays pointing two the right or left rays pointing to the left) and the set $B$ of rays that have the guard interior below (i.e., either right rays pointing two the left or left rays pointing to the right). The set $U$ can be further decomposed into the set of positively sloped rays $U^+$ and negatively sloped rays $U^-$. If a ray from $U^-$ is steeper than $\tilde{h}_i$, it produces a reasonable crossing with $h_i$ or $\tilde{h}_i$. Therefore we may assume that all rays in $U^-$ have slope between 0 and the slope of $\tilde{h}_i$. By assumption all the rays in $B$ have negative slope, because they may intersect $\tilde{h}_i$ uselessly only, hence $B = B^-$. 

Now assume for contradiction that all but at most one ray in $U^-$ have received charge 1. This means each of them intersects $h_j$ or $\tilde{h}_j$ for a different $j > i$: The only way they can have received charge is because they cross a segment reasonably. This segment must be somewhere to the right, therefore $j > i$. Their orientation does not allow them to cross an $h'_j$ or $\tilde{h}'_j$ reasonably for any $j$.

Now consider a sequence of inside/outside point pairs $((s_1, t_1), (s_2, t_2), \ldots)$ defined as follows: $s_1 := p_i$ and $t_1 := q_1$. Pick a ray $r_1 \in B^-$ that distinguishes this point pair (no ray in $U$ can distinguish it). Let $s_2$ be a point infinitesimally close to the intersection of $r_1$ and $f_{i-1}$ such that $s_2$ is slightly above $r_1$ and to the right (i.e., below) of $f_{i-1}$; set $t_2 := t_1$. Note that if there is a natural vertex guard on $w_i$, it does not distinguish $s_2$ and $t_2$. If several rays in $B^-$ could have been chosen, pick one that maximizes the $y$-coordinate of the resulting $s_2$. Again there must be a guard ray distinguishing $s_2$ and $t_2$. No ray in $B^-$ can do that, because we have chosen an extreme $r_1$. A ray in $U^+$ cannot do so either, because $s_2$ is still slightly below $t_2$ (in the extreme case they almost lie on a horizontal line). Therefore, only rays in $U^-$ can distinguish $s_2$ and $t_2$.

Among all rays in $U^-$ that do so, let $r_2$ be one that maximizes the $y$-coordinate of the intersection with $f_{i-1}$ and let $t_3$ be a point infinitesimally close to this intersection slightly outside $H_m$ and above $r_2$; $s_3 := s_2$. Neither $r_1$ nor $r_2$ can distinguish $s_3$ and $t_3$ Now let $r_3$ be the ray from $B^-$ that distinguishes $s_3$ and $t_3$ and maximizes the $y$-coordinate of the intersection with $\overline{f_{i-1}}$, and define $s_4$ to be slightly above $r_3$ and below $\overline{f_{i-1}}$, and so on. See Figure 3.46.
The procedure goes on until $t_k$ lies above the starting point of $\tilde{h}_i$. If a ray $r_l$ ($l$ even) intersects a segment $h_j$ where $j$ is much bigger than $i$, than $r_l$ must be almost horizontal compared to $h_i$, so $t_{l+1}$ has almost the same $y$-coordinate as $s_l$. In other words, we have not gained much in order to reach the starting point of $\tilde{h}_i$. Therefore, even if $m$ goes to infinity and we have an endless supply of rays in $U^-$, all of them coming from a segment $h_j$ or $\tilde{h}_j$ with different $j > i$, there is no way the starting point of $\tilde{h}_i$ can ever be reached, a contradiction.  

If neither $h_i$ nor $\tilde{h}_i$ get crossed reasonably by any guard ray, we find other guard rays we can discharge the point pair $(\tilde{p}_i, \tilde{q}_i)$ to.

**Lemma 3.23.** Each guard ray receives at most charge 1 in total.

**Proof.** A ray $s$ might have received charge at three different stages.

First, it has received charge for any segment $h_i$ or $\tilde{h}_i'$ it crosses reasonably (and that has not already discharged to a natural vertex guard distinguishing the corresponding point pair). A ray can have at most one correctly oriented crossing with $C$ and at most one with $C'$ (see proof of Lemma 3.19). If it has both, one of them must be a useless crossing (Observation 3.21). Therefore, even if a ray does distinguish two different point pairs, it will be charged for only one of them.
Second, there might be new segments $\tilde{h}_i$ or $\tilde{h}_i'$ discharging to $s$. If $s$ receives charge for distinguishing, say, crossing $\tilde{h}_i$ reasonably, it cannot have received any charge in the first stage. (It could only have a reasonable crossing with $h_i$. However, there is no such crossing, or $h_i$ would have discharged to $s$ right away without the detour via $\tilde{h}_i$. Furthermore, using the same argument as in the first stage, $s$ can have at most one reasonable crossing with either a $\tilde{h}_i$ for some $i$ or a $\tilde{h}_i'$ for some $i$, but not both.

Finally, note that a guard ray can at most once receive additional charge as described in Lemma 3.22. After a ray from $U^-$ has crossed $C$, it cannot cross it again. Thus, a ray appears at most once, even if we have to apply the procedure described in the lemma several times.

Give each edge of $H_m$ an initial charge 1. Discharge each edge $e$ evenly to the guards that cover $e$ on the right side, that is, guards that either cover $e$ completely or cover $e$ partly with a ray that contains the right endpoint of $e$.

**Observation 3.24.** If a guard covers two edges of $H_m$ on the right side, then it is a natural vertex guard.

*Proof.* Let $g$ be a guard that covers two edges of $H_m$. If $g$ is not a natural vertex guard, then by Observation 3.15 it follows that $g$ is a 1'-guard, that is, $g$ covers one edge $e$ completely and another edge $e'$ partly. Therefore, $v_g$ lies in the relative interior of $e'$ at the intersection of $e'$ and $\overline{e}$. Looking at the possible locations of $v_g$ on a spike of $H_m$ (see Figure 3.39), we find that all possible lines $\overline{e}$ intersecting one of the two edges of the spike are defined by an edge further to the right. Therefore, $g$ covers $e$ completely using its right ray $r_g$ and it covers $e'$ partly with the left ray $\ell_g$. The orientations of $\ell_g$ and $e'$ only match if $g$ covers the left part of $e'$.

Finally, for each guard $g$, discharge the guard rays $\ell_g$ and $r_g$ to $g$. So now all the charge has been moved the guards of $G$.

**Lemma 3.25.** The final charge to a guard in $G$ is at most 3.

*Proof.* Guards may receive charge from edges they cover or via their rays. If a guard receives charge 2 from the edges it covers, then it is a natural vertex guard and by Lemma 3.20 only one of its rays gets charged at most 1, the other ray gets charged 0 (remember that we excluded the rays of natural vertex guards for the rest the charging scheme). Therefore, the final charge to a natural vertex guard is at most 3. If a guard covers only one edge, it receives at most charge 1 via this edge and at most 2 via its two rays, therefore, it gets charged at most 3. If there are any 0-guards in $G$, they will have final charge at most 2.
Theorem 3.26. For any even natural number $n \geq 2$ there exists a polygonal halfplane with $n$ edges that requires at least $\lceil (2n - 2)/3 \rceil$ guards.

Proof. Consider $H_m, 2m = n$. There are $n - 2$ pairs and $n$ edges with initial charge 1 each, so the total initial charge is $2n - 2$. At the end the charge to the $k$ guards of $G$ is not more than 3 per guard. Therefore, $2n - 2 \leq 3k$. \hfill \Box

Theorem 3.27. For any even natural number $n$ there exists a simple polygon with $n$ edges that requires at least $\lceil (2n - 3)/3 \rceil = \lceil 2n/3 \rceil - 1$ guards.

Proof. For any even $n = 2m$, construct a simple polygon $P_m$ with $n$ vertices by taking $H_m$ and rotating the last ray $f_m$ counter-clockwise until it intersects the first edge $e_1$, which gives an additional vertex $v_1$. Now we use exactly the same charging scheme as for $H_m$, with the only difference that the last pair $(p'_{m-1}, q'_{m-1})$ cannot be defined anymore. The lemmata hold verbatim for $P_m$ as well. So in total there are $n$ edges to be covered and $n - 3$ pairs to be distinguished, which leads to $\delta(P_m) \geq (2n - 3)/3$. \hfill \Box
4 Complexity of the Wireless Localization Problem

Given a polygon $P$, is it possible to find an optimal guarding efficiently? For the classical art gallery problems there is not much hope, as it was shown to be NP-hard almost thirty years ago. In 1983, O’Rourke and Supowit [62] showed that covering a polygonal region with a minimum number of star-shaped polygons is NP-hard, which is equivalent to the art gallery problem with point guards. Using a different reduction, Lee and Lin [50] have shown that the problem is also NP-hard for simple polygons and for other variants of the classical art gallery problem. In 1998, Eidenbenz [27] presented some inapproximability results.

In this light, it seems very plausible that the wireless localization problem is NP-hard as well and that it can be proved in a similar way. But having a closer look, it turns out that the situation regarding the computational complexity is quite different from the classical art gallery problems. First of all, the signal of our guards is not blocked by the edges of the polygon. A basic ingredient in hardness proofs for the classical setting often are small pockets, which can only be guarded from a nearby point because the edges shield it away from the rest of the polygon. Such local arguments do not work anymore, because now guards may have an effect somewhere far away as well. Second, it is not about covering a polygon with certain primitives, that is, describing it as a union, but now also intersections are allowed. This might be why our hardness proofs for the wireless localization problem turn out to be quite different and more complicated. Indeed, we can not answer the question about the computational complexity definitively.

A first attempt to tackle the problem was made by reducing the vertex cover problem of planar graphs to the natural wireless localization problem [14]. This reduction yields polygonal regions with several components, that is, it
only shows NP-hardness for the problem allowing disconnected polygons (or, equivalently, for polygons with holes). Therefore, we tried to reduce directly from SAT, which turned out to be more fruitful. The reduction yields simple polygons and it can be shown to work not only in the natural setting, but also in a more general setting with the only restriction that a guard must be inside the polygon (which includes its boundary). This joint work with Michael Hoffmann [12] is the basis of this chapter. Unfortunately, we have not been able to prove a similar reduction for the general wireless localization problem where guards may also be located outside the polygon. The question whether the general wireless localization problem is in NP remains open as well.

4.1 The Natural Wireless Localization Problem is NP-complete

We use the same notation as in Chapter 3. A natural guarding of a polygon $P$ is a set of natural vertex guards and natural edge guards such that $P$ can be described as a monotone formula over the guards. In this context, we formulate the wireless localization as a decision problem.

The Natural Wireless Localization Problem. Given a simple polygons $P$ and an integer $k$, is there a natural guarding for $P$ using $k$ guards?

**Theorem 4.1.** The natural wireless localization problem is NP-complete.

Given a set of natural guards we can check in polynomial time whether they can describe $P$: Consider the line arrangement defined by the edges of $P$. If for every pair $(C,D)$ of cells of this arrangement with $C \subset P$ and $D \cap P = \emptyset$, we can find a guard $g \in G$ that distinguishes them, that is, $C \subset g$ and $D \cap g = \emptyset$, then $G$ is a guarding of $P$. Therefore the problem is in NP.

To show the NP-completeness we reduce the monotone-SAT problem to the natural wireless localization problem: Let $F$ be a monotone Boolean formula in conjunctive normal form (CNF) with clauses $C_1, \ldots, C_m$ on the variables $x_1, \ldots, x_n$. We call the number of clauses a variable $x_i$ appears in the degree of $x_i$ and denote it as $\deg(x_i) := |\{C_j : x_i \in C_j \text{ or } \overline{x_i} \in C_j\}|$ (without loss of generality, we assume that no variable appears more than once in a clause). Monotone means that in every clause $C_j$ either all literals are positive or all literals are negative. In the first case, we call $C_j$ a positive clause, in the second case a negative clause. We may assume that the first $k$ clauses are all positive and the remaining $m - k$ clauses are all negative.
The basic picture of the reduction is the following. We define different gadgets, which are bi-infinite polygonal chains. In the end we connect these gadgets to form a simple polygon $P(F)$. The variable gadget for a variable $x_i$ will be a merlon-like chain of size $4\deg(x_i) + 3$, which can be guarded optimally in essentially two ways, thus encoding the truth value of $x_i$. There will be a clause gadget for every clause $C_j$ of size 4. The basic property of a clause gadgets is that it can only be guarded with two guards if intersected by a ray of another guard. The crucial idea is that depending on how a variable gadget is guarded, either there are such rays of guards going to the positive clauses in which the variable appears or to the negative clauses in which it appears. We connect the variable gadgets to form one big chain. The variable gadgets are constructed in such a way that after joining them the guardings are still independent in some sense. We connect the clause gadgets to another chain and finally put everything together to a simple polygon using two intermediate chains. See Figure 4.4 to get a first impression. In the following we do not give the explicit construction of these gadgets. A computer program computing explicit coordinates can be found under http://www.inf.ethz.ch/personal/christt/sat_to_guarding.php.

![Figure 4.1: A positive clause gadget $R_j$ and a negative clause gadget $R_k$.](image)

**Clause gadget.** For every clause $C_j$ we define a clause gadget $R_j$, which is a chain with four edges forming two spikes as shown in Figure 4.1. Depending on whether $C_j$ is positive or negative, $R_j$ is of the first form or a reflection of it. Either the line defined by the inner edge of the right spike cuts the more acute left spike or the other way around. Note that such a chain cannot be guarded with two guards if we restrict to natural guards, or more generally, if we do not allow guards outside the polygonal halfplane where $e_1$ and $e_4$ meet. But it can be guarded with two guards if there is a ray of a third guard $g$ intersecting...
it in the right way. Then we have $R_j = v_1 \cup (v_3 \cap g)$ or $R_j = (v_1 \cap g) \cup v_3$, respectively. See Figure 4.2. We put the clause gadgets $R_1, \ldots, R_m$ horizontally one after another, first the gadgets for the positive clauses, then the gadgets for the negative clauses, taking their union.

**Variable gadget.** For every variable $x_i$ we define a variable gadget $Q_i$, which is a polygonal chain with edges $(e_1, \ldots, e_{k_i})$ and vertices $v_1, \ldots, v_{k_i-1}$ where $k_i = 4 \deg(x_i) + 3$. See Figure 4.3. There is a “spike” for every clause $C_j$ in which $x_i$ appears, first for the positive clauses then for the negative clauses. If $C_j$ is positive, then the left topmost vertex $v_k$ with $k = 3 \pmod{4}$ is positioned such that the line $\overline{v_k}$ defined by $e_k$ intersects $R_j$. If $C_j$ is negative, the right downward edge is adjusted so that it points to $R_j$. We connect all variable gadgets horizontally, taking their intersection.

**Connecting the gadgets.** Additionally, we define two intermediate gadgets $I_1$ and $I_2$ to connect everything: $I_1$ is a large rectangle that includes everything and $I_2$ is a chain that starts to the right of the variable gadgets, first goes right until it is to the right of all clause gadgets and then rises all the way up to make the connection to the clause gadget. Then, we define the simple polygon

$$P(F) = I_1 \cap ((R_1 \cup \ldots \cup R_m) \cup (I_2 \cap Q_1 \cap \ldots \cap Q_m)).$$

See Figure 4.4 for an example: The polygon $P(F)$ we get from the formula $F = C_1 \land C_2 \land C_3 = x_1 \land (x_2 \lor x_3) \land (\overline{x_1} \lor \overline{x_2} \lor \overline{x_3})$ by putting together the clause gadgets $R_j$, the variable gadgets $Q_i$ and the two intermediate gadgets $I_1$ and $I_2$. $P(F) = I_1 \cap ((R_1 \cup R_2 \cup R_3) \cup (I_2 \cap Q_1 \cap Q_2 \cap Q_3)).$
4.1. The Natural Wireless Localization Problem is NP-complete

Figure 4.3: A variable-gadget $Q_i$ corresponding to a variable $x_i$ that appears once positively and twice negatively. Depending on where the corresponding clause gadgets are located, the slope of the dotted rays may vary such that they exactly point to this gadget.

**Lemma 4.2.** If $F$ is satisfiable, then $P(F)$ can be guarded with

$$2m + 5 + \sum_{i=1}^{n}(2 \deg(x_i) + 2)$$

natural guards.

**Proof.** Consider a satisfying assignment. Depending on the truth value of $x_i$ we guard $Q_i$ either positively or negatively with $2(\deg x_i + 1)$ guards as shown in Figure 4.5. Consider a clause gadget $R_j$ for a positive clause $C_j = \{x_{j1}, x_{j2}, x_{j3}\}$. At least one of the variables $x_{j1}, x_{j2}, x_{j3}$ is set to true. Thus at least one of the corresponding variable gadgets $Q_{j1}, Q_{j2},$ or $Q_{j3}$ is guarded positively and there is a ray of a guard $g$ from one of those gadgets passing through the clause gadget $R_j$ with correct orientation. Therefore, $R_j$ can be guarded using two natural vertex guards and $g$ (see Figure 4.2). Similarly, we can guard every negative clause gadget using two natural vertex guards and the help of some guard $g$ coming from a variable gadget. Five more guards are needed for $I_1$ and $I_2$. As $P(F) = I_1 \cap ((R_1 \cup \ldots \cup R_m) \cup (I_2 \cap Q_1 \cap \ldots \cap Q_m))$ we get a guarding for $P(F)$ putting the guardings for the gadgets together in the right way. \qed

**Lemma 4.3.** If $P(F)$ can be guarded with $2m + 5 + \sum_{i=1}^{n}(2 \deg(x_i) + 2)$ natural guards, then $F$ is satisfiable.

**Proof.** Let $G$ be a guarding of $P(F)$ consisting of $2m + 5 + \sum_{i=1}^{n}(2 \deg(x_i) + 2)$ natural guards. $P(F)$ has $4m + 10 + \sum_{i=1}^{n}(4 \deg(x_i) + 3)$ edges in total. A
Figure 4.4: A complete example in all its beauty.
4.1. The Natural Wireless Localization Problem is NP-complete

Figure 4.5: A positive guarding of \( Q_i \) (left) and a negative guarding of \( Q_i \) (right). \( Q_i = e_1 \cup (v_2 \cap (v_4 \cup v_6) \cap (v_8 \cup v_{10}) \cap (v_{12} \cup v_{14})) \), or \( Q_i = v_1 \cup (v_3 \cap v_5) \cup \ldots \cup (v_{11} \cap v_{13}) \cup e_{15} \)

guard \textit{belongs} to a variable gadget if it is an edge guard on one of its edges or a natural vertex guard on one of its vertices or if it is a natural vertex guard at the intersection with the next chain to the left. See Figure 4.6.

By Observation 2.7 every edge of a variable gadget \( Q_i \) has to be covered somehow. Except for the last edge, only guards that belong to \( Q_i \) can do so. Since a guard can cover at most two edges, at least \( 2 \deg(x_i) + 1 \) guards belong to the gadget. There is only one way to guard every edge except the last one with that many guards, namely using a natural vertex guard on every other vertex of the chain starting with the first vertex (Figure 4.6). But in this case there is no vertex guard on the last vertex and no edge guard on the last edge, hence there is no guard that can distinguish a point \( p \) near to the second edge of the next chain inside \( P(F) \) and a point \( q \) near to the last edge of this chain outside \( P(F) \). (There may be rays of guards that cross \( pq \), but they cannot have the right orientation to distinguish \( p \) and \( q \).) Therefore, there can be no such guarding and at least \( 2 \deg(x_i) + 2 \) guards belong to the gadget.

Figure 4.6: Natural vertex guards put on the marked vertices and natural edge guards on the fat edges belong to \( Q_i \).

Intuitively, there is some freedom in how to guard a vertex gadget with \( 2 \deg(x_i) + 2 \) guards because we have “half a guard” in excess. We can start with natural vertex guards on every other vertex and put a natural edge guard on the last edge (Figure 4.5 right) or we can start with an edge guard right away
and then continue with natural vertex guards on every other vertex (Figure 4.5 left). Or we can do a combination of both, starting the first way and at some place put a natural vertex guard and continue in the second way. Instead of putting a natural edge guard somewhere we could also employ natural vertex guards only and cover one edge twice.

All possible guardings have one feature in common. Looking from left to right, we can change exactly once from the first pattern to the second pattern. As soon as we are in the second pattern, we cannot change back to the first without “paying” an additional guard. If there is a change to the second pattern within the positive spikes (such that at least one positive ray is emitted towards the corresponding clause gadget), we define the gadget to be guarded positively; otherwise, the gadget is guarded negatively.

A guard belonging to a variable gadget can only cover edges of the variable gadgets. (An exception is the leftmost edge of $P(F)$, which might be covered by a natural vertex guard belonging to $Q_1$. But by considering a pair of points as shown in Figure 4.6, but now on the leftmost spike, we can argue that there must be a second guard covering this leftmost edge.) Thus the remaining $4m + 10$ edges have to be covered by the remaining $2m + 5$ guards. There is only one possible way to achieve this: put a natural vertex guard on every other vertex.

A clause gadget can be guarded with two natural vertex guards if and only if there is another correctly oriented guard ray crossing it. See Figure 4.2. The only rays that might do this are those emanating from guards covering the corresponding edge in a variable gadget of a variable that appears in the clause. At least one of these guard rays must be present, which means that the corresponding variable gadget must be guarded negatively or positively for a negative or positive clause, respectively. Therefore, we obtain a satisfying assignment as follows: If the gadget of a variable is guarded positively, we set the variable to true, if it is guarded negatively, we set it to false. This assignment satisfies every clause, because each of them has at least one helping ray, which means that at least one of its variables gets assigned the right truth value.

4.2 NP-Hardness in a More General Setting

It seems rather difficult to prove a similar statement for the general setting where unrestricted guards are allowed. If guards can be located anywhere in the plane, in particular, on the intersections of two lines of the line arrangement outside the polygon, then the arguments used before break down.
4.2. NP-Hardness in a More General Setting

But this problem can be fixed if we forbid guards outside \( P \). We call a guard whose vertex \( v_g \) is inside \( P \) (which includes the possibility that it is on the boundary of \( P \)) an *internal* guard. Note that this is a very natural restriction having the Internet café model in mind.

**The Internal Wireless Localization Problem.** Given a simple polygon \( P \) and an integer \( k \), is there a guarding for \( P \) using \( k \) internal guards?

**Theorem 4.4.** *The internal wireless localization problem is NP-hard.*

We use a similar reduction as in the natural setting, but we have to change it a little bit. Intuitively, the problem is that for every variable gadget \( Q_i \) there is one guard that covers one edge only and its other ray is not “used”. Now that we allow general guards, this unused ray is free to point to a clause gadget. In this way, clause gadgets could be guarded with two natural vertex guards even though none of its corresponding variable gadgets is guarded in the right way. We overcome this problem by introducing \( n \) additional *special gadgets* to “bind” these free rays.

**Special gadget.** We define \( n \) special gadgets, which are chains with 6 edges. A special gadget looks like a positive clause gadget rotated by \( \pi/2 \) in clockwise direction and with a small spike added at the top. We include the special gadgets to the right. See Figure 4.8.

We define the variable gadgets \( Q_1, \ldots, Q_n \) and the clause gadgets \( R_1, \ldots, R_m \) essentially the same way as in the natural setting. In the variable gadgets we add one additional spike at the beginning, so \( Q_i \) now consists of \( 4 \deg(x_i) + 7 \) edges. We define \( P(F) \) in the same way as in the natural setting except that we include the special gadgets to the right. And now we have three intermediate gadgets \( I_1, I_2, \) and \( I_3 \) having 10 edges in total.

If we consider the line arrangement defined by prolonging the edges of \( P(F) \), we observe that the intersection of two lines either lies on one of the edges or is outside \( P \) or if it is in the interior, the orientations of the lines do not match to form a possible location of a 2-guard. In other words, all possible locations of 2-guards are either vertices of \( P \) or are outside \( P \). For example, see Figure 4.7. This leads to the following observation:

**Observation 4.5.** *The only 2-guards in an internal guarding of \( P(F) \) are natural vertex guards.*
Figure 4.7: Several clause gadgets put together, the first four correspond to positive clauses, the other three to negative clauses. There are no intersections of lines defined by edges in the interior.

Lemma 4.6. If $F$ is satisfiable, then $P(F)$ can be guarded with

$$2m + 3n + 5 + \sum_{i=1}^{n} (2\deg(x_i) + 4)$$

guards.

Proof. Consider a satisfying assignment. Depending on the truth values of $x_i$ in a satisfying assignment we guard $Q_i$ either positively or negatively with $2\deg x_i + 4$ guards similar to the natural setting (see Figure 4.5), but instead of just using natural edge guards we now use the “free” ray to help guarding one of the special gadgets, see Figure 4.8. Then, as in the natural setting, we can guard all the other gadgets using natural vertex guards only. That is, we need 2 guards for every clause gadget, 3 guards for every special gadget, and 5 guards for the intermediate gadgets. \qed

Lemma 4.7. If $P(F)$ can be guarded with

$$2m + 3n + 5 + \sum_{i=1}^{n} (2\deg(x_i) + 4)$$

internal guards, then $F$ is satisfiable.

Proof. Let $G$ be a guarding of $P(F)$ consisting of $2m + 3n + 5 + \sum_{i=1}^{n} (2\deg(x_i) + 4)$ internal guards. Every guard $g \in G$ has two rays $r_g$ and $\ell_g$. In total there are $4m + 6n + 10 + \sum_{i=1}^{n} (4\deg(x_i) + 8)$ rays. Every edge $e$ of $P(F)$ has to be covered (see Observation 2.7), that is, there is at least one ray which is collinear with $e$
4.2. NP-Hardness in a More General Setting

Figure 4.8: A picture of $P(F)$ (turned and the long middle part cut out) for the formula $F = C_1 \land C_2 \land C_3 = x_1 \land (x_2 \lor x_3) \land (\overline{x_1} \lor \overline{x_2} \lor \overline{x_3})$. Shown is a guarding corresponding to the satisfying assignment where $x_1$ and $x_3$ are set to true and $x_2$ to false. The marked vertices are the positions of natural vertex guards.
and there are \(4m + 6n + 10 + \sum_{i=1}^{n}(4 \deg(x_i) + 7)\) edges, that is, exactly \(n\) edges fewer than guard rays. The proof has three parts. First we map every ray to a gadget. Next, we have a look on how the variable gadgets can be guarded and see that this cannot be done with 2-guards only. Finally, we conclude that therefore the clause gadgets have to be guarded with 2-guards and need “help” as in the natural setting, which will lead to a satisfying assignment.

**Claiming the rays.** We say a ray \(r\) of a guard \(g\) gets *claimed* by a gadget *in the first round* if \(g\) covers an edge \(e\) of this gadget (partly or completely) using \(r\), that is, \(g\) covers \(e\) and \(r \subset e\). There are no collinear edges of \(P(F)\), therefore a ray gets claimed at most once.

Now we let each special gadget *claim* an additional ray in a *second round* provided it has not claimed 7 or more rays already: Consider the horizontal edge \(e = uv\) of a special gadget that has claimed 6 rays only, see Figure 4.9. We distinguish three cases. Either \(g\) is covered by a natural vertex guard on \(u\) or a natural vertex guard on \(v\) or it is covered by a guard \(g\) that is not a natural vertex guard, which has to be located somewhere on the line segment \(e \cap P(F)\).

If it is covered by a natural vertex guard on \(u\), then consider the pair of points \((p,q)\) as indicated in the figure. There must be a ray \(r\) of a guard that distinguishes \(p\) and \(q\). Now we let the special gadget claim \(r\).

Similarly, if there is a natural vertex guard on \(v\), then we claim the ray \(r\) that distinguishes the points \(p'\) and \(q'\).

If \(e\) is covered by a \(g\) that is not a natural vertex guard, then we claim the second ray \(r\) of \(g\) in addition to the first ray that covers \(e\) and already was claimed in the first round. Of course, we have to ensure that in any case \(r\) has not got claimed already before.

First, \(r\) cannot have been claimed in the first round, because it cannot be collinear with an edge of \(P(F)\) (the only exception is that in the last case \(g\) might be located on the vertical edge \(f\) of the intermediate gadget \(I_1\) (see Figure 4.9) and \(r\) pointing downwards covers \(f\) partly therefore being claimed by \(I_1\), but then we claim \(r\) anyway taking it away from \(I_1\). There is at least one ray claimed by the edge \(f\) which will never be taken away, namely the one covering the topmost part of \(f\). Second, \(r\) cannot have been claimed by a special gadget in the second round: No ray can distinguish such pairs of points twice, because it can cross the line \(l\) only once. And in the third case the second ray of a 1-guard covering \(e\) cannot distinguish such a pair of points at another special gadget because the orientation does not match, see Figure 4.9.

Summing up, we find that every gadget has exactly claimed as many rays as it has edges in the first round and each special gadget has claimed an additional
ray in the second round if it has not claimed 7 rays in the first round already. In particular, this implies that each variable gadget and each clause gadget has claimed exactly as many rays as it has edges. Therefore all edges of clause gadgets or variable gadgets are covered completely by exactly one guard.

Figure 4.9: The horizontal edge $e$ of a special gadget. We distinguish three cases: $e$ is covered by a natural vertex guard on $u$, $e$ is covered by a natural vertex guard on $v$ or $e$ is covered by a 1-guard.

**Guarding a variable gadget.** Now consider a variable gadget $Q_i$. As in the natural case we say a guard belongs to $Q_i$ if it covers one of its edges except if it is a natural vertex guard covering the last edge and the first edge of the next gadget to the right. Note that this definition generalizes the definition in the natural setting. (Again a guard can belong to at most one variable gadget: There are no guards that cover edges of variable gadgets only partly, therefore there can be only 2-guards that cover two edges of different variable gadgets. The only location for a 2-guard to cover two edges from different variable gadgets is the one in the exception above.)

Assume that only $2 \deg(x_i) + 3$ guards belong to $Q_i$. Then they are natural vertex guards on $v_1, v_3, \ldots$ exactly as in the natural setting. There must be a ray that distinguishes an inside/outside point pair $(p, q)$ close to the endpoints of the line segment $s$ as shown in Figure [4.6]. Let $r$ be a ray that distinguishes the pair, that is, it crosses the line segment $pq$ with correct orientation. If the
ray \( r \) of the guard \( g \) was claimed in the first round because it covers some edge \( e \), then we observe that \( e \) must be the last edge of \( Q_i \) (although there are lines of the line arrangement other than \( \overline{pq} \) intersecting \( pq \), but none of them has the right orientation). Now if \( g \) covers \( e \), it either belongs to \( Q_i \) or it is a natural vertex guard on the tip of the spike where \( Q_i \) meets the next gadget. In the first case, we get a contradiction because the only guards belonging to \( Q_i \) are the natural vertex guards on \( v_1, v_3, v_5, \ldots \) and \( g \) is none of them. In the second case, \( g \) cannot distinguish \( p \) and \( q \) as both points are outside \( g \). Therefore \( r \) cannot have been claimed in the first round. So the ray \( r \) was claimed in the second round and it must intersect both \( Q_i \) and one of the special gadgets, therefore its slope must be between 0 and some value smaller than 1 and the points above \( r \) are outside \( g \). But then, in order to distinguish \( p \) and \( q \), \( r \) must have slope strictly bigger than the slope of the line segment \( pq \), which can be brought arbitrarily close to the slope of \( s \) (see Figure 4.6), which is 1, which leads to a contradiction. We conclude that at least \( 2 \deg(x_i) + 4 \) guards belong to \( Q_i \).

**How the remainder can be guarded.** As in the natural setting we observe that a guard belonging to a variable gadget only covers edges of variable gadgets, therefore the remaining \( 4m + 4n + 10 \) edges have to be covered by the remaining \( 2m + 2n + 5 \) guards and this can only be done by a natural vertex guard on every second vertex. Now this is not as trivial as it was in the natural setting. We crucially need Observation 4.5 (Again we have to deal with the exception of the leftmost edge of \( P(F) \) which might be covered by a natural vertex guard belonging to \( Q_i \) but this can be resolved in the same way.) The rest of the proof goes along the same lines: We define a variable gadget \( Q_i \) to be guarded positively or negatively depending on where the guarding changes from natural vertex guards on odd indices to natural vertex guards on even indices. Then we set the truth values of the variable \( x_i \) accordingly. This yields a satisfying assignment, because looking at a clause gadget \( R_j \) we can again argue that there must be some ray \( r \) distinguishing a pair of points as depicted in Figure 4.2. As in the natural setting such an \( r \) can only come from a natural vertex guard belonging to a variable gadget: The only rays that are not aligned to an edge of \( P(F) \) are exactly those claimed by the special gadgets in the second round. As shown in the second part, such a ray must come from a guard belonging to a variable gadget, because the remaining guards are all natural vertex guards. Apparently, such a ray—coming from some \( Q_i \) and claimed by a special gadget—cannot intersect a clause gadget. We conclude that at least one of the variable gadgets must be guarded in the correct way and therefore the clause \( C_j \) is satisfied by our choice of truth values. \( \square \)
4.3. Is the Wireless Localization Problem in NP?

The Wireless Localization Problem for Vertex Guards. Given a simple polygon \( P \) and an integer \( k \), is there a guarding for \( P \) using \( k \) vertex guards?

**Corollary 4.8.** The wireless localization problem for vertex guards is NP-hard.

*Proof.* The guarding given in Lemma 4.6 uses vertex guards only. Lemma 4.7 trivially remains true if we consider a guarding consisting of vertex guards only.

4.3 Is the Wireless Localization Problem in NP?

There is a difference between Theorem 4.1 and Theorem 4.4 that is maybe not obvious at first glance. We have shown that the natural wireless localization problem is NP-complete, but for the internal wireless localization problem we have only shown NP-hardness and nothing about membership in NP. Indeed, it is a tricky question whether the internal variant is in NP. For the natural setting, proving the problem to be in NP is almost trivial. There is only a limited number of possible natural guards, namely at most \( 2n(P) \) for a polygon \( P \) on \( n(P) \) vertices. Furthermore, all possible guards are easy to construct because they are directly given by features of the polygon. Therefore, we can check in polynomial time whether a set of guards is a guarding of \( P \). In the general setting the situation is more complicated. What is the set of “reasonable” guards for a polygon \( P \)? Even if we restrict our attention to vertex guards, there are still infinitely many possible angular ranges for a guard. So it is not clear if every guard in a solution has a polynomial size description in the input size of \( P \).

To fix this, one might try to prove some statement of the following form: Every polygon \( P \) has an optimal guarding such that each guard lies on a vertex of the line arrangement given by \( P \) and its rays are collinear with lines of the arrangement. But even if we are very generous about what is meant by “the line arrangement given by \( P \)”, it looks difficult to prove such a statement. Let us discuss the strictest variant: Let the line arrangement of \( P \) be the set of lines \( \{\overline{e} | e \in P\} \) defined by the edges of \( P \). The example shown in Figure 4.10 provides a hint that there might be polygons where an optimal guarding is forced to use guards whose rays are not collinear with lines of this arrangement. It is inspired by an example used by O’Rourke [57] to prove that for an optimal convex covering of a polygon one might be forced to use pieces whose vertices do not lie on the arrangement given by the edges of \( P \), and which has become famous as the SoCG logo. The example shows what we believe is an optimal
Figure 4.10: A polygon where it seems to be necessary to use “strange” guards which are not directly given by polygon features: It is not possible to move the dashed quadrangle inside the polygon such that all its sides lie on lines defined by polygon edges and it still contains all uncovered (white) areas.

solution and it uses two guards whose rays are not aligned to any lines given by polygon edges.

For many other geometrical problems the question of membership in NP has remained open as well (for instance, see the remarks in O’Rourke [59], p. 232, or in Culberson and Reckow [22] in the conclusion). We discuss a positive result for a related problem in Section 5.6.
5 Describing a Polygon with Triangles

Ladies and gentlemen, many songs have been written — and this is one of them.

Mo’Horizons, Yes Baby Yes

How can a polygon be described by triangles? Instead of using cones (guards) as primitives, we turn our attention to slightly more complicated primitives, namely triangles. Opposed to cones, triangles are bounded objects. Consequently, we are not only interested in (monotone) descriptions, but it makes also sense to ask for a cover or a partition of a simple polygon into triangles. The main focus of the chapter is on the cover problem; for an extended abstract see [11]

5.1 Introduction

One of the most basic problems in computational geometry is to triangulate a polygon. Very often triangulations are the most important building block for other algorithms. For example, think of the famous proof by Fisk [33] for the classical art gallery theorem. Devadoss and O’Rourke start the first chapter of their book “Discrete and Computational Geometry” [24] by comparing polygons in planar geometry to integers in numerical mathematics and calling triangulations “the prime factorization of polygons”. A triangulation of a polygon $P$ is a set of triangles such that their union is $P$ and the intersection of two triangles is either empty, a common vertex or an edge. It is well-known that triangulations always exist and how to construct one for a given polygon. Furthermore, it is possible to triangulate every polygon without introducing any new vertices, which for a simple polygon always results in a triangulation consisting of $n(P) - 2$ triangles. So usually one forbids new vertices (Steiner points) from the start and insists that the edges of the triangles are either diagonals or polygon edges.
Depending on the application, we might not care about the way the triangles intersect, which leads to the natural problem of describing a polygon as a union of arbitrary triangles. Using the terminology of Chapter 2, we define the objects of interest \( P \) to be all polygonal regions and the set of admissible primitives \( Q(P) \) to be the set of all triangles in the plane independent of \( P \in P \). A cover of \( P \) is a set of triangles \( \{t_1, \ldots, t_k\} \) whose union equals \( P \) and \( \gamma(P) \) denotes the minimum number of triangles needed to cover \( P \).

Obviously, a triangulation is always a solution to the cover problem. Therefore, we know how to cover a simple polygon \( P \) with \( n(P) - 2 \) triangles and conclude \( \gamma(P) \leq n(P) - 2 \) for simple polygons. This first observation immediately carries over to general polygonal regions. If \( P \) is an unbounded polygonal region, it is not possible to cover it with a finite number of triangles. A bounded polygonal region \( P \) with \( k \) connected components and \( h \) holes in total can always be triangulated into \( n(P) + 2h - 2k \) triangles, therefore \( \gamma(P) \leq n(P) + 2h - 2k \) (for a proof see O’Rourke [59], Lemma 5.2). The interesting question is whether we can do any better than that. It is easy to come up with examples where the answer is no. In Section 5.7 it is shown that a convex polygon \( P \) cannot be covered by fewer than \( n(P) - 2 \) triangles, that is, \( \gamma(P) = n(P) - 2 \) if \( P \) is convex. It follows that \( \gamma(n) = n(P) - 2 \). The question remains how to compute \( \gamma(P) \) for a general polygon:

The Triangle Cover Problem: Given a bounded polygonal region \( P \) and a positive integer \( k \), are there triangles \( t_1, \ldots, t_k \) such that \( P = \bigcup_{i=1}^{k} t_i \), that is, \( \gamma(P) \leq k \)?

Refining the proof of Culberson and Reckow [22] for the convex cover problem, we show NP-hardness of the triangle cover problem (Section 5.2). The reduction can be shown to be gap-preserving, implying APX-hardness (Section 5.3). In Section 5.4 we explain how the reduction can be adapted to yield a polygon in general position.

So far, we have not imposed any restrictions on the triangles in a cover. A natural restriction is to allow only triangles the vertices of which are also vertices of \( P \). Formally, \( Q(P) = \{t \in P \mid t \text{ is a triangle and } V(t) \subseteq P\} \). If we use these primitives, we call a cover a triangle cover without Steiner points. In Section 5.5 we modify the reduction to settle the complexity of the convex cover problem without Steiner points, a question not addressed by Culberson and Reckow and appearing as an open problem in the Handbook of Discrete and Computational Geometry [63].

As it is the case for many geometrical problems (e.g., the minimum-weight triangulation problem [55]), it is not clear whether these covering problems
are in NP (see also the remarks in O’Rourke [59], p. 232, or the conclusion Culberson and Reckow [22]). Forbidding Steiner points, the covering problems obviously are in NP, as we can describe a solution in terms of the polygon vertices and check it efficiently. The typical decomposition problems for rectilinear polygons are trivially in NP, too. But for the general variants of the cover problems allowing Steiner points, it is not obvious how a solution can be described in terms of the input.

The situation changes if the primitives are required to cover the boundary of the input region only. Again formulating it as a decision problem, we define

**The Boundary Cover Problem:** Given a bounded polygonal region $P$ and an integer $k$, are there triangles $t_1, \ldots, t_k \subseteq P$ such that $\partial P \subset \bigcup_{i=1}^{k} t_i$?

This is the problem treated in Section 5.6. As for the convex boundary cover problem [22], the boundary cover problem with triangles remains NP-hard. But interestingly, it can be shown to be in NP. The question remains whether we can find efficient approximation algorithms. The usual approaches for covering problems boil down to finding an abstract set cover greedily and therefore lead to factor $O(\log n)$ approximation algorithms [36, 28]. But in the context of triangles, this is not satisfactory. If the input polygon does not have any collinear edges, a triangle can cover at most three polygon edges. So simply triangulating is a factor 3 approximation algorithm for the problem of minimizing the number of triangles in a cover, both for the boundary and the general cover problem. Having shown APX-hardness on the other side, there remains a small range for interesting approximation ratios. For the boundary cover problem, we give an efficient factor 2 approximation algorithm.

### 5.2 NP-hardness of the Triangle Cover Problem

To show NP-hardness we reduce the Boolean satisfiability problem (SAT) to the triangle cover problem. Let $F$ be a CNF formula with clauses $C_1, \ldots, C_m$ on the variables $x_1, \ldots, x_n$. We call the number of clauses a variable $x_i$ appears in the degree of $x_i$ and denote it by $\deg(x_i) := |\{C_j \mid x_i \in C_j \text{ or } \overline{x_i} \in C_j\}|$ (we assume that no variable appears more than once in a clause).

We define different gadgets, all of which are simple polygons. In the end we take these gadgets and glue them to the boundary of a big triangle to form one big simple polygon.

The first basic ingredient are the switch gadgets, see Figure 5.1. Such a switch gadget can be covered optimally in essentially two ways. Thus, it can be used
to encode the truth value of a variable. Depending on how it is covered, one of the triangles in its cover can be extended through the opening (that is, where it is going to get glued to the rest) to cover additional points far away. More precisely, we need at least 4 triangles to cover such a gadget and exactly one of these can be stretched out and cover an additional area far away, but only in a specific direction. Exactly one of two triangles in the cover can be stretched, not both at the same time.

The other basic ingredient is the clause gadget. We define a clause gadget $R_j$ for every clause $C_j$. They all look exactly the same. If we use two triangles to cover it, there remains a small spot uncovered (depicted by a circle). So we have to waste a third triangle to do that, unless it is possible to extend the triangle of a switch gadget to do the job.

![Figure 5.1: A switch gadget (A) with its four special points (disks) and its central point (cross); a clause gadget (B) with its two special point and the uncovered spot marked with a circle.](image)

For every variable $x_i$, we define a variable gadget $Q_i$, which consists of a big main switch gadget and $\deg(x_i)$ many small switches. To the opposite of every small switch, we put a small clause-gadget-like structure. The small switches corresponding to clauses $C_j$ where $x_i$ appears as a positive literal are put to the bottom left of the main switch, the small switches corresponding to clauses $C_j$ where $x_i$ appears as a negated literal are put to the right. Depending on how the main switch is covered, namely which of the possible triangles gets stretched out, either the uncovered spots of the left clause-gadget-like structures or the uncovered spots of the right clause-gadget-like structures get covered. If the spots at the bottom right get covered, the small switches at the bottom left that correspond to positive occurrences of $x_i$ can be used to cover the small spot left by the corresponding clause gadgets, but the small switches at the top right (corresponding to negative occurrences of $x_i$) cannot cover their corresponding clause gadget, because we are forced to extend their other free triangle so they
can cover the small spots left by their own clause-gadget-like structures at the top left. If the spots at the top left get covered by the main switch, then the exact opposite happens: the small switches at the top right can be used to cover their corresponding clause gadgets, whereas the small switches at the bottom left are bound to stay inside the variable gadget. (In addition to the clause-gadget-like structures leaving an uncovered spot, there are two more smaller such structures that do not leave an uncovered spot and are merely used to connect the others with the main switch leaving room for the triangle of the main switch if it should get extended.)

Figure 5.2: Variable gadget $Q_i$ for $x_i$ with $\deg x_i = 7$, $x_i \in C_{i_1}, C_{i_2}, C_{i_3}, C_{i_4}$ and $\bar{x}_i \in C_{i_5}, C_{i_6}, C_{i_7}$; special points marked by disks, uncovered spots by circles.

Finally, glue the variable gadgets to the upper edge and the clause gadgets to the lower edge of a huge triangle. Adjust the small switches in the variable gadgets such that their free triangle can cover what it is supposed to cover, namely exactly the uncovered spot of the specific clause gadget the small switch
stands for. Looking at a switch gadget, we notice that there is no freedom where
the triangle can be extended to, because one edge of an extended triangle is
determined by an edge of the switch gadget. So we have to adjust these edges
such that each of them exactly points to the spot intended for it. The whole
procedure results in a simple polygon $\phi(F)$.

In the following we do not give any explicit constructions. But with some
extra effort one can find explicit coordinates, in particular, the construction
can be done on an $O((n + m)^2)$-grid.

**Lemma 5.1.** If $F$ is satisfiable, then $\gamma(\phi(F)) \leq \sum_{i=1}^{n}(6 \deg(x_i) + 10) + 2m + 1$.

**Proof.** Depending on the truth value of $x_i$ we cover the variable gadget $Q_i$ either
extending the left or the right triangle of the main switch with $6 \deg(x_i) + 10$
triangles as shown in Figure 5.2. Then, we cover the clause gadgets with two
triangles each. As $F$ is satisfied by the assignment, for every clause there is
at least one variable gadget that is covered in such a way that there is a long
thin triangle that extends to the uncovered spot of the corresponding clause
gadget. The huge space between variable and clause gadgets can be covered
by one additional triangle. So summing up, we get a cover with $\sum_{i=1}^{n}(10 +
6 \deg(x_i)) + 2m + 1$ triangles.

**Lemma 5.2.** If $\gamma(\phi(F)) \leq \sum_{i=1}^{n}(\deg(x_i) + 10) + 2m + 1$, then $F$ is satisfiable.

**Proof.** Assume there is a cover of $\phi(F)$ consisting of $k := \sum_{i=1}^{n}(\deg(x_i) +
10) + 2m + 1$ triangles. Fix special points as shown in the figures by disks.
Additionally, put one special point somewhere into the huge triangle connecting
everything such that it cannot be seen by any other special point. Notice that
all special points are pairwise invisible. So no two of them can be covered by
the same triangle. There are exactly $k$ special points, so each triangle covers
exactly one special point.

We say a triangle belongs to a gadget if it is the triangle corresponding to one
of the gadget’s special points. Now we want to derive a satisfying assignment.
Look at $Q_i$. In the middle of its main switch there is a central point denoted
by a cross in the figure. At least one triangle has to cover the central point.
Now the only special points that are visible from there are the leftmost and
the rightmost convex vertex of the main switch. So the only triangles that are
able to cover the cross are those corresponding to these two special points. At
most one of the two does not cover the central point, so we define the truth
value of $x_i$ accordingly: If the triangle corresponding to the left special point
does not cover the central point, we define $x_i$ to be positive, else we set $x_i$ to
negative.
Now we claim that this yields a satisfying assignment of $F$. Look at a clause gadget $R_j$. We are going to argue that the corresponding clause $C_j$ is satisfied. There is a triangle $t$ in the cover that covers the central point of $R_j$, again depicted by a cross. Now the only special points visible from the central point of $R_j$ are those belonging to small switches of corresponding variable gadgets, that is, variable gadgets $Q_i$ such that $x_i \in C_j$ or $\overline{x}_i \in C_j$. So $t$ corresponds to a small switch gadget of a variable gadget, say $Q_i$, and assume without loss of generality $x_i \in C_j$. The central point of the small clause-gadget-like structure that lies opposite of this small switch to the right must be covered by a triangle, call it $t'$. Now because $t$ covers the central point of the clause gadget, it cannot cover the central point of the small switch gadget, which therefore has to be covered by the other free triangle of the small switch gadget, which cannot be $t'$. So $t'$ must belong to the main switch of $Q_i$ and it cannot cover the central point of the main switch, which implies that the variable $x_i$ is positive. 

**Theorem 5.3.** The triangle cover problem is NP-hard.

**Proof.** The reduction $\phi$ maps CNF-formulas to simple polygons in polynomial time. The triangle cover problem for $\phi(F)$ with $k = \sum_{i=1}^{n} (6 \deg(x_i) + 10) + 2m + 1$ has a solution if and only if $F$ is satisfiable. 

**5.3 Inapproximability**

We have shown that if $F$ is satisfiable, then $\phi(F)$ can be covered with $k = \sum_{i=1}^{n} (6 \deg(x_i) + 10) + 2m + 1$ triangles and if $F$ is not satisfiable, we need strictly more triangles. Now we strengthen Lemma 5.2 to get an inapproximability result. If $F$ is a 3-CNF where each variable has degree at most 13, then the reduction $\phi$ is also gap-preserving:

**Lemma 5.4.** Let $F$ be a 3-CNF of bounded degree 13. If at most a $1 - \varepsilon$ fraction of the clauses of $F$ can be satisfied simultaneously for an $\varepsilon > 0$, then we need at least $(1 + \varepsilon/144)k$ triangles to cover $\phi(F)$.

**Proof.** Let $F$ be a 3-CNF where every variable has degree at most 13 and at most $(1 - \varepsilon)m$ of the clauses can be satisfied. Assume for contradiction that $\phi(F)$ can be covered by less than $(1 + \varepsilon/144)k$ triangles. Consider such a cover. We proceed as in the proof of Lemma 5.2. We can find a mapping between the triangles of the cover and the special points. But now there might be up to $\varepsilon k/144$ more triangles than special points. So not all triangles belong to a gadget, but a some of them remain “uncontrollable”.
Formally, we define a triangle to be *uncontrollable* if it does not cover any special point. If there are several triangles covering the same special point, we arbitrarily pick one to be responsible for the special point and call the other triangles uncontrollable as well.

Consider a variable gadget $Q_i$. If one of the uncontrollable triangles intersects $Q_i$, it might happen that $Q_i$ is both positive and negative at the same time, in the sense that both triangles of the small switches to the left and of the small switches to the right extend down to the corresponding clause gadgets. But there are strictly less than $\varepsilon k/144$ uncontrollable triangles so at most that many variable gadgets are *ambiguous* in that they cover both clause gadgets they appear positively in and clause gadgets they appear negatively in. So from the guarding we only get a partial truth value assignment $\alpha$. But intuitively, already $\alpha$ satisfies most of the clauses.

Now we go through the remaining unset variables and set them either to true or false depending on which choice satisfies more of the remaining unsatisfied clauses. Each variable appears in at most 13 clauses. So there are at most 13 unsatisfied clauses $x_i$ appears in. So after choosing the better truth value for $x_i$, at most 6 of the clauses remain unsatisfied.

This leads to a complete assignment $\alpha'$, which satisfies at least a $1 - \varepsilon$ fraction of the clauses, that is, there remain at most $\varepsilon m$ clauses unsatisfied: Look at a clause gadget $R_j$. There must be a triangle covering its central point. Either there is one of the uncontrollable triangles directly covering it (which it can do for at most one clause gadget) or there is a triangle of a small switch of a variable gadget that covers it. If the latter is the case, then either this variable gadget is unambiguous and so the clause already got satisfied by the partial assignment $\alpha$ or the variable gadget is ambiguous and then there is one of the uncontrollable triangles responsible for it. An uncontrollable triangle can either cover at most one central point of a clause gadget or it can cause at most one variable gadget to be ambiguous. If a variable is ambiguous, after choosing the better truth value for it, it leaves at most 6 clauses unsatisfied. Therefore, an uncontrollable triangle is responsible for at most 6 clauses that remain unsatisfied by $\alpha$. So in the worst case, there remain $6 \varepsilon k/144 = \varepsilon k/24$ clauses unsatisfied. An easy calculation shows $k \leq 24m$, therefore at most $\varepsilon m$ remain unsatisfied.

**Theorem 5.5.** There is a constant $c > 0$ such that it is NP-hard to approximate $\gamma(P)$ within a factor smaller than $1 + c$.

**Proof.** For every $\varepsilon' > 0$, there is a polynomial time reduction from 3SAT to MAX3SAT such that satisfiable formulas get mapped to satisfiable formulas
and unsatisfiable formulas get mapped to formulas where at most a \(7/8 + \varepsilon'\) fraction of the clauses can be satisfied \([2]\).

Furthermore, there is a reduction from MAX3SAT to MAX3SAT(13) such that satisfiable formulas remain satisfiable and if at most a \(1 - \zeta\) fraction is satisfiable, at most a \(1 - \zeta/19\) fraction is satisfiable in the reduced formula, for any \(\zeta > 0\) \([2]\).

So putting these two reduction and \(\phi\) together, we get a reduction from SAT to triangle cover with the following two properties: First, satisfiable formulas get mapped to polygons coverable with \(k\) triangles. Second, unsatisfiable formulas get mapped to polygons such that every cover needs at least \((1+\varepsilon/144)k = (1+\zeta/(19\cdot144))k = ((1+(1/8-\varepsilon')/2736)k = (1+1/21888-\varepsilon'')k\) triangles, for some constant \(\varepsilon''\), which we can make arbitrarily small.

\[\square\]

### 5.4 Restricting to general position

It is possible to refine the reduction given in Section \([5.2]\) to get a simple polygon in general position. Note that the simple polygon \(\phi(F)\) produced by the reduction sketched above already is in general position if we look at individual gadgets. The only collinearities in \(\phi(F)\) stem from the fact that constructing the variable gadgets and then again \(\phi(F)\) itself, we glued a lot of gadgets to the same edge, thus producing a lot of collinear edges. To avoid them we do the following: Whenever there are several edges on the same line, we retain one of them and replace the other edges by small gadgets with the property that it needs two additional triangles to cover them that cannot be used to cover anything else. More specifically, we can replace every edge by a very flat clause-gadget-like structure as they already appear in the variable gadgets, but of course in such a way that they do not leave an uncovered spot. See Figure \([5.3]\). We do not give the details of this adapted reduction. The proofs exactly work the same, only that we have to introduce two additional special points for every flat clause-gadget-like structure we have added. If we do that as before, placing them into the two narrow corners, all special points will still be invisible to each other.

**Theorem 5.6.** There is a constant \(\varepsilon > 0\) such that it is NP-hard to approximate the triangle cover problem for polygons in general position within a factor smaller than \(1 + \varepsilon\).
5.5 Triangle cover without Steiner points

So far we have put no restrictions on the coordinates of the vertices of the covering triangles. What happens if Steiner points are forbidden, that is, the vertices of the triangles coincide with polygon vertices? Indeed, with this restriction, the problem changes considerably.

The Non-Steiner Triangle Cover Problem. Given a bounded polygonal region $P$ and an integer $k$, are there triangles $t_1, \ldots, t_k$ with $V(t_i) \subset V(P)$ for all $i$ such that $P = \bigcup_{i=1}^{k} t_i$?

It turns out that if we consider a polygon in general position, we cannot save anything compared to triangulations in the Non-Steiner setting. In other words, allowing the triangles to overlap does not help.

**Theorem 5.7.** If $P$ is a polygon in general position with $h$ holes, then a Non-Steiner triangle cover of $P$ contains at least $n(P) + 2h - 2$ triangles. If it has exactly $n(P) + 2h - 2$ triangles, then it is a triangulation of $P$.

**Proof.** Assume we have a Non-Steiner triangle cover of $P$ consisting of $k$ triangles. Consider a vertex $v \in V(P)$ and a triangle $t$ of the cover with $v \in t$.

Choose $\varepsilon$ small enough such that every triangle which covers some point in $B_{\varepsilon}(v)$ also covers $v$ itself and such that $v$ is the only vertex in $B_{\varepsilon}(v)$. Because $t \subset P$ and $v \in \partial P$, $v \in \partial t$. Because $t$ is a Non-Steiner triangle, all vertices of $t$
are in \( V(P) \), so if \( v \) is not a vertex of \( t \), but in the relative interior of one of its edges, then there are three polygon vertices on a common line, contradicting the general position assumption. Therefore, \( v \in V(t) \).

Define \( \alpha(v) \) as the interior angle of the polygon at a vertex \( v \). The (interior) angles at \( v \) of all triangles covering \( v \) summed up must be at least \( \alpha(v) \). Otherwise we could find some point in every \( \varepsilon \)-neighborhood of \( v \) which does not get covered by any triangle. Summing up this inequality over all polygon vertices we get

\[
(n + 2h - 2)\pi = \sum_{v \in V(P)} \alpha(v) \leq k\pi.
\]

If we have equality, then this means that no two triangles overlap at any vertex \( v \). So if we look at the situation locally at \( v \), it looks like a triangulation.

Assume there are two triangles \( t \) and \( t' \) that overlap. A vertex of \( t' \) cannot be in the interior of \( t \), neither can it be in the relative interior of an edge of \( t \). So we find a vertex \( v \in V(t) \) such that there is an edge \( e \in E(t') \) intersecting both edges of \( t \) incident to \( v \). For a fixed \( t \), pick the edge \( e \) of another triangle \( t' \) such that there is no other triangle edge crossing \( t \) between \( e \) and \( v \). Now look at the edge \( e \). It must be a diagonal of \( P \). On one of its sides there is \( t' \), on the other side, namely the side where \( v \) lies, there must be another triangle, call it \( t'' \), it is an edge of (because at the endpoints of \( e \), the cover looks like a triangulation). Then \( t'' \) has a third vertex \( w \) not incident to \( e \). If \( w = v \), we have a contradiction, because then \( t \) and \( t'' \) overlap at \( v \), if \( w \neq v \) we get a contradiction because we chose \( e \) such that there is no other triangle edge crossing \( t \) between \( v \) and \( e \).

This theorem shows that in the case of general position, an optimal Non-Steiner cover is equivalent to a triangulation. A triangulation can be found in linear time in the case of simple polygons \[7\] or in time \( O(n \log n) \) for a polygon with holes \[3\]. However, if we drop the general position assumption, there are examples of polygons that can be covered by fewer than \( n - 2 \) triangles — the Star of David for example. It turns out that finding an optimal Non-Steiner cover is NP-hard if we allow collinear edges. To prove this, we basically use the same reduction as before, we just have to get rid of all Steiner points. We adjust \( \phi(F) \) such that every triangle that appears in the solution as given in Lemma 5.1 has all vertices on \( \partial \phi(F) \). Instead of only allowing polygon vertices to be triangle vertices, we fix a finite set \( W \) with \( V(P) \subset W \subset \partial P \), which are the allowed triangle vertices in a solution. We call \( W \) the set of generalized vertices. (Of course, another way to look at it would be to think of the polygon as the ordered list of its vertices and allow consecutive collinear polygon edges.)
It is possible to reduce this variant of the problem to the standard Non-Steiner Triangle Cover Problem.

**Theorem 5.8.** If \( P \) is a simple polygon and \( W \) a finite set of generalized vertices, that is, \( V(P) \subset W \subset \partial P \), then we can compute another simple polygon \( P' \) in polynomial time such that \( P \) has cover consisting of \( k \) triangles using only points in \( W \) if and only if \( P' \) has a Non-Steiner triangle cover consisting of \( k + m \) triangles, where \( m = |W \setminus V(P)| \).

**Proof.** First scale \( P \) such that its diameter becomes 1. Consider the line arrangement we get by drawing a line through every pair of points in \( W \). Let \( \varepsilon > 0 \) be smaller than the area of every 2-dimensional cell of this arrangement. Note that we can compute such an \( \varepsilon \) in polynomial time. Then we construct \( P' \) as follows. For every \( w \in W \setminus V(P) \) we attach a small spike, as shown in Figure 5.4, such that the interior angle of the spike is equal to \( \alpha \), where \( \alpha = \arctan(\varepsilon/(k + m)) \), and the width of the spike is at most \( \varepsilon/(3(k + m)) \) and small enough to make the spike fit without intersecting the polygon. If necessary, tilt it a little bit such that the sector we get by prolonging the two edges of the spike (dashed in the figure) does not contain any point of \( W \). First we observe that a \( k \)-cover of \( P \) using only vertices in \( W \) carries over to a \((k + m)\)-cover of \( P' \): Keep all the triangles as they are and add one for every small spike.

Now assume that \( P \) has no \( k \)-cover on \( W \). This means that every possible set of \( k \) triangles using only vertices from \( W \) and being inside \( P \) leaves at least one cell of the arrangement uncovered. Assume for contradiction that \( P' \) has a Non-Steiner \((k + m)\)-cover. At least one triangle has to cover the vertex at the tip of every spike. No triangle can cover two such vertices at the same time. All \( m \) triangles covering a spike tip have area strictly less than \( \varepsilon/(k + m) \). After removing them, we adjust the remaining triangles that have vertices on a spike-vertex by replacing these vertices by the point \( w \) the corresponding spike came from (if it is not on \( w \) already). Note that the resulting triangle still is contained in \( P \): if one of the edges of the new triangle were leaving \( P \), then this would imply a cell in the arrangement of area less than \( \varepsilon \). Because we chose the spikes to be very small, a vertex gets moved by at most \( \varepsilon/(3(k+m)) \). Therefore, the area of a triangle \( t \) can decrease by at most \( \varepsilon/(k + m) \): the perimeter of \( t \) is at most 3, so the area of \( t \) decreases by less than \( 3\varepsilon/(3(k + m)) \). So we get a potential cover of \( P \) that for sure leaves less than \( (k + m)\varepsilon/(k + m) = \varepsilon \) area uncovered, a contradiction.

**Theorem 5.9.** The Non-Steiner triangle cover problem is NP-complete.
5.5. Triangle cover without Steiner points

Proof. We adapt the reduction in Section 5.2 such that all triangle vertices are going to lie on ∂φ(F). See Figure 5.5. Now each switch gadget has five special points and each clause gadget has got four special points (shown by disks). Note that the switch gadgets can be covered by five triangles whose vertices are either on vertices of the gadget or on one of the four additional points on the boundary depicted by boxes—the generalized vertices. One of two triangles, either the triangle associated with the leftmost special point or the one associated with the rightmost special point can be extended (provided that there is a vertex or generalized vertex we can extend to). But note that extending both triangles would leave the central point of the switch gadget (cross) uncovered. A clause gadget can be covered by four triangles whose vertices lie on vertices of the gadget. As in the original setting, covering the gadget in this obvious way using four triangles leaves an uncovered spot (depicted as usual by a fat circle). The variable gadgets are constructed the same way as before, the only difference being the the clause-gadget-like structures, which might look pretty weird now in order to make them coverable by triangles with vertices on the boundary. Now it can be shown using the same proof that F is satisfiable if and only if φ(F) can be covered with \( \sum_{i=1}^{n} (7 \deg(x_i) + 11) + 4m + 1 \) triangles the vertices of which are in \( V(\phi(F)) \) or additional generalized vertices on ∂φ(F).

This reduces SAT to an instance of the triangle cover problem with a set of generalized vertices. So in polynomial time we get \( \phi(F) \) and a set \( W \) which contains the vertices of \( \phi(F) \) and the additional generalized vertices as shown in the figure. Then we apply the reduction given in Theorem 5.8 yielding a polygon \( \phi'(F) := (\phi(F))' \) where all the generalized vertices have been replaced by spikes. Then, \( \phi'(F) \) can be covered with a certain number \( k \) of triangles if and only if the formula \( F \) is satisfiable.

Finally we observe that this reduction works for convex cover setting as well, thus resolving a question stated in the Handbook of Discrete and Computational Geometry [63].

**Theorem 5.10.** The problem of covering a simple polygon with a minimum
number of convex polygons without Steiner points is NP-complete: Given a polygon \( P \) and an integer \( k \) the problem of deciding whether there is a set of \( k \) convex polygons with all vertices in \( V(P) \) whose union equals \( P \) is NP-complete.

Proof. We can replace the term “triangle” by “convex polygon” in the relevant proofs and all statements still hold. (Of course, this is also true in the general setting allowing Steiner points.)

5.6 Covering the Boundary

Let us consider the setting where only the edges of a polygonal region have to be covered by triangles, which, however, still have to be inside \( P \). Recall the definitions in Section 2.3: a line segment \( s \) is covered by a triangle \( t \subseteq P \) if \( t \) intersects \( s \) in more than just one point. If \( s \subseteq t \), \( t \) covers \( s \) completely. Otherwise, if only a subsegment of \( s \) is contained in \( t \), \( t \) covers \( s \) partly. If we have a cover of \( P \), then every edge has to be covered either completely by some triangle or partly by at least two triangles (see Observation 2.5).

**Theorem 5.11.** The boundary cover problem is in NP.

Proof. We show that the problem is in NP by restricting the set of possible triangles in a solution. We define a triangle contained in a polygonal region
5.6. Covering the Boundary

Figure 5.6: A problematic triangle and how its free edge (fat) gets rotated to make it fair.

$P$ to be *nice* if each of the three lines defined by its edges contains at least two vertices of $P$. So nice triangles have a compact description in terms of the input. We call a triangle *problematic* if it covers two polygon edges $e, f \in E(P)$ only partly and one of its edges $a$ starts in the relative interior of $e$ and ends in the relative interior of $f$ and does not cover any polygon edges. Note that problematic triangles are not nice in general.

Two of their defining lines are given by polygon edges (namely $e$ and $f$) whereas the third triangle edge $a$ can be assumed to contain one polygon vertex $v \in V(P)$ (if not, just move $a$ outwards until it does) but not more. If we rotate $a$ around $v$ keeping the other two defining lines fixed, the triangle covers more of, say, $e$ while it covers less of $f$, see Figure 5.6, so there is no unique way to make a problematic triangle nice without possibly ruining the boundary cover. We call $a = a(t)$ the *free edge* of the problematic triangle $t$.

Let $t$ be a triangle in a boundary cover of $P$. We can replace $t$ by a nice or problematic triangle $t'$ such that $t \cap (\partial P \setminus V(P)) \subseteq t' \subseteq P$, that is, $t'$ covers at least as much of the boundary as $t$ with the exception of the polygon vertices. (If all boundary points that are not vertices of $P$ are covered, then so are the vertices.) If each edge of $t$ covers a polygon edge (partly or completely), then $t$ is already nice and we set $t' = t$.

If an edge of $t$ does not cover any polygon edge, then its interior is completely contained in the interior of $P$ or the only boundary points it contains in its interior are vertices of $P$. So if only one of the edges of $t$ covers a polygon edge, then we can replace $t$ by the degenerate triangle $t'$ which only consists of this one edge, and therefore is nice.

If $t$ has exactly one edge $a$ that does not cover any polygon edge, then either $t$ is problematic and we directly set $t' = t$ or we can shrink $t$ by moving $a$ and keeping the other two defining lines fixed until at least one of the endpoints of $a$
is on a polygon vertex \( v \in V(P) \). In a second step we enlarge \( t \) again by rotating \( a \) around \( v \) and keeping the rest fixed, until it contains a second polygon vertex. This results in a triangle \( t' \) which is nice and \( t \cap (\partial P \setminus V(P)) \subseteq t' \). So either \( t \) is nice or problematic from the very beginning or we can make it nice as described. So from now on we may assume that the boundary cover contains nice or problematic triangles only.

If there are no problematic triangles or if all problematic triangles happen to be nice, we are done. Otherwise we take care of them as follows: Let \( t \) be a problematic (and non-nice) triangle whose free edge \( a(t) = pq \) does not cover any edge and the other two edges cover the polygon edges \( e, f \in E(P) \) partly, \( p \) is in the relative interior of \( e \) and \( q \) in the relative interior of \( f \) as shown in Figure 5.6. In the cover there has to be another triangle \( t' \) also covering \( e \) and \( p \in t' \). And there has to be a third triangle \( t'' \) covering \( f \) and \( q \in t'' \). We may assume that \( a \) contains exactly one polygon vertex \( v \in V(P) \) in its relative interior. (If it contains no vertex, blow \( t \) up until it does; if it contains already two, it is nice.) Now if we rotate \( a \) around \( v \), as shown in Figure 5.6, we might have to move \( t' \) and \( t'' \) along in order to keep \( \partial P \) covered. If one of the triangles \( t' \) or \( t'' \) is nice, we shrink the part of \( e \) covered by \( t \) (of \( f \), respectively) rotating \( a \) around \( v \) as far as possible, that is, until \( t \cap t' \cap e \) (\( t \cap t'' \cap f \), respectively) is a single point, thus increasing the part covered on the other side. Then, \( t \) has not necessarily become nice, but at least it has got a compact description: two of its defining lines are directly given by polygon edges and the third defining line through \( a \) is now defined by a polygon vertex and a vertex of another triangle in the cover. See Figure 5.6. We call such a triangle that can be described by five polygon vertices and one vertex of another triangle \textit{fair}.

It might happen that both \( t' \) and \( t'' \) are problematic as well. In this case, we say that \( t \) \textit{depends} on both \( t' \) and \( t'' \). More formally, we call two problematic triangles \( t \) and \( t' \) \textit{dependent}, if they cover a common edge \( e \in E(P) \) and \( t \cap t' \cap e = pp' \), where \( p = e \cap a(t) \) and \( p' = e \cap a(t') \) are the endpoints of the free edges of \( t \) and \( t' \) (possibly \( p = p' \)). This defines an abstract graph on the problematic triangles, in which each triangle has degree at most 2.

If a problematic triangle has degree at most one in this graph, we can make it fair as described above, and then remove it from the graph. Let \( t_1, \ldots, t_k \) be a cycle in the dependency graph. We replace the triangles \( t_1, \ldots, t_k \) by degenerate triangles \( b_1, \ldots, b_k \) without ruining the cover, where \( b_i \) is defined as the line segment which is the union of the two triangle edges of \( t_i \) and \( t_{i+1} \) that together cover a polygon edge.

So we may assume that every triangle in a boundary cover is either nice or fair. And even though fair triangles may use vertices of other fair triangles in their description, there are no cyclic dependencies. Note that constructing
5.6. Covering the Boundary

the triangles as shown above involves constructing the intersection of lines several times. We have to be careful that the size of the coordinates does not explode when constructing the fair triangles. But every time we construct the missing coordinates of a fair triangle using the coordinates of another fair triangle, only one of the points comes from this other fair triangle and the other points involved in the construction are polygon vertices. So the bit size of the coefficients of the defining lines only grows by an additive term in each step (and this term is bounded by the input size):

So we have replaced the arbitrary boundary cover by a cover consisting of triangles, which can be described in polynomial space. And if such a set of triangles is given, it can be checked in polynomial time if it covers $P$. It follows that the boundary cover problem is in NP.

Theorem 5.12. The boundary cover problem is NP-hard

Proof. The NP-hardness proof is again very similar to the one given for the general cover problem. See Figure 5.7 to see how the gadgets can be changed to work for the boundary cover problem: The clause gadgets have two special points and can be covered with two triangles leaving the fat edge uncovered. A switch gadget has five special points (shown by disks) that are invisible to each other, so we need at least five triangles to cover the boundary of a switch gadget. Now there are two points taking over the role of the central point (denoted by crosses). At most one of them can be covered by the triangle that covers the special point in the middle. Which one is covered by this triangle in the main switch of a variable gadget stands for to the truth value of the variable. The other of the two crosses has to be covered by the triangle that covers the special point on the middle right (or middle left, respectively), hence preventing it from being extended. The other triangle can be extended and thus cover the “uncovered edges” (depicted as fat edges) between the “clause-gadget-like structures” (if the switch is a main switch) or of the corresponding clause gadget (if it is a small switch). The “clause-gadget-like structures” are even simpler than in the original reduction: The just consist of three edges that can be covered by one triangle extending from the main switch, those uncovered fat edges have to be collinear. Hence, this reduction does not work for polygons in general position. The rest of the proof works similarly.

Finally, we consider an approximation algorithm for the problem of finding a cover with as few triangles as possible. Note that if the polygon is in general position, just covering any edge by its own triangle yields a trivial factor 3 approximation algorithm for the boundary cover problem. Similarly, just taking
any triangulation is a trivial factor 3 approximation algorithm for the general triangle cover problem. In the boundary cover case, there is a pretty simple factor 2 approximation algorithm.

**Theorem 5.13.** Given a bounded polygonal region $P$ (not necessarily in general position), we can find a boundary cover for $P$ in time polynomial in $n(P)$ that uses at most twice as many triangles as an optimal solution.

**Proof.** Let $P$ be a polygonal region with $n = n(P)$ edges. If two or more polygon edges are on the same line and have the same orientation (that is, the interior of $P$ to the same side) and the line segment between them is in $P$, we merge them to one edge. Formally, We define two edges $e, f \in E(P)$ to be equivalent if there is a line segment $s \subset P$ such that both $e \subset s$ and $f \subset s$. This defines an equivalence relation on $E(P)$. We call the equivalence classes **merged edges** and denote the set of merged edges by $E', n' := |E'|$. The first step is to define an abstract graph $G$ on $E'$: $e, f \in E'$ are adjacent in $G$ if there...
is a triangle $t \subset P$ such that both $e \subset t$ and $f \subset t$.

If $P$ is a simple polygon, the abstract graph $G$ can be constructed in time $O(n^2)$ by constructing the visibility graph of $V(P)$ first. If there are holes, in time $O(n^2 \log n)$: Fix a merged edge $e$. We are going to find the neighbors of $e$ in $G$ in time $O(n \log n)$. Pick a point $p$ in the interior of $e$ such that $p \in \partial P$. Construct the visibility polygon $W$ with respect to $p$, which can be done using an angular sweep around $p$ in time $O(n \log n)$ \[69\]. Then preprocess $W$ to allow ray-shooting queries in $O(\log n)$ time, which can be done in time $O(n \log n)$ as well, using a geodesic triangulation \[9\]. Finally, go through the edges of $W$. If we find an edge $f$ of $W$ which is also a complete edge of $P$, we have to find out if there is a triangle in $W$ containing $e$ and the merged edge $f$ is part of. This can be done doing two ray-shooting queries along the two directions of $f$ in logarithmic time.

The second step is to find a maximum matching $M$ in $G$, which can be done in $O(n^{5/2})$ time using the algorithm by Micali and Vazirani \[53\] or in time $O(n^{2.376})$ using the algorithm by Mucha and Sankowski \[54\]. Now $M$ corresponds to a set of triangles $T$ such that in total, the triangles in $T$ cover at least $2|M|$ merged edges completely. For all the remaining $n' - 2|M|$ merged edges, we choose its own triangle to cover it and add it to $T$. Now the triangles in $T$ cover all merged edges completely and therefore the whole boundary of $P$. So we have found a boundary cover of $P$ consisting of $n' - |M|$ triangles. The running time of the algorithm is dominated by the computation of the matching.

Finally, we show that $n' - |M|$ is at most twice as much as there are triangles in an optimal solution. Let $a$ be the number of triangles in a fixed optimal solution that cover at least 2 merged edges exclusively (that is, such a triangle completely covers two edges that no other triangle covers). Consider the following charging scheme. Initially, every merged edge gets a charge of 1. Now each merged edge discharges evenly to all triangles that cover it. A triangle can cover at most 3 merged edges (partly or completely), namely at most one on each side. Therefore a triangle contributing to $a$ gets charged at most 3 and all other triangles get charged at most 2 (namely at most 1 from a merged edge possibly covered completely, and at most 1/2 each from up to 2 merged edges covered partly). Summing up, we get $3a + 2(OPT - a) \geq n'$, where $OPT$ denotes the number of triangles in an optimal solution. This implies $a \geq n' - 2OPT$. We observe $|M| \geq a$ and conclude $n' - |M| \leq 2OPT$. \[\square\]
5.7 CSG-Representations Using Triangles

Recall the general framework introduced in Chapter 2. What else can we say about CSG-representations of simple polygons using triangles (allowing Steiner points)? It is easy to show that for simple polygons, \( \beta(n) = \gamma(n) = n - 2 \) by observing that a convex polygon cannot be covered by fewer triangles. We do not know of any way to improve on that using monotone descriptions. However, for general descriptions the corresponding result for guards as primitives implies \( \delta^c(n) \leq 3n/4 \) for polygons without parallel edges.

**Theorem 5.14.** [72] Any simple polygon \( P \) can be partitioned into \( n(P) - 2 \) triangles. A convex polygon \( P \) cannot be covered by fewer than \( n(P) - 2 \) triangles. Consequently, \( \beta(n) = \gamma(n) = n - 2 \).

**Proof.** The proof is reminiscent of the proof of Theorem 5.7. A simple polygon \( P \) can be triangulated into \( n(P) - 2 \) triangles, so \( \gamma(n) \leq \beta(n) \leq n(P) - 2 \).

Let \( S \) be a triangle cover of a convex polygon \( P \) consisting of \( k \) triangles. Consider a vertex \( v \in V(P) \). Choose \( \varepsilon > 0 \) small enough such that every triangle \( t \in S \) which covers some part of \( B_\varepsilon(v) \) also contains \( v \) itself and such that \( v \) is the only vertex of \( P \) in \( B_\varepsilon(v) \). If a triangle \( t \) covers \( v \), then \( v \in V(t) \). Define \( \alpha(v) \) as the interior angle of the polygon at a vertex \( v \). The interior angles at vertex \( v \) of all triangles covering \( v \) summed up must be at least \( \alpha(v) \). The sum of the interior angles of a triangle is \( \pi \) and the sum of the angles of \( P \) is \( (n(P) - 2)\pi \). Summing up the inequality over all vertices of \( P \) we observe

\[
(n(P) - 2)\pi = \sum_{v \in V(P)} \alpha(v) \leq k\pi
\]

and conclude \( k \geq n(P) - 2 \). (If equality holds, then no two triangles overlap at any vertex \( v \) and it follows that \( S \) is a triangulation, see the proof of Theorem 5.7 for details.)

**Theorem 5.15.** Any simple polygon \( P \) without parallel edges can be described by \( \delta(P) \leq \lfloor (3n - 2)/4 \rfloor + 1 \) triangles.

**Proof.** Let \( t \) be a big triangle such that \( P \subset t \). By Corollary 3.14 there is a guarding \( \mathcal{G} \) for \( P \) using at most \( \delta(P) \leq \lfloor (3n - 2)/4 \rfloor + 1 \) guards. For any convex guard \( g \in \mathcal{G} \), pick a triangle \( t_g \) such that \( t_g \cap t = g \cap t \). For any reflex guard \( h \in \mathcal{G} \), construct a triangle \( t_h \) with the property that \( t_h^c \cap t = h \cap t \). (using regularized set operations, see Section 2.1). Let \( \mathcal{G}' \) be the formula obtained by replacing each convex guard \( g \) by \( t_g \) and each reflex guard \( h \) by \( t_h^c \). Then a description of \( P \) is given by \( t \cap \mathcal{G}' \).
6 Wireless Localization within Orthogonal Polyhedra

They consulted the computer. It said: “I regret that I have been temporarily closed to all communication. Meanwhile, here is some light music.” They turned off the light music.

D. Adams, Hitchhiker’s Guide to the Galaxy

In this chapter we turn our attention back to the *wireless localization problem* as treated in Chapter 3. However, we consider the analogous problem in 3D: given a polyhedron $P \subset \mathbb{R}^3$, place guards—which now are polyhedral cones—that collectively describe $P$. Generalizing a known result for 2-dimensional orthogonal polygons, we show that for any given 3-regular orthogonal polyhedron $P \subset \mathbb{R}^3$ with $n$ vertices it suffices to put a *natural vertex guard* onto every other vertex. Furthermore, we show how to describe $P$ with $3n/8$ (general) vertex guards. Note that in this chapter $n$ does not denote the number of faces $|F(P)|$ but the number of vertices of $P$, $n = |V(P)|$. The basis of this chapter is a joint work with Michael Hoffmann [13].

6.1 Introduction

The definition of guards used in Chapter 3 directly carries over to the 3-dimensional case. A guard is a polyhedral cone, that is, an unbounded polyhedron with at most one vertex and a connected 1-skeleton.

In accordance with our definitions in 2D, we call a guard that is placed at a vertex of $P$ a *vertex guard*. A vertex guard on a vertex $v$ is called *natural* if its shape is given by the shape of $P$ at $v$. More precisely, if the intersection of an $\varepsilon$-ball around $v$ with $g$ equals the intersection with $P$. A guard placed anywhere on an edge $e$ of $P$ and the shape of which is given by the shape of $P$ at $e$ is called a *natural edge guard*. Note that natural edge guards are wedges the only edge of which is a line through $e$. Obviously, now there is a third class of natural guards: We call the closed halfspace defined by a face $f$ of
Chapter 6. Wireless Localization within Orthogonal Polyhedra

Let $P$ a natural face guard on $f$. Note that both edge and face guards are cones without apex, so their exact position is undefined. We can think of them to be place anywhere on their own edge (or anywhere on their bounding plane, respectively).

For the 3D case, Dobkin et al. [26] observed that putting a face guard onto every face of a polyhedron suffices. (Of course, they did not speak of guards but used different notions, see Section 2.4.) The naive way of deriving a formula in disjunctive normal form (see Section 2.3) results in a formula of size $O(n^4)$. In [25] it is shown that there is always a formula of size $O(n^2\alpha(n))$ and furthermore, they give a worst-case lower bound of $\Omega(n^2)$ on the size of a monotone description formula. ($\alpha(n)$ denotes the inverse Ackermann function.) To our knowledge, the 3-dimensional wireless localization problem has not been studied since then. In this work we focus on orthogonal polyhedra and prove that $n/2$ natural vertex guards suffice to guard a polyhedron with $n$ vertices. If we allow general vertex guards, we can improve the bound to $3n/8$. On the other end, an argument in the spirit of Theorem 2.6 shows that there are orthogonal polyhedra that cannot be guarded by fewer than $n/4$ guards.

### 6.2 Notation and Basic Observations

Recall that an orthogonal polyhedron $P$ is a polyhedron where all faces are orthogonal to one of the coordinate axes (see Section 1.4). Faces orthogonal to the $x$-axis ($y$-axis, $z$-axis) are called $x$-faces ($y$-faces, $z$-faces, respectively). Consequently, all edges of $P$ are parallel to one of the coordinate axes and are called $x$-edges, $y$-edges and $z$-edges accordingly. Think of the $x$-axis as being oriented from left to right, the $y$-axis front to back, and the $z$-axis bottom up. We define its vertex set $V(P)$, its edge set $E(P)$ and its set of faces $F(P)$ in the usual way. Let $n = |V(P)|$ be the number of vertices.

Furthermore, we restrict our attention to bounded orthogonal polyhedra with the additional property that exactly three edges meet at every vertex. In other words, the graph of $P$ has to be 3-regular. Eppstein and Mumford [30] use a similar definition and additionally require the polyhedral surface bounding a polyhedron to have the topology of a sphere. They call this class of polyhedra simple orthogonal polyhedra. We do not use this notation, as in this work, we do not need any topological conditions and allow the polyhedra to form handles (that is, their genus might be greater than 0) and to contain cavities (that is, their surface might be disconnected). Therefore, we use the term 3-regular instead. From now on, we only consider 3-regular polyhedra. By the graph $G(P)$ of a polyhedron $P$ we mean the abstract graph defined by the 1-skeleton.
of $P$: The vertex set of $G(P)$ are the vertices of $P$ and two vertices $v$ and $w$ are adjacent in $G(P)$ if there is a polyhedral edge $e \in E(P)$ joining $v$ and $w$. The graph of a 3-regular orthogonal polyhedron (beside being 3-regular by definition) is a bipartite graph (cf. 30, where this is observed for simple polyhedra). We define the type of a vertex $v$ as follows. Assuming $v$ to be the origin, the type of $v$ is the set of octants $P$ locally occupies around $v$. We call $v$ convex, if only one octant is inside $P$. There are eight different possible types of convex vertices. We call $v$ reflex if all but one octant around $v$ are in $P$. There are eight possible types of reflex vertices. Furthermore, there are vertex types where exactly three octants are occupied, so two of the adjacent edges are convex and one is reflex. We call such a vertex semiconvex. There are 24 different semiconvex types. Finally, there are vertex types where all but three octants are occupied, denoted as semireflex. In Figure 6.2 we see two convex vertices of different type in the top row and two semiconvex vertices in the middle row and two semireflex vertices in the bottom row. See Genc 35, p. 38, for a classification of possible vertex types of general orthogonal polyhedra.

We denote each of the eight octants by a string in $\{+, -\}^3$. For example, the octant $-++$ is the set of points with negative $x$-coordinates, positive $y$-coordinates and negative $z$-coordinates. By definition, guards are (unbounded) polyhedral regions. In this context we restrict ourselves to 3-regular orthogonal guards. So from now on, a guard is an unbounded orthogonal cone with at most one $x$-edge, at most one $y$-edge and at most one $z$-edge. Consequently, a guard has at most one $x$-face, at most one $y$-face and at most one $z$-face.

The type of an edge $e \in E(P)$, is given by its direction (parallel to the $x$-axis or $y$-axis or $z$-axis) and by which quadrants around $e$ are occupied by $P$ in the plane orthogonal to $e$. Either one quadrant is occupied, in which case we call $e$ a convex edge or three quadrants are occupied, in which case we call $e$ a reflex edge. For example, we say an edge $e$ is a convex $z$-$(++)$ edge if $e$ is vertical and locally around $e$, the points in $P$ are the points with both higher $x$- and $y$-coordinate. Or we say $e$ is a reflex $x$-$(+-)$ edge if $e$ is parallel to the $x$-axis and $P$ occupies all but one quadrant around $e$, namely it leaves out the quadrant that lies behind $e$ (higher $y$-coordinates) and below $e$ (lower $z$-coordinates). There are 4 convex and 4 reflex edge types in any of the three directions, so totally, there are 24 different edge types. For each type, we fix a direction of the edge: We define the convex $x$-$(++)$ edges to be directed in negative $x$-direction (to the left). Similarly, we define the reflex $x$-$(++)$ edges to be oriented in positive $x$-direction (to the right). If we rotate $P$ around the $x$-axis, an $x$-$(++)$ edge either becomes a $x$-$(+-)$ or a $x$-$(--)$ edge. So rotating several times around all possible axes, each time by $\pi/2$, an edge can change from any type to any other type. We define the directions of all types in such
a way that rotating by $\pi/2$ around any of the three coordinate axes flips the orientation of an edge.

**Observation 6.1.** The edges of an orthogonal polyhedron can be oriented according to their type such that rotating by $\pi/2$ around any coordinate axis reverses the orientation.

See Figure 6.1 for a possible orientation of all edge types. There are exactly two ways to orient the convex types such that the observations holds and, independently, exactly two ways to orient the reflex types such that the observation holds. From now on, we think of every edge to be oriented as shown.

**Observation 6.2.** At a vertex $v \in V(P)$ either all adjacent edges are pointing into $v$ or all adjacent edges are pointing out of $v$.

First consider just one convex vertex type. After observing the property for this type, it follows for all other convex vertex types directly. Repeatedly rotating the vertex by $\pi/2$ around any coordinate axis, we can go from one convex type to any other convex type. With each single rotation, the orientation of all three adjacent edges flip. So if the edges were pointing towards the vertex before, they are all pointing away after the rotation and vice versa, see Figure 6.2.

Hence it suffices to consider one convex, one reflex, one semiconvex, and one semireflex type and the observation follows for the rest.

With this observation we have reproved that the graph of a 3-regular orthogonal polyhedron is bipartite, as observed for simple orthogonal polyhedra by Eppstein and Mumford [30]. Their proof is somewhat easier, as it is a direct consequence of the fact that the graph is planar and the numbers of edges of every face is even.

**Corollary 6.3.** The graph of a 3-regular orthogonal polyhedron is bipartite.

**Theorem 6.4.** For any integer $k \geq 2$, there are 3-regular orthogonal polyhedra with $4k$ vertices that cannot be guarded by fewer than $k$ guards.

**Proof.** Take an arbitrary 2-dimensional orthogonal polygon $Q$ with $2k$ pairwise non-collinear edges. Let $P$ be a (right) prism with base $Q$. $P$ has $2k + 2$ faces,
$k$ of which are $x$-faces. Consider a guarding $G$ of $P$. Each face $f$ of $P$ has to be covered by at least one guard. No two $x$-faces are coplanar, so any guard can cover at most one $x$-face. Therefore, there are at least $k$ guards in $G$. \hfill \Box$

### 6.3 The Proof in 2D

In 2D, putting a natural vertex guard onto every other vertex of an orthogonal polygon gives a valid guarding, see [29], Theorem 9, where this is proved for
Figure 6.2: How the orientations of the edges adjacent to a convex and a semiconvex vertex flip when the polyhedron gets rotated by \( \pi/2 \) around a coordinate axis.

simple polygons. Because guarding a polygon and guarding its complement are equivalent problems (Observation 3.1), the same holds for polygons with holes.

**Theorem 6.5.** [29] An orthogonal polygon with \( n \) vertices (possibly containing holes) can be guarded by \( n/2 \) guards putting a natural vertex guard onto every other vertex.

As a warm-up, we reprove this result using a different proof, which anticipates the basic idea of the proof in 3D in Section 6.4.

As in the 3D-case, we start by orienting the edges of \( P \). In the plane, there are only four different edge types: a horizontal edge can have the interior of \( P \) above or below and a vertical edge can have the interior of \( P \) to its left or to its right. We define all horizontal edges with the interior of \( P \) below to be pointing to the right, the horizontal edges with \( P \) above to be pointing to the left. The vertical edges having \( P \) on the right are defined to point downwards, those with \( P \) on the left upwards. We observe that at each vertex \( v \) of \( P \) either both adjacent edges are directed towards \( v \), or both are directed away from \( v \).
Put a natural vertex guard onto every vertex of $P$ that has incoming adjacent edges. Now we show that for every pair of generic points $(p, q)$ with $p \in P$ and $q \notin P$, there is at least one guard $g$ that distinguishes $p$ and $q$, that is, $p \in g$ and $q \notin g$. By Observation 2.4 this shows that we have a valid guarding.

Let $(p, q)$ be an inside/outside pair and let $Q$ be the axis-parallel rectangle spanned by $p$ and $q$ (possibly degenerated to a line segment). First consider the case that $p$ lies to the lower left of $q$: $p_x \leq q_x$ and $p_y \leq q_y$. See Figure 6.3 on the left. Consider the point $r \in P \cap Q$ that maximizes $r_x + r_y$ (the closest point to $q$ in the Manhattan metric). If $r$ is in the interior of $Q$, it must be a convex vertex of $P$ with the interior of $P$ in the lower left quadrant. Therefore, there is a guard on $r$, and this guard distinguishes $p$ and $q$. If $r$ lies on the boundary of $Q$, it either must be on the top edge of $Q$ or on the right edge of $Q$. If it is on the top edge, $r$ lies on a vertical edge $e \in E(P)$ which has the interior of $P$ to the left. Therefore, $e$ is oriented upwards and following the edge upwards we find a vertex of $P$ with a guard on it. The key observation is, that no matter how this vertex looks (it either is a convex vertex as before or a reflex vertex with the exterior of $P$ to the lower right), the guard on it distinguishes $p$ and $q$. Similarly, if $r$ lies on the right edge of $Q$, it must lie on a horizontal edge of $P$ pointing to the right, where we find a guard that distinguishes $p$ and $q$.

If $p$ lies to the top right of $q$, the proof works in the same way: Pick the point $r \in P \cap Q$ that is closest to $q$ in the Manhattan metric and either observe it is a vertex of $P$ and we find the guard we need right there, or it is on an edge of $Q$ adjacent to $q$ and we have to follow the orthogonal $P$-edge $r$ lies on (this edge is always going to point away from $Q$) to find a guard who does the job.
In the remaining two cases where \( p_x \leq q_x \) and \( p_y \geq q_y \) \((p_x \geq q_x \) and \( p_y \leq q_y \), respectively), we can apply almost the same idea, see Figure 6.3 on the right. But instead of picking the point \( r \) closest to \( q \), now let \( r \) be the point closest to \( p \) in (the closure of) \( Q \setminus P \). Again, if \( r \) is in the interior of \( Q \), it must be a reflex vertex with a guard on it. If it is on the boundary of \( Q \), we can again follow the corresponding edge of \( P \) to find a guard.

### 6.4 Guarding with \( n/2 \) Natural Vertex Guards in 3D

**Theorem 6.6.** A 3-regular orthogonal polyhedron \( P \) with \( n \) vertices can be guarded with \( n/2 \) natural vertex guards.

**Proof.** Put a guard onto every vertex where all edges are pointing inwards. We show that for every inside/outside point pair \((p, q)\) there is a guard \( g \) that distinguishes \( p \) and \( q \) (see Observation 2.4). Denote the axis-parallel cube spanned by \( p = (p_x, p_y, p_z) \) and \( q = (q_x, q_y, q_z) \) by \( Q \). Edge orientations and hence guardings are symmetric under a rotation by an angle of \( \pi \) around a coordinate axis. Therefore we may suppose without loss of generality that either \( q_x \leq p_x \), \( q_y \leq p_y \) and \( q_z \leq p_z \), or \( p_x \leq q_x \), \( p_y \leq q_y \) and \( p_z \leq q_z \).

First consider the case that \( q_x \leq p_x \), \( q_y \leq p_y \) and \( q_z \leq p_z \). Look at \( Q \cap P \). Pick the point \( r = (r_x, r_y, r_z) \in P \cap Q \) which minimizes \( r_x + r_y + r_z \). (In other words, report the first point we hit sweeping the \( x + y + z = c \) plane from \( q \) towards \( p \).) The point \( r \) can arise in three different ways, as depicted in Figure 6.4. If \( r \in V(P) \), \( r \) is a convex vertex such that \( P \) occupies the \((+ + +)-octant\). Therefore, there is a guard on it which distinguishes \( p \) and \( q \). If \( r = e \cap f \) is the intersection of an edge \( e \in E(P) \) and a face \( f \) of \( Q \), \( e \) must be adjacent to \( q \) and \( e \) must be a convex edge orthogonal to it. Therefore, \( e \) is pointing outward of \( Q \) to a guard \( g \) distinguishing \( p \) and \( q \). If \( r \) is on an edge of \( Q \) and in the interior of a face \( f \) of \( P \), we observe that (looking at \( f \) as a 2-dimensional polygon) there must be a convex vertex in every quadrant of \( f \) defined by \( r \). So we find a convex vertex \( v \) of \( f \) in the quadrant opposite to the one containing \( f \cap Q \). No matter which type \( v \) has as a vertex of \( P \), it is going to distinguish \( p \) and \( q \).

In the case where \( p_x \leq q_x \), \( p_y \leq q_y \) and \( p_z \leq q_z \), we have to use a slightly different argument. Let \( A \) be the face of \( Q \) that is adjacent to \( p \) and orthogonal to the \( z \)-axis. Consider the 2-dimensional orthogonal polygon \( P' \) we get by intersecting \( P \) with the plane \( \overline{A} \) and picking the connected component of the intersection that contains \( p \). Let \( r \) be the point in \( P' \cap A \) that maximizes \( r_x + r_y \). As in the first case, there are three sub-cases to consider. If \( r = A \cap e \), \( e \in E(P) \), we observe that \( e \) is a convex \( z \)-edge of type \((- -)\), so it is oriented...
downwards. So we can follow \( e \) to its end point outside \( Q \) where we find a guard \( g \) that distinguishes \( p \) and \( q \), see Figure 6.5. If \( r \) is the intersection of an edge of \( A \) with a face \( f \) of \( P \), then \( r \) divides \( f \) into four quadrants, in each of which we find a convex vertex of \( f \) (thought of as \( 2D \)-polygon). In particular, there is a convex vertex \( v \) of \( f \) in the quadrant opposite to the one containing \( f \cap Q \) and there is a guard on \( v \) that distinguishes \( p \) and \( q \). Finally, if \( r \) is a vertex of \( A \), which means that \( A \) is completely contained in \( P \), then pick another face \( B \) of \( Q \) adjacent to \( p \), and repeat the argument. If all faces adjacent to \( p \) are completely inside \( P \), then look at the top face \( C \) of \( Q \) adjacent to \( q \) and observe that \( C \cap P \) must have at least one reflex vertex \( r \). This vertex \( r \) lies on a reflex \( z \)-edge of \( P \), which is pointing upward to a guard outside of \( Q \) that distinguishes \( p \) and \( q \), see the example to the right in Figure 6.5.  

\[ r \in V(P) \]
\[ r \in e \in E(P) \]
\[ r \in f \in F(P) \]
\[ v \text{ convex} \]
\[ v \text{ reflex} \]
\[ v \text{ semiconvex} \]
\[ v \text{ semireflex} \]
Figure 6.5: Case 2: Following $e$ we find a guard $g$, which has one of several possible types.

6.5 Improving the Bound to $3n/8$

Let $P$ be an arbitrary 3-regular orthogonal polyhedron with $n$ vertices. Every vertex is incident to one $x$-edge, one $y$-edge and one $z$-edge. Thus, there are exactly $n/2$ edges in each direction. Let $f(P)$ be the number of faces of $P$. A face of $P$ is bounded by at least 4 edges and each edge bounds exactly two faces. This implies $4f(P) \leq 2e(P) = 3n$. So in total, there are at most $3n/4$ faces, some of them orthogonal to the $x$-axis, some of them to the $y$-axis and the rest to the $z$-axis. We call them $x$-faces ($y$-faces or $z$-faces, respectively).

Let $E$ be horizontal plane, i.e., orthogonal to the $z$-axis. Let $Q := E \cap P$. $Q$ is a 2-dimensional orthogonal polygon and its vertices correspond to $z$-edges of $P$. Now go back to the 2-dimensional case. We have shown that the natural way to guard $Q$ is to put a natural vertex guard onto every other vertex. More specifically, use exactly the guards as described in Section 6.3. If there is a natural vertex guard on a vertex $v \in V(Q)$, we do the same thing in the 3-dimensional case, namely, we put a natural edge guard onto the $z$-edge $e$ that
corresponds to $v$ if and only if $e$ is oriented downwards.

So far, we have put more or less $n/4$ guards, but in general not exactly that many, because there can be more $z$-edges of the types that do get guards than $z$-edges with types that do not get guards. If we pick a random rotation around the $z$-axis (by a multiple of $\pi/2$), the probability that a fixed $z$-edge gets a guard is exactly $1/2$, so the expected number of guards is exactly $n/4$, which implies that there is rotation such that we do put at most $n/4$ by the above rule.

Such a guarding certainly covers all $x$– and $y$-faces, but we have not done anything about the $z$-faces yet. An easy solution would be to put a natural face guard $g_f$ onto every $z$-face $f$. This would yield a valid guarding, but the number of $z$-faces could be as large as $n/4$ (even after permuting the coordinate axes). So instead, we replace every natural edge guard $g_e$ on a $z$-edge $e$ pointing down to a vertex $v$ of $f$ by a vertex guard $g_v$: $g_v := g_e \cap g_f$ if $f$ has the interior of $P$ above and $g_v := g_e \cup g_f$ if $f$ has the interior of $P$ below. So—in some sense—we combine the natural edge guards and the natural face guards to vertex guards. (Note that these new guards are not necessarily natural vertex guards.) However, some $z$-faces may not be covered still. We call a $z$-face $f$ good, if the guards we put onto vertices of $f$ cover $f$ completely, and we call $f$ bad, otherwise. It turns out that if a $z$-face is good (hence the name), it is not necessary to waste an additional face guard there. If a $z$-face $f$ is bad, then we put a natural face guard $g_f$ onto $f$. So we have to make sure that this does not happen too often and that we will not put more than roughly $n/8$ face guards in this way. Let $G$ be the set of vertex guards as described above together with the face guards on bad $z$-faces.

**Lemma 6.7.** $G$ is a valid guarding of $P$.

**Proof.** We use a sweep argument. For simplicity, we assume that $P$ has no coplanar $z$-faces. Imagine sweeping a plane $E$ orthogonal to the $z$-axis upwards and look at the intersection polygon $Q = P \cap E$. Whenever $E$ is coplanar with a $z$-face $f$ of $P$ (called the event face), the intersection polygon $Q$ changes: the new intersection polygon $Q'$ is either bigger or smaller, $Q' = Q \cup f$ or $Q'$ is (the closure of) $Q \setminus f$. We call the first case an increasing event, the second case a decreasing event. The claim is that using guards encountered so far only, we are able to guard $P$ as far as we have seen it: At any moment the set $\tilde{G}$ of guards that lie below the sweep plane $E$, together with an imaginary face guard $g_E$ that is the closed halfspace below $E$, is a guarding of the part $\tilde{P} = P \cap g_E$ of $P$ below $E$. We prove this claim by induction on the number of event faces processed. At the beginning $\tilde{P}$ is empty. At some point, we hit the first $z$-face $f$ of $P$. The vertices of $f$ correspond to $z$-edges of $P$ starting at
and going upwards. According to our rule, there is a guard on every second vertex of \( f \), which we can think of as a 2-dimensional guarding for \( f \) extending to the region orthogonally above \( f \). If we take the 2-dimensional guarding we get for \( f \) and intersect it with \( g_E \), we get a guarding for \( \tilde{P} \), which remains valid until we hit the second event face. After the first event, which for sure was an increasing event, the second event can be both increasing or decreasing, and so on, until the last event, which is decreasing.

**Figure 6.6:** An increasing event.

**Increasing Event.** Let \( f \) be an increasing event face, that is, the interior of \( P \) lies above \( f \) (Figure 6.6). We claim that after \( E \) has passed \( f \) (but no other event face yet), we still have a valid guarding for \( \tilde{P} \). Consider an inside/outside pair \( p \in \tilde{P} \) and \( q \notin \tilde{P} \). We may suppose without loss of generality that both \( p \) and \( q \) lie in the closed halfspace below \( \overline{f} \):

If one of the points, say, \( p \) lies above \( \overline{f} \), then consider the orthogonal projection \( p' \) of \( p \) onto \( \overline{f} \) instead. As all guards in \( \tilde{G} \) are located in the closed halfspace below \( \overline{f} \), none of them distinguishes \( p \) and \( p' \).

If \( p \) and \( q \) are both below \( \overline{f} \), then by induction there is a guard that distinguishes them.

If both \( p \) and \( q \) lie in \( \overline{f} \), then we are in a 2-dimensional situation and find a guard that distinguishes them because our guarding contains a 2-dimensional guarding of \( Q' \). (The vertices of \( Q' \) correspond to \( z \)-edges. There is a guard on every other \( z \)-edge that—intersected with \( \overline{Q'} \)—is a natural 2D vertex guard of \( Q' \). See Theorem 6.5.) If \( p \in \overline{f} \) and \( q \) lies below, then either there is a face guard \( g_f \) that does the job or \( f \) is a good face. In the latter case, the new vertex
6.5. Improving the Bound to 3n/8

guards (i.e., those put when handling the event face $f$) collectively cover $f$. If $p \in f$, then one of these new guards distinguishes $p$ and $q$. Otherwise, a point $p'$ slightly below $p$ lies within $P$ as well. By induction there is some old guard (i.e., a guard put before handling the event face $f$) to distinguish $p'$ from $q$. As such a guard cannot distinguish between $p$ and $p'$, it also distinguishes $p$ and $q$. Symmetrically, if $q \in f$ and $p$ is below, consider a point $q'$ located slightly below $q$ and note that $q' \notin P$ because the event is increasing. By induction, there is an old guard that distinguishes $p'$ and $q'$ but cannot distinguish $q$ and $q'$. Hence this guard also distinguishes $p$ and $q$.

**Decreasing Event.** Consider a point pair $p \in \tilde{P}$ and $q \notin \tilde{P}$. As above, we may suppose without loss of generality that $p$ lies in the closed halfspace below $\tilde{f}$. However, if $q$ lies above the event face $f$—that is, the orthogonal projection $q'$ of $q$ onto $\tilde{f}$ lies in $f$—we cannot simply replace $q$ by $q'$, because $q' \in P$. But we know that some guard that was placed when handling the event face $f$ distinguishes $q$ and $q'$, and every guard placed when handling $f$ contains the closed halfspace below $\tilde{f}$. Therefore, if $p$ lies below $\tilde{f}$, then this guard distinguishes $p$ and $q$. If $p \in \tilde{f}$, then recall that we have a 2-dimensional guarding for $Q'$, which must contain a guard that can distinguish $p$ and $q'$. This guard classifies $q'$ as outside and so it does with $q$. Hence it distinguishes $p$ and $q$. We may thus assume that $q$ lies in the closed halfspace below $\tilde{f}$ as well. It follows inductively that there exists an old guard that distinguishes $p$ and $q$.

**Lemma 6.8.** Under a random rotation around the $z$-axis by a multiple of $\pi/2$ and independently, a reflection with respect to the plane $z = 0$ with probability $1/2$, a $z$-face with 4 or 6 vertices is good with probability at least $1/2$.

**Proof.** Each vertex $v$ of $f$ corresponds to a $z$-edge $e_v$ of $P$. $e_v$ either starts at $v$ going upwards or it ends at $v$. We call $v$ a starting vertex or an ending vertex, respectively. If $v$ is a starting vertex and $e_v$ is pointing to $v$, there is a guard $g_v$ on $v$. Else, if $e_v$ is oriented upwards or $v$ is an ending vertex, there is no guard on $v$. Under a reflection in the $xy$-plane, starting vertices turn into ending vertices and vice versa, see Figure 6.7. Consider $f$ as a 2D-polygon. If it has 6 vertices, exactly 5 are convex and one is reflex. One of the convex 2D-vertex-types appears twice, the other three appear exactly once and are referred to as unique, therefore. If $f$ has 4 vertices, they are all unique. So in any case we have at least three unique (convex) vertices. Moreover, these vertices appear consecutively along the boundary of $f$, which implies that for at least one of them the incident $z$-edge is directed towards the vertex. If there is a guard on some unique vertex of $f$, then $f$ is good because this guard covers
Chapter 6. Wireless Localization within Orthogonal Polyhedra

page 126

Let $f$ completely. (Note that $f$ may be bad if there is a guard at some non-unique vertex only, see Figure 6.7 (B).)

If two adjacent unique vertices of $f$ are starting, then—by the remark above—at least one of them has a guard. When reflecting $P$ at the plane $z = 0$, all ending vertices turn into starting vertices and vice versa. Hence, if two adjacent unique vertices of $f$ are ending, then at least one of them has a guard after this reflection and so $f$ is good with probability at least 1/2. It remains to consider the case that the three unique vertices of $f$ follow the pattern starting–ending–starting or ending–starting–ending and neither of the starting vertices, with and without reflection, has a guard (Figure 6.7 (C)). When rotating around the $z$-axis by $\pi/2$ or $3\pi/2$, starting vertices remain starting and ending vertices remain ending. But edge orientations flip and so a starting vertex without guard turns into a starting vertex with guard. As a result, both the original face and the reflected variant turn good after such a rotation. So again with probability at least 1/2, the face $f$ appears as a good face.

Theorem 6.9. Let $P$ be a 3-regular orthogonal polyhedron with $n$ vertices. Then $P$ can be guarded with $3n/8$ guards.

Proof. Guard $P$ as described at the beginning of the section. Whenever we have to put a face guard $g_f$, we charge it to the edges of $f$. The edges of a $z$-face $f$ are also edges of $P$ and each $x$- or $y$-edge appears exactly once as an edge of a $z$-face. So the edges of a $z$-face $f$ of degree $d \geq 8$ each get charged at most $1/d \leq 1/8$. The edges of a face $f$ with degree $d = 4$ or $d = 6$ get charged $1/d \leq 1/4$ if we have to put a face guard $g_f$. If we pick a random rotation around the $z$-axis and independently decide to reflect $P$ with respect to the plane $z = 0$ with probability 1/2, we have shown that $f$ ends up as a bad face with probability at most 1/2. So the $x$- or $y$-edges gets charged at most 1/4 with probability at most 1/2 and 0 otherwise. So the expected charge of an $x$- or $y$-edge is at most 1/8. The vertex guards get charged to their corresponding $z$-edge. Under a random rotation, a $z$-edge gets charged 1 with probability 1/2 and 0 otherwise. Therefore, the expected total charge is going to be at most $\frac{1}{8}n + \frac{1}{2}n/2 = 3n/8$, so there is a rotation (possibly combined with a reflection) such that at most $3n/8$ guards are used. \qed
Figure 6.7: (A) a bad face that turns good after reflection; (B) a face that is bad even though there is a vertex \( v \) with a guard \( g_v \) on it: \( g_v \) does not cover the face completely; (C) a face that stays bad after reflecting, but both itself and the reflected version turn good after rotating around the \( z \)-axis by \( \pi/2 \).
7 Consistent Digital Line Segments

The Dude: I mean, it’s not just, it might not be just such a simple... uh, you know?
The Big Lebowski: What in God’s holy name are you blathering about?
The Dude: I’ll tell you what I’m blathering about...I’ve got information man! New shit has come to light!

The Dude, The Big Lebowski

In this chapter we study consistent digital line segments in the plane and in 3D. First it is shown how a system of consistent digital lines segments (CDS) in the plane can be derived from any total order on the integers. Then we use a special total order to construct consistent digital line segments that are close to the Euclidean segments, thus rediscovering an old result by Michael Luby [52]. We characterize the CDSes that are derived from total orders and show that there are also “exotic” CDSes that cannot be derived from a total order. In the last section of this chapter we define consistent digital line segments in higher dimensions and conclude with some partial results in 3D. This joint work with Dömötör Pálvölgyi and Miloš Stojaković [17] [18] originates from a problem posed by Jiří Matoušek at the 7th Gremo Workshop on Open Problems - GWOP 2009.

7.1 Introduction

One of the most fundamental challenges in digital geometry is to define a “good” digital representation of a geometric object. Of course, the meaning of the word “good” here heavily depends on particular conditions we may impose. Looking at the problem of digitalization in the plane, the goal is to find a set of points on the integer grid $\mathbb{Z}^2$ that approximates well a given object. The topology of the grid $\mathbb{Z}^2$ is commonly defined by the graph whose vertices are all the points of the grid, and each point is connected by an edge to each of the four points that are either horizontally or vertically adjacent to it.
Knowing that a straight line segment is one of the most basic geometric objects and a building block for many other objects, defining its digitalization in a satisfying manner is vital. Hence, it is no wonder that this has been an important scientific topic in the last few decades, see [47] for a survey and [38], [37], and [68] for related work, dealing with the problem of representing objects in digital geometry without causing topological and combinatorial inconsistencies.

For any pair of points \( p \) and \( q \) in the grid \( \mathbb{Z}^2 \) we want to define the digital line segment \( S(p,q) \) connecting them, that is, \( \{p, q\} \subseteq S(p,q) \subseteq \mathbb{Z}^2 \). Chun et al. [19] propose the following four axioms that arise naturally from properties of line segments in Euclidean geometry.

(S1) **Grid path property:** For all \( p, q \in \mathbb{Z}^2 \), \( S(p,q) \) is the vertex set of a path from \( p \) to \( q \) in the grid graph.

(S2) **Symmetry property:** For all \( p, q \in \mathbb{Z}^2 \), we have \( S(p,q) = S(q,p) \).

(S3) **Subsegment property:** For all \( p, q \in \mathbb{Z}^2 \) and every \( r \in S(p,q) \), we have \( S(p,r) \subseteq S(p,q) \).

(S4) **Prolongation property:** For all \( p, q \in \mathbb{Z}^2 \), there exists \( r \in \mathbb{Z}^2 \), such that \( r \notin S(p,q) \) and \( S(p,q) \subseteq S(p,r) \).

Note that (S3) is not satisfied by the usual way a computer visualizes a line segment. A natural definition of the digital straight segment between \( p = (p_x, p_y) \) and \( q = (q_x, q_y) \), where \( p_x \leq q_x \) and \( 0 \leq q_y - p_y < q_x - p_x \) is

\[
\left\{ \left( x, \frac{(x - p_x)(q_y - p_y)}{q_x - p_x} + p_y + 0.5 \right) \mid p_x \leq x \leq q_x \right\},
\]

see [47].

This also does not satisfy (S1), but it could be easily fixed by a slight modification of the definition. Still, it also does not satisfy (S3), for example, for \( p = (0,0), r = (1,0), q = (4,1) \), the digital subsegment from \( r \) to \( q \) is not contained in the digital segment from \( p \) to \( q \), see Figure 7.1.

![Figure 7.1: Axiom (S3) not satisfied for the usual way a computer visualizes a segment: The subsegment from \( r \) to \( q \) is not contained in the segment from \( p \) to \( q \).](image-url)
The set of axioms (S1)–(S4) seems rather natural, but there are still some fairly exotic examples of digital segment systems that satisfy all four of them. For example, let us fix a double spiral $D$ centered at an arbitrary point of $\mathbb{Z}^2$, traversing all the points of $\mathbb{Z}^2$. As it is a spanning path of the grid graph, we can set $S(p, q)$ to be the path between $p$ and $q$ on $D$, for every $p, q \in \mathbb{Z}^2$. It is easy to verify that this system satisfies axioms (S1)–(S4).

Another condition was introduced in [19] to enforce the monotonicity of the segments, ruling out pathological examples like the one above.

(S5) **Monotonicity property:** If both $p, q \in \mathbb{Z}^2$ lie on a line that is either horizontal or vertical, then the whole segment $S(p, q)$ belongs to this line.

We have phrased the monotonicity axiom differently, but still, the system of axioms (S1)–(S5) remains equivalent to the one given in [19]. The monotonicity axiom in [1] is given in the following form: “(R5) For any $r \in S(pq)$, $|\overline{pr}| \leq |\overline{pq}|$, where $|\overline{ab}|$ is the length of the Euclidean segment from $a$ to $b$”. Now (S5) follows immediately from (S1), (S3) and (R5), whereas (S5) together with (S1) and (S3) readily implies (R5).

We call a system of digital line segments that satisfies the system of axioms (S1)–(S5) a **consistent digital line segments system (CDS)**. It is straightforward to verify that every CDS also satisfies the following three conditions.

(C1) If the slope of the Euclidean line going through $p$ and $q$ is non-negative, then the slope of the Euclidean line going through any two points of $S(p, q)$ is non-negative. The same holds for non-positive slopes.

(C2) For all $p, q \in \mathbb{Z}^2$, the grid-parallel box spanned by points $p$ and $q$ contains $S(p, q)$.

(C3) If the intersection of two digital segments contains two points $p, q \in \mathbb{Z}^2$, then their intersection also contains the whole digital segment $S(p, q)$.

We give a simple example of a CDS, where the segments follow the boundary of the grid-parallel box spanned by the endpoints. Let $p, q \in \mathbb{Z}^2$ be two points with coordinates $p = (p_x, p_y)$ and $q = (q_x, q_y)$. If $p_y \leq q_y$, we define $S(p, q) = S(q, p) = \{(x, p_y) \mid \min\{p_x, q_x\} \leq x \leq \max\{p_x, q_x\}\} \cup \{(q_x, y) \mid p_y \leq y \leq q_y\}$. If $p_y > q_y$, we swap the points $p$ and $q$, and define the segment as in the previous case.

Observe that this system of digital line segments fulfills all axioms, hence it is a CDS, which we call the **bounding box example**. (In [52] this CDS is introduced under the name *Square geometry.*) But the digital segments in this system visually do not resemble well the Euclidean segments.
One of the standard ways to measure how close a digital segment is to a Euclidean segment is to use the Hausdorff distance. Recall that for $v, w \in \mathbb{R}^2$ and $A \subseteq \mathbb{R}^2$, $d(v, w)$ and $d(v, A)$ denote the usual Euclidean distances between two points, and between a point and a set. The Hausdorff distance between two sets $A$ and $B$ is denoted by $H(A, B)$, see Section 1.4.

The main question raised in [19] is whether it is possible to define a CDS such that a Euclidean segment and its digitalization have a reasonably small Hausdorff distance. More precisely, the goal is to find a CDS satisfying the following condition.

(H) Small Hausdorff distance property: For every $p, q \in \mathbb{Z}^2$, we have that $H(pq, S(p, q)) = O(\log d(p, q))$.

Note that in the bounding box example above, the Hausdorff distance between a Euclidean segment of length $d$ and its digitalization can be as large as $d/2$.

While this question was not resolved in [19], a clever construction of a system of digital rays emanating from the origin of $\mathbb{Z}^2$ that satisfy (S1)–(S5) and (H) was presented. Furthermore, it was shown that already for rays emanating from the origin, the log-bound imposed in condition (H) is the best bound we can hope for, directly implying the following theorem.

**Theorem 7.1.** (Chun et al. [19], Luby [52]) There exists a constant $c > 0$, such that for any CDS and any $d > 0$, there exist $p, q \in \mathbb{Z}^2$ with $d(p, q) > d$, such that $H(pq, S(p, q)) > c \log d(p, q)$.

Apparently, Chun et al. were not aware of the results of Michael Luby [52] who considered the same problem more than twenty years ago and neither were we when publishing [17]. Luby considers grid geometries, which in our terminology are systems of digital line segments satisfying (S1), (S2), and (S5). Then he introduces a further constraint called smoothness defined via the following notion of distance between digital line segments: The antidiagonal distance $\text{dist}(S_1, S_2, r)$ between two digital line segments $S_1$ and $S_2$ at a point $r \in S_1$ is the distance from $r$ to $S_2$ measured along the antidiagonal through $r$. (An antidiagonal is any line that is parallel to the line defined by the equation $y = -x$.) If the antidiagonal through $r$ does not intersect $S_2$, then $\text{dist}(S_1, S_2, r)$ is undefined. The points $r \in S_1$ for which $\text{dist}(S_1, S_2, r)$ is defined is a (possibly empty) subsegment of $S_1$. A grid geometry is smooth if for each pair $S_1$ and $S_2$ of digital line segments, $\text{d}(S_1, S_2, r)$ is either monotonically increasing or monotonically decreasing varying $r$ along its domain of definition. It is shown in [52] that a smooth grid geometry also satisfies (S3) and (S4), hence a smooth grid geometry is a CDS.
We introduce an approach for the construction of a CDS, which turns out to be related to the approach in [52] but stresses the concept of total orders: For any total order \( \prec \) on \( \mathbb{Z} \), we show how to derive a CDS from \( \prec \). (By total order we always mean a strict total order.) This process is described in Section 7.2. As a consequence, in Section 7.3, we manage to define a CDS that satisfies (H), deriving it from a specially chosen order on \( \mathbb{Z} \), thus reproving the following main result:

**Theorem 7.2.** (Luby [52]) There is a CDS that satisfies condition (H).

Note that Theorem 7.1 ensures that such a CDS is optimal up to a constant factor in terms of the Hausdorff distance from the Euclidean segments. In Section 7.4 we make a step toward a characterization of CDSes, demonstrating their natural connection to total orders on \( \mathbb{Z} \). It turns out that CDSes derived from total orders are exactly the same as the smooth grid geometries as introduces by Luby.

### 7.2 Digital Line Segments Derived from a Total Order on \( \mathbb{Z} \)

Let \( \prec \) be a total order on \( \mathbb{Z} \). We are going to define a CDS \( S_{\prec} \). Let \( p, q \in \mathbb{Z}^2 \), \( p = (p_x, p_y) \) and \( q = (q_x, q_y) \). If \( p_x > q_x \), we swap \( p \) and \( q \). Hence, from now on we may assume that \( p_x \leq q_x \).

If \( p_y \leq q_y \), then \( S_{\prec}(p, q) \) is defined as follows. We start at the point \( p = (p_x, p_y) \) and we repeatedly go either up or to the right, visiting points from \( \mathbb{Z}^2 \), until we reach \( q \). Note that in each step the sum of the coordinates \( x + y \) increases by 1. In total we have to make \( q_x + q_y - p_x - p_y \) steps and in exactly \( q_y - p_y \) of them we have to go up. The decision whether to go up or to the right is made as follows: if we are at a point \( (x, y) \) for which \( x + y \) is among the \( q_y - p_y \) greatest elements of the interval \( [p_x + p_y, q_x + q_y - 1] \) according to \( \prec \), we go up, otherwise we go to the right. We refer to this interval as the *segment interval*.

If \( p_y > q_y \), that is, if \( p \) is the top-left and \( q \) the bottom-right corner of the grid-parallel box spanned by \( p \) and \( q \), then we define \( S_{\prec}(p, q) \) as the mirror reflection of \( S_{\prec}((-q_x, q_y), (-p_x, p_y)) \) over the \( y \)-axis.

**Example.** Suppose \( p = (0, 0) \) and \( q = (2, 2) \). The segment interval consists of four numbers, 0, 1, 2, 3. If \( \prec \) is the natural order on \( \mathbb{Z} \), then the two greatest elements of the segment interval are 2 and 3. Since 0 + 0 is not one of these, at (0, 0) we go right, to (1, 0). At (1, 0) we again go to right, to (2, 0), from...
there to $(2, 1)$ (since $2 + 0$ is one of the greater elements) and finally to $(2, 2)$, see Figure 7.2. In fact, using the natural order on $\mathbb{Z}$ we get the CDS called the bounding box example mentioned in Section 7.1, where segments always follows the boundary of the box spanned by the endpoints.

![Figure 7.2: Segments obtained from the natural order on $\mathbb{Z}$: Going from $(0, 0)$ to $(2, 2)$.](image)

**Theorem 7.3.** $S_\prec$, defined as above, is a CDS.

**Proof.** We will verify that $S_\prec$ satisfies the axioms (S1)–(S5).

(S1) The condition (S1) follows directly from the definition of $S_\prec$.

(S2) Let $p, q \in \mathbb{Z}^2$. If the first coordinates of $p$ and $q$ are different, then condition (S2) follows directly. Otherwise, $p$ and $q$ belong to the same vertical line, and from the construction we see that both $S_\prec(p, q)$ and $S_\prec(q, p)$ consist of all the points on that line between $p$ and $q$.

(S3) For a contradiction, assume that there are points $p = (p_x, p_y), q = (q_x, q_y)$ and $r = (r_x, r_y)$, with $r \in S_\prec(p, q)$, such that $S_\prec(p, r) \not\subseteq S_\prec(p, q)$. Without loss of generality, we may assume that $\overline{pq}$ has a non-negative slope.

*Case 1.* $p_x \leq q_x$ and $p_y \leq q_y$. We also have $p_x \leq r_x$ and $p_y \leq r_y$, and going on each of the segments $S_\prec(p, r)$ and $S_\prec(p, q)$ point-by-point starting from $p$, we move either up or right. By assumption, these two segments separate at some point $(a, b)$ and then meet again, for the first time after this separation, at some other point $(c, d)$, see Figure 7.3. One of the segments goes up at $(a, b)$ and arrives at $(c, d)$ horizontally coming from the left, which implies that $a + b$ is among the greater numbers of the segment interval of this segment, while $c + d - 1$ is not, thus $c + d - 1 \prec a + b$. But the other segment goes horizontally at $(a, b)$
and enters \((c,d)\) vertically coming from below, which similarly implies \(a + b \prec c + d - 1\), a contradiction.

\(\begin{align*}
&\quad \quad \quad \quad \quad (c,d) \\
&\quad \quad \quad \quad \quad (a,b)
\end{align*}\)

**Figure 7.3:** Two paths splitting up at \((a,b)\) and meeting again at \((c,d)\).

**Case 2.** \(q_x \leq p_x\) and \(q_y \leq p_y\). We also have \(q_x \leq r_x\) and \(q_y \leq r_y\). By assumption, the two segments starting at \(q\) and \(r\), \(S_\prec(q,p)\) and \(S_\prec(r,p)\), separate at some point \((a,b)\) and then meet again, for the first time after this separation, at some other point \((c,d)\). Using the same argument as before, we get a contradiction.

(S4) To show that condition (S4) holds, consider the segment from \(p = (p_x, p_y)\) to \(q = (q_x, q_y)\). Without loss of generality, we can assume that \(p_x \leq q_x\) and \(p_y \leq q_y\). We distinguish two cases.

**Case 1.** If \(q_x + q_y\) is among the \(q_y - p_y + 1\) greatest numbers of \([p_x + p_y, q_x + q_y]\) according to \(\prec\), then we can prolong the segment going one step vertically up, that is, the segment \(S_\prec((p_x, p_y), (q_x, q_y + 1))\) contains the segment \(S_\prec((p_x, p_y), (q_x, q_y))\) as a subsegment.

**Case 2.** If, on the other hand, \(q_x + q_y\) is not among the \(q_y - p_y\) greatest numbers of \([p_x + p_y, q_x + q_y]\), we can prolong the segment horizontally to the right, that is, \(S_\prec((p_x, p_y), (q_x, q_y)) \subset S_\prec((p_x, p_y), (q_x + 1, q_y))\).

Note that if \(q_x + q_y\) is exactly the \((q_y - p_y + 1)\)th number in \([p_x + p_y, q_x + q_y]\), then the conclusions of both cases are true, and indeed the rays starting at \((p_x, p_y)\) split at \((q_x, q_y)\).

(S5) Condition (S5) follows directly from the definition of \(S_\prec\).

For segments with non-negative slope, in the definition of \(S_\prec(p,q)\), only the sum of the coordinates of the points plays a role, so if we translate \(p\) and \(q\)
along the antidiagonal, that is, by a vector \((t, -t)\), for any integer \(t\), the digital line segment will look the same.

**Observation 7.4.** Let \(t \in \mathbb{Z}\) be an integer, and let \(p\) and \(q\) be two points in \(\mathbb{Z}^2\) determining a line with non-negative slope. Then, \(S_\prec (p + (t, -t), q + (t, -t)) = S_\prec (p, q) + (t, -t) = \{(x + t, y - t) \in \mathbb{Z}^2 \mid (x, y) \in S_\prec (p, q)\}\).

This means that the segments in \(S_\prec\) are translation invariant in one specific direction: We say \(S_\prec\) is translation invariant along the antidiagonal. It might be desirable to have translation invariance in any direction. But if we insist on that and impose the condition that \(S_\prec (p + (a, b), q + (a, b)) = S_\prec (p, q) + (a, b)\) for any vector \((a, b) \in \mathbb{Z}^2\), the problem becomes trivial. It is easy to see that the bounding box example defined in Section 7.1 following the grid-parallel box, and its counterpart – the CDS we get by always following the box on the other side, are the only CDSes that are translation invariant in every direction.

### 7.3 CDS with Small Hausdorff Distance

For an integer \(k \neq 0\) and \(l \geq 2\), let \(|k|_l\) denote the number of times \(k\) is divisible by \(l\), that is,

\[|k|_l = \max \{m \mid l^m \mid k\}.\]

We define \(|0|_l := \infty\), that is, \(|0|_l\) is defined to be greater than \(|k|_l\) for any \(k \neq 0\). A total order on \(\mathbb{Z}\) is defined as follows: Let \(a \prec b\) if and only if there exists a non-negative integer \(i\) such that \(|a - i|_2 < |b - i|_2\), and for all \(j \in \{0, \ldots, i - 1\}\) we have \(|a - j|_2 = |b - j|_2\). In words, for two integers \(a\) and \(b\), we say that the one that contains a higher power of 2 is greater under \(\prec\). In case of a tie, we repeatedly subtract 1 from both \(a\) and \(b\), until at some point one of them contains a higher power of 2 than the other. Thus, for example,

\[-1 \prec -5 \prec 3 \prec -3 \prec 5 \prec 1 \prec -2 \prec 6 \prec -6 \prec 2 \prec -4 \prec 4 \prec 0.\]

For positive integers, this is equivalent to ordering them based on the least significant difference in the binary representation. Note that if we take the elements of an interval of the form \((0, 2^n)\) in \(\prec\)-decreasing order and divide them \(2^n\), then we get the first few elements of the binary Van der Corput sequence [75]: \(\frac{1}{2}, \frac{1}{4}, \frac{3}{8}, \frac{5}{8}, \frac{7}{8}, \frac{1}{16}, \frac{9}{16}, \frac{5}{16}, \frac{13}{16}, \frac{3}{16}, \frac{11}{16}, \frac{7}{16}, \frac{15}{16}, \ldots\)

We will prove that using this total order to define the system of digital line segments \(S_\prec\), as described in the previous section, we obtain a CDS that satisfies condition (H). In Figure 7.4 we give some examples of digital segments in this CDS, and Figure 7.5 shows the segments emanating from \((0, 0)\), as well as the segments from \((2, 3)\).
At first sight it may be surprising to observe that all the segments emanating from the origin in our construction coincide with the ones given in the construction of digital rays in [19] (after turning by $\pi/2$). However, it is not a coincidence, as the construction from [19], though different in nature, also relies on the same total order on the integers. Similarly, the rays segments emanating from the origin resemble the up doubling tree defined in [52].

Recall that for points $v, w \in \mathbb{R}^2$ and $A \subseteq \mathbb{R}^2$, $d(v, w) = |v - w|$ and $d(v, A) = \inf_{a \in A} d(v, a)$ denote the usual Euclidean distances between points and between a point and a set, respectively. For $p, q, r, s \in \mathbb{Z}^2$, by $pqrs$ we denote the union of the Euclidean line segments from $p$ to $q$, from $q$ to $r$, and from $r$ to $s$.

**Observation 7.5.** For any $p, q \in \mathbb{Z}^2$, $H(S_{<}(p, q), pq) = \max\{d(r, pq) \mid r \in S_{<}(p, q)\}$.

We proceed by proving three statements that we will ultimately use to prove Theorem 7.2. Moving from a digital segment to one of its subsegments, the
Hausdorff distance to the corresponding Euclidean segment may increase. But the following lemma shows that the distance can at most double.

**Lemma 7.6.** Let \( p, q \in \mathbb{Z}^2 \) and \( r, s \in S_\prec(p, q) \). Then

\[
H(rs, S_\prec(r, s)) \leq 2H(pq, S_\prec(p, q)).
\]

**Proof.** We know that \( d(r, pq) \leq H(pq, S_\prec(p, q)) =: h \) and \( d(s, pq) \leq h \), therefore \( H(prsq, pq) \leq h \). Hence, for all \( v \in pq \), \( d(v, prsq) \leq h \). Let \( t \in S_\prec(r, s) \subseteq S_\prec(p, q) \) and \( v \in pq \) be such that \( d(t, v) = d(t, pq) \leq h \). Using the triangle inequality we conclude \( d(t, prsq) \leq d(t, v) + d(v, prsq) \leq 2h \). Because of (C2), we have \( d(t, rs) = d(t, prsq) \leq 2h \), and therefore \( H(S_\prec(r, s), rs) \leq 2h \). \( \square \)

The following lemma is a purely geometrical statement that will be used in the proof of Theorem 7.2 to derive the exact bound.

**Lemma 7.7.** Let \( p, q, r, r' \in \mathbb{Z}^2 \), such that \( r_x - p_x = q_x - r'_x + \varepsilon \), \( r_y - p_y = q_y - r'_y - \varepsilon \), with \( \varepsilon \in \{0, 1, -1\} \), \( r'_x = r_x \) and \( r'_y = r_y + 1 \) (see Figure 7.6). Then

\[
H(pq, prr'q) \leq \sqrt{5}/2.
\]

**Proof.** Without loss of generality, let \( p = (0, 0) \), \( q_x \geq 0 \), and \( q_y \geq 0 \). We have

\[
H(pq, prr'q) = \max\{d(r, pq), d(r', pq)\}.
\]
By assumption, $2r_x - \varepsilon = q_x$ and $2r_y + \varepsilon + 1 = q_y$. So we get

$$d(r, \overline{pq}) = \left| \frac{q_y r_x - q_x r_y}{\sqrt{q_x^2 + q_y^2}} \right| = \left| \frac{q_y q_x + \varepsilon - q_x q_y - \varepsilon - 1}{\sqrt{q_x^2 + q_y^2}} \right| = \left| \frac{1}{2\sqrt{q_x^2 + q_y^2}} (q_y \varepsilon + q_x \varepsilon + q_x) \right|.$$  

Similarly,

$$d(r', \overline{pq}) = \left| \frac{1}{2\sqrt{q_x^2 + q_y^2}} (q_y \varepsilon + q_x \varepsilon - q_x) \right|.$$  

Setting $\xi := q_x/q_y$, we observe that no matter whether $\varepsilon = 0, \varepsilon = 1$, or $\varepsilon = -1$

$$H(\overline{pq}, prr'q) \leq \frac{1}{2\sqrt{(\xi q_y)^2 + q_y^2}} (q_y + 2\xi q_y) = \frac{\xi + 1/2}{\sqrt{\xi^2 + 1}} \leq \sqrt{5}/2,$$

as the function

$$f(\xi) = \frac{\xi + 1/2}{\sqrt{\xi^2 + 1}}$$

attains its global maximum at $\xi = 2$.  

The following lemma is a statement about the order $\prec$ and will be the key ingredient of the proof of Theorem 7.2.
Lemma 7.8. Let \( \{x \in \mathbb{Z} \mid A \leq x < B\} \) be an interval of integers with the following properties:

(i) Its number of elements is \( B - A = 2^{k+1} - 1 \) for some number \( k \).

(ii) \( |(A + B - 1)/2|_2 \geq k \) and \( |x|_2 < k \), for any other \( A \leq x < B \), \( x \neq (A + B - 1)/2 \).

Let \( x_1 < x_2 < \ldots < x_{2^k+1-1} \) be the elements of the interval sorted in increasing order according to \( \prec \). Then elements from the left half and elements from the right half of the interval alternate, that is, if \( x_i < (A + B - 1)/2 \) for some \( 1 \leq i < 2^{k+1} - 1 \), then \( x_{i+1} \geq (A + B - 1)/2 \).

Proof. Define \( M := (A + B - 1)/2 \). Assume for a contradiction that there is an \( 1 \leq i < 2^{k+1} - 1 \) such that both \( x_i < M \) and \( x_{i+1} < M \). First, we look at the case that \( |A - 1/2| < |M|_2 \). Then on one hand, \( x_i < x_i + 2^k \) because \( |x_i - j|_2 = |x_i + 2^k - j|_2 \) for all \( 0 \leq j \leq x_i - A \) and \( |x_i - j|_2 = |A - 1/2| < |M|_2 = |x_i + 2^k - j|_2 \) for \( j = x_i - (A - 1) \). (We use the simple observation that if \( |x|_2 < k \), then \( |x + 2^k|_2 = |x|_2 \).) But on the other hand, \( x_i + 2^k < x_{i+1} \), because there is a 0 \( \leq j_0 \leq x_i - A \) such that \( |x_i + 2^k - j_0|_2 = |x_i - j_0|_2 < |x_{i+1} - j_0|_2 \) and \( |x_i + 2^k - j|_2 = |x_i - j|_2 = |x_{i+1} - j|_2 \) for all \( 0 \leq j < j_0 \) (if the first \( j_0 \) such that \( |x_i - j_0|_2 \) and \( |x_{i+1} - j_0|_2 \) differ were \( j_0 = x_i - A + 1 \), then \( x_{i+1} < x_i \), because \( |A - 1/2| \geq k \)). So \( x_i < x_i + 2^k < x_{i+1} \), a contradiction.

In the case \( |A - 1/2| > |M|_2 \), we can argue similarly that \( x_i < x_{i+1} + 2^k < x_{i+1} \). Note that equality never occurs, as \( |A - 1/2| = |M|_2 \) implies \( |(A - 1 + M)/2|_2 \geq |M|_2 \geq k \), but \( |(A - 1 + M)/2|_2 < k \) by assumption.

We have shown that if \( x_i < M \), then \( x_{i+1} \geq M \). Similarly, \( x_i > M \) implies \( x_{i+1} \leq M \). This follows directly by the pigeon-hole principle: Every \( x_i < M \) has a unique successor \( x_{i+1} > M \) and exactly half of the elements of \( [A, B] \setminus \{M\} \) are smaller than \( M \), so there cannot be any \( x_j > M \) whose successor is also greater than \( M \).

Now we are ready to prove the main result of this chapter, Theorem 7.2, by showing that the CDS \( S_{\prec} \) satisfies the condition (H).

Proof. (of Theorem 7.2) Let \( p,q \in \mathbb{Z}^2 \). We may assume that \( p_x < q_x \) and \( p_y < q_y \). We are going to prove that \( H(pq, S_{\prec}(p,q)) \leq 2c \log(q_x + q_y - p_x - p_y) \) for \( c = \sqrt{5}/2 \) and the logarithm to the base 2. Let \( r \in S_{\prec}(p,q) \) be the point with the property that \( r_x + r_y \) is the greatest element of the segment interval, that is, \( r_x + r_y > s \) for all \( s \in [p_x + p_y, q_x + q_y] \), \( s \neq r_x + r_y \), see Figure 7.7 for an example. Now let \( s' \) be the second greatest element of the segment interval according to \( \prec \). Define \( k := |s'|_2 + 1 \).
We can extend the segment $S_{≺}(p, q)$ over both endpoints, moving both $p$ and $q$ such that $|p_x + p_y - 1|_2 \geq k$ and $|q_x + q_y|_2 \geq k$, that is, we extend the segment as far as we can, so that $k$, defined as above, remains unchanged. From Lemma 7.6 we get that by this extension we decreased the Hausdorff distance by at most a factor of 2. Now the segment interval contains exactly $2^{k+1} - 1$ elements and $r_x + r_y$ is the element in the very middle. We call such a segment normalized.

We are going to proceed by induction on $k$ to prove that for all normalized digital line segments $S_{≺}(p, q)$, we have $H(\overline{pq}, S_{≺}(p, q)) \leq ck$ with $c = \sqrt{5}/2$. This will prove the theorem, as $k + 1 = \log(q_x + q_y - p_x - p_y + 1)$, and then the distance of the unnormalized original segment (we started from) is at most $2ck = 2c(\log(q_x + q_y - p_x - p_y + 1) - 1) \leq 2c\log(q_x + q_y - p_x - p_y)$.

![Figure 7.7:](image)

In the base case $k = 1$, the segment interval consists of 3 numbers, so $S_{≺}(p, q)$ is a path of length 3 and by checking all possibilities we see that $H(\overline{pq}, S_{≺}(p, q)) < c$.

If $k > 1$, the idea is to split the segment at $r$ into two subsegments that are similar in some sense and apply induction. Let $r' = (r_x, r_y + 1)$ be the point that comes after $r$ in the segment $S_{≺}(p, q)$. (We know that we go up at $r$, because we go up at least once and $r_x + r_y$ is the greatest element of the segment interval). Consider the subsegments $S_{≺}(p, r)$ and $S_{≺}(r', q)$ and partition the segment interval accordingly. The key observation is that picking the elements of the interval according to $≺$ starting with the greatest, we first get $r$, and then alternately an element of the left and the right subsegment interval. This is shown in Lemma 7.8 setting $A = p_x + p_y$, $B = q_x + q_y$. Therefore, up to a difference of at most one, half of the $q_y - p_y - 1$ great-
est elements \((r_x + r_y, r_x + r_y)\) belong to \([p_x + p_y, r_x + r_y]\) and half of them to \([r_x + r_y + 1, q_x + q_y]\). This implies that \(p, q, r, r'\) meet the conditions of Lemma 7.7 leading to \(H(pq, prr'q) \leq c\). By the induction hypothesis we have \(H(pr, S_{<}(p, r)) \leq c(k - 1)\) and \(H(r'q, S_{<}(r', q)) \leq c(k - 1)\). So \(H(prr'q, S_{<}(p, q)) = \max\{H(pr, S_{<}(p, r)), H(r'q, S_{<}(r', q))\} \leq c(k - 1)\). Using the triangle inequality we conclude \(H(pq, S_{<}(p, q)) \leq ck\).

### 7.4 Characterization of CDSes

Now we approach the same problem from a different angle, taking arbitrary CDSes and trying to find some common patterns in their structure. Knowing that condition (C1) holds for all CDSes, we can analyze the segments with non-positive and non-negative slopes separately, as they are completely independent. More precisely, the union of any CDS on segments with non-positive slopes and another CDS on segments with non-negative slopes is automatically a CDS. Having this in mind, in this section we will proceed with the analysis of only one half of a CDS, namely of segments with non-negative slope.

We will show that, in a CDS, all the segments with non-negative slope emanating from a fixed point must be derived from a total order. However, as we will show later, these orders are not necessarily the same for different points.

**Theorem 7.9.** For any CDS and for any point \(p = (p_x, p_y) \in \mathbb{Z}^2\), there is a total order \(\prec_p\) that is uniquely defined on both \((-\infty, p_x + p_y - 1]\) and \([p_x + p_y, +\infty)\), such that the segments with non-negative slope emanating from \(p\) are derived from \(\prec_p\) (in the way described in Section 7.2).

**Proof.** We fix a CDS \(S\) and a point \(p\). The segments with non-negative slope with \(p = (p_x, p_y)\) as their upper-right point will induce an order on the integers smaller than \(p_x + p_y\), and the segments for which \(p\) is the lower-left endpoint will induce an order on the rest of the integers. In the following we will just look at the latter type of segments. First, we prove an auxiliary statement: Two positively sloped segments both starting at a point \(p\), either run in parallel or split (further) up, but they cannot come closer to each other again. Note that this is exactly the smoothness condition (see Section 7.1) in the special case of what Luby calls an origin geometry.

**Lemma 7.10.** It cannot happen that for two segments with non-negative slope having the same lower-left endpoint \(p\), one of them goes up at \((a, C - a)\) and the other goes right at \((b, C - b)\), for some \(C\) and \(a > b\).

**Proof.** (of Lemma 7.10) Let us assume the opposite, see Figure 7.8. Look at
the \( a - b + 1 \) segments between point \( p \) and each of the points on the line \( x + y = C \) between the points \((a, C-a)\) and \((b, C-b)\). It is possible to extend all of them through their upper-right endpoints, applying (S4). Note that each of the extended segments goes through a different point on the line \( x + y = C \), and hence, because of condition (C3), no two of them can go through the same point on the line \( x + y = C + 1 \). But, there are only \( a - b \) available points on the line \( x + y = C + 1 \) between the points \((a, C - a + 1)\) and \((b + 1, C - b)\), one fewer than the number of segments, a contradiction.

\[ p \quad x + y = C \]

\[ (a, C - a) \]
\[ (b, C - b) \]

**Figure 7.8:** Two segments having \( p \) as their lower-left endpoint, one of them going up at \((a, C - a)\) and the other going right at \((b, C - b)\), with \( a > b \).

Returning to the proof of the theorem, we define the relation \( \prec_p \) in the following way. Whenever there is a segment in \( S \) with non-negative slope starting at \( p \), going right at a point \((x, D - x)\) and going up at a point \((x', E - x')\), for some \( x \) and \( x' \), we set \( D \prec_p E \). This way we defined a relation on \([p_x + p_y, +\infty)\), which is obviously irreflexive. To show that \( \prec_p \) is a total order, it remains to prove that it is asymmetric, transitive and total.

To show asymmetry, assume for a contradiction that for some integers \( D \) and \( E \) we have both \( D \prec_p E \) and \( E \prec_p D \). That can happen only when there are two segments with non-negative slope having \( p \) as their lower-left endpoint, such that on the line \( x + y = D \) one of them goes up, the other right, and then on \( x + y = E \) they both go in different direction than at \( x + y = D \). But then the situation described in Lemma \[7.10\] must occur on one of the two lines, a contradiction.
Next, if $C \prec_p D$ and $D \prec_p E$, then we also have $C \prec_p E$ – we just take a segment starting from $p$ that goes right at $C$ and up at $D$, and (if necessary) extend it until it passes the line $x + y = E$. It must also go up at $E$, because of $D \prec_p E$ and the asymmetry of $\prec_p$. Hence, the relation $\prec_p$ is transitive.

It remains to prove the totality of $\prec_p$. That is, for any pair of integers $p_x + p_y \leq D < E$, either $D \prec_p E$ or $E \prec_p D$ holds. Consider a segment from $p$ to some point $q$ on the line $x + y = E$, such that this segment splits at $q$, that is, there are two extensions of the segment, one going up and another one going right. Such segment exists since in the upper-right quadrant of $p$, the line $x + y = E + 1$ contains one more point than the line $x + y = E$. If we look at all the segments between $p$ and the points on $x + y = E + 1$, the pigeonhole principle ensures that two of them will contain the same point $q$ on the line $x + y = E$. Now the segment $S(p, q)$ crosses the line $x + y = D$ at some point $q'$. Depending on whether it goes up or right at the point $q'$, either $E \prec_p D$ or $D \prec_p E$ holds.

To see that these orders can differ for different points, consider the following example of a CDS, which we call the waterline example because of the special role of the $x$-axis. To connect two points with a segment, we do the following. Above the $x$-axis we go first right, then up, below the $x$-axis we go first up, then right, and when we have to traverse the $x$-axis, we go straight up to it, then travel on it to the right, and finally continue up, see Figure 7.9. It is easy to check that this construction satisfies all five axioms.

Now, if we consider a point $p = (a, b)$ below the waterline, $b < 0$, the induced total order $\prec_p$ on $[a + b, +\infty)$ is $a \prec_p a + 1 \prec_p \ldots \prec_p (+\infty) \prec_p a - 1 \prec_p a - 2 \prec_p \ldots \prec_p a + b$, and the order on $(-\infty, a + b - 1]$ is $a + b - 1 \prec_p a + b - 2 \prec_p \ldots \prec_p -\infty$. If $p$ is above the waterline, $b \geq 0$, the induced total orders are $a - 1 \prec_p a - 2 \prec_p \ldots \prec_p (-\infty) \prec_p a \prec_p a + 1 \prec_p \ldots \prec_p a + b - 1$ and $a + b \prec_p a + b + 1 \prec_p \ldots \prec_p +\infty$. Obviously, there is no total order on $\mathbb{Z}$ compatible with these orders for all possible choices of $p$. For example, if we choose $p = (a, b) = (1, -1)$, we get $1 \prec_p 0$, but if we choose $p = (a, b) = (-1, 1)$, we get $0 \prec_p 1$.

The special role played by the $x$-axis in the waterline example can be fulfilled by any other monotone digital line with a positive slope; above the line go right, then up, below the line go up, then right, and whenever the line is hit, follow it until either the $x$- or the $y$-coordinate matches that of the final destination, see Figure 7.10. Again, it is straightforward to show that this way we obtain a CDS.

A way to see that such a CDS cannot be derived from a total order is to observe that now the antidiagonal translation of a digital line segment does not
always yield another digital line segment, that is, these examples do not satisfy the condition from Observation 7.4. Actually, we can prove that it suffices to add this condition to force a unifying total order on all integers.

**Theorem 7.11.** Let \( S \) be a restriction of a CDS to the set of all segments with non-negative slopes, such that for any \( t \in \mathbb{Z} \) and any \( p, q \in \mathbb{Z}^2 \),

\[
S(p + (t, -t), q + (t, -t)) = S(p, q) + (t, -t).
\]

Then there is a unique total order \( \prec \) on \( \mathbb{Z} \) such that \( S = S_\prec \).

**Proof.** Let \( p, q \in \mathbb{Z}^2 \). By Theorem 7.9 the segments starting at \( p \) define a unique total order \( \prec_p \) on \( [p_x + p_y, \infty) \) and similarly the segments starting at \( q \) define a unique total order \( \prec_q \) on \( [q_x + q_y, \infty) \). Let \( q_x + q_y \geq p_x + p_y \). We want to check whether these two orders agree on \( [q_x + q_y, \infty) \). Assume for contradiction there are integers \( q_x + q_y \leq A < B \) such that there is a segment \( S \) starting at \( p \) going to some point \( r \) implying \( B \prec_p A \) and another segment \( T \) starting at \( q \) implying \( A \prec_q B \), see Figure 7.11 for an example. We translate the segment \( S \) diagonally by a vector \((t, -t)\) until \( q \) lies on the translated segment \( S' \) from...
Figure 7.10: A more exotic example of a CDS with an arbitrary “special” line (bold and dotted).

$p'$ to $r'$. Then the subsegment from $q$ to $r'$ still goes up at level $A$ and to the right at level $B$, implying $B \prec_q A$, a contradiction.

Therefore we can define a unique total order $\prec$ as follows. To compare two integers $A$ and $B$ take any point $p$ with $p_x + p_y \leq A$ and define $A \prec B$ if and only if $A \prec_p B$. By the argument above, this definition is independent of the choice of $p$ and the arguments in the proof of Theorem 7.9 carry over to $\prec$, which shows it is a total order.

Observation 7.12. The CDSes that are translation invariant along the antidiagonal are exactly the smooth grid geometries in Luby’s terminology. This means that CDSes derived from total orders as explained in Section 7.2 and smooth grid geometries as defined in [52] are equivalent concepts.

Proof. Translation invariance along the antidiagonal together with Lemma 7.10 implies smoothness (see Section 7.1 for a definition). A smooth grid geometry is a CDS as shown in [52]. Additionally, smoothness implies that the antidiagonal distance between $S(a,b)$ and $S(a + (t,-t),b + (t,-t))$ is either monotonically increasing or decreasing. But apparently,

$$d(S(a,b), S(a + (t,-t),b + (t,-t)), a) = d(a,a + (t,-t)) = \sqrt{2}t,$$
and similarly,

\[ d(S(a, b), S(a + (t, -t), b + (t, -t)), b) = d(b, b + (t, -t)) = \sqrt{2}t. \]

So \( d(S(a, b), S(a + (t, -t), b + (t, -t)), r) = \sqrt{2}t, \) for all \( r \in S(a, b), \) which implies that \( S(a + (t, -t), b + (t, -t)) = S(a, b) + (t, -t). \)

### 7.5 Digital Lines

Our focus is on the digitalization of line segments, but the present setup can be conveniently extended to a definition of digital lines. We say that a digital line is the vertex set of a path infinite in both directions in the \( \mathbb{Z}^2 \) base graph, such that the digital line segment between any two points on the digital line belongs to the digital line.

In this section we restrict our attention to CDSes that are derived from a total order as described in Section 7.2. Furthermore, for simplicity, we are only going to consider lines with non-negative slope, meaning that the Euclidean segment between any two points of the line has non-negative slope (including zero and infinity).
Consider a digital line \( \ell \) derived from \( S_{\prec} \). We define \( A_\ell \subseteq \mathbb{Z} \) to be the set of numbers \( x + y \) for which \( \ell \) goes upward at \((x, y)\) and call it the slope set of \( \ell \). Note that the slope set \( A_\ell \) is an interval in \((\mathbb{Z}, \prec)\) which is unbounded in the increasing direction, i.e., if \( x \in A_\ell \), then for any \( y \succ x \) we have \( y \in A_\ell \). This implies that there is a natural total order on the set of possible slope sets given by inclusion.

The following observation follows directly from the definition of \( S_{\prec} \).

**Observation 7.13.** Every line \( \ell \) can be described by its slope set \( A_\ell \) and a point \( p \) it contains. Also, given a point \( p \), every \( \prec \)-interval \( A \) that is unbounded in the increasing direction is a valid slope set of a line through \( p \), that is, there is a line \( \ell \) such that \( p \in \ell \) and \( A_\ell = A \).

It follows directly from Theorem 7.3 that two lines having a point in common cannot split and then meet later. Similarly, two lines cannot “touch” without crossing. (If they did, by Observation 7.4 we could translate one of them diagonally to produce a contradiction with axiom (S3).)

**Observation 7.14.** If two different lines intersect, then they either cross (having a common segment), or they have a common halfline.

**Lemma 7.15.** Consider two different slope sets \( A \) and \( B \) with \( A \subset B \). Let \( I = B \setminus A \) be the difference of the slope sets. We distinguish three cases.

1) If \( I \) is finite, then there are lines \( l \) and \( s \) such that \( A_{l} = A \), \( A_{s} = B \) and \( l \) and \( s \) intersect in a lower-left halfline and there are lines \( l' \), \( s' \) with slope sets \( A_{l'} = A \), \( A_{s'} = B \) intersecting in an upper-right halfline.

2) If \( I \) is infinite and bounded in one direction in the natural order on \( \mathbb{Z} \), then we can find lines \( l \) and \( s \) with slope sets \( A \) and \( B \) intersecting in a lower-left or an upper-right halfline depending on whether \( I \) is lower- or upper-bounded with respect to the natural order.

3) If \( I \) is unbounded in both directions, then all lines \( l \) and \( s \) with slope sets \( A \) and \( B \) do intersect in a finite segment.

**Proof.** If \( I \) is finite, let \( c \) be its smallest element according to the natural order. Let \( p \) be a point with \( px + py = c \). Construct the line \( l \) with slope set \( A \) through \( p \) and the line \( s \) through \( p \) with slope set \( B \). To the lower left of \( p \), \( l \) and \( s \) coincide, because for all \( z \leq px + py \) (in the natural order), \( z \in A \) if and only if \( z \in B \). Similarly, picking a point \( q \) such that \( qx + qy \) equals the biggest element of \( I \) in the natural order, one can construct the line with slope set \( A \) and the line with slope set \( B \) through \( q \). These two lines are going to coincide to the upper right of \( q \).
If \( I \) is infinite and there is a \( k \in \mathbb{Z} \) such that for all \( i \in I, i < k \), then pick a point \( p \) with \( x_p + y_p = k \). The line \( l \) with slope set \( A \) through \( p \) and the line \( s \) with slope set \( B \) through \( P \) have the halfline starting at \( p \) in common, because for all \( z \in \mathbb{Z} \) with \( z \geq k \), \( z \) is in \( A \) if and only if \( z \) is in \( B \).

Finally, consider lines \( l \) and \( s \) with slope sets \( A \subset B \) such that \( I = B \setminus A \) is unbounded in both directions according to the natural order. For any point \( p \in l \), look at the corresponding point \( p' \in s \) on the same level, that is, with \( x_p + y_p = p'x + p'y \). Moving \( p \) on \( l \) toward the upper right, the difference \( p'y - p'y = p'x - p_x \) decreases monotonically. (Whenever \( x_p + y_p \in I \), \( p \) is moving to the right, but \( p' \) is moving up, so \( p'y - p'y \) decreases by one.) The same way, the difference increases from time to time if we move \( p \) on \( l \) into the other direction. Because \( I \) is unbounded both ways, there is no point after which the difference remains constant moving to the upper right, nor a point after which the difference remains constant moving \( p \) to the lower left. Therefore, \( l \) and \( s \) have to intersect in a common line segment.

We define two lines to be parallel if they do not cross, that is, according to Observation 7.14, two lines are parallel if they are disjoint or if they agree on a halfline. We distinguish two possible cases for the slope sets \( A_\ell \) of a line \( \ell \). If there is a \( c \in \mathbb{Z} \) such that \( A_\ell = [c, \infty)_\prec = \{ a \in \mathbb{Z} \mid a \succ c \} \cup \{ c \} \) or \( A_\ell = (c, \infty)_\prec = \{ a \in \mathbb{Z} \mid a \succ c \} \), that is, if its boundary can be described by the smallest element, either including or excluding this element, then we call the slope set \( A_\ell \) rational. In the special cases \( A_\ell = \mathbb{Z} \) and \( A_\ell = \emptyset \) we define \( A_\ell \) as rational, too. If no such \( c \) exists, or, in other words, if \( A_\ell \) does not have a smallest element and its complement \( \mathbb{Z} \setminus A_\ell \) does not have a greatest element (and they are not empty), then we call \( A_\ell \) irrational.

We proceed by analyzing the digital lines derived from the total order that we defined in Section 7.3, using the powers of 2. We denote this order by \( \prec^\ast \). With the help of the previous lemma, we can characterize which lines do have unique parallels and which do not.

**Theorem 7.16.** Let \( \ell \) be a line with respect to \( S_{\prec^\ast} \), and let \( p \in \mathbb{Z}^2 \) be a point with \( p \notin \ell \).

1) If the slope set \( A_\ell \) is irrational, then there is a unique line \( \ell' \) through \( p \) that is parallel to \( \ell \). Furthermore, \( A_{\ell'} = A_\ell \).

2) If \( A_\ell \) is rational, then \( \ell \) has exactly two parallels \( \ell' \) and \( \ell'' \) through \( p \), one of them with the same slope set as \( \ell \), the other with a slope set that differs by one element. Consequently \( \ell' \) and \( \ell'' \) either have a common halfline in one direction, and split at one point to run parallel at distance one in the other direction.
Proof. Note that \( \prec^* \) is a dense order on \( \mathbb{Z} \), that is, for any integers \( a \prec^* b \), we find a \( c \in \mathbb{Z} \) such that \( a \prec^* c \prec^* b \). Equivalently, this means that every \( \prec^* \)-interval \([a, b]\) is infinite. Let \( s \) be an arbitrary line through \( p \) parallel to \( \ell \). Consider the symmetric difference of the slope sets \( I = \mathcal{A}_\ell \triangle \mathcal{A}_s \). If \( I \) is empty, the slope sets are equal and \( s \) is the diagonal translate of \( \ell \) going through \( p \). If \( I \neq \emptyset \), by density of the order, \( I \) is either infinite or consists only of one element \( a \). If \( I \) is infinite, then it is unbounded according to the natural order in both directions. (Besides density, this is the only property of \( \prec^* \) that we use.) According to Lemma 7.15, \( s \) and \( \ell \) do intersect in a finite segment in this case, which is a contradiction because \( s \) is parallel to \( \ell \). Therefore, \( I \) consists of one element. If \( A_\ell \) is irrational, by adding one element to \( A_\ell \) or removing one element from \( A_\ell \) we do not get a \( \prec^* \)-interval. Hence, in this case, there is only one possible slope set for any line parallel to \( \ell \), namely, \( A_\ell \). If \( A_\ell \) is rational, then either \( A_\ell = \{ a \in \mathbb{Z} \mid c \prec^* a \} \cup \{ c \} \) or \( A_\ell = \{ a \in \mathbb{Z} \mid c \prec^* a \} \), for some \( c \in \mathbb{Z} \), so adding \( c \) or removing \( c \), respectively, yields the only different possible slope set for a parallel through \( p \). As the slope sets of these two parallels differ by only one element, they have either a common halfline to the left or a common halfline to the right.

\[ \square \]

7.6 CDS in Higher Dimensions

Are there CDSes in more than two dimensions? The definition of a CDS directly carries over to the higher dimensional spaces. Instead of \( \mathbb{Z}^2 \) we now consider \( \mathbb{Z}^d \), for a fixed integer \( d \geq 3 \), with the usual graph structure, that is, two points \( p \) and \( q \) are adjacent if they differ in exactly one coordinate by exactly one. For every pair \( p, q \in \mathbb{Z}^d \) we want to define a subset \( S(p, q) \subset \mathbb{Z}^d \) such that the following axioms hold:

- **(S1) Grid path property:** For all \( p, q \in \mathbb{Z}^d \), \( S(p, q) \) is the vertex set of a path from \( p \) to \( q \) in the grid graph \( \mathbb{Z}^d \).
- **(S2) Symmetry property:** For all \( p, q \in \mathbb{Z}^d \), we have \( S(p, q) = S(q, p) \).
- **(S3) Subsegment property:** For all \( p, q \in \mathbb{Z}^d \) and every \( r \in S(p, q) \), we have \( S(p, r) \subseteq S(p, q) \).
- **(S4) Prolongation property:** For all \( p, q \in \mathbb{Z}^d \), there exists \( r \in \mathbb{Z}^d \), such that \( r \notin S(p, q) \) and \( S(p, q) \subseteq S(p, r) \).
- **(S5) Monotonicity property:** If \( p, q \in \mathbb{Z}^d \) agree in a coordinate, then the whole segment \( S(p, q) \) agrees in this coordinate.

The axioms (S1)–(S4) are the same as in 2D, verbatim, and the monotonicity axiom (S5) gives the same as in 2D setting \( d = 2 \). A system of digital line
7.6. CDS in Higher Dimensions

segments that satisfy these axioms is called consistent or a CDS for short.

The bounding box example already introduced in 2D can be generalized to higher dimensions. Let \( p, q \in \mathbb{Z}^d \) and without loss of generality let \( p \) be lexicographically smaller than \( q \). If \( p, q \) only differ in one coordinate, then we define \( S(p, q) \) to be the straight axis-parallel segment connecting them. (Actually because of axiom (S5) we have no other choice.) If they differ in more than one coordinate, we start at \( p \) increasing the first coordinate they differ until this coordinate agrees with \( q \), and from there we define the path recursively. This defines again a valid CDS in higher dimensions, again called the bounding box example. (The segment between \( p \) and \( q \) follows the 1-skeleton of the axis-parallel cuboid spanned by \( p \) and \( q \).) This shows that there are CDSes in any dimension.

Again we are interested in higher dimensional CDSes satisfying

\[(H) \quad \text{Small Hausdorff distance property: For every } p, q \in \mathbb{Z}^2, \text{ we have that } H(\overline{pq}, S(p, q)) = O(\log d(p, q)).\]

The slope type of a pair of points \((p, q)\) is the vector \((\sigma_1, \ldots, \sigma_d) \in \{+, -\}^d\) with \( \sigma_i = + \) if \( p_i \leq q_i \), and \( \sigma_i = - \) if \( p_i > q_i \). The slope type of a digital segment \( S(p, q) \) is defined as the slope type of \((p, q)\). We say a segment has strictly positive slope if its slope type is \((+, \ldots, +)\), that is, the coordinates are monotonically increasing in each coordinate.

7.6.1 Strictly Positive Sloped Segments

It is not hard to see that we can again derive a consistent system from an arbitrary total order \( \prec \) on \( \mathbb{Z} \) if we consider only segments that have strictly positive slope, that is, slope type \((+, \ldots, +)\); the only difference is that now we have to cut the segment interval \([p_1 + \ldots + p_d, q_1 + \ldots + q_d - 1]\) into \( d \) parts, according to \( \prec \). Let \( p, q \in \mathbb{Z}^d \) be two points, such that \( p_i \leq q_i \) for all \( 1 \leq i \leq d \). We can define the segment \( S_\prec(p, q) \) in a similar way as in the 2D case — starting at \( p \) and repeatedly going in one of the \( d \) possible directions, collecting points from \( \mathbb{Z}^d \), until reaching \( q \). If we are at a point \((r_1, \ldots, r_d)\) for which \( r_1 + r_2 + \ldots + r_d \) is among the \( q_d - p_d \) greatest elements of the segment interval \([p_1 + \ldots + p_d, q_1 + \ldots + q_d - 1]\) according to \( \prec \), we proceed in direction \( d \); if it is among the \( q_{d-1} - p_{d-1} \) greatest elements of the segment interval that remain after removing the \( q_d - p_d \) elements that were the greatest, we proceed in direction \( d - 1 \), and so on. Finally, if \( r_1 + r_2 + \ldots + r_d \) is among the \( q_1 - p_1 \) smallest elements of the segment interval, we proceed in the first direction.

**Theorem 7.17.** The definition above yields a CDS of segments with strictly positive slope.
Proof. We start by proving the subsegment property (S3). As in the proof of Theorem 7.3, assume there are two segments with strictly positive slope, splitting at some point \( p \) and meeting again for the first time at \( q \) (see Figure 7.3). The two subsegments from \( p \) to \( q \) can be seen as words over the alphabet \([d] = \{1, 2, \ldots, d\}\), where 1 stands for going in the first direction, 2 for going in the second direction, and so on. So let \( \alpha = a_1a_2 \ldots a_k, \beta = b_1b_2 \ldots b_k \in [d]^k \) be these two words.

By assumption they differ at the beginning and at the end and they contain the same number of each of the letters, that is, for any \( l \in [d], |\{i \mid a_i = l\}| = |\{i \mid b_i = l\}|. \) Without loss of generality we may assume that \( a_1 > b_1 \). Then there must be an \( 1 < i \leq k \) such that \( a_i < a_1 \) and \( b_i > b_1 \): If there were no such \( i \), then for any \( 1 < i \leq k \) with \( a_i < a_1 \) we have \( b_i \leq b_1 < a_1 \). Now looking at all letters that are strictly smaller than \( a_1 \) in both words, we see that there is at least one such letter more in \( b \), namely at the first position, a contradiction.

This leads to a contradiction, as translated back to the original setting it implies \( p_1 + \ldots + p_d > p_1 + \ldots + p_d + i - 1 \), if we look at the interval of the first segment, and at the same time \( p_1 + \ldots + p_d < p_1 + \ldots + p_d + i - 1 \), if we look at the interval of the second segment. This proves (S3).

(S1), (S2), and (S5) follow directly from the construction. In order to prove the prolongation property (S4) we can argue as in the planar case. Consider the strictly positive sloped segment \( S(p, q) \). If \( q_1 + \ldots + q_d \) is among the \( q_d - p_d + 1 \) greatest numbers of \([p_1 + \ldots + p_d, q_1 + \ldots + q_d]\) according to \( \prec \), then we can prolong the segment going one step into direction \( d \). Else, if it is among the \((q_d - p_d) + (q_{d-1} - p_{d-1}) + 1 \) greatest elements, we can prolong the segment by going one step into direction \( d - 1 \), and so on.

Of course, we can use the same construction to define segments for all the remaining slope types. However, unlike in the 2-dimensional case, putting them all together in an attempt to construct a complete CDS fails in general, as the axiom (S3) might be violated, see Figure 7.12 for a possible conflict.

As long as we are interested in a system of digital rays with a common starting point \( o \) only, the different slope types do not interfere and our method can be used to derive such a system, thus reproving a result by Chun et al. [19], Theorem 4.4. But we are not going to elaborate on this.

### 7.6.2 Constrained CDSes in 3D

In this section we show how to combine two planar CDSes to a CDS in 3D that works for four of the eight slope types.

Take two planar CDSes \( S_1 \) and \( S_2 \) and define two projections \( \pi : (x, y, z) \mapsto (x, y) \) and \( \phi : (x, y, z) \mapsto (x + y, z) \). Then we define a system of digital line
Figure 7.12: A digital line segment from $p$ to $q$ (solid) and a segment from $r$ to $s$ (dashed) causing a contradiction to the subsegment property (S3).

segments by the following rules:

\[ \pi(S(p, q)) = S_1(\pi(p), \pi(q)) \]

and

\[ \phi(S(p, q)) = S_2(\phi(p), \phi(q)), \]

where $S_i(p, q)$ denotes the digital segment between $p$ and $q$ in $S_i$. If the projection $\pi(pq)$ has positive slope, there is a unique digital 3D segment $S(p, q)$ that satisfies the two rules. If, however, $\pi(pq)$ has negative slope, then in general there is no digital segment that satisfies them. We call a system of digital line segments in 3D satisfying axioms (S1)–(S5) only for pairs $(p, q)$ with the property that their projection onto the $xy$-plane has positive slope a constrained CDS.

**Theorem 7.18.** Given two planar CDSes $S_1$ and $S_2$, we can combine them to a constrained CDS satisfying the two rules above. Furthermore, using $S_1 = S_2 = S_{\prec}$ (the CDS derived from the total order that we defined in Section 7.3), we get a constrained CDS satisfying (H): for every $p, q \in \mathbb{Z}^3$ such that $\pi(pq)$ has positive slope, $H(\overline{pq}, S(p, q)) = O(\log d(p, q))$.

**Proof.** Let $p, q \in \mathbb{Z}^3$ a pair of grid points such that $(p, q)$ has slope type $(+++)$, $(++-)$, $(---)$, or $(--+)$. First we show that there is a unique segment $S(p, q)$ satisfying the two rules. Move from $p$ to $q$ in $Z^3$ following $S_1(\pi(p), \pi(q))$ and $S_2(\phi(p), \phi(q))$ at the same time. Imagine drawing $S(p, q)$ in 3D while following $S_1(\pi(p), \pi(q))$ with the index finger of your left hand and $S_2(\phi(p), \phi(q))$ with the index finger of your right hand. Follow $S_2(\phi(p), \phi(q))$ with your right finger. If it makes a move in the second coordinate we make the same move vertically in 3D (and the left finger stays where it is), else, if it moves in the
first coordinate, we also move our left finger and at the same time we move horizontally in 3D according to what our left finger does. No matter how we move exactly horizontally, \( x + y \) is either always increasing (in the cases \((+++)
\) and \((++-))\) or always decreasing, so the two rules stay satisfied after each step. See Figure 7.13. It is easy to show that all axioms are satisfied.

Second, we show that setting \( S_1 = S_2 = S_{\prec} \) the segments we get satisfy (H). By the proof of Theorem 7.2

\[ H(\pi(S(p, q)), \pi(\overline{pq})) = H(S_1(\pi(p), \pi(q)), \pi(p)\pi(q)) = O(\log d(\pi(p), \pi(q))) \]

and similarly

\[ H(\phi(S(p, q)), \phi(\overline{pq})) = H(S_2(\phi(p), \phi(q)), \phi(p)\phi(q)) = O(\log d(\phi(p), \phi(q))). \]

Therefore

\[ H(S(p, q), \overline{pq}) \leq O(\log d(\pi(p), \pi(q))) + O(\log d(\phi(p), \phi(q))) = O(\log d(p, q)). \]
7.7 Conclusion

The main result of this chapter is the construction of a CDS in the plane with the property that the digital segments are reasonably close, in the Hausdorff metric, to their Euclidean counterparts, thus reproving an older result by Luby \cite{52}. The key idea is to derive a CDS from a total order on $\mathbb{Z}$.

Adding an additional constraint, namely translational invariance along the antidiagonal, there is a one to one correspondence between CDSes and total orders on $\mathbb{Z}$. It turns out that CDSes with this additional property are the same as the smooth grid geometries studied by Luby. However, there remain “exotic” examples not satisfying this condition, like the waterline example presented in Section 7.4, which consequently are not derived from a total order (not smooth). Are there other “exotic” examples? How can we completely characterize all consistent digital line segment systems in the plane?

Moving to higher dimensional spaces, the trivial CDSes following the axis-parallel box spanned by the two end points naturally generalize. Also, one can generalize the waterline example. In three dimensions, for example, we can define a CDS as follows. Let $p$ be a point which is farther away from the plane $z = 0$ than $q$. Starting at $p$, first move towards the $z = 0$ plane vertically until either the $z$-coordinate equals $q_z$ or we hit the water level $z = 0$. Then move on the horizontal plane $z = q_z$ or on the water level $z = 0$, respectively, following some planar CDS, and finally, if necessary, move away from the $z = 0$ plane vertically to reach $q$. Unfortunately, just like their planar variants, these systems do not satisfy the small Hausdorff distance property.

Our approach using total orders remains applicable if we restrict to consistent digital rays emanating from one common point or if we only allow segments of a certain slope type. However, when trying to combine systems with different slope types, it seems inevitable to violate property (S3), as shown in Figure 7.12, unless we use systems derived from very special total orders — in which case the system does not satisfy property (H) — or we retreat to constrained systems where only some slope types are allowed.

The question whether there is a CDS in 3D or higher dimensions satisfying the small Hausdorff distance property (H) remains open.
Bibliography


Curriculum Vitae

Jakob Alfred Werner Tobias Christ

born May 8, 1981 in Frauenfeld, Switzerland, citizen of Basel, Switzerland, and Germany

1994 - 2001 Kantonsschule Frauenfeld, Switzerland
Degree: Matura, Type B (Latin)

2001 - 2006 Universität Basel, Switzerland
Degree: Master of Science in Mathematics

since 2007 Ph.D. student at ETH Zürich, Switzerland
Institute of Theoretical Computer Science