On-line Target Searching in Bounded and Unbounded Domains

by

Alejandro López-Ortiz

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 1996

I hereby declare that I am the sole author of this thesis.

I authorize the University of Waterloo to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize the University of Waterloo to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

In this work we study the problem of a robot searching for a target on bounded and unbounded domains, specifically in one and two dimensions. We present some relevant results in the field, and within this framework, the contributions of this work. We show that one-sided searches on the real line have an average competitive ratio of 9 and we apply this result for the case of known destination searches on $\mathcal{G}$-street polygons. For this class of polygons we also show that the best known upper bound of $\sqrt{82}$ is indeed optimal for the case of unknown destination searches. For orthogonal street polygons we prove that knowing the location of the destination does not reduce the competitive ratio.

We present the first robust algorithm for searches inside street polygons under navigational errors, as well as for other important classes of impaired robots such as robots lacking triangulation and depth measurement mechanisms. In particular, we propose a $(\pi+1)$-competitive strategy which is robust under navigational error, and equally efficient for oblivious robots. We also present an 1.92-competitive strategy which does not require depth perception from the robot.

We also give the best-known strategy for searching inside street polygons, at a competitive ratio of 1.76. This significantly improves over the best previously known ratio of 2.8.

We give the first non-trivial lower bound for searching and walking into the kernel of a polygon. We provide upper and lower bounds for searching for a target inside street polygons. These results are the first constant competitive ratio strategies inside a class of polygons which is independent of the location of the start and target points.

We also prove negative results regarding the optimality of several well-known family of strategies for searching inside simple polygons. In particular we show that Kleinberg's strategy is 2.6-competitive, and that continuous bisector, which is in many respects a good strategy, is at least 1.68-competitive. Lastly we show that any strategy which does not respond to the absence of new extreme points has a competitive ratio strictly larger than $\sqrt{2}$.

Acknowledgements

I wish to express my gratitude to all those who made possible the successful completion of this work. First and foremost to my supervisor Professor Prabhakar Ragde who by his actions has set an example for all to follow. His wide range of interests in the computer sciences and beyond has made this learning experience all the richer.

I was fortunate to meet my fellow researcher and coauthor of many articles, Sven Schuierer. Working with him has been a most fruitful and enjoyable experience. I am indebted to the other members of my thesis committee: Profs. I. Munro, M. Li, J. Cheriyan and a special mention to J.-R. Sack, whose detailed reading of this thesis and thoughtful comments greatly contributed to the quality of this work.

I would also like to thank my friends Atsushi Yamada and Arun Achyuthan from the International Graduate Student Committee, as well as my office neighbours and friends Andrej Brodnik, Igor Benko, Marzena Makuta, Luke O'Connor and Peter Yamamoto for many enjoyable corridor "seminars", which provided much needed breaks during a hard day's work. I must thank my Latin American friends Desireé Castillejos, René Mayorga, Juan Pablo and Diana Soriano, Luis and Lucia Vargas, and Alfredo and Graciela Viola for all the memories; as well as Daniel M. Germán, Eva Wieser, Peter Yamamoto and Rob Zvan, for most interesting discussions on many topics.

Last, but not least, I wish to thank my parents for all those Sunday morning calls, and my girlfriend Claudia Iturriaga Velázquez for her support and many happy times together.

Contents

List of Figures

Chapter 1

Introduction

> The US Coast Guard uses a Computer Assisted Search Planning System to provide a probability distribution for search target location [...] This tool, available to all search planners, takes into account the many uncertainties associated with a search as well as the results of prior searches.
>
> —*U.S. Coast Guard Fact File 1995*

1.1 Motivation and Introduction

The problem of target searching involves an agent or robot exploring a given domain, with the purpose of guarding it or finding and possibly reaching a given target under some conditions of uncertainty.

On-line target searching, as opposed to path-planning, is the class of target searching problems in which the robot is only partially acquainted with some of

- the configuration of the search domain,
- the topology of the terrain,
- the position of the target and
- its own position.

These conditions introduce a degree of uncertainty in the search that is remedied as the robot traverses the terrain. Thus, searches can be viewed as an on-line problem because the robot can only attain more information while active, as opposed to the off-line version in which all the information is given at once before the start of the computation.

While in this work we refer to the search agent as a "robot", it should be noted that searching for a target is a problem that predates the era of computer technology. Examples include **Search and Rescue** missions on land and sea as well as locating of buried objects by construction and service crews.

For example, Canadian Coast Guard manuals [12] describe the following procedures:

> In the planning of the search, the Co-ordinator Surface Search (CSS) must plot the *datum*, which is the most probable position of the search target at a given time, and the initial most probable area.
>
> Individual search patterns have been designed with the aim of providing a ready made framework to enable a search by one or more ships to be initiated rapidly by the CSS [...] These patterns have been selected for simplicity of execution.

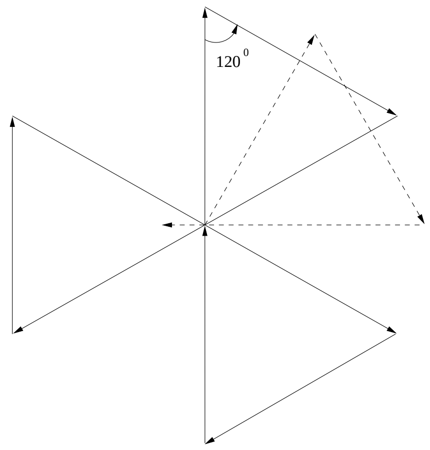

Figure 1.1: Sector search, Canadian Coast Guard SAR procedures.

For a single ship searching an unbounded region it recommends an *expanding square search* similar to the spiral search introduced in Chapter 3. For the case of exhaustive search of a circular region (i.e. the neighbourhood of a datum), the coast guard proposes the "sector search" displayed in Figure 1.1. Recently, manual search planning systems have been replaced with computer-assisted programs.

Until recently, co-ordinators performed maritime search planning by hand. Co-ordinators had to calculate the search area and direct rescue units to the survivor's most likely location. In the manual calculations, assumptions were made that enlarged the search area to ensure that there was at least a 50% chance the target would be contained.

Using the Canadian Search and Rescue Planning program search co-ordinators create multiple "what if scenarios" and within minutes have a graphic display of the search area and search plan [51] which optimizes the probability of finding the target within a short time span given the search resources available. Similarly, the US Coast Guard uses Monte-Carlo methods for search target location [65].

Another common form of target searching is the children's game hide-and-seek, which in the past presumably prepared its players for more advanced forms of target search such as hunting and other survival activities.

In on-line target searching the target is assumed to be static or immobile. In pursuit (differential) games, the target is mobile and the robot must "tag" the target [4, 23]. Differential games assume that the target as well as the robot is mobile and usually the target is visible at all times. Even though these games share their origin with search games, we consider them to be sufficiently different in technique and approach as to deserve separate treatment, and they are thus not included in this survey.

1.2 Historical Overview

In this work we focus on on-line static search problems. Amongst the best-known forms of on-line target searching is the popular mathematical game of a sailor swimming to shore. In this game, one is asked to device a strategy for a sailor to reach the shoreline, which is located at a given distance from the current position but in an unknown direction. The aim is to minimize the distance swum by the sailor. A common variant is the case in which the distance to shore is unknown. In either case, the target to be "tagged" is a line sitting in an unknown planar domain. A similar problem consists of finding a buried utility line sometimes assumed to be located at a bounded distance from a given point [60].

The duration and extent of the search, be it in the case of the sailor or any other search depends on the geometry of the terrain and the search tools and information available to the searcher. Henceforth we consider that the searcher or robot is an agent aided with tactile or visual sensors, partial information on the search domain or position of the target, navigational tools, and computing resources.

In this form, on-line searching becomes an optimization problem in which the robot aims to minimize the distance traversed in the average or worst case as compared to the shortest path.

Work on this area can be broadly classified into three different epochs: naval research during 1940-1960, the game theoretical approach during 1970-1980 (most of which forms the basis of what is now known as differential games), and the 1987-present period, in which research has focused on computational aspects of on-line searching problems.

Results from the early period are not readily available. For example, a monograph by Koopman [40] covering most of the work from the war time period was classified as CONFIDENTIAL until 1958. This secrecy has led to some duplication of work by researchers in the same and/or different fields, which continues to various degrees to date[1]. It is in this early period when the first solution to the sailor swimming to shore problem seems to have been devised.

Most advances from the 1970-1980 period are recapitulated in Gal's comprehensive survey[2] "Search Games" in 1980 [24]. At this stage the main thrust of work moved away from static searches and towards differential games.

Since the last fifteen years have seen heightened activity in the field, Gal's work, thorough as it is, is out of date. This thesis will summarize many of the main results in the field and within this framework present original research done by the author in collaboration with Sven Schuierer.

[1]As a curious historical fact, J. M. Dobbie notes that two selected 1966 bibliographies covering the field had less than 30% articles in common [21].

[2]Ruckle published a subsequent survey in 1983, apparently unaware of Gal's expository work.

1.2.1 Types of Search

The quality of a search varies significantly depending on the searching abilities of the robot, as well as the type and amount of information known about the terrain. For this reason it is important to consider many similar cases independently.

Targets for searches are classified, as we mentioned before, into two main types: **mobile** and **immobile**.

In turn, domains are divided into **bounded** and **unbounded**. Among the first, we consider searches inside a given rectangular or circular region, inside a polygon, and on line segments, among others. For unbounded domains we consider the real line and the Euclidean plane.

Robots are also classified according to their abilities in several classes: **tactile, visual, navigational tools,** and **computing resources.** A tactile robot identifies obstacles in the terrain and the target when the robot is located at an ϵ distance or less of the object in question. A robot with vision is equipped with a system that provides a visibility map of its local environment. Among the navigational tools, the robot may be aided by a compass that identifies a preferred direction. Computing resources are characterized by size of memory and number of computation steps. A robot moving under restricted memory conditions may not have access to previous visibility maps. A robot with no memory is termed **oblivious**. A robot with a limited amount of computational resources performs a stroboscopic search in which it is allotted only a limited number of mappings of the terrain to be explored.

Information available to the robot is also relevant to the problem. The type of information given, if any, is **location** or **distance** to the target and partial information on the **terrain** and **type of search**.

1.3 Overview

This thesis is concerned with searches for immobile targets in various domains. In Chapter 2 we study searches on the real line. We provide the first thorough proofs of some "folk" theorems in the field such as the *doubling strategy*. As well, we introduce some new average case results for asymmetric strategies. In the same chapter we explore the impact of some practical constraints such as limited speed and acceleration on the strategies proposed.

In Chapter 3, we present some related results in the field which place in context results from later chapters. In particular, we introduce the spiral search, which is used in Chapter 5 for lower bounds, and searches with obstacles which are of use for searches on polygons with holes.

In Chapter 4, we study some of the simplest forms of searching inside a polygon. We consider the case of searches in star polygons as well as the problem of walking into the kernel of a star polygon. We propose upper and lower bounds for target searches in star polygons.

More complex forms are studied in Chapter 5 and 6. We explore in detail searches on street polygons, both with known and unknown destination. We prove matching upper and lower bounds for searches with known destination in orthogonal polygons. In Chapter 6 we also provide a strategy with the best known competitive ratio for the case of street polygons. We explore several families of strategies, which, while suboptimal, are robust under realistic constraints on the robot. In this regard we propose a $(\pi + 1)$-competitive strategy which is robust under navigational error, and equally efficient for oblivious robots. We also present an 1.92-competitive strategy which does not require depth perception from the robot. In this chapter we show a lower bound on a particular strategy due to Kleinberg that matches the best known upper bound, and propose several variations on this strategy which result in better competitive ratios. An important result is that there are cases in which a robot must change its trajectory even on the absence of new extreme points if it aims to be $\sqrt{2}$-competitive, which is widely believed to be achievable. Interestingly most strategies do not react to the absence of new features.

In Chapter 7 we show that searches in the class of orthogonal $\mathcal{G}$-street polygons with known destination are also at least 9-competitive. We also prove a lower bound which matches the upper bound for searching in this class of streets, effectively completing the study of this class. Lastly, we present and analyze the first strategy for searches in general $\mathcal{G}$-streets.

Chapter 2

Searching on the Real Line

The situation of a robot searching for a point on the real line is the most basic form of search. We assume that the robot "finds" the target either when it reaches the position of the target or, in some cases, when it comes within a distance of ϵ or less (an ϵ-neighbourhood) of the target.

This problem has been studied in many different settings, and partial solutions to it have been rediscovered independently by several researchers [5, 24, 1]. In its full generality, it consists of a robot searching for a point on the infinite line.

Without loss of generality, we assume that the robot is located at the origin and searches for an immobile target located at the point $p \in R$ in the real line. The target selects its hiding place either according to a (possibly unknown) probability distribution or according to an ad-hoc strategy designed to counter the robot's searching strategy.

Definition 2.1 *For a given deterministic search strategy S, we denote the cost of finding the point p as $\widehat{C}_S(p)$, which is defined as the total distance traversed by a robot following strategy S until it finds a target located at point p.*

2.1 A Game Theoretical Approach

On-line search problems can be approached from the perspective of game theory. In this setting there are two players, namely, the robot and the target. The robot selects a search strategy while the target chooses a hiding strategy. The aim of the game for the robot is to minimize the search time.

Definition 2.2 *Let $F(x) = Pr(g \leq x \,|\, g$ is the target position$)$ denote the cumulative distribution function (c.d.f.) of the target hiding selection mechanism. Let $f(x)$ be the probability density function (p.d.f.) corresponding to $F(x)$.*

Definition 2.3 *The* **value of the game** *for the target is the expected distance traversed by the robot in search of the target $\int_R \widehat{C}_S(x) f(x)\, dx = \int_R \widehat{C}_S(x)\, dF(x)$.*

The value of the game for the target is the same as the average (expected) cost of the search for the robot. Note that on-line searching on the real line is, by definition, a positive-sum game for the target player.

The robot wishes to minimize the expected distance traversed in search of the target. However, this approach is not very meaningful as the value of the game might be infinite, even for a robot that knew its destination beforehand.

More formally, there exist distribution functions $f(x)$ on the real line such that the average (expected) distance from the origin to a point, $\int_R f(x)|x|\, dx$ is an infinite quantity. Consider the probability distribution induced by the distribution function given by:

$$f(x) = \begin{cases} 2^{-n-1} & \text{if } x \in [2^{2^n}, 2^{2^n}+1] \text{ and } n \text{ is an integer,} \\ 0 & \text{otherwise.} \end{cases}$$

with $\int_R f(x)\,dx = \sum_{n=1}^{\infty} 1/2^n = 1$, while $\int_R f(x)|x|\,dx \geq \sum_{n=1}^{\infty} 2^{2^n}/2^n = \infty$. The function $f(x)$ is well defined as $\int_R f(x)\,dx = \sum_{n\in N} 2^{-k} = 1$. The expected distance is bigger than $\sum_{n=1}^{\infty} 2^{2^n}/2^n$ which diverges.

The distribution above can be discretized, while still having an infinite expected distance.

This implies that the expected traversed search distance for a robot under an arbitrary distribution can be infinite. To see this, notice that $\widehat{C}_S(p) \geq |p|$ and thus $\int_R \widehat{C}_S(p) f(p)\,dp \geq \int_R f(x)|x|\,dx = \infty$.

Furthermore, it is possible to have two strategies which are essentially equivalent and resulting in different values for the game. This is due to the fact that, for many hiding strategies, the value of the game is increased by simply "hiding farther away". Even the expected distance (that is, a perfect search) can be increased arbitrarily without any qualitative changes to the hiding strategy. Given any hiding strategy with p.d.f. $f(x)$ one can define a new hiding strategy $g(x) = cf(cx)$. The function $g(\cdot)$ is a p.d.f. as $\int_R g(x)\,dx = \int_R c\,f(cx)\,dx = \int_R f(y)\,dy = 1$ where $y = cx$ and $\int_R g(x)|x|\,dx = \int_R cf(cx)|x|\,dx = \int_R f(y)|y|/c\,dy = 1/c \int_R f(y)|y|\,dy$. Thus, for $c < 1$, the hiding strategy g (which is essentially an enlargement of f) is $1/c$ times better for the target.

There are other differences or disadvantages of the game-theoretical model. The real-life robot, as opposed to the traditional game-theoretical player, is constrained to a single deterministic strategy which is chosen deterministically. It is also assumed (for worst case analysis) that the target has foreknowledge of the strategy ultimately chosen by the robot and can optimize against it. That is, the robot is restricted to a pure strategy which is known by the adversary.

2.2 The Competitive Framework

A natural alternative is to normalize the expected search time by the actual distance of the target. This normalized figure is termed the competitive ratio of the search strategy.

Definition 2.4 *The* **competitive ratio** *for any on-line search problem is the largest possible value of the quotient of the distance traversed by the robot before reaching its target and the distance it would have traversed under perfect information, for all possible locations of the target.*

In the case of the real line the competitive ratio *for a given point* p is the distance traversed by the robot divided by the distance from origin to the target position, that is $C_S(p) = \widehat{C}_S(p)/|p|$.

We are interested in both worst case and average case performance under this measure.

For the case of competitive searches on the real line we study the case where the target point is not located within a distance ϵ of the robot starting position. This is a natural restriction, as otherwise the target can hide infinitesimally close to the starting point and on the opposite side of the first search move by the robot, resulting in an unbounded competitive ratio.

As well, without loss of generality, we can assume that the robot starts from the origin. This simplifies the description of a strategy.

As the robot can only choose the "turn points" at which it changes its direction of movement, any search strategy S can be described by the sequence of turn points $\{0, s_1, s_2, s_3, \ldots\}$.

Definition 2.5 *Let $\mathcal{C}_S$ denote the maximum (worst case) competitive ratio for strategy S, i.e. $\mathcal{C}_S = \sup_{x \in R}\{C_S(x)\}$.*

Observation 2.1 *Any search strategy is dominated by a search strategy such that $0 < s_1 < s_3 < s_5 < \ldots$ and $0 > s_2 > s_4 > s_5 > \ldots$ with no larger competitive ratio.*

Proof. Consider a strategy $S = \{0, s_1, s_2, s_3, \ldots\}$ such that $s_1 < 0$. As $C_S(x) = \widehat{C}_S(x)/|x|$, it follows that the mirror strategy $S' = \{0, -s_1, -s_2, -s_3, \ldots\}$ has the cost function $C_{S'}(x) = C_S(-x)$ and thus $\mathcal{C}_{S'} = \mathcal{C}_S$. This proves that we can assume $s_1 > 0$.

Now suppose that strategy S does not move in a sequence of increasingly large alternating moves. This means there exist an i such that at least one of

$$0 < s_{i+1} < s_i, \tag{2.1}$$

$$0 > s_{i+1} > s_i, \tag{2.2}$$

$$s_{i+1} < 0 < s_{i+2} < s_i \qquad \text{or} \tag{2.3}$$

$$s_{i+1} > 0 > s_{i+2} > s_i \tag{2.4}$$

hold. In cases (2.1) or (2.2) the robot can simply move from s_i to s_{i+2}, thus obviating the move s_{i+1} without a subsequent decrease in the competitive ratio. For cases (2.3) and (2.4) the robot moves from s_{i+1} to s_{i+3} once again obtaining a strategy with a better competitive ratio. This shows that one can avoid steps that do not follow the pattern described in observation 2.1 without penalty for the competitive ratio, as required. $\square$

Lemma 2.1 *The competitive ratio of a search strategy for the real line, as described in observation 2.1, is*

$$\mathcal{C}_S = 1 + 2 \sup_{k \in Z} \left\{ \sum_{i=1}^{k} \frac{|s_i|}{|s_{k-1}|} \right\}$$

Proof. From the definition,

$$
\begin{aligned}
\mathcal{C}_S &= \sup_{x \in R}\{C_S(x)\} = \sup_{x \in R}\{\widehat{C}_S(x)/|x|\} \\
&= \sup_{x \in R}\left\{\frac{\sum_{i=1}^{k} 2|s_i| + |x|}{|x|} \mid x \in [s_{k-1}, s_{k+1}]\right\} \\
&= 1 + \sup_{x \in R}\left\{\sum_{i=1}^{k} \frac{2|s_i|}{|x|} \mid x \in [s_{k-1}, s_{k+1}]\right\} \\
&= 1 + \sup_{k \in Z}\left\{\sum_{i=1}^{k} \frac{2|s_i|}{|s_{k-1}|}\right\}
\end{aligned}
$$

□

2.3 Doubling Strategies

Although this theorem has been rediscovered many times[1], all known proofs involve restrictions on the search domain or target probability distribution function (p.d.f). The following version is more general as it assumes a malicious target which does not necessarily hide using a p.d.f. It also applies to the entire real line and not only to integer hiding points. Finally, the theorem is described here in the competitive framework.

Theorem 2.1 [7, 43, 1] *The real line can be searched at an optimal 9-competitive ratio.*

Proof. First we show that the line can be searched at a 9-competitive ratio. Consider the following search strategy.

[1]The earliest version found yet dates to Beck in 1970 [5], though considering the intense level of research on this subject during the 1940-1960 period for the Office of Naval Research, it would not be surprising to learn that it was already known back then [40].

Doubling Strategy

$D = \{0, 1, -2, 4, \ldots, (-1)^i\, 2^i, \ldots\}$

The competitive ratio for this strategy is, by Lemma 2.1,

$$\begin{aligned} \mathcal{C}_D &= 1 + 2 \sup_{k \in Z} \left\{ \sum_{i=1}^{k} \frac{2^i}{2^{k-1}} \right\} = 1 + 2 \sup_{k \in Z} \left\{ \frac{2^{k+1} - 1}{2^{k-1}} \right\} \\ &= 1 + 2 \sup_{k \in Z} \left\{ 4 - \frac{1}{2^{k-1}} \right\} = 9 \end{aligned}$$

Notice that the expression above denotes the supremum of a sequence converging from below to 9. This means that all points are found at a competitive ratio which is slightly better than 9.

Now we proceed to show that this ratio is indeed optimal. Let $\mathcal{C}_S$ be the competitive ratio of a given strategy $S = \{s_1, s_2, \ldots\}$. We are required to prove that $\mathcal{C}_S \geq 9$ for any strategy S. We compare the strategy S with the doubling strategy by studying the sequence $\{c_n\}$, where $c_n = \frac{1}{2^n} \sum_{i=1}^{n} |s_i|$. Notice that

$$\begin{aligned} \tfrac{1}{2}\left(c_{n+1} + c_{n-1}\right) &= \frac{1}{2^{n+2}} \sum_{i=1}^{n+1} |s_i| + \frac{1}{2^n} \sum_{i=1}^{n-1} |s_i| \\ &= \frac{1}{2^n} \left(\frac{1}{4} \sum_{i=1}^{n+1} |s_i| + \sum_{i=1}^{n-1} |s_i| \right) \end{aligned} \tag{2.5}$$

From Lemma 2.1, we have

$$\mathcal{C}_S \geq 1 + 2 \sum_{i=1}^{n+1} \frac{|s_i|}{|s_n|} \quad \text{which implies} \quad \sum_{i=1}^{n+1} |s_i| \leq (\mathcal{C}_S - 1)/2\, |s_n|$$

Substituting in Equation 2.5 we obtain

$$\begin{aligned} \tfrac{1}{2}\left(c_{n+1} + c_{n-1}\right) &\leq \frac{1}{2^n} \left[\sum_{i=1}^{n-1} |s_i| + \frac{1}{4} \left(\frac{\mathcal{C}_S - 1}{2} \right) |s_n| \right] \\ &= \frac{1}{2^n} \left[\sum_{i=1}^{n} |s_i| + \left(\frac{\mathcal{C}_S - 1}{8} - 1 \right) |s_n| \right] \\ &= c_n + \frac{1}{2^n} \left(\frac{\mathcal{C}_S - 9}{8} \right) |s_n| \end{aligned} \tag{2.6}$$

Now, either $\mathcal{C}_S > 9$ in which case there is nothing to show, or $\mathcal{C}_S \leq 9$. In the latter case, $(\mathcal{C}_S - 9)/8 \leq 0$ and again from Lemma 2.1 and Equation 2.6, we have

$$\begin{aligned}\tfrac{1}{2}(c_{n+1} + c_{n-1}) &\leq c_n + \frac{1}{2^n}\frac{2\sum_{i=1}^{n+1}|s_i|}{\mathcal{C}_S - 1}\left(\frac{\mathcal{C}_S - 9}{8}\right) \\ &= c_n - r c_{n+1} \quad \text{where} \quad r = \frac{1}{2}\left(\frac{9 - \mathcal{C}_S}{\mathcal{C}_S - 1}\right) > 0\end{aligned}$$

Equivalently,

$$\tfrac{1}{2}(c_{n+1} - c_n) \leq \tfrac{1}{2}(c_n - c_{n-1}) - r c_{n+1} \tag{2.7}$$

The strictly positive sequence $\{c_n\}$ is either monotone non-decreasing or it is not. If it is not then there exists N such that $c_{N-1} > c_N$. Equation 2.7 implies $c_N - c_{N+1} \geq c_{N-1} - c_N$, and in general $c_N - c_{N+k} \geq k(c_{N-1} - c_N)$, which implies $c_N - k(c_{N-1} - c_N) \geq c_{N+k}$ for all $k \geq 1$. However the left-hand side of this equation is negative for all $k > \left\lceil \frac{c_{N-1}}{c_{N-1} - c_N} \right\rceil$, implying $0 > c_{N+k}$, which is a contradiction as $c_n \geq 0$ for all n.

Thus we know that $c_n \geq c_{n-1} \geq c_1$ for all $n \geq 2$. Equation 2.7 implies then $c_{n+1} - c_n \leq c_n - c_{n-1} - 2rc_1$, and iteratively, $c_{n+k} - c_{n+k-1} \leq c_n - c_{n-1} - 2krc_1$. But once again, the right hand side of this equation is negative for $k \geq \left\lceil \frac{c_n - c_{n-1}}{2rc_1} \right\rceil$ unless r is zero. So r must be 0, which implies $\mathcal{C}_S = 9$, as required. □

Corollary 2.1 *All search strategies are at least $9 - \epsilon$ competitive infinitely often, for any $\epsilon > 0$.*

Proof. In the proof of Theorem 2.1 we considered the competitive ratio at all turn points s_j. The same proof holds if we restrict to turn points s_j with $j \geq M$ for any given M. □

Theorem 2.2 *Any search strategy for a target at most n units away is at least $(9 - f(n))$-competitive, where $f(n) \leq 24/\log_4 n$, and n is sufficiently large.*

Proof. Consider a strategy $S = \{s_1, s_2, \ldots, s_N\}$ for searching the interval $[-n, n]$ starting from the origin. We will show that $\mathcal{C}_S \geq 9 - 24/\log_4 n$.

First, notice that if $\mathcal{C}_S \leq 9$, then

$$1 + 2\sum_{j=1}^{i} \frac{|s_j|}{|s_{i-1}|} \leq 9 \Longrightarrow \sum_{j=1}^{i} |s_j| \leq 4|s_{i-1}|$$

and in particular $|s_i| \leq 4|s_{i-1}|$. Thus $N \geq \log_4 n$.

Let $\{c_i\}$ be defined as in the proof of Theorem 2.1 for $1 \leq i \leq N$ and let $M = \lceil N/2 \rceil$. Now either $\{c_j\}_{1 \leq j \leq M}$ is an increasing sequence or not. If it is increasing, then from Equation 2.7 in Theorem 2.1 we have that

$$c_{2+k} - c_{1+k} \leq c_2 - c_1 - 2krc_1.$$

Notice as well that, since $|s_i| \leq 4|s_{i-1}|$, then

$$\frac{3}{2} \geq \frac{|s_2| - |s_1|}{2|s_1|} = \frac{\frac{1}{4}(|s_1| + |s_2|) - \frac{1}{2}|s_1|}{\frac{1}{2}|s_1|} = \frac{c_2 - c_1}{c_1}$$

Let $\mathcal{C}_S = 9 - f(n)$ for some positive function $f(n)$. Once again, we have two possibilities: either $f(n) \leq 24/\log_4 n$ in which case there is nothing to show or $f(n) \geq 24/\log_4 n$. In this case we have

$$\begin{aligned} f(n) &\geq \frac{24}{\log_4 n} \geq \frac{12}{M} \quad \Longrightarrow \\ M &\geq \frac{8 \cdot 3}{2f(n)} \geq \frac{8}{f(n)} \cdot \frac{c_2 - c_1}{c_1} \geq \frac{8 - f(n)}{f(n)} \cdot \frac{c_2 - c_1}{c_1} = \frac{c_2 - c_1}{2rc_1} \end{aligned}$$

where r is as defined in the proof of Theorem 2.1. In short, we obtained $M \geq \frac{c_2 - c_1}{2rc_1}$; this implies that, for all k such that $M \leq k \leq N$,

$$c_{2+k} - c_{1+k} \leq c_2 - c_1 - 2krc_1 < 0$$

which is a contradiction as we know that $\{c_j\}$ is an increasing sequence.

Now, if $\{c_j\}_{1\leq j\leq M}$ is not increasing then from Equation 2.7 it follows that there exists an i such that $c_j \geq c_{j+1}$ for $i \leq j \leq N$. Once again if $\lceil \frac{c_{j-1}}{c_{j-1}-c_j} \rceil < M$ this would imply $c_{M+k} \leq 0$ which is a contradiction. Thus $M \leq \lceil \frac{c_{j-1}}{c_{j-1}-c_j} \rceil$, equivalently $M/(M-1) \geq c_{j-1}/c_j$. Now if there exists k such that

$$1 + 2\sum_{i=1}^{k} \frac{|s_i|}{|s_{k-1}|} \geq 9 - \frac{16}{\log_4 n}$$

there is nothing to show; thus we can assume that

$$1 + 2\sum_{i=1}^{k} \frac{|s_i|}{|s_{k-1}|} < 9 - \frac{16}{\log_4 n}$$

for all $1 \leq k \leq N$. This implies $2\sum_{i=1}^{k} |s_i| < [8 - 16/\log_4 n]|s_{k-1}|$. From the definition of c_j we get

$$\frac{M}{M-1} \geq \frac{c_{j-1}}{c_j} = \frac{2\sum_{i=1}^{j-1}|s_i|}{\sum_{i=1}^{j}|s_i|} > \frac{4\sum_{i=1}^{j-1}|s_i|}{(8-16/\log_4 n)|s_{j-1}|}$$

But we know that $c_{j-1} \leq c_{j-2}$ which implies $\sum_{i=1}^{j-1}|s_i| \leq 2\sum_{i=1}^{j-2}|s_i|$ implying $s_{j-1} \leq \sum_{i=1}^{j-2}|s_i|$. Thus

$$\frac{M}{M-1} > \frac{8}{8-16/\log_4 n}$$

from this we obtain

$$\frac{M-1}{M} < 1 - \frac{2}{\log_4 n} \quad \Longrightarrow \quad M < \frac{\log_4 n}{2}$$

which is a contradiction as $M \geq N/2$. □

The following theorem completes a proof first outlined in [1].

Theorem 2.3 [1, 43] *The optimal competitive ratio is 9 for a target hiding on the integer points of the real line.*

Proof. The Doubling Strategy is also 9-competitive for hiding strategies on integer points, which proves the upper bound. Now, let us assume that there exists an integer point search strategy S which is $9-\epsilon$ competitive, where $\epsilon > 0$. The competitive ratio of the integer point strategy is given by

$$\mathcal{C}^i_S = 1 + 2 \sup \left\{ \sum_{i=1}^{k} \frac{|s_i|}{|s_{k-1}|+1} \right\}.$$

Note the additive factor of 1 in the denominator. This reflects the fact that if the robot did not find the target at the ith turn point then the next possible position of the target on that side is at least one unit farther away from the turn point.

The same integer strategy S applied to searches on the real line has competitive ratio

$$\mathcal{C}_S = 1 + 2 \sup \left\{ \sum_{i=1}^{k} \frac{|s_i|}{|s_{k-1}|} \right\}.$$

Let k be such that $(9-\epsilon)/|s_{k-1}| < \epsilon/4$, and $|s_{k+j}| \geq |s_k|$, for $j \geq 0$. We know such k exists as s_n is composed of two unbounded monotone-increasing sequences. Then, this implies

$$\begin{aligned}
\epsilon/4 &> \frac{9-\epsilon}{|s_{k-1}|} = \frac{1}{|s_{k-1}|} + 2\frac{\sum_{i=1}^{k}|s_i|}{(|s_{k-1}|+1)|s_{k-1}|} \\
&= \frac{1}{|s_{k-1}|} + 2\frac{(|s_{k-1}|+1-|s_{k-1}|)\sum_{i=1}^{k}|s_i|}{(|s_{k-1}|+1)|s_{k-1}|} \\
&\geq 2\left[\frac{\sum_{i=1}^{k}|s_i|}{|s_{k-1}|} - \frac{\sum_{i=1}^{k}|s_i|}{|s_{k-1}|+1}\right],
\end{aligned}$$

which implies, that $C_S(s_k) - C^i_S(s_k) \leq \epsilon$, and thus $C_S(s_k) \leq 9-\epsilon$, and similarly $C_S(s_n) \leq 9-\epsilon$ for all $n \geq k$.

Let us define a new search strategy $S' = \{s'_i\}$ as follows, $s'_i = (-1)^i c 2^i$ for $i < k$ and $s'_i = s_i$ for all $i \geq k$, with $c = \sum_{i=1}^{k-1} |s_i|/2^k$. Let ℓ be such that $c2^\ell \leq 1 \leq c2^{\ell+1}$. We claim that $\mathcal{C}_{S'} = \max\{9-\epsilon, 9-1/2^{k+\ell-2}\}$. Indeed, consider the expression of the competitive ratio given by Lemma 2.1

$$\mathcal{C}_{S'} = 1 + 2 \sup_{j \in Z} \left\{ \sum_{i=1}^{j} \frac{|s_i|}{|s_{j-1}|} \right\}$$

If $j < k$ then $1 + 2\sum_{i=\ell}^{j} c2^i/(c2^{j-1}) = 1 + 2(4 - 1/2^{j+\ell-1}) = 9 - 1/2^{j+\ell-2} \leq 9 - 1/2^{k+\ell-2}$. If $j \geq k$ we have

$$\begin{aligned} 1 + 2\sum_{i=\ell}^{j} \frac{|s_i|}{|s_{j-1}|} &= 1 + 2\frac{\sum_{i=\ell}^{k-1} c2^i + \sum_{i=k}^{j} |s_i|}{|s^{j-1}|} = 1 + 2\frac{c(2^k - 1) + \sum_{i=k}^{j} |s_i|}{|s^{j-1}|} \\ &\leq 1 + 2\frac{\sum_{i=0}^{k-1} |s_i| + \sum_{i=k}^{j} |s_i|}{|s^{j-1}|} = 1 + 2\frac{\sum_{i=0}^{j} |s_i|}{|s^{j-1}|} \quad \leq \quad 9 - \epsilon \end{aligned}$$

which implies $\mathcal{C}_{S'} = \max\{9 - \epsilon, 9 - 1/2^{k+\ell-2}\}$ as required. But this means that S' is a general strategy for the real line with competitive ratio under $9 - \epsilon$ which contradicts Theorem 2.1. □

2.4 Random Strategies and Average Case Results

Average case analysis assumes a "natural" distribution over the input space against which the performance of an algorithm can be tested. Thus an efficient average case algorithm may perform badly on rarely occurring inputs.

However, as the real line has no natural distribution function it is not possible to perform an average case analysis in the standard sense. Thus we have to assume that the target selects a distribution which is unknown to the robot. The robot then selects a searching strategy randomly as well.

This constitutes what is called a mixed strategy, in game-theoretical terms, or a random strategy, in computer science terminology. For this type of approach, the cost of finding a target hiding at point p is, in this case, a random variable

$\widehat{C} : S \times R \rightarrow R$ which maps a strategy S_r and a hiding point p to the distance traversed by the robot under S_r before reaching the point p.

Definition 2.6 *The competitive ratio for a point p, in the case of a robot using a randomized selection of strategies* $\{S_r\}_{r\in\Omega}$ *is defined as* $C_{S_r}(p) = \int_\Omega \widehat{C}(S_r, p)/|p| d\mu$ *where* μ *is the distribution function imposed by the robot over the set of strategies* $\{S_r\}$.

The competitive ratio of a set $\mathcal{S}$ of randomized strategies is defined, as in section 2.3, to be the supremum of all possible values of the competitive ratio, i.e. $\sup_{p\in R}\{C_{S_r}(p)\}$.

The following theorem was first proven by Beck [5, 6, 7] and was independently rediscovered in a weaker form by Kao et al. [36].

Theorem 2.4 **[24] Trade-off Theorem for the Real Line** *There exists a family of random (mixed) strategies* $\mathcal{S}$ *which are average and worst-case optimal. That is, given a strategy* $S \in \mathcal{S}$ *in the family, and an arbitrary strategy* S', *then if* $avg(C_{S'}) < avg(C_S)$ *then* $worst(C_{S'}) > worst(C_S)$ *and if* $worst(C_{S'}) < worst(C_S)$ *then* $avg(C_{S'}) > avg(C_S)$.

The proof is constructive [24, 36] which results in the following corollary.

Corollary 2.2 **[24, 36]** *The strategy with the best average case ratio has an average case competitive ratio of* $1 + 1/W(1/e) \approx 4.5911214$, *where* $W()$ *is the Lambert function determined by the functional equation* $W(x)e^{W(x)} = x$.

It is tempting then to claim that this strategy is significantly better than the doubling strategy. The following three lemmas dispel this notion.

Lemma 2.2 [24] *The strategy in the corollary above has a worst case competitive ratio of* $1 - 2/(W(e^{-1})^2 - W(e^{-1})) \approx 10.95410950$.

Lemma 2.3 *Assume that the hiding target selects a random distance* d *and that it hides using a uniform distribution on the interval* $[d, -d]$. *Then the doubling strategy has a worst case competitive ratio of* ≈ 5.27.

Proof. The robot searches intervals of the form $[2^i, 2^{i+2}]$ on the left and right side. Thus, save possibly for the last two steps, the competitive ratio is given by

$$\left[\int_0^1 \frac{3+x}{1+x} d\,x + \sum_{i=0}^{D-2} \int_0^{3\cdot 2^i} \left(\frac{2(2^{i+2}-1)}{2^i+x} + 1\right)\, dx + + \int_0^{d-2^{D-1}} \left(\frac{2(2^{D+1}-1)}{2^{D-1}+x} + 1\right)\, dx + \right.$$
$$\left. + \int_0^{d-2^D} \left(\frac{2(2^{D+1}-1)+2d}{2^D+x} + 1\right) d\,x\right] \left(\frac{1}{2d-2}\right)$$
$$= \frac{-1+6\ \ln(2)(2^D-1)+4\ \ln(d)2^D-2\ \ln(d)+d-8\,D\ln(2)2^D+\ln(d)d-D\ln(2)d}{d-1}$$

where D is such that $2^D \leq d \leq 2^{D+1}$. Solving numerically, we obtain that the competitive ratio above is ≈ 5.27 as required. $\square$

Lemma 2.4 [24] *There exists a randomized strategy which is 9-competitive in the worst case, and* $1 + 3/\ln(2) \approx 5.328085$ *in the average case.*

The sequence of uniform distributions over intervals of increasing length centered over the origin is often used as an alternative natural "distribution" over the real line. However, it is important to keep in mind that the sequence of uniform distribution functions described above does not converge, as d goes to infinity, to a distribution function in any standard sense.

Now we consider the case in which a strategy "favours" one side of the search. As Theorem 2.4 shows, the average case is significantly better than 9-competitive. This points out the idea that perhaps favouring one side over the other in a search may result in a competitive ratio better than 9 when averaged over worst case left and right hiding positions.

The following theorem shows otherwise. That is, if a strategy "favours" one side of the search then it must pay a penalty on the cost of searches on the opposite side of at least the same amount by which it favoured the search on the former.

Theorem 2.5 *Let C_S^L (C_S^R) be the competitive ratio for finding a target point on the left (right) under a given strategy S. Then $(C_S^L + C_S^R)/2 \geq 9$.*

Proof. Let $\ell_i = -s_{2i}$, $r_i = s_{2i-1}$ for $i \geq 1$ denote the left and right turn points respectively. Let $\overline{L_k} = \sum_{i=1}^{k} \ell_i$ and $\overline{R_k} = \sum_{i=1}^{k} r_i$ be the distance traversed on the left and the right. Let L_k (R_k) be the competitive ratio after the kth step. That is

$$L_k = \frac{2\overline{L_k} + 2\overline{R_k}}{\ell_k} + 1 \quad \text{and} \quad R_k = \frac{2\overline{L_{k+1}} + 2\overline{R_k}}{r_k} + 1$$

Then $C_S^L = \sup_k\{L_k\}$ and $C_S^R = \sup_k\{R_k\}$.

If there exist k_1 and k_2 such that $L_{k_1} \geq 9$ and $R_{k_2} \geq 9$ then there is nothing to prove. Thus we can assume, without loss of generality, that $R_k < 9 - \eta$. If $L_k < 9 - \eta$ this would contradict Theorem 2.1, as we would have a better than 9 competitive strategy for the real line, thus $L_k = 9 + \epsilon$ for $\epsilon \geq 0$, and

$$\begin{aligned} L_k &= \frac{2\overline{L_k} + 2\overline{R_k}}{\ell_k} + 1 = 9 + \epsilon \qquad \Longrightarrow \quad 2\overline{L_k} + 2\overline{R_k} = (8 + \epsilon)(\ell_k). \\ L_k &= \frac{2\overline{L_k} + 2\overline{R_k}}{\ell_k} + 1 = \frac{2\overline{L_{k-1}} + 2\overline{R_{k-1}}}{\ell_k} + 3 + \frac{2r_k}{\ell_k}. \end{aligned}$$

The second equation above implies

$$r_k = (6 + \epsilon - \Omega_k)\,\ell_k/2 \qquad \text{where } \Omega_k = \frac{2\overline{L_{k-1}} + 2\overline{R_{k-1}}}{\ell_k}$$

Moreover $\Omega_{k+1} = (8+\epsilon)\,\ell_k/\ell_{k+1}$. It follows then that

$$\begin{aligned}
L_k + R_k &= 9 + \epsilon + \frac{2\overline{L_{k+1}} + 2\overline{R_k}}{r_k} + 1 \\
&= 10 + \epsilon + \frac{2\,\ell_{k+1} + 2\overline{L_k} + 2\overline{R_k}}{r_k} \\
&= 10 + \epsilon + \frac{2\,\ell_{k+1} + (8+\epsilon)\,\ell_k}{(6+\epsilon-\Omega_k)\,\ell_k/2} \\
&= 10 + \epsilon + \frac{(8+\epsilon)\,2}{6+\epsilon-\Omega_k} + \frac{4\,\ell_{k+1}}{(6+\epsilon-\Omega_k)\,\ell_k} \\
&= 10 + \epsilon + \frac{(8+\epsilon)\,2}{6+\epsilon-\Omega_k} + \frac{4\,(8+\epsilon)}{(6+\epsilon-\Omega_k)\,\Omega_{k+1}} \\
&\geq 10 + \frac{16}{6-\Omega_k} + \frac{32}{(6-\Omega_k)\,\Omega_{k+1}} \\
&= 10 + \frac{16\,(\Omega_{k+1}+2)}{(6-\Omega_k)\,\Omega_{k+1}}
\end{aligned}$$

Now, we have several cases:

1. $\Omega_{k+1} < 4/3$.

   (a) $\Omega_k > 1$. Then $L_k + R_k \geq 10 + \frac{16\cdot 10}{5\cdot 4} = 18$.

   (b) $\Omega_k \leq 1$. Then $L_{k-1} + R_{k-1} \geq 10 + \frac{16\cdot 3}{6} = 18$.

2. $\Omega_{k+1} \geq 4/3$.

   (a) There exists j such that $\Omega_j{+}1 < 4/3$, in which case, $\max\{L_j{+}R_j, L_{j-1}{+}R_{j-1}\} \geq 18$ follows from the case above.

   (b) There exists j such that $\Omega_j > 4 - 4/\Omega_{j+1}$. Thus

$$\begin{aligned}
L_j + R_j &\geq 10 + \frac{16\,(\Omega_{j+1}+2)}{(6-4+4/\Omega_{j+1})\,\Omega_{j+1}} \\
&= 10 + \frac{16\,(\Omega_{j+1}+2)}{(2\,\Omega_{j+1}+4)} \quad = \quad 18.
\end{aligned}$$

(c) For all j, $4/3 \leq \Omega_j \leq 4 - 4/\Omega_{j+1} < 4$. Let $f(x) = 4 - 4/x$. Note that if $x < y$ then $f(x) < f(y)$. Inductively, we have $\Omega_j \leq f^{(n)}(\Omega_{j+n}) < f^{(n)}(4)$. For any initial value $\Omega_{j+n} < 2$, the sequence $f^{(n)}(\Omega_{j+n})$ goes to $-\infty$ which is a contradiction as $4/3 \leq \Omega_j < f^{(n)}(\Omega_{j+n})$. This implies then that for all sufficiently large j, $\Omega_j \geq 2 - \epsilon'$.

Now, for all initial values $\Omega_{j+n} > 2$ the sequence $f^{(n)}(\Omega_{j+n})$ converges to the fixed point $f(2) = 2$. Thus, for all sufficiently large j, we also have that $\Omega_j \leq 2 + \epsilon'$. Implying

$$\begin{aligned} L_j + R_j &\geq 10 + \frac{16\,(4+\epsilon')}{(4+\epsilon')\,(2+\epsilon')} \\ &\geq 18 - \frac{8\,\epsilon'}{2+\epsilon'} \\ &\geq 18 - 4\,\epsilon' \qquad \text{since } \epsilon > 0. \end{aligned}$$

As ϵ' can be arbitrarily small, $\sup_j\{L_j + R_j\} \geq 18$.

In all cases we have shown that $\sup_j\{L_j + R_j\} \geq 18$, and thus

$$\sup_j \left\{ \frac{L_j + R_j}{2} \right\} \geq 9$$

as required. $\square$

Now let us consider a kinetic model in which the robot cannot suddenly stop and change direction. In this case we assume that the robot is moving at a certain speed and acceleration ratio.

2.5 Kinetic Models

Consider a robot traveling at a speed $|v| = 1$. The robot decelerates at a constant rate when it approaches the turn point. In this case the competitive ratio is given by the distance of the target to the origin divided by the time taken by the robot to reach the target.

Theorem 2.6 *A target on the real line can be found at a kinetic competitive ratio of 9 which is optimal.*

Proof. Let us assume first that the robot does not stop upon reaching the target; rather, it suffices to overrun it. In this case, the optimal trajectory takes time proportional to the distance to the target, while the search takes longer, as the robot must brake at each turn point and change direction. In this case then, no strategy can search at a competitive ratio better than 9, as the same strategy would have an even better competitive ratio for the non-kinetic model contradicting Theorem 2.1.

Let S be the strategy defined by $s_1 = 3$, $s_2 = 7$, and $s_n = 2^{n+1} - 1/2$ for $n \geq 3$. The robot traverses each segment in time $2^{n-2}+1$, as the robot requires one unit of time and half a unit of distance to come to a halt. Similarly, the robot takes time $2^{n-2}+1$ to return past the origin, as the robot accelerates at a unit constant rate.

The competitive ratio of this strategy is

$$\begin{aligned} c_S = 1 + \sup_{n \geq 3} \left\{ 2 \sum_{i=1}^{n} \frac{2^{i-2}+1}{2^{n-3}+1/2} \right\} &= 1 + \sup_{n \geq 3} \left\{ 4 \frac{2^{n-1}+1+n}{2^{n-2}+1} \right\} \\ &= 1 + \sup_{n \geq 3} \left\{ 8 + 4 \frac{n-1}{2^{n-2}+1} \right\} \\ &\leq 9. \end{aligned}$$

Now if we consider the case in which the robot must stop and "pick up" the target, we see that, for each point, the competitive ratio is improved while the final competitive ratio (i.e. the supremum of each competitive ratio) is still 9.

This is optimal as well, because it follows from the proof of Theorem 2.1 that the competitive ratio must be larger than 9 infinitely often, even for strategies which have competitive ratios worse than 9 in a vicinity of the origin. Thus, as the braking time is of no relevance for large n, we know that the competitive ratio must be at least 9 as well. □

2.6 Searches on Several Lines at Once

Above we studied some interesting variations of searches on the real line, such as bounded distance, integer coordinates for the target, and kinetic considerations. Another important variation is given by the search of n rays, for $n \geq 1$, where the robot is located on the apex of the rays. This problem is sometimes referred to in the literature as the "cow-path" problem in which a cow reaches an m-fork on the road, with one (and only one) of the paths leading to a green pasture [36].

The case $n = 1$ is trivial. In this case, if we assume that the robot is **not** located in the apex, but at a distance d from it, we obtain, from Theorem 5 that the competitive ratio of the search goes to 9 as d grows.

The case $n = 2$ reduces to a search on the real line, which we know to be 9-competitive.

For $n \geq 3$, the optimal strategy is no longer readily apparent. In fact, we have

Theorem 2.7 **[24]** *A robot can search m rays for a target located at an arbitrary distance from the origin at a*

$$1 + 2\frac{m^m}{(m-1)^{m-1}}$$

competitive ratio, which is optimal.

As the proof for the lower bound is cumbersome and unenlightening [24], we will limit ourselves to pointing out that an extension of the doubling strategy as follows achieves this ratio.

Consider a strategy based on the sequence $S = \{s_1, s_2, \ldots\}$ where $s_i = \left(\frac{m}{m-1}\right)^i$, consisting on traversing, at step i, a distance s_i and back on the ray $j \equiv i \bmod m$ for $j \in [1..m]$. Notice as well that for $m = 2$ we obtain a 9-competitive ratio, which is consistent with Theorem 2.1 for the real line (two semi-infinite rays).

For randomized strategies we have the following theorem, first proposed by Gal [24] and independently rediscovered by Kao et al. [36].

Theorem 2.8 [24] *Let $S = \{s_i\}_{i \in N}$ be a randomized strategy for searching m rays such that $s_i = a^{i+u}$, where u is random value uniformly distributed in $[0, m)$. Then S has a competitive ratio of*

$$1 + \frac{2(a^m + 1)}{m(a-1)\ln(a)}$$

where the value of a is suitably chosen to minimize the expression above.

Chapter 3

Searching in the Plane

There are essentially two types of searches on the plane. One is a search for a target that is identifiable on contact; the second is a search for a visually identifiable target that is hiding behind some obstacles in either a bounded or unbounded domain in the plane.

In its full generality, the problem of searching consists of a robot searching for a target on a *scene* on the Euclidean plane.

Definition 3.1 [11] *A* **scene** $\mathcal{S}$ *consists of a start point* s *and a target* t*, together with a set of opaque, impenetrable, non-overlapping obstacles, none of which contain* s *or* t*.*

The target is usually a line, a point, or a polygon. Obstacles are often assumed to be polygons, possibly convex.

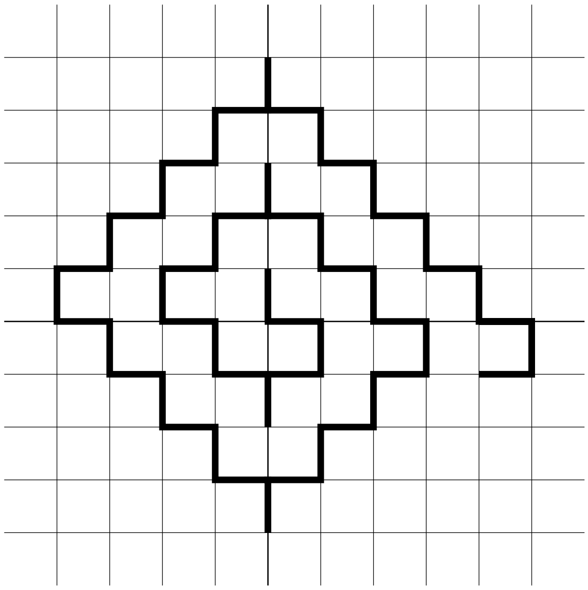

Figure 3.1: A strategy for searching lattices.

3.1 Contact Searches

It is not possible for a zero dimensional robot to search for a point in a two dimensional plane, even if the target is located at a bounded distance. Thus, we have to restrict the possible locations of the target, say, to an integer lattice, or to extend the search capabilities of the robot in such a way that it "touches" an area around its current position.

The first case is that of a robot searching for a point on the integer lattice.

Theorem 3.1 [1] *A robot moving on an integer lattice traverses, in the worst case, at least* $2n^2+4n+1$ *steps to find a point at some unknown distance* n*. Furthermore,* $2n^2+5n+2$ *steps are always sufficient. The competitive ratio is then* $2n+4+K$ *for* $1/n < K \leq 1+2/n$*.*

Proof. Consider the strategy illustrated in Figure 3.1. The robot starts at the origin and moves to the point $(0,1)$. At this point it moves south on a southwest staircase 3 links long, then it moves west on a southwest staircase 2 links long. From there the general sequence is as follows:

Starting from the previous position,

1. traverse $5+4i$ links starting northwards in a northwest staircase,
2. traverse $4i$ links starting eastwards in an northeast staircase,
3. traverse $3+4i$ links starting southwards in an southeast staircase,
4. traverse $2+4i$ links starting westwards in a southwest staircase, and
5. increment i, go to step 1.

Notice that there are $4n$ points at L_1 distance n from the origin. These are the integer points on the ball of radius n in the L_1 metric. We claim that, from the moment the last point at distance $n-1$ was visited to the time when the last point at distance n was visited, the robot takes $4n+3$ steps, except on the initial case when nine steps are required. This is so, as it takes two consecutive "staircases" for the robot to move from the last point in the ball of distance $n-1$ to the last point on the n-ball. Each of these two staircases is four units longer than the one just below on the lattice. The staircases below correspond to points at distance $n-2$ which implies that the length of the staircase for points at distance n is $4(n-2)+3+8=4n+3$. Thus the total distance traversed is $2+\sum_{i=1}^{n} 4i+3 = 2n^2+5n+2$ as claimed.

To prove the lower bound, we first note that an n-ball contains $2n^2+2n+1$ points. As it takes at least one step to visit each point, this immediately provides a lower bound within $2n$ of the one claimed. This last additive factor comes from the fact that to move from a point at distance n to another point at the same distance the robot must visit a point at either distance $n+1$ or $n-1$. Intuitively, visiting points at distance $n+1$ is "extra work" which worsens the competitive ratio. But if the robot visits only points at distance $n-1$, then the next time around, when it visits points at distance $n+1$, it must either revisit points at distance n or visit "extra" points at distance $n+2$, for a total waste of $4n$.

Formally, let $P_{+1}(n)$ be the number of points at distance $n+1$ which were visited **before** the robot visits the last point at distance n. Let $C(n)$ be the distance traversed before this last point is reached. Then, from the discussion above it follows:

$$C(n) \geq 2n^2 + 2n + 1 + P_{+1}(n)$$

Now, either $P_{+1}(n) \geq 2n$ and there is nothing to show, or $P_{+1}(n) \leq 2n$. In the latter case, we know that, after visiting the last point at distance n there remain $4(n+1) - P_{+1}(n)$ points to be visited at distance $n+1$. Between every two of these points, the robot either revisits at least one point at distance of n or visits at least one point at distance of $n+2$. In both of these cases, there are at least two extra steps per point which are not included in the count of $C(n)$. Thus

$$\begin{aligned}
C(n+1) &\geq C(n) + 2(4(n+1) - P_{+1}(n)) - 1 \\
&\geq [2n^2 + 2n + 1 + 2(4(n+1) - 1)] - P_{+1}(n) \\
&\geq [2(n+1)^2 + 6n + 6] - 2n \\
&= 2(n+1)^2 + 4(n+1) + 2
\end{aligned}$$

as required.

□

It is also interesting to note that search patterns might change significantly if more than one searcher is allowed. This is often the case for actual searches such as sea rescues. Baeza et al. [2] have proposed some strategies for parallel searching. Some of these techniques have been previously studied [63].

3.2 Searches Amongst Obstacles

An instance of a search with obstacles in the plane is defined by an unbounded planar region with impenetrable obstacles, a start point s for the robot and a target point t of the hider, such that the distance $d(s,t) = n$.

Theorem 3.2 [11, 57] *There is no strategy that achieves a constant competitive ratio for searches among obstacles, even if some or all of the following hold:*

- *the location of the target is known to the robot,*
- *the shape of the obstacles is limited to be nonintersecting rectangles with sides parallel to the axis,*
- *the robot learns the shape of an entire obstacle by observing a* **single** *point of the obstacle.*

In fact, the competitive ratio is $\Omega(\sqrt{n})$*, and if all of the above hold is* $\Theta(\sqrt{n})$*.*

Theorem 3.3 [57] *The competitive ratio for searches with known destination in the case of square obstacles of unit size is at least 3/2.*

Theorem 3.4 [24] *For arbitrary obstacles, the competitive ratio in searches for a target of unknown location is* $\Theta(n)$*.*

If the aspect ratio of the longest and shortest side of an obstacle is bounded there are better bounds possible. Also, if the obstacles are relatively small as compared to the distance to the target, the competitive ratio can also be improved.

For the randomized case we have,

Theorem 3.5 [8] *There exists a* $(n^{\frac{4}{9}} \log n)$*-competitive randomized strategy for searching for a target of known location on a scene with rectangular objects.*

Chapter 4

Searching Inside a Simple Polygon

In many cases, searches often occur within bounded regions of irregular shape. Polygons effectively represent these type of environments. In this chapter we consider the problem of a robot aided by vision searching inside a polygon, particularly the case of star polygons.

Definition 4.1 [58] *A* **simple polygon** *is a finite sequence of co-planar line segments* $L_1, L_2, \ldots, L_n$ *called* **edges** *such that every segment extreme is shared by exactly two edges, no subset of edges has the same property, and there is no other intersection among them.*

In general, for simple polygons is not hard to see that a constant competitive ratio is not achievable. The polygon of Figure 4 shows an example of a polygon that is $\Omega(n)$-competitive, where n is the number of vertices. In this case, if an adversary places the target into the spike that is last explored by the strategy, then the robot will travel a distance of $\Omega(nd)$ while a shortest path has length at most $d + \epsilon$. Hence the competitive ratio is $\Omega(n)$ for large d and small ϵ.

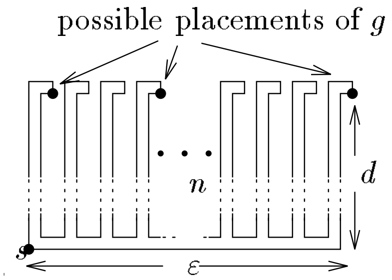

Figure 4.1: A lower bound for searching simple polygons.

As we are interested in relatively low competitive ratios, researchers have proposed measures of "complexity" of a polygon for the purposes of on-line searching. These measures are such that, while being relatively natural, they divide polygons into classes in which constant competitive ratios are achieved.

Among the obvious ones are searches inside the class of convex polygons, which is a trivial task, as the target is visible from the starting point.

A natural question then is to find a more complex class of polygons which the robot might still search at a constant competitive ratio. Since the target might be hiding anywhere inside the polygon, a natural choice is to explore the class of polygons which can be seen on its entirety from a single point.

4.1 Searching for the Kernel

Definition 4.2 [58] *A simple polygon P is a* **star polygon** *if there exists a point z not external to P, such that for all points p of P the segment $\overline{zp}$ lies entirely within P.*

Star polygons are often referred to as star-**shaped** polygons [58], however as star polygons are, in general, not star-shaped (see Figure 4.9) we have chosen the equally common but shorter name of star polygons.

Definition 4.3 [58] *The locus of all points inside a simple polygon P such that the entire polygon is visible from any of them is the* **kernel** *of P.*

Thus for the robot to see the entire polygon from a single point, it must reach the kernel of P.

The kernel of a n vertex polygon can be computed off-line in $\theta(n)$ time [42].

Definition 4.4 *The distance from a point p in the plane to a set Q is defined as $d(p, Q) = \inf_{q \in Q} d(p, q)$, where $d(p, q)$ is the standard Euclidean distance between two points in the plane.*

Icking and Klein studied the problem of on-line kernel searching on a star polygon. In this case, the competitive ratio is given by the ratio of the length traversed by the robot from the starting point to a kernel point and the optimal distance, which is the the distance from the starting point to the kernel set.

As it turns out, a careful study of searches for the kernel suggests a natural classification of some types of polygons in which point target searches have a constant competitive ratio.

As well, several of the techniques used in this problem can be applied for point searches.

To study kernel searches we need to introduce some new concepts.

Definition 4.5 *At any given point p the* **visibility polygon** *of the robot at p, denoted $V_P(p)$, is the subset of P visible to the robot.*

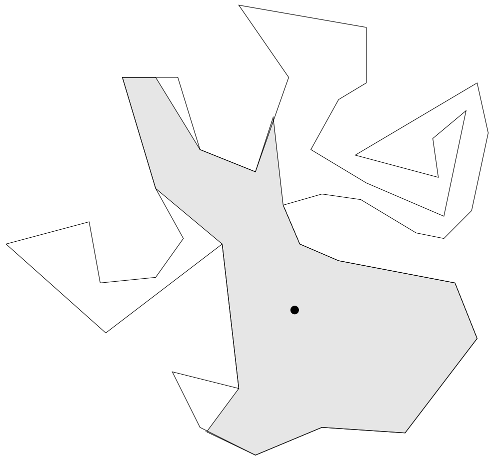

Figure 4.2: Visibility polygon.

Note that we have defined a subset of P as the visibility **polygon**. The following observation justifies the choice of terminology.

Observation 4.1 [58] *Given any polygon P and an interior point p, the visibility polygon $V_P(p)$ is a polygon (in fact, it is a star polygon).*

Definition 4.6 *The connected components of the difference $P - V_P(p)$ are called* **pockets***. The boundary of a pocket is made of some polygon edges and some line segments not belonging to the boundary of P. The pocket's edges which are not polygon edges are called* **windows***.*

Definition 4.7 *A* **pocket edge** *is a ray emanating from the robot's current position and containing a window.*

Definition 4.8 *Pocket edges pass through at least one reflex vertex v_i of the polygon, which is also an end point of the window associated with the pocket edge. These type of reflex vertices are called* **extreme entrance points***.*

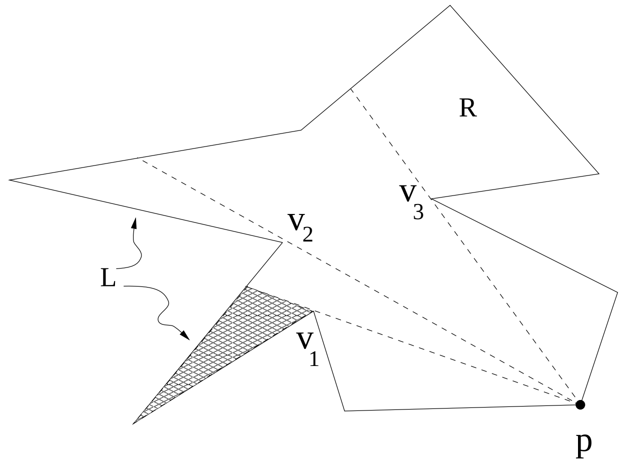

Figure 4.3: Left and right pockets.

Definition 4.9 *A pocket is said to be a* **left pocket** *(right pocket) if it lies to the left (right) of the oriented window edge under the orientation induced by the* $\overrightarrow{pv_i}$ *ray containing it, where* v_i *is an extreme entrance point of the pocket edge.*

Definition 4.10 *A pocket edge is said to be a* **left pocket edge** *(right pocket edge) if it defines a left (right) pocket.*

Observation 4.2 *The kernel lies to the right (left) of a left (right) pocket edge.*

This is so, as any point inside a pocket cannot see, by definition, the current position of the robot, and thus cannot possibly belong to the kernel of the polygon. For example, in the polygon of Figure 4.3, the kernel, if it exists, must lie to the right of $\overrightarrow{pv_1}$ and $\overrightarrow{pv_2}$ and to the left of $\overrightarrow{pv_3}$.

Observation 4.3 *The kernel must lie to the right of all left pocket edges and to the left of all right pocket edges.*

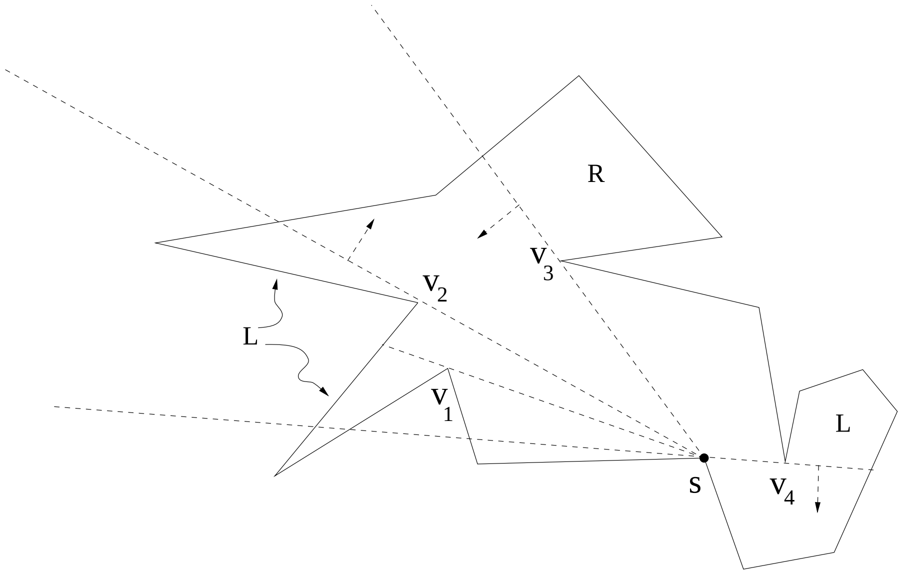

Figure 4.4: Alternating left and right pockets.

This also implies that, for star polygons, starting from a left pocket and moving clockwise, all left pocket edges appear consecutively; at some point, the first right pocket edge is seen and from then onwards all pocket edges are right pocket edges, until it reaches full circle back to the sequence of left pocket edges. This is so as the extension of each pocket defines a half plane in which the kernel of the polygon must lie. If the pockets alternate between left and right such intersection is necessarily empty (see Figure 4.4).

If the robot is initially located on a point s on the boundary of the polygon, the robot sees all left pocket edges by starting from the edge on which s lies, and scanning on the clockwise direction the interior of the polygon. At some point, a right pocket edge is seen and from then onwards all pocket edges are right pocket edges and eventually the robot reaches the edge containing s again, which completes the scanning process.

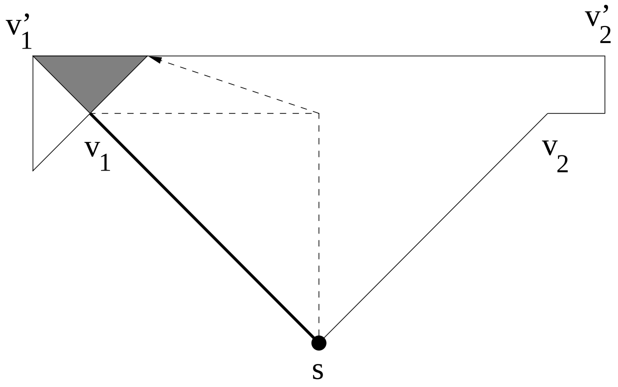

Figure 4.5: Kernel of a polygon.

Definition 4.11 *Let v^l denote the extreme entrance point to the last left pocket edge and v^r the extreme entrance point to the next right pocket edge in the clockwise direction.*

Thus a natural search strategy for the robot is to move in the angular sector $\angle v^l p v^r$, which is known to contain the kernel and towards the pockets until it either sees a new pocket in which case it updates the candidate region, or until it sees the interior of all pockets, in which case the robot is in the kernel.

With these observations at hand, we can now study the efficiency of kernel searches.

As shown in Figure 4.5, the problem of on-line search for the kernel of a polygon is at least $\sqrt{2}$ competitive [34]. In this case, a robot located at s observes two pockets into which it cannot see. At this time the whole interior of the triangle $\triangle s v_1' v_2'$ is a candidate for the kernel. As the robot moves upwards, the potential region is the intersection of all triangles $\triangle p_t v_1' v_2'$, where p_t is a parameterization of the path followed by the robot thus far. It is not until the robot reaches the line $\overline{v_1 v_2}$ when it can determine which of the two pockets, if any, is triangular, and thus

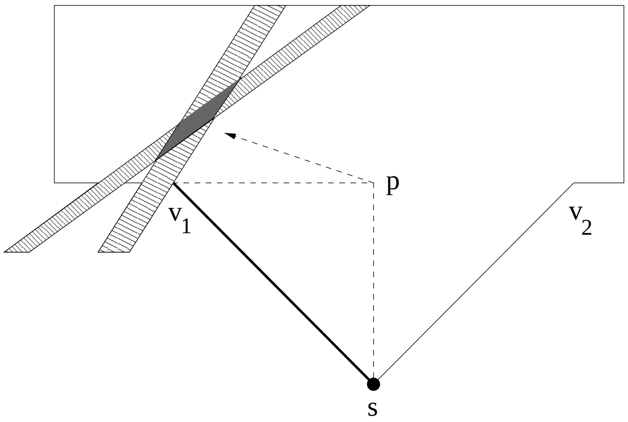

Figure 4.6: Polygon with two beams.

determine to which side move.

Now, using an adversary argument, the $\sqrt{2}$ lower bound follows. If the robot does not reach the line $\overline{v_1v_2}$ on the midpoint, the adversary places the triangular pocket on the opposite side to the position of the robot. If the robot reaches that line in the midpoint the adversary locates the kernel on either pocket.

Notice that this behaviour of the adversary is consistent with the information acquired by the robot. Furthermore, if the angle $\angle v_1 s v_2 = \pi/2$, it is easy to see that the competitive ratio for the robot tends to $\sqrt{2}$ as $\overline{v_1'v_2'}$ becomes closer to $\overline{v_1v_2}$.

The next theorem shows that kernel searches in the best case are worse than $\sqrt{2}$-competitive. This result stands out against several other lower bounds for searching in simple domains, for which it seems that a robot can find an optimal path on-line for the L_1 metric (see, for example Theorem 5.1 in the next chapter).

Definition 4.12 *The visibility region of a subset B of a polygons is the set of all points in the polygon which see all points in B.*

Definition 4.13 *Given a point p where the robot is located and a pocket B with respect to that point, the* **beam** *of the pocket is the visibility region of B.*

Notice that if the pocket is a trapezoid, the visibility region resembles a search light beam (see Figure 4.6).

Observation 4.4 *The kernel lies in the intersection of all beams.*

Theorem 4.1 *Searching for the kernel of a polygon is at least* $1/2 + 3/8\sqrt{2} + (2 + \sqrt{2})/8\sqrt{10\sqrt{2} - 13} \approx 1.486429521$*-competitive.*

Proof. Consider the polygon of Figure 4.6. Notice that, as before, the robot must reach the the line segment $\overline{v_1 v_2}$ before it reaches the kernel. As well, the robot must reach $\overline{v_1 v_2}$ at its midpoint p, as otherwise the following construction can be made on the opposite side and it follows from the triangle inequality that the competitive ratio would only worsen. Again, from p it is not yet clear where the kernel is located. In fact, depending upon the specific angle and location of the pockets, the **beams** might specify a small kernel located anywhere in the visibility polygon region of s which is above $\overline{v_1 v_2}$.

We use an adversary argument. After the robot reaches p the adversary closes one side, and selects two candidate kernels, illustrated by the large dots in Figure 4.7, such that one is next to v_1 the other right above the midpoint, and the line joining them is at a $\pi/4$ angle to the horizontal. This can be achieved by locating a beam A along the line joining the two candidate regions, and a second one, B, nearly parallel and to the right of A (see Figure 4.8). The intersection of both beams defines the kernel of visibility.

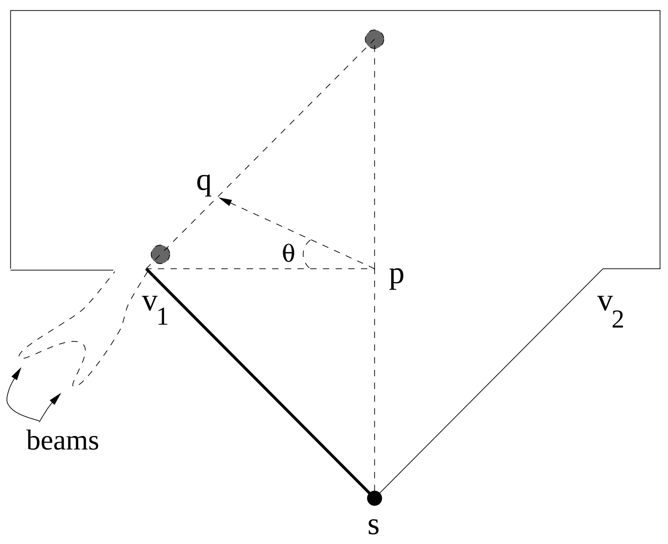

Figure 4.7: Lower bound configuration.

At this point, we assume that the robot learns of this decision and thus can restrict itself, to its benefit, to determining which of the two regions is the kernel.

In this case, the robot cannot decide which of the candidates is the kernel before it reaches at least one of A or B. As the beams become progressively thinner, the robot reaches either beam at an ϵ distance of the $\pi/4$ line joining the two candidate regions (that is, the right edge of the A beam).

Assume this happens at a point q located, as indicated in the previous paragraph, arbitrarily close to the $\pi/4$ line. Let θ be the angle given by $\angle v_1pq$. Without loss of generality, let the distance $d(s, v_1) = \sqrt{2}$. Thus the competitive ratio for the kernel on the left side is given by (detailed computations for these expressions can be found in page 92)

$$\frac{\sqrt{2}\sin(\theta) + \sqrt{2}\cos(\theta) + \sqrt{2} + 2\ \sin(\theta)}{2\ \sin(\theta) + 2\ \cos(\theta)}$$

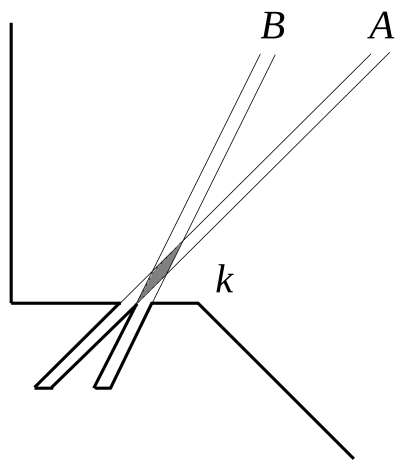

Figure 4.8: Progressively thinner beams.

and on the right side

$$\frac{\sin(\theta) + \cos(\theta) + 1 + \sqrt{2}\cos(\theta)}{2\,\sin(\theta) + 2\,\cos(\theta)}$$

As the competitive ratio is the maximum of both quantities above, the robot selects θ such that the competitive ratio on either side is the same. Solving the equation we obtain,

$$\theta = \arctan\left(1/4 + 1/8\sqrt{2}\left(1 - \sqrt{10\sqrt{2} - 13}\right)\right).$$

For this value, the competitive ratio is

$$(2+\sqrt{2})/8\sqrt{10\sqrt{2} - 13} + 1/2 + 3/8\sqrt{2} \approx 1.486429521$$

as required. □

The best known search strategy for the kernel, is by Icking and Klein [34].

Theorem 4.2 **[34]** *There exists a strategy for searching for the kernel which is no more than* $\sqrt{4 + (2+\pi)^2} \approx 5.5168$*-competitive.*

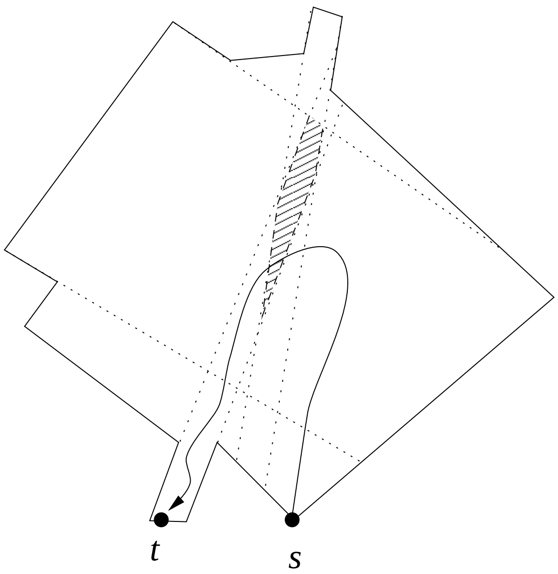

Figure 4.9: Searching for a target via the kernel.

4.2 Target Searching in Star Polygons

As we shall see in this section, there many similarities between kernel searches and target searching. Note that, when searching for a target, it might not be an efficient strategy to reach the kernel first and from there move to the target as illustrated in Figure 4.9. However, if the target is located on the "other side" of the polygon, then it is possible to find it at a constant competitive ratio. More formally,

Definition 4.14 *A point p in a polygon P is said to be* **on the other side** *of a set $B \subseteq P$ with respect to a point s if the normal through q to the line segment joining a point $q \in B$ and s divides the plane in two regions, one containing p, the other s. Equivalently p is on the other side of B with respect to s if $d(s,p) \geq \max_{q \in B} d(p,q)$.*

Lemma 4.1 *There exists a $\sqrt{4+(2+\pi)^2}$-competitive strategy for searching for a target on the other side of the kernel with respect to the starting position.*

Proof. A robot can reach the kernel at a $\sqrt{4+(2+\pi)^2}$-competitive ratio using the strategy from Theorem 4.2 [34]. When the robot enters the visibility polygon of the target t it sees t and can move directly to it, Since the kernel of the polygon is contained in the visibility polygon of t, the robot is certain to reach the visibility polygon of t, by walking into the kernel. In the worst case, the robot reaches the kernel and the visibility polygon of t simultaneously.

Without loss of generality, let the start point s be located at $(0, -1)$ and the kernel point reached at $(0, 0)$. As the target is located on the other side of s it must lie on the opposite side of the x-axis. That is, the target is located at a point (x, y) where $x \in R, y \in R^+$. The competitive ratio is given by:

$$\frac{\sqrt{4+(2+\pi)^2}+\sqrt{x^2+y^2}}{\sqrt{x^2+(y+1)^2}}$$

Differentiating with respect to x and y one can see that the expression above is maximized in the domain of interest when $x = y = 0$, which gives the desired upper bound. □

Searching for a target of unrestricted location inside a star polygon is a provably harder problem. The next two theorems give upper and lower bounds for this problem.

Theorem 4.3 *There exists a 17-competitive strategy for searching for a target inside a star polygon.*

We introduce a definition and an observation which will be of use in the proof of the theorem.

Consider the set of pocket edges seen by the robot from the starting position. We extend this set as follows.

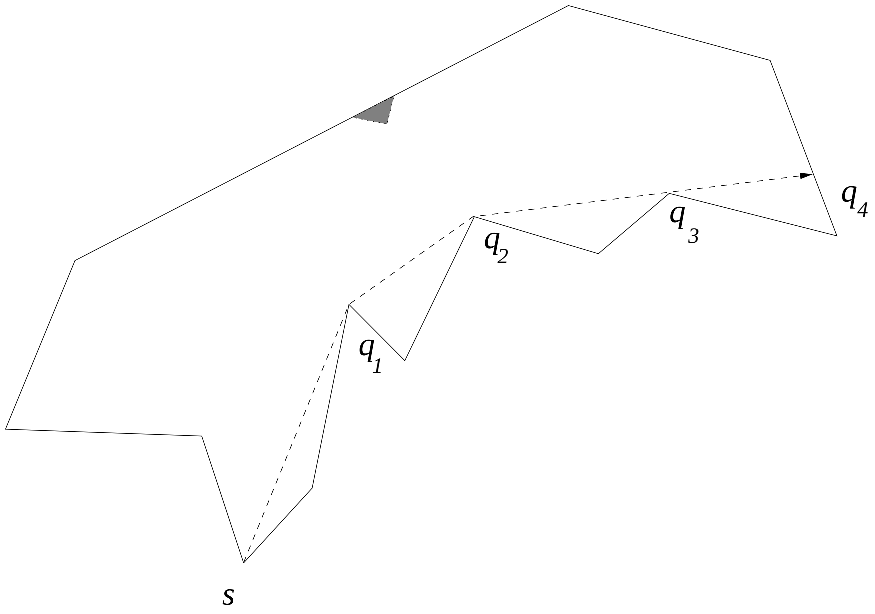

Figure 4.10: An extended pocket edge.

Definition 4.15 *Given a polygon P, an* **extended pocket edge** *from a point s is a concave polygonal chain $q_0, q_1, q_2, \ldots, q_k$ such that $q_0 = s$, and each of q_i is a vertex of P, save possibly for q_k. Furthermore q_{k-2}, q_{k-1} and q_k are collinear and must form a pocket edge with $\overline{q_{k-1}q_k}$ as associated window. If $\overline{q_{k-2}q_k}$ is a left (right) pocket edge, then each of $\angle q_{i-1}q_iq_{i+1}$ is a counterclockwise (clockwise) reflex angle (see Figure 4.10).*

Definition 4.16 *A subset A of a polygon P is said to be* **weakly visible** *from a subset $B \subseteq P$ if for any point $a \in A$ there is at least one point $b \in B$ such that the line segment $\overline{ab}$ is completely contained in the polygon.*

Observation 4.5 *Any chord splits a star polygon in two parts, one of which is weakly visible from the chord, the other containing at least one point in the kernel*

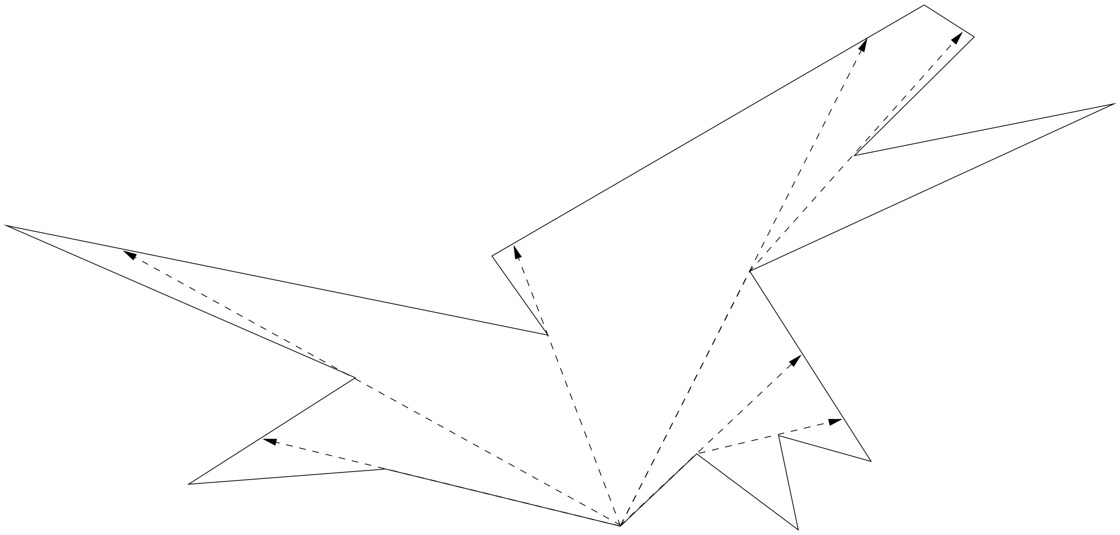

Figure 4.11: Inefficient searches through the kernel.

Proof. Clearly a chord splits a polygon in two. Further, a given point in the kernel must be contained in one of the two parts. As this point is in the kernel, the entire opposite part of the star polygon can be seen from it. But any line joining a the point on the kernel and a point on the opposite part must intersect the chord. This implies that the chord weakly sees all points on the opposite side as well. □

Proof. [Theorem 4.3] Let $\mathcal{F}$ denote the set of all extended pocket edges starting from s. From the definition it follows that, in general, the robot may not see all of $\mathcal{F}$ from s (see for example the star polygon of Figure 4.11). The robot thus uses a strategy that starts with a subset F of $\mathcal{F}$. This set is enlarged as the robot sees new pocket edges. Given an extended pocket edge E, let l_E denote the last point in the chain, and p_E denote the penultimate point of E.

Let $D \in \{left, right\}$ be one of the directions. If $D = right$ then $\neg D = left$ and vice versa.

Target Searching on Star Polygons

1. Let F denote the set of extended pocket edges currently seen but not explored. Initially F contains only standard pocket edges.

2. Let $\{p_{E_i}, l_{E_i}\}$ denote the set of last and second to last points of for all edges $E_i \in F$. Let p_E be the closest extreme point to the starting position, say at a distance d.

3. Set $D \leftarrow right$.

4. The robot traverses d units on E starting from s.

5. New pocket edges seen in this trajectory are added to F as extended pocket edges starting from s.

6. Remove from $\mathcal{F}$ all extended D pocket edges to the D side of the line segment $\overline{sp_E}$, including E if l_E is reached.

7. Let $d \leftarrow 2d$. The robot changes direction $D \leftarrow \neg D$, and moves back to s. *Invariant: All pockets at a distance of $d/4$ or less on the D side have been explored.*

8. If $D = left$ $(right)$, let p_E be the rightmost (leftmost) extreme point on a $left$ (right) pocket edge such that the length of the extended pocket edge from s to p_E is less than d. If there is no such edge, select as E any $left$ (right) edge in F arbitrarily.

9. If F is not empty go to step 4.

10. Lastly, if at anytime the robot sees the target, it moves directly to it.

We must show that when the algorithm terminates, it must have seen the target, and that it did so at a 17-competitive ratio.

The correctness of the algorithm follows from Observation 4.5. As the robot visits extended pocket edges, it must eventually visit the leftmost right pocket edge and the rightmost left pocket edge. This is so as no other left (or right) edge could possibly mark the rightmost left pocket edge, except this edge itself (recall that left edges only mark edges which are to the left of themselves).

Once the robot has visited the extreme leftmost and rightmost pocket edges, it has explored the part to the left of the extreme left pocket edge, and to the right of the extreme right pocket edge. Further, the part of the polygon contained in between the two extreme pocket edges has no hidden regions as it contains no pockets. Thus the entire polygon has been explored, and the target must have been found.

We claim that this algorithm is 17-competitive. At the end of case 7, the invariant holds because if there was a, say, left pocket at a distance of less than $d/4$ it means it was part of the set F two steps before. Thus if it is unexplored it could not have been traversed as in step 8, which means there is another left pocket of length at most $d/4$ to the right of it which was explored instead. But exploring this second edge entails exploring the earlier edge as shown in Observation 4.5.

As a consequence of the invariant we have that, while searching at a distance of at most d, the target cannot be at a distance of less than $d/4$. Then the worst case competitive ratio occurs when the robot sees the target at a distance of $d/4 + \epsilon$, at the very last step of a d-unit long search. This means that the robot traversed, according to the doubling strategy, a distance of $9d/4 + 3d/4$ to reach the point from which it sees the target. In the worst case we have that the target might be located on the opposite side of the starting point, at a distance $d/4$ from it, which means that the robot has to move another $d + d/4$ units to reach the target for a

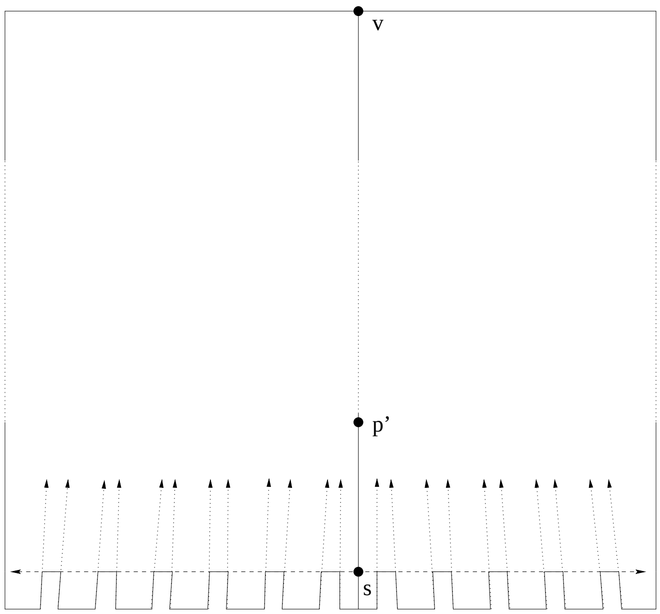

Figure 4.12: Lower bound for searching for a target.

total competitive ratio of

$$\mathcal{C} = \frac{12\,d/4 + d + d/4}{d/4} = 17$$

as required. $\square$

Theorem 4.4 *Any strategy for searching for a target inside a star polygon is at least 9-competitive.*

Proof. Consider the polygon of Figure 4.12. Let s be located at the origin. This polygon is made of $(n-1)2^{n-1}+1$ teeth with $(n-1)/2^{n-3}+4$ vertices attached to a rectangle of height n^2 and width $2n$. Teeth are equally spaced at a distance $1/2^n$, and of width $1/2^{n+1}$ save for the tooth containing s which is of width $2-1/2^n$. Each tooth defines a beam (see proof of Theorem 4.1). All beams intersect at the point $v=(0,n^2)$ which sees the entire interior of the polygon.

We claim that the robot must essentially do a doubling search on the teeth, in which case Theorem 2.2 for intervals gives a 9 lower bound. However, in this case there are several differences that must be considered. First, the movement of the robot is not restricted to a line; second, the lower bound is for searches on any point of the interval rather than on discrete positions. Thus the proof proceeds as follows: first we argue that any search strategy is sufficiently close to a search on the real line, and secondly we show that the bound for the continous case implies a similar bound for the discrete case.

For the robot to explore a tooth it must reach the beam above it. Number the beams symmetrically, and consecutively starting from the origin; thus beam b_i is at the same distance from the origin as beam $-b_i$.

The distance from s to the the base of the ith beam on either side is $d_i = 1 + (i-1)/2^n$. The distance from s to the closest point in the beam is (see Figure 4.13)

$$d(b_i, s) = \frac{d_i}{\sqrt{1 + (1 + (d_i)^2/n^4}} \geq \frac{d_i}{\sqrt{1 + 1/n^2}}$$

However the robot is not forced to move back to s after each search. The robot cannot reach a height past $9n$ as that alone would imply a competitive ratio above 9. Let p' be located at $(0, 9n)$. Thus we know that

$$d(b_i, p') \geq d_i \frac{n^2 - 9n}{\sqrt{n^4 + d_i^2}} \geq d_i \frac{n^2 - 9n}{\sqrt{n^4 + n^2}} = d_i \frac{n - 9}{\sqrt{n^2 + 1}}$$

Now, the order in which beams are visited can be denoted by the sequence $S = \{s_j\}_{1 \leq j \leq N}$ of the distances d_i from the origin to the base of those beams in which the robot changed direction (turn points).

Consider the beams associated to two consecutive terms in the sequence S above, say b_{k_i} and $b_{k_{i+1}}$. Without loss of generality, let us assume that b_{k_i} is on the left side

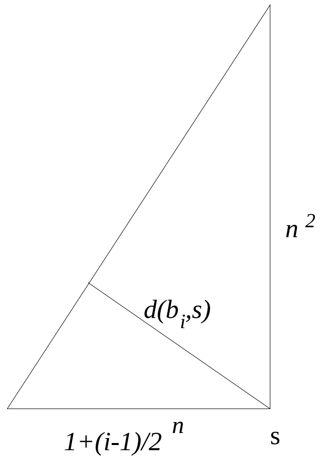

Figure 4.13: Distance to a beam.

and $b_{k_{i+1}}$ on the right side. Then, the distance traversed by the robot from beam b_{k_i} to beam $b_{k_{i+1}}$ is at least $d(q_i, q_{i+1}) \geq d(q_i, p_i) + d(p_i, q_{i+1})$, where q_j denotes the position of the robot in b_{k_j} for $j = \{i, i+1\}$, and p_i is the intersection of $\overline{q_i q_{i+1}}$ with the y-axis. Furthermore, $d(q_j, p_j) \geq d(b_{k_j}, p_j) \geq d(b_{k_j}, p')$.

Now we will show that the search strategy S, applied to a target hiding in a point at distance d_i on the real line is

$$\mathcal{C}_S^D = \sup_{1 \leq j \leq N} \left\{ 1 + 2 \frac{\sum_{i=1}^{j} |s_i|}{|s_{j-1}| + 1/2^n} \right\} \geq 9 - 25/\log_4 n.$$

Assume that, to the contrary, $\mathcal{C}_S^D < 9 - 25/\log_4 n$. We know from Theorem 2.2 that any sequence S visiting the interval $[-n, n]$ and searching for a target located in any interior point has a competitive ratio greater or equal to $9 - 24/\log_4 n$. Since S is such a sequence, we have then that

$$\mathcal{C}_S = 1 + 2 \frac{\sum_{i=1}^{k} |s_i|}{|s_{k-1}|} \geq 9 - 24/\log_4 n$$

for some k such that $1 \leq k \leq N$.

Now let $C_S(s_k)$ and $C_S^D(s_k)$ denote the competitive ratio of strategy S to find a target hiding at s_k for the real and discrete case respectively. Note that $\mathcal{C}_S = C_S(s_k) \geq 9 - 24/\log_4 n$ and that

$$C_S^D(s_k) = 1+2\frac{\sum_{i=1}^{k}|s_i|}{|s_{k-1}|+1/2^n} < 9-\frac{25}{\log_4 n} \Longrightarrow \sum_{i=1}^{k}|s_i| < \left(8-\frac{25}{\log_4 n}\right)\left(|s_{k-1}|+\frac{1}{2^n}\right).$$

Note that, similarly to the proof of Theorem 2.3, there is an additive factor of $1/2^n$ in the denominator, denoting the next possible position of the target on that side.

We claim that $0 \leq C_S(s_k) - C_S^D(s_k) < 1/\log_4 n$. Indeed,

$$\begin{aligned} C_S(s_k) - C_S^D(s_k) &= 1+2\frac{\sum_{i=1}^{k}|s_i|}{|s_{k-1}|} - 1 - 2\frac{\sum_{i=1}^{k}|s_i|}{|s_{k-1}|+1/2^n} \\ &= 2\sum_{i=1}^{k}|s_i|\left[\frac{1}{|s_{k-1}|} - \frac{1}{|s_{k-1}|+1/2^n}\right] \\ &< \left(8-\frac{25}{\log_4 n}\right)\frac{1}{2^{n-1}|s_{k-1}|} \\ &\leq \frac{1}{2^{n-3}|s_{k-1}|} \leq \frac{1}{2^{n-3}} \leq \frac{1}{\log_4 n} \qquad \text{for } n > 3 \end{aligned}$$

as claimed. Thus, as $C_S(s_k) \geq 9 - 24/\log_4 n$ it follows that $\mathcal{C}_S^D \geq C_S^D(s_k) > 9 - 25/\log_4 n$, which is a contradiction.

Thus it follows that $\mathcal{C}_S^D \geq 9 - 25/\log_4 n$. Now, we know that the robot traversed, for each s_j a distance $d(q_j, p_j)$ which is at least $|s_j|\frac{n-9}{\sqrt{n^2+1}}$, for a total competitive ratio of at least

$$1 + 2\sup_{k\in Z}\left\{\frac{n-9}{\sqrt{n^2+1}}C_S^D(s_k)\right\} \geq 1 + 2\sup_{k\in Z}\left\{\left(\frac{n-9}{\sqrt{n^2+1}}\right)\left(8-\frac{25}{\log_4 n}\right)\right\}$$

Thus the value above is a lower bound for the competitive ratio of the robot searching a polygon. Now, as the construction of the polygon of Figure 4.12 is valid for any n, we have that, in the limit, the competitive ratio is bounded by

$$\lim_{n\to\infty} 1 + \left(\frac{n-9}{\sqrt{n^2+1}}\right)\left(8-\frac{25}{\log_4 n}\right) = 9$$

as claimed. □

Chapter 5

Searching Orthogonal Street Polygons

One can relax the conditions of star polygons and still search at a constant competitive ratio. The robot does not need to see the whole polygon from a single point to efficiently find the target. It suffices for the polygon to have a trajectory that the robot may traverse efficiently and from which it can see the entire polygon.

A fruitful and interesting class of polygons for which there are constant competitive ratio strategies is given by street polygons and related families.

Street polygons can be used to model searches along a non-convex corridor. Orthogonal polygons are often used to represent environments in which most objects are lined up against the wall, and the configuration might change on a daily basis[1].

[1] Informally, these types of scenarios are sometimes referred in the literature as "offices".

5.1 Street Polygons

The next class of polygons we consider are those polygons containing a "direct" trajectory from the start point s to the target point t such that the complete polygon is visible from it.

As the concept of "direct" is somewhat vague, a better definition is given by

Definition 5.1 [37] *Let P be a simple polygon with two distinguished vertices, s and t, and let L and R denote the clockwise and counterclockwise, resp., oriented boundary chains leading from s to t. If L and R are mutually weakly visible, i.e. if each point of L sees at least one point of R and vice versa, then (P, s, t) is called a* **street**.

This class, first introduced by Klein in 1990 [37] has proven to be quite natural and rich. We will devote the rest of the chapter to the study of this class.

Note that a given polygon might or might not be a street depending upon the selection of the pair (s, t). The polygon in Figure 5.1 is a street for the pair (s, t_2) but not for the pair (s, t_1).

Street polygons can be recognized off-line in linear time [18]. In fact, given a polygon P it is possible to determine in linear time all pairs (s, t), if any, such that (P, s, t) is a street.

An equivalent formulation is that a triple (P, s, t) is a street polygon if the entire interior of the polygon is visible from any simple trajectory Γ from s to t. This is so as a trajectory inside P divides the polygon in two parts, one containing L the other R, thus any line joining a point from L to R must intersect Γ.

Interestingly, Definition 4.8 and Observation 4.3 hold for street polygons as well, in the sense that the shortest path to the target lies to the left of all right pockets

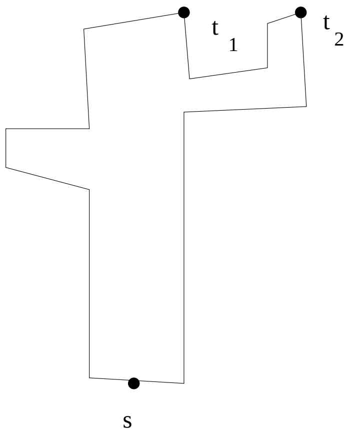

Figure 5.1: A possible street polygon.

and to the right of all left pockets. The shortest path may only "exit" the triangular sector defined by $v^l p v^r$ through the last windows of the pocket edges $\overline{pv^l}$ and $\overline{pv^r}$. However, notice that, in general, star polygons are not necessarily street polygons. For example the shortest path from s to t in the polygon of Figure 4.9 intersects only two of the three beams depicted and thus does not sees the entire interior of the polygon.

As the robot advances some areas previously explored are no longer in the field of vision of the robot, while other regions not currently in its field of vision remain unexplored. These unexplored pockets are called **true pockets**.

Definition 5.2 *A strategy A is said to* **dominate** *a strategy B if for all polygons P and target positions t, $C_A^P(t) \leq C_B^P(t)$.*

Any strategy for searching street polygons is dominated, in the worst case by the following high-level strategy.

High Level Strategy

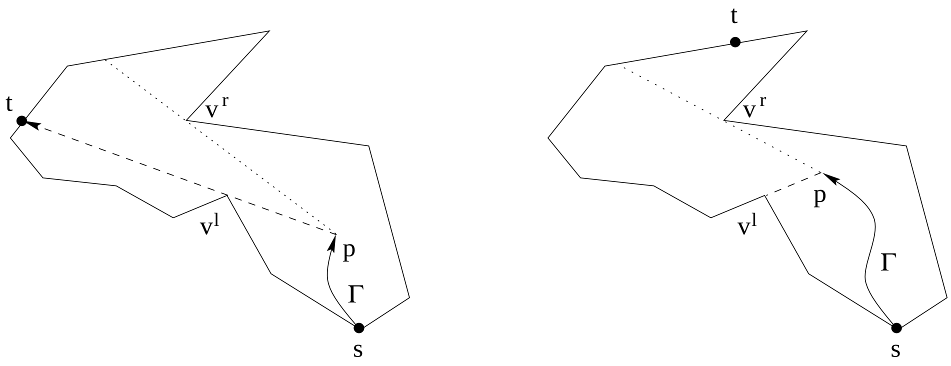

Figure 5.2: Cases 1 and 2.

While the target has not been reached, follow a path inside the triangle $\triangle v^l p v^r$ until one of the following occurs:

1. the target becomes visible in which case the robot moves directly to it,
2. the pocket on v^l (v^r) becomes completely visible and the robot moves to v^r (v^l resp.),
3. v^l (v^r) is no longer an extreme entrance point, and the new extreme point $\hat{v}^l$ ($\hat{v}^r$) is not collinear with $\overline{pv^l}$ ($\overline{pv^r}$), in which case the robot moves directly to v^r (v^l resp.), or
4. v^l (v^r) is no longer an extreme entrance point, and the new extreme point $\hat{v}^l$ ($\hat{v}^r$) **is** collinear with $\overline{pv^l}$ ($\overline{pv^r}$), in which case the robot continues moving on a path inside the triangle $\triangle \hat{v}^l p v^r$ ($\triangle v^l p \hat{v}^r$ resp.) until either one of cases 1-3 occurs, or the line $\overline{v^l v^r}$ is reached.

For cases 2 and 3, we need the following observation, sometimes referred to as the "rubber band lemma".

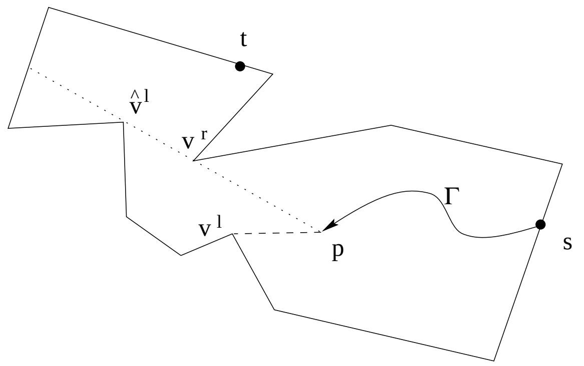

Figure 5.3: Case 3.

Observation 5.1 *The shortest path connecting two points s and t inside a polygon is a polygonal chain $s = q_0, q_1, q_2, \ldots, q_k = t$ such that q_i is a reflex vertex for $0 < i < k$.*

Physically, this observation states that a tight rubber band joining s and t forms a sequence of straight line segments and that only vertices of more that $\pi/2$ aperture can create a "bend" on the band.

Clearly, in case 1, a strategy that moves to the target as soon as it is identified dominates any strategy that does otherwise.

In cases 2 and 3, we know that the optimal trajectory from s (and from the current position p as well) visit v^l (or v^r in the symmetric case). To see this, one can apply Observation 5.1 above to each of the pairs (s, v^l), and (v^l, t). From the definition of the street and of the strategy it follows then that the optimal path from s to t is the catenation of the two paths above and that v^l is a reflex vertex. Thus, by the same argument as in case 1 it is optimal to visit extreme points that are part of the optimal trajectory as soon as it becomes clear that they are so.

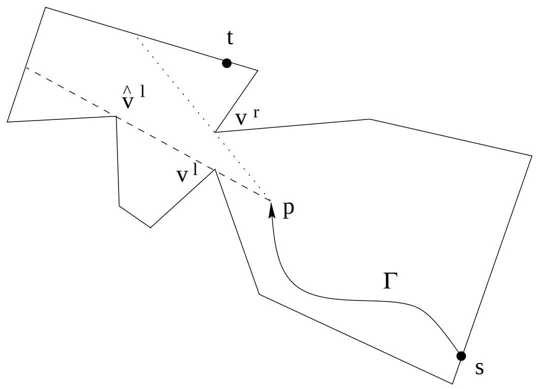

Figure 5.4: Case 4.

Case 4 is essentially a catch-all as the robot must follow some path within the zone of visibility for it to eventually see the target. A sequence of case 4 steps results in a "funnel-like" structure, as illustrated by Figure 5.5.

Observation 5.2 *At the end of case 4 only one of v^l or v^r is an extreme point.*

If the robot switched from case 4 to one of cases 1-3, the observation follows trivially as those cases by definition imply that one of v^l, v^r is no longer an extreme point. On the other hand, if the robot has reached $\overline{v^l v^r}$ then the observation follows by way of contradiction from the fact that the polygon is a street. Indeed, if both v^l and v^r were extreme points it implies that $\overline{pv^l}$ and $\overline{pv^r}$ are pocket edges; thus the associated pocket regions cannot see each other, and as only one of them contains the target, the other cannot see the other side of the street, which is a contradiction.

In the following sections we will consider several different strategies for selecting a path in Case 4.

Definition 5.3 *The high level strategy selects a case every time an extreme point changes. This is called a* **step** *of the strategy. Thus the robot moves on a sequence* *of steps $i = 1 \ldots k$. Let p_i denote the position of the robot at the beginning of step i. In particular, $s = p_1$. Let v_i^l, v_i^r denote the extreme points after step i.*

Notice that in most cases only one of the extreme points changes from step to step, thus in general one of $v_{i+1}^l = v_i^l$ or $v_{i+1}^r = v_i^r$ holds.

Figure 5.5 shows a typical street being searched by the robot. Notice that the path Γ moves inside a "funnel-like" structure until it reaches the bold dashed line joining the two extreme points. At this time, the robot has now determined which of the two extreme points belongs to the optimal path and moves to it. In the polygon shown, the target is seen immediately after reaching this extreme point, but this need not be the case; the robot might find itself at the start of a new funnel in which searching for the target continues. In the same figure, the thin dotted lines mark the points on the trajectory in which the robot sees new extreme points on either the left or right side.

Thus, a search inside a street can be analyzed as a sequence of independent funnel searches, between consecutive robot visits to the boundary of the polygon. The competitive ratio in this case is bounded by the maximum competitive ratio on each funnel search, which follows from the fact that $\max\{\sum a_i / \sum b_i\} \leq \max\{a_i/b_i\}$ for any two sequences of positive numbers $\{a_i\}$ and $\{b_i\}$.

At the beginning of a funnel search, the remaining unexplored parts of the polygon together with a consecutive pair of visits to the boundary form a street polygon. In this case Observation 4.3 applies to true pockets.

For the purposes of this analysis, we denote by Γ the path followed by the robot, and its length by $\lambda(\Gamma)$. Given two points q and r on Γ, $\Gamma(q,r)$ denotes the part of the path from q to r. The competitive ratio then is the ratio of the length of Γ to the length of $sp(s,t)$, the shortest path from s to t.

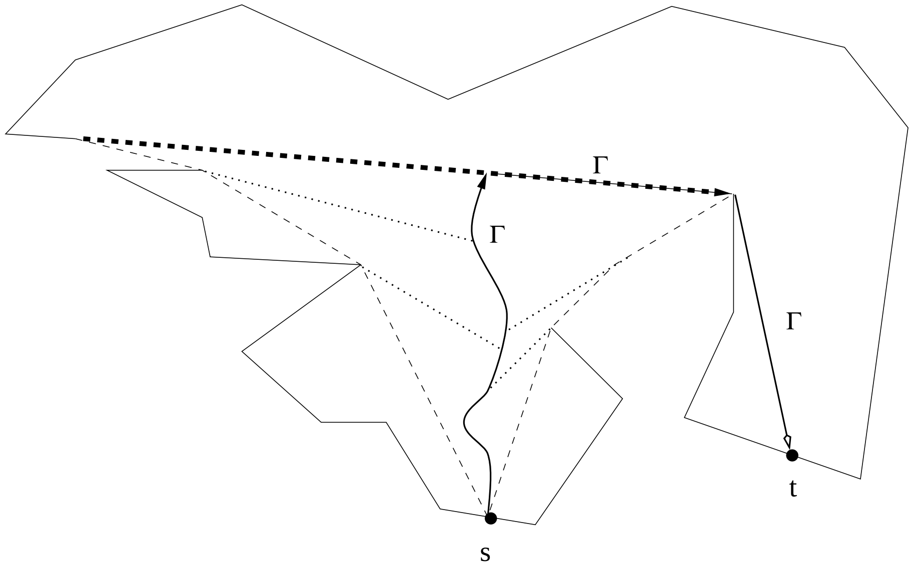

Figure 5.5: A street.

5.2 Orthogonal Street Polygons

Let us first study the case of orthogonal street polygons. In general, orthogonal polygons are useful representations of real life scenes, such as city blocks and buildings, which consist mostly of straight segments meeting at orthogonal angles.

Definition 5.4 [54] *An* **orthogonal polygon** *is a polygon in which each pair of adjacent edges are perpendicular to each other.*

Definition 5.5 *An* **orthogonal street polygon** *is a triple* (P, s, t) *such that* P *is an orthogonal polygon and* (P, s, t) *is a street.*

5.2.1 Searching with Unknown Destination

In this section we present a rigorous proof of the following theorem.

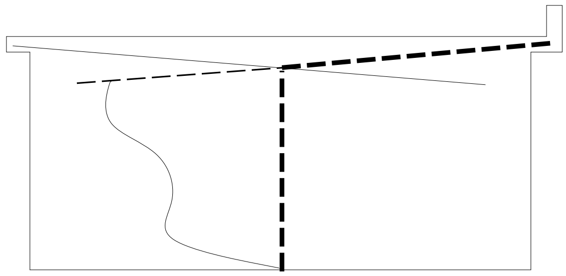

Figure 5.6: Eared rectangle, with walk inside.

Theorem 5.1 **[37, 38]** *There exists a strategy S such that for any orthogonal street polygons it finds the target point t with competitive ratio, in the worst case, of $\sqrt{2}$. Further, this ratio is optimal.*

To prove this theorem we introduce some concepts and an auxiliary lemma.

Definition 5.6 *An* **eared rectangle** *is composed of a rectangle two units wide and one unit tall. The center of the base has an* **entry point** *and on the top left and right corners there are two small alleys (ears) attached to it (see Figure 5.6). One of the alleys is* **connecting** *(to another part of the polygon), the other is* **a dead alley***.*

Lemma 5.1 *An eared rectangle may be traversed from the entry point to the connecting alley at a $(\sqrt{2} - \epsilon)$-competitive ratio, which is optimal.*

Proof. First we show that a $\sqrt{2}$-competitive ratio is attainable. The robot walks up the middle of the rectangle, until it sees the either the upper corner of the dead alley or the opening on the connecting alley. At this time the robot determine which of the two is open, and proceed to walk in that direction (see bold dashed lines in Figure 5.6). The length of the trajectory is $1 - \tan\theta + 1/\cos\theta$, where θ is the angle of the line between the extreme upper and the closer lower end point of the alleys. Notice that θ can be made arbitrarily small by means of reducing the height of the alley. Thus, this strategy gives a walk of length arbitrarily close to $\sup_{\theta\to 0}\{1+1/\cos\theta - \tan\theta\} = 2$. The optimal walk is of length $\sqrt{2}$ for a competitive ratio of $\sqrt{2}$ in the limit.

This strategy is optimal as well. We use an adversary argument to show this. The adversary simply opens the first alley to be looked into by the robot, and closes the other alley. Clearly the alley opened is always in the opposite half of the rectangle in which the robot is currently located (see curvy path plus dashed line in Figure 5.6). A simple application of the triangle inequality shows that the path in bold is shorter, and thus has a better competitive ratio. □

With this lemma we can now prove Theorem 5.1.

Proof. [Theorem 5.1] As Lemma 5.1 shows, a $\sqrt{2}$ ratio is sometimes necessary regardless of the strategy. Now we need to show that there exist a strategy that searches any orthogonal street polygon at a $\sqrt{2}$ competitive ratio. Consider the following strategy [38]:

- **Case 1** If the target t is visible, the robot moves directly to it.
- **Case 2** If there is no left (right) pocket, the robot moves directly to v^r (v^l).
- **Case 3** Both v^l and v^r are visible. The robot moves, on either the horizontal or vertical direction in such a way that the L_1 distance to both v^l and v_r decreases.
- **Case 4** The point v_l (or v_r) jumps and becomes collinear with v_r (v_l). The robot moves to v_r (v_l).

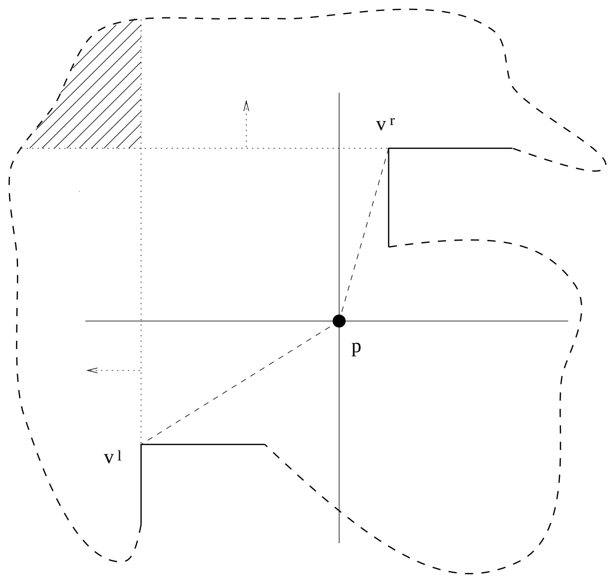

Figure 5.7: Extreme points on an orthogonal polygon.

We claim that the robot always meets one of the conditions for cases 1-4. For this we show that if case 3 does not apply then one of 1,2 or 4 should. First notice that, at any time, there exists a vertical or horizontal line passing through robot's current position and such that v^r and v^l lie on the same half plane among the two determined by the such line. Equivalently, if we consider a relative system of coordinates in which the origin is always located on the robot's current position, we claim that it is not possible for v^l and v^r to lie on diagonally opposite quadrants of the plane.

Let us assume otherwise. Without loss of generality, let v^r be located on the first quadrant (I) and v^l on the third quadrant (III).

Extreme points are, necessarily, reflex vertices. Figure 5.7 shows the only possible edge orientation that allows vertices on the I and III quadrant to be extreme points. However, in this case the target must lie in the shaded area as points on R must be visible from the vertical edge through v^l and points on L must be visible from the horizontal edge through v^r. But if the target were to be on this shaded are either is visible from the robot's current position or there is a pocket in which the target is hidden, and the entrance points v^l and v^r are not extreme points.

Now, if both points lie on adjoining quadrants or in the same quadrant, it is always possible for the robot to move on the horizontal or vertical direction in such a way that the L_1 distance to both extreme points decreases by the same amount traveled, except in two possible cases: (1) when the robot's current position and the two extreme points are collinear, (2) when the angle $\angle v^l p v^r$ is a right angle. Neither of these cases can actually occur. When a situation like these is reached one of the extreme points must no longer be an extreme point and thus we are no longer in case 3.

Having shown that one of cases 1-4 always applies to the robot, we notice that in case 3 the robot moves at a $\sqrt{2}$-competitive ratio towards the target even if the movement between the start and end point of a case 1, 2 or 4 was on a rectilinear path. In fact, the movement between the start and end point in any of latter three cases occurs on a single straight line segment joining these two points, which only betters the competitive ratio. $\square$

5.2.2 Searching with Known Destination

As we saw in Chapter 3, in many cases knowing the location of the target (destination) betters the search of the robot (for example, in [1], all six cases studied have better competitive ratios for known distance alone). Thus it is natural to consider the case of Known Destination Searches (KDS) for orthogonal street polygons.

Theorem 5.2 *Known destination searches in orthogonal street polygons are $\sqrt{2}$-competitive in the worst case.*

Proof. As proved by Kleinberg [38], there exists a $\sqrt{2}$-competitive strategy for UDS which can be applied in a straightforward manner to the KDS problem and gives a strategy of the same competitive ratio for all orthogonal street polygons in the KDS problem. What remains to be shown is that this competitive ratio is optimal. In this case the example of Figure 5.6 obviously no longer provides a lower bound.

The lower bound shall follow from the proof of the next lemma. □

The following definition applies to eared rectangles as defined in page 64.

Definition 5.7 *A* **left (right) move** *consists of a robot trajectory which forces the adversary of page 64 to open the* right *(left) alley. (Recall that, per the adversary strategy, the alley seen is always in the opposite half of the rectangle in which the robot is currently located.)*

Lemma 5.2 *There exists an orthogonal street which can only be searched, in the best case, at a $\sqrt{2}$-competitive ratio.*

Proof. We construct a family of polygons which are $(\sqrt{2} - \epsilon)$-competitive for KDS, for any $\epsilon > 0$. First, we define some widgets which will be used in the general construction.

Eared rectangles can be connected to create paths. In Figure 5.8 we can see how the alleys between eared rectangles are interconnected. A connecting alley leads to the entry point of another eared rectangle.

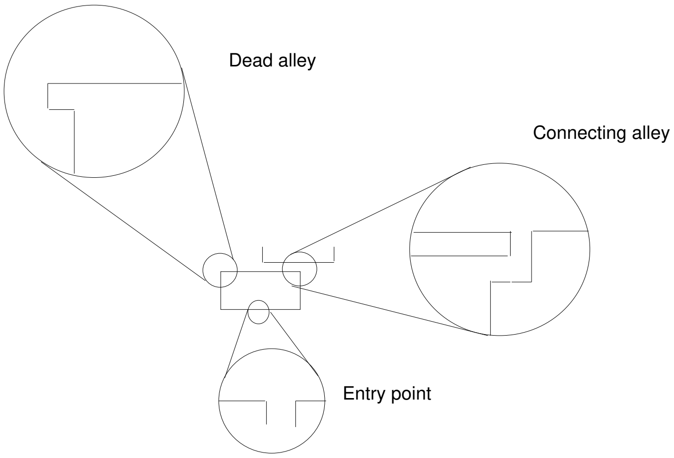

Figure 5.8: Interconnecting eared rectangles.

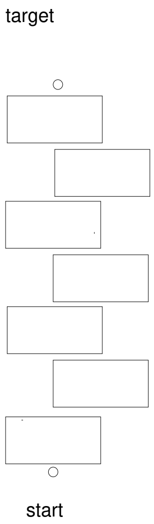

Figure 5.9: Walk the middle policy.

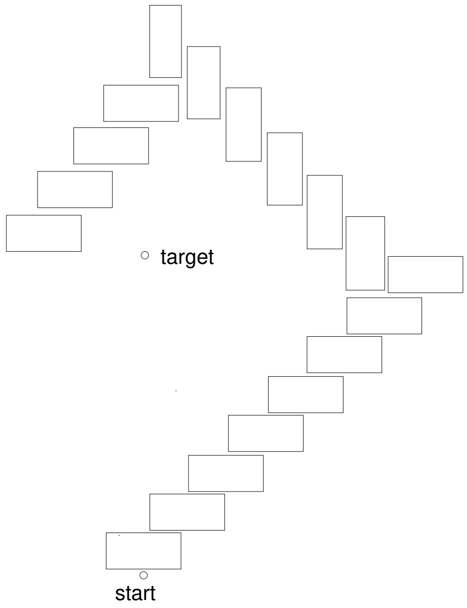

Figure 5.10: Always to the right policy.

We assume that the target is at some distance n directly above the start point as shown in Figure 5.9. To prove a lower bound of $\sqrt{2}$, we first consider two extreme cases of interconnecting eared rectangles, namely the *Walk the Middle Policy* and the *Always to the Right Policy*.

If the algorithm uses a strategy such as the one proposed in Lemma 5.1, the construction of Figure 5.9 shows an example of a polygon with a competitive ratio of $(2n+1)/(\sqrt{2}n+1)$.

Thus, an algorithm needs to deviate from the *Walk the Middle Policy*. In this case, the adversary presents the algorithm with an eared rectangle and it opens and closes the alleys according to the strategy proposed in Lemma 5.1. If we assume that the algorithm always makes left moves then the adversary consistently opens the right alley (see Figure 5.10). This creates a staircase-like path climbing to the right of the start point.

Definition 5.8 *A* **staircase** *is a chain of connected monotonic eared rectangles with entry points at a constant distance d in the L_1 metric from the target t.*

Alternatively, a staircase can be thought of eared rectangles on one side of a d-ball in the L_1 metric centered in t.

Notice that in the case of the *Always to the Right Policy*, when the robot reaches the end of the staircase, the adversary has forced the algorithm to move at a $\sqrt{2}$ competitive ratio, but the target is no closer than before. At this point the current connecting alley is horizontally aligned with the target, and the adversary moves one unit closer to the target (we assume that the algorithm moves optimally in this part, since it knows the position of the target) and the adversary then proceeds to construct a new staircase. This results in a spiraling set of staircases converging to the start point. The spiral is of length quadratic in n (see [1]) and, thus, the competitive ratio is $O((2n^2+n)/(\sqrt{2}n^2+n))$ which goes to $\sqrt{2}$ as n goes to infinity.

Having analyzed these extreme cases, we now consider the general case in which the algorithm may neither walk up the middle, nor consistently slant either way (see Figure 5.11). The *Walk the Middle Policy* and *Always to the Right Policy* can be viewed as extreme instances of this *Wavering Policy.*

In the case of a wavering algorithm, the adversary maintains the strategy described above. If the algorithm moves towards the left, the adversary opens the right alley, and conversely, if the algorithm deviates from the *Always to the Right Policy*, then the adversary opens the left alley. More precisely:

Input: A target at a distance of exactly n units above the start point in the vertical direction. A visibility polygon for the start point s corresponding to an eared rectangle with s at the entry point.

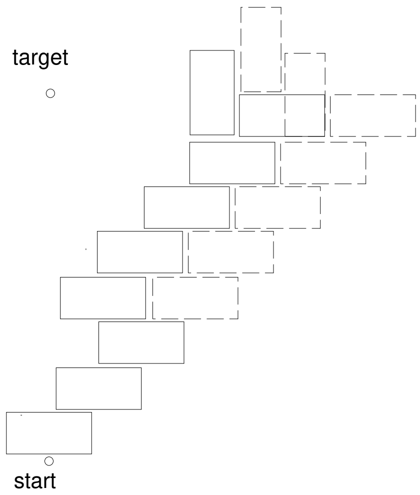

Figure 5.11: Wavering policy.

Opposition Strategy

1. If the robot first sees into the right alley (i.e. the robot makes a left move) then the adversary opens the right alley, conversely,

2. if the robot first sees into the left alley (i.e. the robot makes a right move) then the adversary opens the left alley,

3. **unless** the extra rectangle would increase the L_1 distance from the entry point (that is, the robot has reached the endpoint of a staircase) in which case the adversary opens the alley that is closer to the target under the L_1 metric. This is called a turn point of the staircase.

The following observations are immediate from the strategy's definition:

Observation 5.3 *The L_1 distances from the target to each entry point of the eared rectangles encountered by a robot traveling against an Opposition Strategy form a* *non-increasing sequence.*

Observation 5.4 *The robot traverses at most d eared rectangles with entry points at a distance d, under the L_1 metric, from the target. Furthermore, those rectangles are traversed consecutively in a staircase.*

Lastly, notice that if the robot is located in the southeastern quadrant, and traveling upwards, then a right move forces the adversary to open the left alley, with the entry point of the next eared rectangle at a distance of 2 units less than the previous entry position the L_1 metric; otherwise, if the robot makes a left move the adversary opens the right alley and the L_1 distance to the entry point remains unchanged. (Analogous observations can be made for each of the other three quadrants and traveling directions.)

Observation 5.5 *At any time, a left move followed by a right move, or vice versa, causes a decrease of two units in the distance to the target t, in the L_1 metric.*

In terms of the *Always to the Right Policy*, one may think of a right move causing a "jump" from the original spiral associated to an *Always to the Right Policy* (such as in Figure 5.10) to a spiral associated to a start point two units closer to the target in the L_1 metric (see Figure 5.11 with the polygon in solid lines, and the older staircase in dashed lines).

Note that any strategy can be decomposed into left and right sequences, and thus the *Wavering Policy* is indeed the general case.

With this definition at hand, we can now analyze precisely the competitive ratio of a robot moving under the *Wavering Policy*. Assume $n = 2m$ is even. Let k be the number of alternating sequences of left and right robot moves. As per Observation

5.5, we know that $k \leq m$. As indicated above, the robot moves in staircases, some of which may consist of a single rectangle. We number the staircases from 1 to m. The entry points of eared rectangles in staircase i are at a L_1-distance of $n-2(i-1)$ units from the target.

Let $a_1, a_2, \ldots, a_k$ be the staircases in which the robot jumped from one level of the spiral to another (as opposed to the rest of the staircases in which the robot traverses them until reaching the turn point).

Consider the distance traversed by the robot from the start point until the first turn point. There are exactly n rectangles from the start point to the first turn point. Since the optimal algorithm traverses each of those n rectangles on paths of length $\sqrt{2}$ we have that the optimal trajectory to the first turn point is of length $(\sqrt{2})n$. In turn, the robot traverses each of these rectangles save the last one, at a competitive ratio which is, at best, $\sqrt{2}$. The last rectangle is assumed to be traversed optimally. Thus the total distance traversed to the first turn point by the robot is $2(n-1)+\sqrt{2}$ regardless of the number of jumps of the robot. Notice that regardless of the number of jumps it is always possible to construct a staircase, as jumps to a smaller staircase cannot be reversed and turn points also result in a move to a staircase one unit closer to the target.

Let the starting point s also be considered as a turn point. In general,

Observation 5.6 *A sequence of staircases starting from a turn point at distance d from the target and finishing at the next consecutive turn point contains d eared rectangles, regardless of the number of jumps taken by the robot.*

Thus, if the ℓ-th staircase starts at a turn point, the length of the trajectory traversed by the robot until the next turn point is $2(n-2\ell-1)+\sqrt{2}$. Lastly, we note that the total number of turning points is equal to $n-2k$.

Then, the total length of the path traversed by the algorithm is

$$\sum_{i=0}^{k} \sum_{\ell=a_i+1}^{a_i-1} \sqrt{2} + 2(n-2\ell-1)$$

where $a_0 = -1$ for convenience of notation. Equivalently,

$$\begin{aligned} \sum_{j=1}^{a_1}(2n-4j-2+\sqrt{2}) \quad &+ \sum_{j=1}^{a_2-a_1-1}(2n-4a_1-4j-2+\sqrt{2}) \\ &+ \sum_{j=1}^{a_3-a_2-1}(2n-4a_2-4j-2+\sqrt{2}) + \cdots \end{aligned} \tag{5.1}$$

Notice that some of the summations above might be empty, i.e. they sum to zero.

Lemma 5.3 *Consider two strategies for walking up the staircase. Strategy A turns left in staircases $\{a_i\}_{1\le i\le k}$, and Strategy B turns left in the staircases $\{b_i\}_{1\le i\le k}$, such that $b_i = a_i - 1$, for all i with $a_i > 1$, and $b_i = 1$ otherwise. Then strategy B has a better competitive ratio than strategy A.*

Proof. Since $a_i \ge b_i$ it follows that summation (5.1) above is, term by term, larger for strategy A than for strategy B. That is, the distance traversed by the robot under strategy A is $(n-2k)\sqrt{2} + 2\,\mathcal{S}_{\mathcal{A}}$ where

$$\mathcal{S}_{\mathcal{A}} = \sum_{j=1}^{a_1}(n-2j-1) + \sum_{j=1}^{a_2-a_1-1}(n-2a_1-2j-1) + \sum_{j=1}^{a_3-a_2-1}(n-2a_2-2j-1) + \cdots$$

and the optimum is $(n-2k)\sqrt{2} + \sqrt{2}\,\mathcal{S}_{\mathcal{A}}$, for a competitive ratio of $((n-2k)\sqrt{2} + 2\,\mathcal{S}_{\mathcal{A}})/((n-2k)\sqrt{2} + \sqrt{2}\,\mathcal{S}_{\mathcal{A}})$. The competitive ratio for strategy B is $((n-2k)\sqrt{2} + 2\,\mathcal{S}_{\mathcal{B}})/((n-2k)\sqrt{2} + \sqrt{2}\,\mathcal{S}_{\mathcal{B}})$, which is better than for strategy A since $\mathcal{S}_{\mathcal{B}} \le \mathcal{S}_{\mathcal{A}}$. $\square$

Thus, setting $a_i = 1$, for all i, is optimal. So we have

$$\begin{aligned}
\text{Length of shortened spiral} &= n + \sum_{i=0}^{m-k} (n - 2k - 2i) = n + (m-k)(m-k+1) \\
\text{Length of optimal walk} &= \sqrt{2}\,(n + (m-k)(m-k+1)) \\
\text{Distance traversed by the alg.} &= 2\,(n + (m-k)(m-k+1)) - (2-\sqrt{2})(m-k) \\
\text{Competitive ratio} &= \frac{2\sqrt{2}m + \sqrt{2}m^2 - 2\sqrt{2}mk + \sqrt{2}k^2 + m - k}{3\,m + m^2 - 2\,mk + k^2 - k}
\end{aligned}$$

To improve its competitive ratio, the algorithm can select the optimal value of k which for a given m minimizes the competitive ratio. To simplify calculations we maximize $1/\sqrt{2}$ times the inverse of the competitive ratio, which is equivalent. Let

$$\begin{aligned}
d(m,k) &= \frac{3\,m + m^2 - 2\,mk + k^2 - k}{4m + 2m^2 - 4mk + 2k^2 + \sqrt{2}m - \sqrt{2}k} \\
\frac{\partial}{\partial k} d(m,k) &= \frac{\left(2-\sqrt{2}\right)\left(k - m - \sqrt{2m}\right)\left(k - m + \sqrt{2m}\right)}{\left(4\,m + 2\,m^2 - 4\,mk + 2\,k^2 + \sqrt{2}m - \sqrt{2}k\right)^2}
\end{aligned}$$

The critical points of $d(m,k)$ lie at $m \pm \sqrt{2m}$. Since $k \leq m$, we need only study $k_1 = m - \sqrt{2m}$.

Consider $d(m,\cdot)$ as a function of k. We see that the second derivative at $k_1 = m - \sqrt{2m}$ is $(\sqrt{m} - \sqrt{2m})/\,(4\,m + \sqrt{m})^2$ which is negative. Thus $d(m,\cdot)$ is maximized at k_1 as required. This implies that the best competitive ratio any algorithm may achieve is

$$\frac{1}{\sqrt{2}}\ d(m, m - \sqrt{2m})^{-1} = \frac{\sqrt{2}\left(4\,m + \sqrt{2}\sqrt{m}\right)}{4\,m + \sqrt{2}\sqrt{m}} + \frac{-2 + \sqrt{2}}{4\sqrt{m} + \sqrt{2}}$$

which is $\sqrt{2} - O(1/\sqrt{m})$. Since each eared rectangle is traversed at a $\sqrt{2}$ competitive ratio, we have that the adversary strategy described above forces any algorithm into a $\sqrt{2} - O(1/\sqrt{n})$ competitive ratio, which in the limit is $\sqrt{2}$.

Thus we have shown that regardless of the policy, a $\sqrt{2}$ inefficiency factor is necessarily introduced, even in the case where the robot knows where it is going, but is ignorant of the terrain in which is moving. $\square$

Chapter 6

Searching General Street Polygons

Strategies for street traversal can be classified broadly into two classes. One is *lad* (local absolute detour) in which the robot attempts to locally balance the overall detour of the strategy, and the other comprises oblivious strategies, in which the robot ignores some of the information learned along the way.

In the following subsection we study strategies of both types which are of further interest.

6.1 Oblivious Strategies For Street Polygons

There are several strategies which do not make use of information learned in the past. In this section we study the ones that result in surprisingly efficient searches.

6.1.1 Kleinberg's Strategy

The following strategy, first presented by Kleinberg [38], is the general version of the strategy presented in Theorem 5.1.

Kleinberg's Strategy

- **Case 1** If the target g is visible, the robot moves directly to it.
- **Case 2** If there is no left (right) pocket, the robot moves directly to v^r (v^l).
- **Case 3** Both v^l and v^r are visible. The robot moves, on a straight line in any direction within the angular sector $\angle v^l p v^r$ until the line of movement becomes perpendicular to one of $\overline{v^l p}$ or $\overline{v^r p}$.
- **Case 4** If the line of trajectory is perpendicular to, say, $\overline{v^l p}$, the robot moves at a $\pi/4$ angle to this line, still within the angular sector $\angle v^l p v^r$. The robot stops when the angle between the line of movement and either of $\overline{v^l p}$ or $\overline{v^r p}$ equals $3\pi/4$.
- **Case 5** The point v_l (or v_r) jumps and becomes collinear with v_r (v_l). The robot moves to v_r (v_l).

Figure 6.1 shows a search on a polygon using Kleinberg's strategy. In this case, we illustrate the two possible situations of case 4, when the angle of the current trajectory with the line $\overline{v^l p}$ and/or $\overline{v^r p}$ becomes $3\pi/4$. The following observations are key to the correctness of the strategy proposed above.

Observation 6.1 *At the end of Case 4, one of v^l or v^r is no longer an extreme point.*

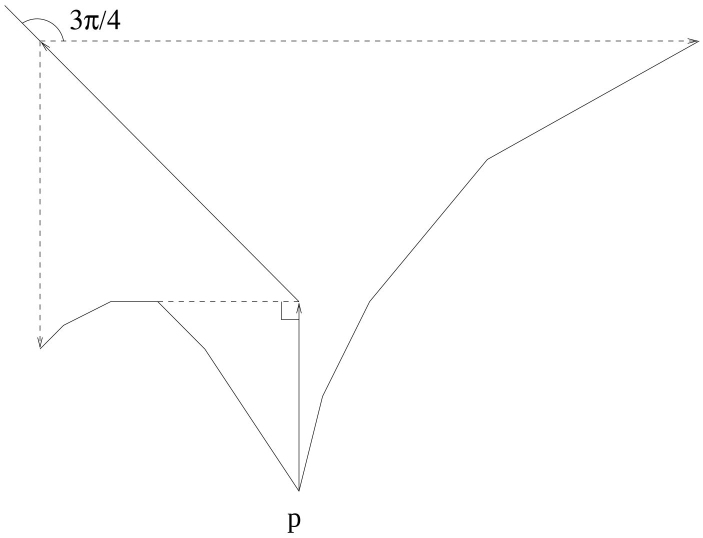

Figure 6.1: A robot under Kleinberg's strategy.

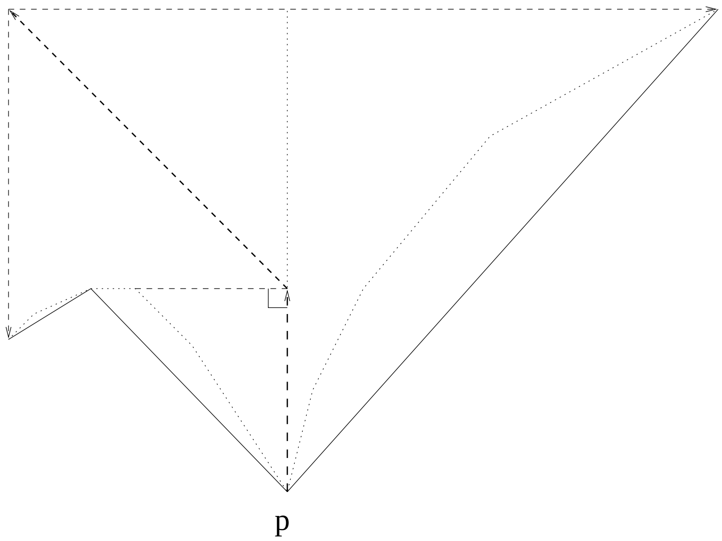

Figure 6.2: Worst case of Kleinberg's strategy.

Indeed, if v^l and v^r were both extreme points, it means that those points in the pocket they determine cannot see to the left of the vertical line $\overline{pv^l}$, if the robot is located above v^l or points below the horizontal line $\overline{pv^r}$ if located at the same height as v^r; this contradicts the fact that (P, s, t) is a street. It follows then that at most one of them is an extreme point.

Observation 6.2 *The angles between the current trajectory and the lines* $\overline{pv^l}$ *and* $\overline{pv^l}$ *increase monotonously and continuously.*

For the following theorem, a weaker upper bound was proven in [38] where it is also claimed that it can be improved to $2\sqrt{1+1/\sqrt{2}}$ while no proof is provided. In this case, we present the first proof in the literature that we know of and present a matching lower bound for the analysis.

Theorem 6.1 *Kleinberg's strategy has an exact* $2\sqrt{1+1/\sqrt{2}}$*-competitive ratio.*

Proof. Figure 6.2 shows a worst case situation for Kleinberg's strategy. First, we note that if the dotted lines from the original polygon are replaced with the solid lines, the robot traverses the same path as before, while the optimal trajectory has been reduced in size.

Now we must determine the value of the angles α, β, and η, as defined in Figure 6.3, that result in the worst case competitive ratio for the robot under this strategy.

The total distance traversed by the robot is, if the target is on the left side

$$\begin{aligned} d(p,p_1)+d(p_1,p_3)+d(p_3,v_2^l) &= d(p,p_1)+d(p_1,p_2)+d(p_2,p_3)+ \\ &\quad d(p_3,p_4)+d(p_4,v_2^l) \\ &= d(p,p_1)+d(p_1,p_2)+d(v_1^l,p_4)+ \\ &\quad d(p_2,v_1^l)+d(p_4,v_2^l) \end{aligned}$$

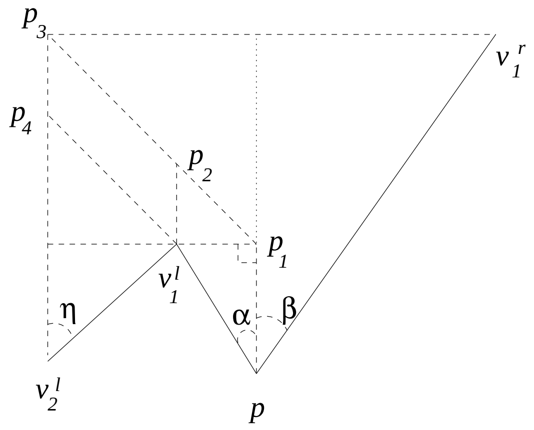

Figure 6.3: Worst case analysis for Kleinberg's strategy.

where $\overline{v_1^l p_4}$ is parallel to $\overline{p_1, p_3}$. Notice that the competitive ratio $C(v_2^l)$ is given by $(d(p,p_1)+d(p_1,p_2)+d(v_1^l,p_4)+d(p_2,v_1^l)+d(p_4,v_2^l))/(d(p,v_1^l)+d(v_1^l,v_2^l))$, thus

$$C(v_2^l) \leq \max\left\{\frac{d(p,p_1)+d(p_1,p_2)+d(p_2,v_1^l)}{d(p,v_1^l)}, \frac{d(v_1^l,p_4)+d(p_4,v_2^l)}{d(v_1^l,v_2^l)}\right\}$$

Without loss of generality, let $d(p,v_1^l) = 1$. The first term is given by $d(p,p_1) + d(p_1,p_2) + d(p_2,v_1^l) = \cos(\alpha) + \sqrt{2}\sin(\alpha) + \sin(\alpha)$ which is maximized in $[0,\pi]$ when $\alpha = 3/8\pi$ for a competitive ratio of $2\sqrt{1+1/\sqrt{2}}$. The second term is $(d(v_1^l,p_4) + d(p_4,v_2^l))/d(v_1^l,v_2^l)$. Once again, we can assume, without loss of generality, that $d(v_1^l,v_2^l) = 1$, and we have $d(v_1^l,p_4) + d(p_4,v_2^l) = \sin(\eta)\sqrt{2} + \sin(\eta) + \cos(\eta)$ which also has a maximum for $\eta = 3/8\pi$ for a competitive ratio of $2\sqrt{1+1/\sqrt{2}}$.

On the opposite side, the worst case is given by

$$C(v_1^r) = \frac{d(p,p_1)+d(p_1,p_3)+d(p_3,v_1^r)}{d(p,v_1^r)}.$$

Without loss of generality let $d(p,v_1^r) = 1$ and $d(p,p_1) = x$. Thus

$$C(v_1^r) \quad = \quad d(p,p_1) + d(p_1,p_3) + d(p_3,v_1^r)$$

$$\begin{aligned}
&= x + (\cos(\beta) - x)\sqrt{2} + (\cos(\beta) - x + \sin(\beta)) \\
&= (\cos(\beta) - x)\sqrt{2} + \cos(\beta) + \sin(\beta)
\end{aligned}$$

Clearly, for any given value of β the expression above is maximized when $x = 0$. Thus we need only maximize $\cos(\beta)\sqrt{2} + \cos(\beta) + \sin(\beta)$ which occurs when $\beta = \arctan(1/(1+\sqrt{2}))$, for which $C(v_1^r) = 2\sqrt{1 + 1/\sqrt{2}}$ as required. This proves that Kleinberg's strategy is at most $2\sqrt{1 + 1/\sqrt{2}}$-competitive. However this does not show that such a large competitive ratio is in fact required. For example, in Figure 6.3 the robot never reaches the point p_3 or even p_2 for that matter. Indeed, after crossing the line $\overline{v_2^l, v_1^r}$, the left pocket disappears and the robot moves to v_1^r as per case 2.

To show that a $2\sqrt{1 + 1/\sqrt{2}}$-competitive ratio is necessary, we need to exhibit a family of polygons in which this ratio is attained, or at least approximated. Figure 6.4 illustrates such family of polygons. First notice that $x = d(s, p_1)$ can be arbitrarily small and (P, s, t) is still a street. Secondly, the point p_2 becomes arbitrarily close to p_3 as the point v_2^l moves on the horizontal direction to $-\infty$. But this implies that the configuration on the right side can become arbitrarily close to the worst right side case of Figure 6.4 which occurs when p_2 and t are horizontally aligned. From the upper bound analysis, we know that in such situation the competitive ratio is maximized when $\overline{sv_1^r}$ forms an angle $\beta = \arctan(1/(1+\sqrt{2}))$ with the horizontal. In this case the competitive ratio is $2\sqrt{1 + 1/\sqrt{2}}$ as claimed. □

6.1.2 Bisector Strategies

The first step of Kleinberg's strategy consists of arbitrarily choosing a direction vector within the triangle $\triangle p v_1^l v_1^r$ and obliviously moving on it until $\angle v_1^l p v_1^r$ is a

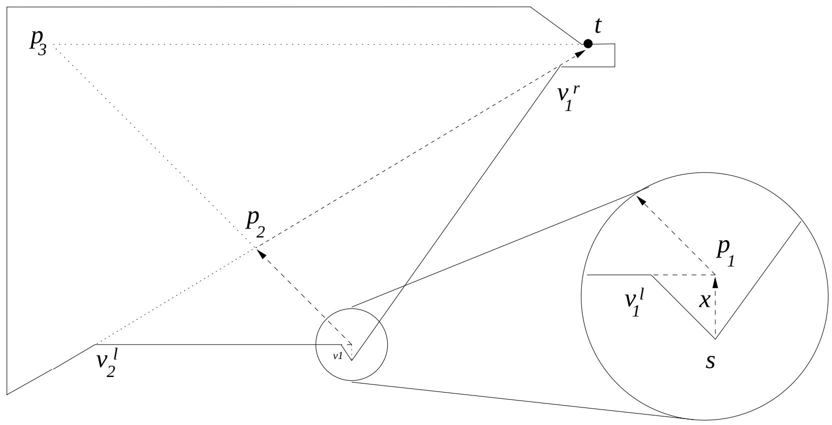

Figure 6.4: Actual worst case polygon for Kleinberg's strategy.

right angle.

Thus it may be possible to improve the competitive ratio by selecting the direction of movement more carefully. Indeed, it seems that a natural path to follow would be simply to walk on the bisector of this angle until the robot reaches the line $\overline{v^l v^r}$.

Lemma 6.1 *A bisector strategy is at least 3-competitive in the worst-case.*

Proof. If we consider a polygon which the robot can search on a single step (see Figure 6.5), we can see that if the target is on the left side, for a fixed starting angle 2θ the distance traversed by the robot increases as the angle α does likewise, while the optimal trajectory remains unchanged.

Thus for a given starting angle the worst case in a single step is reached in the limit, when $\alpha = \pi - 2\theta$ and the point v^l is at infinity (see Figure 6.6).

Since in the limit the lines $\overline{v^l v^r}$ and $\overline{pv^r}$ are parallel, we have that $\angle q v^l p = 2\theta$. Without loss of generality, let $d(p, v^l) = 1$. From this it follows that $d(p, q) = \sin(2\theta)$

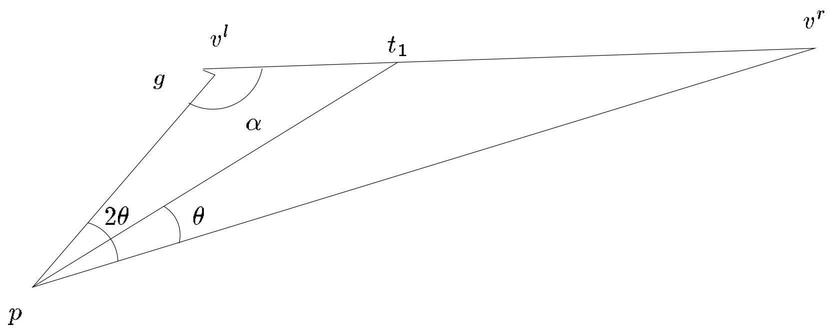

Figure 6.5: A polygon requiring a single step.

and $d(q, v^l) = \cos(2\theta)$. Also, we have $\angle ptv^l = \theta$, thus $d(p, q) = \sin(\theta)d(p, t)$, $d(p, t) = \sin(2\theta)/\sin(\theta) = 2\cos(\theta)$, $d(t, q) = 2\cos^2(\theta)$, and $d(t, v^l) = 2\cos^2(\theta) - \cos(2\theta) = 1$. The total length of the robot's trajectory is then $2\cos(\theta) + 1$, while the optimal length is 1, for a competitive ratio of $2\cos(\theta) + 1$. This expression is maximized for $\theta = 0$, with a competitive ratio of 3. $\square$

Another possibility is to compute the competitive ratio of the local progress, that is, the ratio $d(p, t)/(d(p, v^l) - d(t, v^l))$. This ratio is unbounded, as shown in Figure 6.7. In this case, we have that the triangle $\triangle tv_i^l p$ is isoceles, and thus $d(p, v^l) - d(t, v^l) = 0$ which implies that the local progress is unbounded. Notice that, if $\angle v^l tp = \angle tpv^r$ as indicated in Figure 6.7 then $\overline{v^l v^r}$ and $\overline{pv^r}$ must be parallel, which may only happen in the limit. In fact, it is not hard to see that the local progress measure is unbounded for all strategies as θ goes to π. Thus it might seem idle to even compute such a measure. However, this measure is useful for the strategies analyzed in the next two subsections.

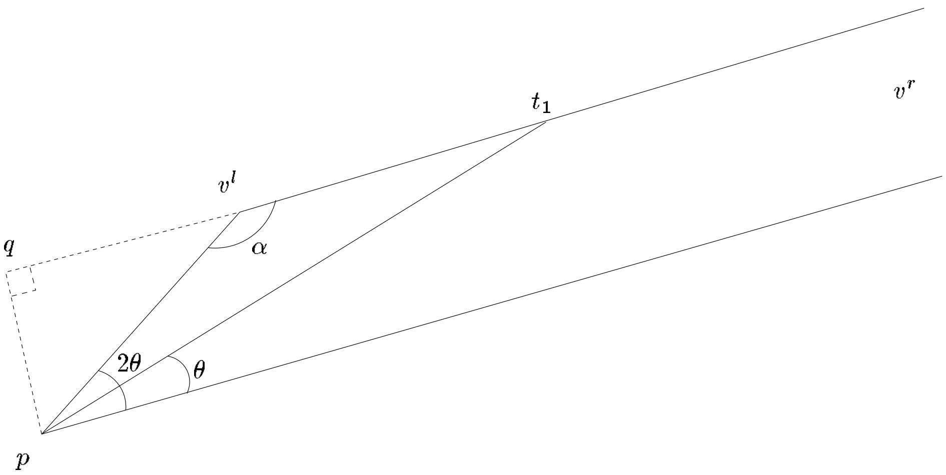

Figure 6.6: The polygon in the limit.

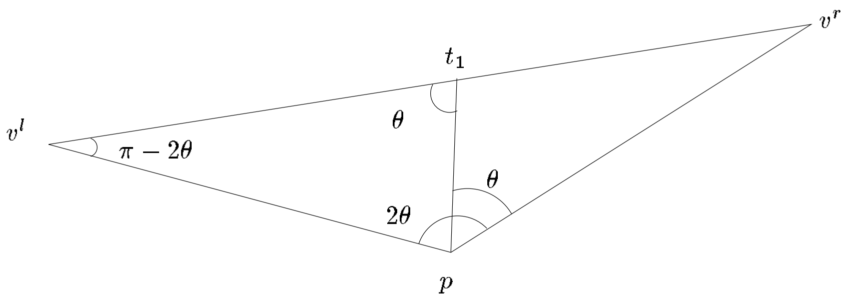

Figure 6.7: Unbounded ratio for local progress.

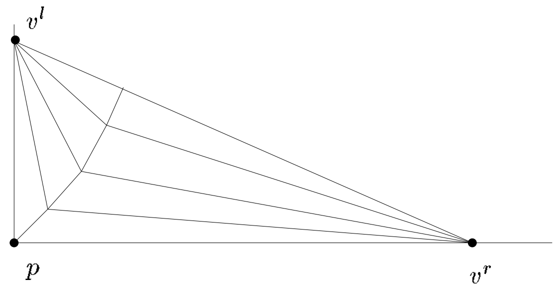

Figure 6.8: A sequence of steps.

Continuous Bisector

It is tempting then to devise a strategy in which the robot moves at all times on the bisector of the current angle. In fact this modification seems quite promising. However, as we will show in Lemma 6.3, a continuous bisector strategy is likely not the best possible strategy.

Consider a robot that advances on the bisector of $\angle v^l p v^r$ for a short length d. At this point, the robot recomputes the bisector of $\angle v^l p' v^r$, where p' is the current position of the robot. This procedure is repeated until the robot reaches the line $\overline{v^l v^r}$, at which time, for a single step, the robot can determine on which side is the target (see Figure 6.8). The trajectory described by the robot is a polygonal curve of line segments of length d. When d goes to zero, the curve becomes a continuous trajectory such that its tangent on a point p bisects the angle $\angle v^l p v^r$.

Definition 6.1 Continuous Bisector *is the strategy in which, for all points p in the trajectory, the tangent to the trajectory bisects the angle $\angle v^l p v^r$, where v^l, v^r, are the current extreme entrance points.*

Lemma 6.2 *The curve of points bisecting the angle formed by the current position*

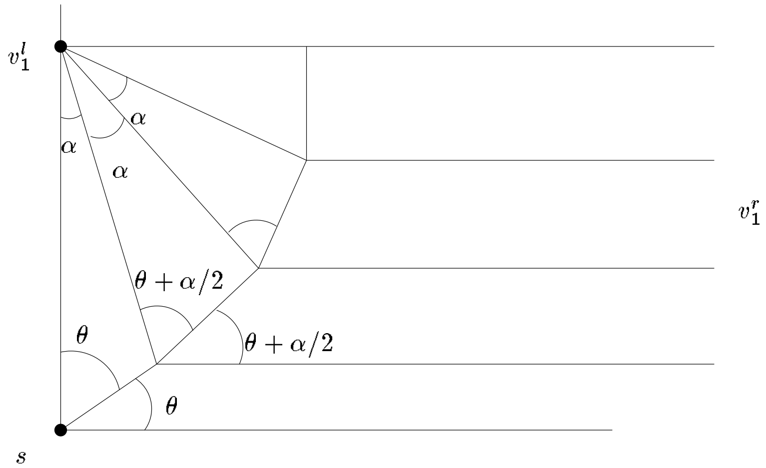

Figure 6.9: One step in the limit.

and two fixed points is a hyperbola.

This lemma can be shown directly, by establishing the differential equation and solving it, or one can refer to basic properties of the conic sections, as discussed in geometry texts (see for example [22]).

To compute a lower bound on the competitive ratio of this strategy first we compute the length of the trajectory in the worst case for a single step, where the robot is at $(0,0)$, one extreme entrance point is at $(0,1)$ and the other is at $(\infty, 1)$. In this case the robot initially moves at a $\pi/4$ angle. When the angle $\angle pv^l p'$ reaches a given threshold value $\Delta\alpha$ it recomputes the bisector and moves again (see Figure 6.9). Let $\lambda(\mathcal{P}_i)$ be the length of each step, for $i = 0 \ldots N-1$, and d_i^l the distance from p_i to v_i^l. Then the distance traversed by the robot is given by

$$\lim_{\Delta\alpha \to 0} \left(\sum_{i=0}^{N-1} \lambda(\mathcal{P}_i) + d_N^l \right)$$

where $N = \lfloor (\pi - 2\theta)\Delta/\alpha \rfloor$. If the target is on the left the optimum trajectory has length one and thus the competitive ratio is also described by the limit above. It follows then,

$$\lambda(\mathcal{P}_0) = \frac{\sin(\Delta\alpha)}{\sin(\theta+\Delta\alpha)} \qquad\qquad d_0^l = \frac{\sin(\theta)}{\sin(\theta+\Delta\alpha)}$$

$$\begin{aligned} \lambda(\mathcal{P}_1) &= \frac{\sin(\Delta\alpha)}{\sin(\theta+3/2\,\Delta\alpha)}\cdot d_0^l = \frac{\sin(\Delta\alpha)}{\sin(\theta+3/2\,\Delta\alpha)}\cdot\frac{\sin(\theta)}{\sin(\theta+\Delta\alpha)} \\ d_1^l &= \frac{\sin(\theta+\Delta\alpha/2)}{\sin(\theta+3/2\,\Delta\alpha)}\cdot d_0^+ = \frac{\sin(\theta+\Delta\alpha/2)}{\sin(\theta+3/2\,\Delta\alpha)}\cdot\frac{\sin(\theta)}{\sin(\theta+\Delta\alpha)} \end{aligned}$$

In general we have that

$$d_i^l = \frac{\sin(\theta+i/2\,\Delta\alpha)}{\sin(\theta+(i+2)/2\,\Delta\alpha)}\cdot d_{i-1}^l = \frac{\sin(\theta+\Delta\alpha/2)}{\sin(\theta+(i+2)/2\,\Delta\alpha)}\cdot\frac{\sin(\theta)}{\sin(\theta+(i+1)/2\,\Delta\alpha)}$$

$$\lambda(\mathcal{P}_i) = \frac{\sin(\Delta\alpha)}{\sin(\theta+(i+2)/2\,\Delta\alpha)}\cdot d_{i-1}^l$$

Thus

$$\begin{aligned} \sum_{i=0}^{N-1}\lambda(\mathcal{P}_i)+d_N^l &= \sum_{i=0}^{N-1}\frac{\sin(\Delta\alpha)}{\sin(\theta+(i+2)/2\,\Delta\alpha)}\frac{\sin(\theta+\Delta\alpha/2)}{\sin(\theta+(i+1)/2\,\Delta\alpha)}\frac{\sin(\theta)}{\sin(\theta+i/2\,\Delta\alpha)} \\ &\quad+\frac{\sin(\theta+\Delta\alpha/2)}{\sin(\theta+(N+1)/2\,\Delta\alpha)}\frac{\sin(\theta)}{\sin(\theta+N/2\,\Delta\alpha)} \\ &= \sum_{i=0}^{N-1}\frac{\sin(\Delta\alpha)}{\sin(\theta+(i+2)/2\,\Delta\alpha)}\frac{\sin(\theta+\Delta\alpha/2)}{\sin(\theta+(i+1)/2\,\Delta\alpha)}\frac{\sin(\theta)}{\sin(\theta+i/2\,\Delta\alpha)} \\ &\quad+\frac{\sin(\theta+\Delta\alpha/2)}{\sin(\pi/2+\Delta\alpha/2)}\frac{\sin(\theta)}{\sin(\pi/2)} \end{aligned}$$

In the limit,

$$\lim_{\Delta\alpha\to 0}\left(\sum_{i=0}^{N-1}\lambda(\mathcal{P}_i)+d_N^l\right) = \sin^2(\theta)\cdot\lim_{\alpha\to 0}\left(\sum_{i=0}^{N-1}\frac{\sin(\Delta\alpha)}{\sin^3(\theta+i/2\,\Delta\alpha)}\right)+\sin^2(\theta)$$

To evaluate the summation above we transform it into a Riemman summation. Consider the Taylor expansion of order 5 of $\sin(x)$ at zero, i.e., $\sin(x) = x - \frac{1}{6}x^3 +$

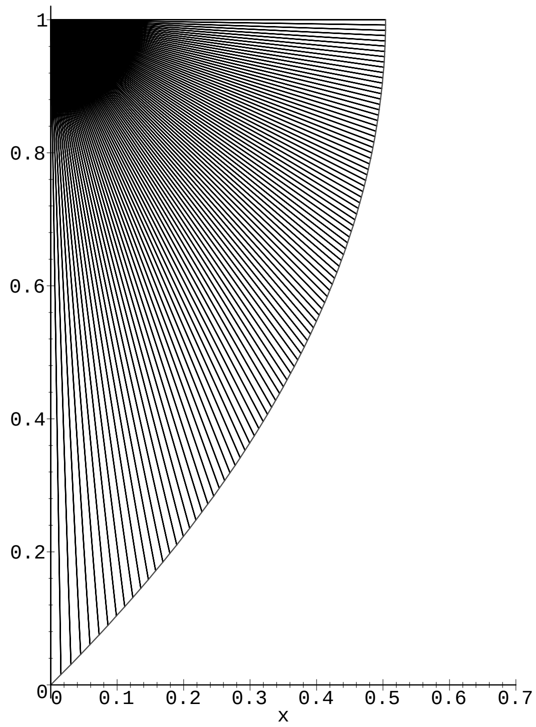

Figure 6.10: Continuous bisector.

$O(x^5)$ If we substitute this expression for $\sin(\Delta\alpha)$ in the summation above we see that,

$$\begin{aligned}
\lim_{\Delta\alpha\to 0}\sum_{i=0}^{N-1}\lambda(\mathcal{P}_i)+d_N^l &= \sin^2(\theta)\cdot \\
&\quad \left[1+\lim_{\Delta\alpha\to 0}\left(1-\frac{1}{6}\Delta\alpha^2+O(\Delta\alpha^4)\right)\sum_{i=0}^{N-1}\frac{\Delta\alpha}{\sin^3(\theta+i/2\,\Delta\alpha)}\right] \\
&= \sin^2(\theta)\left[1+\lim_{\Delta\alpha\to 0}\sum_{i=0}^{N-1}\frac{\Delta\alpha}{\sin^3(\theta+i/2\,\Delta\alpha)}\right] \\
&= \sin^2(\theta)\left[1+\int_0^{\pi/2-\theta}\frac{2}{\sin^3(\theta+\alpha)}\,d\alpha\right] \\
&= \sin^2(\theta)\cdot[1+\cot(\theta)\csc(\theta)-\ln(\csc(\theta)-\cot(\theta))]
\end{aligned}$$

At $\theta=\pi/4$, we have that the competitive ratio is then, $1/2\,(\sqrt{2}-\ln(\sqrt{2}-1)+1)\sim 1.6477$. In general, from its derivative, it can be seen that this function has a unique critical point, which is a maximum, in the interval $[0,\pi/2]$.

Lemma 6.3 *A single step in continuous bisector has a worst case competitive ratio of at least* 1.6837*, which occurs when* $\theta_{max}=0.9348$.

Bisector for Small Initial Angles

The strategy Bisector for small angles follows cases 1-3 as in the high level strategy described on page 58. Case 4 is as follows:

> If $\angle v_i^l p_i v_i^r \le \Phi$ then the robot moves on the bisector of $\angle v_i^l p_i v_i^r$. The robot advances until either a new vertex v_i^l or v_i^r is seen in which case the robot starts anew from the current position or until the local progress reaches a competitive ratio that is bigger (worse) than a predetermined value $\mathcal{C}$.

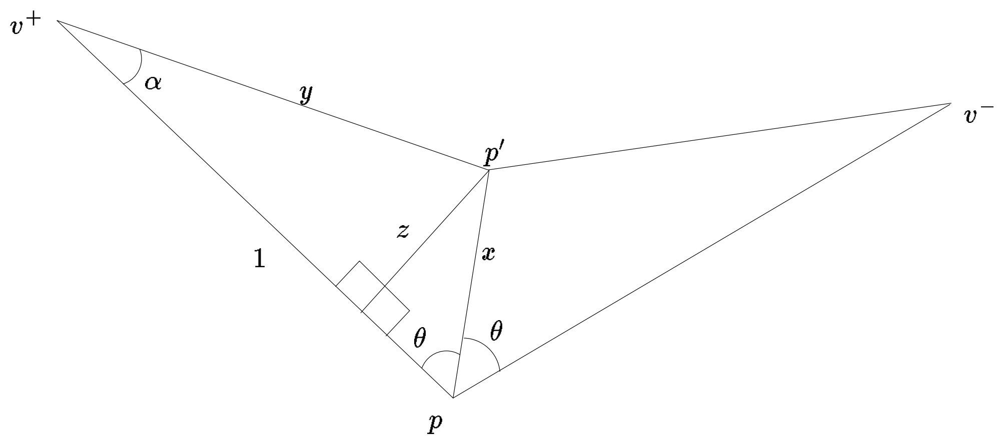

Figure 6.11: Starting position.

With notation as in Figure 6.11, the competitive ratio on the distance advanced from p to p' under the bisector strategy is determined by $x/(1-y)$. Since $z = y\sin(\alpha)$ and $z = x\sin(\theta)$, then $y = x\sin(\theta)/\sin(\alpha)$. Lastly we note that $1 - y\cos(\alpha) = x\cos(\theta)$ and $y^2 = (y\cos(\alpha))^2 + (y\sin(\alpha))^2$. From this we obtain

$$x = \frac{\sin(\alpha)}{\sin(\alpha+\theta)}, \qquad y = \frac{\sin(\theta)}{\sin(\theta+\alpha)}$$

with competitive ratio

$$\frac{x}{1-y} = \frac{\sin(\alpha)}{\sin(\alpha+\theta)-\sin(\theta)}$$

First, we consider the function $C(\alpha,\theta) = x/(1-y)$ as a function of θ

$$\frac{\delta}{\delta\theta}C(\alpha,\theta) = \frac{\sin(\alpha)(\cos(\theta)-\cos(\theta+\alpha))}{(\sin(\alpha+\theta)-\sin(\theta))^2}$$

which is positive for all $0 \leq \theta,\ \alpha \leq \pi/2$. This implies that, for a fixed α, the competitive ratio worsens as θ increases. If we then consider $C(\alpha,\theta)$ as a function of α, we can see that

$$\frac{\delta}{\delta\alpha}C(\alpha,\theta) = \frac{\sin(\theta)}{\cos(\alpha+2\theta)+1}$$

which is positive for all $0 \le \theta \le \pi$. Together these equations imply that if for a given θ_0, $C(\alpha,\theta_0) \le \mathcal{C}$, then $C(\alpha,\theta) \le \mathcal{C}$ for $\theta \ge \theta_0$, and that $C(\alpha,\theta) < C(\alpha_0,\theta)$ for $0 \le \alpha < \alpha_0$. Now, we compute the largest angle θ_{max} for which $C(\alpha,\theta_{max}) \le \mathcal{C}$. From the discussion above it follows that θ_{max} is such that $C(0,\theta_{max}) = \mathcal{C}$. As the robot is moving on a fixed bisector at angle θ it starts at a competitive ratio of $\lim_{\alpha\to 0}\sin(\alpha)/(\sin(\alpha+\theta)-\sin(\theta)) = 1/\cos(\theta)$, and thus $\theta_{max} = \arccos(1/\mathcal{C})$.

For $\mathcal{C} = 2$, we have $\theta_{max} = \pi/3$. In general, at the end of case d), the new starting angle can be computed from the equality $\sin(\alpha)/(\sin(\alpha+\theta)-\sin(\theta)) = \mathcal{C}$. Solving for α we obtain $\alpha = 2\arctan((\cos(\theta)-1/\mathcal{C})/\sin(\theta))$. This implies that the starting angle after case d) is

$$\begin{aligned} \text{starting angle after case d)} &\ge 2\theta+\alpha \\ &= 2\theta + \arctan\left(\frac{\cos(\theta)-1/\mathcal{C}}{\sin(\theta)}\right) \end{aligned}$$

We claim that at the end of a bisector step, the starting angle is always at least $2\theta_{max}+\alpha$. I.e, for all $\theta \le \theta_{max}$, $2\theta+\alpha \ge \theta_{max}$ (notice that this property is satisfied trivially for starting angles larger than θ_{max}). Indeed, $2\theta+\arctan((\cos(\theta)-1/\mathcal{C})/\sin(\theta))$ is a decreasing function, as its derivative is given by $-2(-1+\mathcal{C}\cos(\theta))/(\mathcal{C}^2+1-2\mathcal{C}\cos(\theta))$, which is negative for all $\theta < \theta_{max}$. This implies that at the end of the bisector step, the starting angle for the next step is always at least θ_{max}.

Theorem 6.2 *Let S be a hybrid strategy consisting of the bisector strategy for angles smaller than θ_{max} and a second strategy A for when the angle becomes larger than θ_{max}. Then the competitive ratio of strategy S is given by $C_S \le \max\{1/\cos(\theta_{max}), C_A\}$, where C_A is the competitive ratio of exploring the initial* *polygon as if the starting point was the switch point between the bisector strategy and strategy A.*

Proof. Follows from the fact that $(a+b)/(c+d) \leq \max\{a/c, b/d\}$ where $a, b, c, d \geq 0$, and the discussion above. □

6.1.3 Hybrid Kleinberg-Bisector Strategy

For the reasons described above, no strategy can rely solely on walking on the bisector, and any competitive robot that uses the bisector strategy must at some point use a different strategy. One such hybrid is the Kleinberg-Bisector Strategy. This strategy consists of using bisector up until $\theta_{max} = \pi/2$, and then switching to Kleinberg's strategy.

Lemma 6.4 *A Hybrid Kleinberg-Bisector Strategy has a* $1+\sqrt{2}$*-competitive ratio.*

Proof. From Theorem 6.2 it follows that for $\theta_{max} = \pi/2$ the competitive ratio of a hybrid Kleinberg-Bisector strategy is bounded by $\max\{1/\cos(\pi/2), C_K\} = \max\{\sqrt{2}, C_K\}$. Now we can apply a similar analysis for Kleinberg's strategy as in Theorem 6.1, with one important difference. In Case 4 of Kleinberg's strategy the robot advances at a $\pi/4$ angle until the angle between this trajectory and one of $\overline{pv^r}$ or $\overline{pv^l}$ equals $3\pi/4$. For this hybrid strategy we modify the angle at which the robot stops in case 4.

- **Case 4** If the line of trajectory is perpendicular to, say, $\overline{v^l p}$, the robot moves at a $\theta = f(\alpha, \beta) = \pi/4$ angle to this line, still within the sector $\angle v^l p v^r$. The robot stops when the angle between the line of movement and either of $\overline{v^l p}$ or $\overline{v^r p}$ equals $3\pi/4 - f(\alpha, \beta)$.

Recall that, from Observations 5.2, 6.1 and 6.2, it follows that one of v^l and v^r can no longer be an extreme point when $\overline{p_t v_t^l} \parallel \overline{pv^r}$ or $\overline{p_t v_t^r} \parallel \overline{pv^l}$, where p_t, v_t^l and v_r^t are the position of the robot and extreme points at the time it switched from Case 3 to Case 4.

Now we need to analyze Kleinberg's strategy under the new restricted conditions. As in the case of Theorem 6.1, we split the analysis into two cases: when the target is on the left and when it is on the right (see Figure 6.12).

If the target is on the left, the adversary can choose the values of η and β that maximizes the competitive ratio for the robot, while the robot chooses the value of $\theta = f(\alpha, \beta)$. First, we show that this occurs when $\eta = \pi/2$. The competitive ratio on the left side is given by

$$C(v_2^l) = \frac{d(p,p_1) + d(p_1,p_2) + d(p_2,p_3) + d(p_3,p_4) + d(p_4,v_2^l)}{d(p,v_1^l) + d(v_1^l,v_2^l)}$$

with $d(p_2,p_3) = d(v_1^l,p_4)$ and $d(p_3,p_4) = d(p_2,v_1^l)$. Thus, we can maximize $(d(v_1^l,p_4) + d(p_4,v_2^l))/d(v_1^l,v_2^l)$ and $(d(p,p_1) + d(p_1,p_2) + d(p_2,v_1^l))/d(p,v_1^l)$ independently, and select the maximum of both.

For $(d(v_1^l,p_4) + d(p_4,v_2^l))/d(v_1^l,v_2^l)$ we can see that, if we arbitrarily rescale so that $d(p_5,v_2^l) = 1$ we have that $d(v_1^l,v_2^l) \cdot \cos(\eta) = \cos(\pi/2 - \beta)$ which implies $d(v_1^l,v_2^l) = \sin(\beta)/\cos(\eta)$. Similarly, we obtain the system of equations

$$\begin{aligned}
d(p_5,v_1^l) &= \sin(\eta) \cdot d(v_1^l,v_2^l) - \sin(\pi/2 - \beta) = \tan(\eta)\sin(\beta) - \cos(\beta) \\
\sin(\theta) \cdot d(v_1^l,p_4) &= \sin(\beta) \cdot d(p_4,p_5) \\
d(p_5,v_1^l) &= \cos(\theta) \cdot d(v_1^l,p_4) + \cos(\beta) \cdot d(p_4,p_5)
\end{aligned}$$

which implies

$$d(v_1^l,p_4) = \frac{\sin(\beta)\cos(\beta+\eta)}{\sin(\beta+\theta)\cos(\eta)}$$

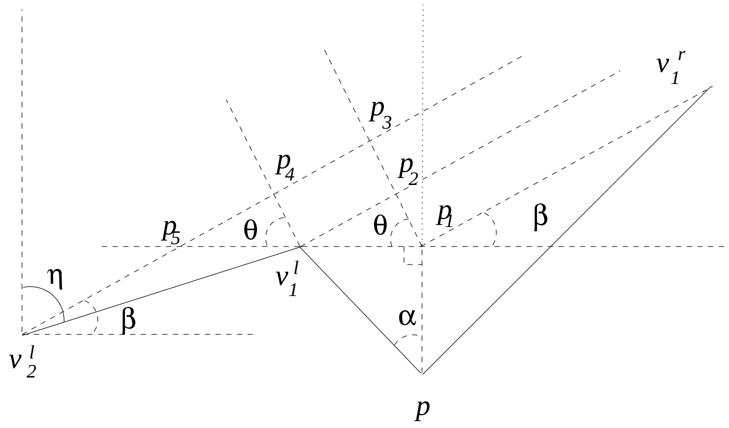

Figure 6.12: Analysis of Kleinberg-Bisector strategy.

$$d(p_4, v_2^l)) = 1 + \frac{\sin(\theta)\cos(\beta+\eta)}{\sin(\beta+\theta)\cos(\eta)}$$

for a total competitive ratio of

$$C(v_1^l, v_2^l) = \frac{\cos(\eta-\theta) - \cos(\beta+\eta)}{\sin(\theta+\beta)},$$

where $\pi/4 \leq \eta, \theta \leq \pi/2$, $0 \leq \beta \leq \pi/4$ and $\pi/2 \leq \eta + \beta \leq 3\pi/4$. Now, we proceed to show that this expression is maximized when η is maximum. Consider the derivative

$$D_\eta(C(v_1^l, v_2^l)) = \frac{-\sin(\eta-\theta) + \sin(\beta+\eta)}{\sin(\theta+\beta)}$$

with respect to η. Clearly $D_\eta(C(v_1^l, v_2^l)) \geq 0$ only if $-\sin(\eta-\theta) + \sin(\beta+\eta) \geq 0$, as $\pi/4 \leq \beta + \theta \leq 3\pi/4$. As a function of β, we have that $\sin(-\eta+\theta) + \sin(\eta+\beta)$ is a decreasing function (recall that $\eta + \beta \in [\pi/2, 3\pi/4]$). Thus the minimum value of $D_\eta(C(v_2^l)) \cdot \sin(\theta+\beta) = -\sin(\eta-\theta) + \sin(\beta+\eta)$ in the interval of interest is obtained when $\beta = 3\pi/4 - \eta$, with value $2\sin(-\eta+\theta) + \sqrt{2}$. Thus, if $D_\eta(C(v_1^l, v_2^l))$ is positive at this point, it must also be positive on the rest of the domain of interest. This is so as $\theta - \eta \in [-\pi/4, \pi/4]$ where $|\sin(\theta-\eta)| < 1/\sqrt{2}$. Thus we have shown that the competitive ratio increases as a function of η.

Thus the competitive ratio is maximized when $\eta \in [\pi/4, \pi/2]$ is maximum. In this case the competitive ratio is then

$$C(v_1^l, v_2^l) = \frac{\sin(\theta) + \sin(\beta)}{\sin(\beta + \theta)}$$

We consider then the derivative of this as a function of β or θ, as the function is symmetric, and obtain $D_\beta(C(v_1^l, v_2^l)) = \sin(\theta)/(1 + \cos(\beta + \theta))$ which is positive for $\theta \in [\pi/4, \pi/2]$ as it is the case. For these reasons, for a given β, the competitive ratio is determined by the largest possible $\theta = f(\alpha, \beta)$ that the robot may choose.

Recall that $C(v_2^l) = C(p, v_2^l) = \max\{C(p, v_1^l), C(v_1^l, v_2^l)\}$ with $C(p, v_1^l) = (d(p, p_1) + d(p_1, p_2) + d(p_2, v_1^l))/d(p, v_1^l)$. Let $d(p, v_1^l) = 1$, then $d(p, p_1) = \cos(\alpha)$, $d(p_1, p_2) = \sin(\alpha)\sin(\beta)/\sin(\beta + \theta)$ and $d(p_2, v_1^l) = \sin(\alpha)\sin(\theta)/\sin(\beta + \theta)$. So we have

$$C(p, v_1^l) = \cos(\alpha) + \frac{(\sin(\beta) + \sin(\theta))\sin(\alpha)}{\sin(\beta + \theta)}$$

Let us study first the case where θ is a function of β alone. Then the adversary can select an α that maximizes $C(p, v_1^l)$. This happens when $D_\alpha(C(p, v_1^l)) = -\sin(\alpha) + (\sin(\beta) + \sin(\theta))\cos(\alpha)/\sin(\beta + \theta) = 0$ and $D_\alpha^{(2)}(C(p, v_1^l)) = -\cos(\alpha) - (\sin(\beta) + \sin(\theta))\sin(\alpha)/\sin(\beta + \theta) \leq 0$. The first condition implies $\alpha = \arctan((\sin(\theta) + \sin(\beta))/\sin(\beta + \theta))$, while the second always holds in the domain of interest (recall that $\alpha \in [\pi/4, \pi/2]$). Substituting the value of α computed above we have that the competitive ratio is bounded, in the worst case, by

$$C(p, v_1^l) = \sqrt{1 + \left(\frac{\sin(\theta) + \sin(\beta)}{\sin(\theta + \beta)}\right)^2}$$

Now we substitute $\theta = \pi/4$, which gives $C(p, v_1^l) = \sqrt{1 + \frac{\sin(\beta) + 1/\sqrt{2}}{\sin(\beta + \pi/4)}}$. This expression is maximized when $\beta = \pi/4$, where $C(p, v_1^l) = \sqrt{3}$.

$$C(p, v_2^l) = \max\{C(p, v_1^l), C(v_1^l, v_2^l)\} = \max\{\sqrt{3}, \sqrt{2} + 1\} = \sqrt{2} + 1$$

The analysis on the right side is the same as for the original Kleinberg's strategy, where the competitive ratio is given by

$$C(p, v_1^r) = \cos(\pi/2 - \beta)(\sqrt{2} + 1) + \sin(\pi/2 - \beta)$$

which is maximized[1] when $\beta = \pi/4$, where $C(p, v_1^r) = \sqrt{2} + 1$. To conclude then we have that the worst case competitive ratio on the left and on the right is $1 + \sqrt{2}$ as required. □

Notice that this ratio can be improved in two ways. First, the worst case competitive ratios on the left and on the right side do not occur on the same trajectory. Thus, in principle it is possible to modify the search strategy to favour the left and right worst cases, while improving the competitive ratio. The second improvement is simply to note that increasing the value of θ_{max} in bisector results in a better combined strategy with smaller competitive ratio.

Let *Extended Bisector* be the strategy for which the value of θ_{max} is optimized, i.e. the second case above. Thus the competitive ratio for the Kleinberg's part of the Hybrid strategy is given by

$$\begin{aligned} \mathcal{C} &= \max\{C(v_1^l, v_2^l), C(p, v_1^l), C(p, v_1^r)\} \\ &= \max\left\{\frac{\sin(\theta) + \sin(\beta)}{\sin(\beta + \theta)}, \sqrt{1 + \left(\frac{\sin(\theta) + \sin(\beta)}{\sin(\theta + \beta)}\right)^2}, \sin(\beta)(\sqrt{2} + 1) + \cos(\beta)\right\} \end{aligned}$$

while the competitive ratio of extended bisector is $\mathcal{C} = 1/\cos(\theta_{max})$, where $\alpha, \beta \leq \pi/4 - \theta_{max}$.

For the values of interest, the minimum of the maximum of all the functions above occurs for $\max\{\cos(\theta_{max})^{-1}, \sin(\beta)\left(\sqrt{2} + 1\right) + \cos(\beta)\}$, and $\beta = \pi/2 - \theta_{max}$.

[1] Notice that the angles β in Theorem 6.1 and β in this theorem are complementary.

Thus we need the value of θ_{max} such that

$$\cos(\theta_{max})\sqrt{2} + \cos(\theta_{max}) + \sin(\theta_{max}) = \cos(\theta_{max})^{-1}.$$

This occurs when $\theta_{max} = \arccos\left(1/4\sqrt{4+2\sqrt{2}+2\sqrt{-26+20\sqrt{2}}}\right)$. Thus we have

Lemma 6.5 *The Hybrid Extended Bisector-Kleinberg strategy is* $\mathcal{C} = 4/\sqrt{4+2\sqrt{2}+2\sqrt{-26+20\sqrt{2}}} \approx 2.0504$*-competitive.*

For this case, we selected $\theta = \pi/4$. But, in fact, we can now select a better value for θ. For this, we need to reexamine the analysis of the right side competitive ratio.

Definition 6.2 *A* **variable Kleinberg strategy** *is a variation of Kleinberg's strategy in which the robot chooses a different angle θ depending on the value of the angle β.*

Thus, for general θ we have

$$C(p, v_1^r) = \frac{\sin(\beta) + \sin(\theta + \beta)}{\sin(\theta)}$$

Numerical analysis reports an optimal competitive ratio when $\theta \approx 1.0809$ and $\theta_{max} = 0.9693$.

Theorem 6.3 *A variable-Kleinberg extended-bisector strategy has a competitive ratio of less than* 1.7674*.*

6.1.4 Suboptimality of Oblivious Strategies

All strategies proposed thus far for street traversal rely on visual cues from *extreme points* to update the current strategy. That is, the robot moves on a line, until a new extreme point is seen. In this section we exhibit a polygon for which any robot moving on a straight line until a new extreme point has been identified performs worse than the optimal online path and the $\sqrt{2}$ lower bound for street traversal[2]. That is, we show that all oblivious strategies have a competitive ratio worse than $\sqrt{2}$.

Theorem 6.4 *An oblivious robot has a competitive ratio no better than* 1.4152507.

Figure 6.13 shows the parts of the polygon initially seen by a robot with the origin as starting point. We call this the initial signature of a polygon. Clearly several polygons may share the same initial signature. Given this signature the robot must choose a straight path $y = \tan\alpha \cdot x$, for some $-\pi/4 < \alpha < \pi/4$.

In turn the adversary, depending on the angle α chosen, presents the robot with either the polygon of Figure 6.14 or 6.15. If the angle of movement is larger than a threshold α_0, the adversary presents the robot with the *almost symmetric* polygon in Figure 6.15, otherwise, if $\alpha \leq \alpha_0$ then the adversary presents the robot with the *silent* polygon of Figure 6.14.

In both figures, we indicated a possible line of movement at an angle α up to the point where a new extreme point is seen.

Let us study the case of the silent polygon in Figure 6.14. In this case the adversary places the target either very close to the extreme point $(1, 1)$ or $(2, -2)$. The total distance traversed by the robot before reaching a target located at $(1 -$

[2] It has been noted by psychologists and behavioural scientists that humans are rather inept at detecting features that are oddly absent from a visual composition. In contrast, humans can single out an odd element in a picture with ease. From this perspective then, it is not surprising that all strategies proposed thus far for street traversal react only to the presence of visual cues, rather than to the absence of them.

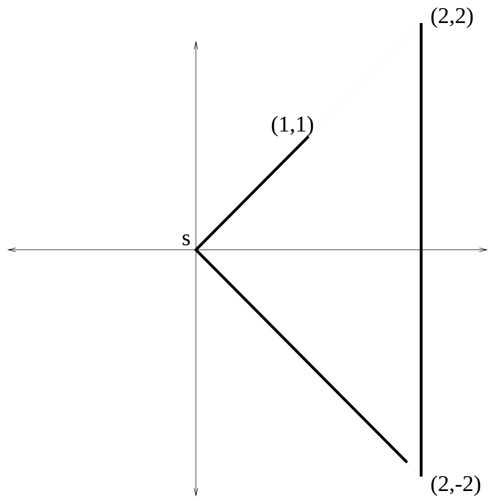

Figure 6.13: The signature of a polygon.

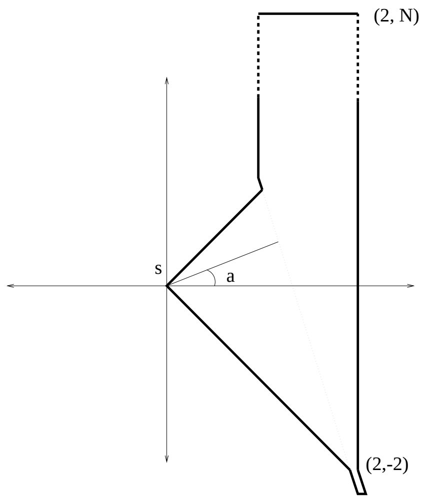

Figure 6.14: Silent polygon.

$\epsilon, 1+\epsilon)$ is given by $\sqrt{x_0^2+y_0^2}+\sqrt{(x_0-1)^2+(y_0-1)^2}$ as compared to the optimal which is $\sqrt{2}$. The point (x_0, y_0) is located on the intersection of the lines $y = -3x+4$ and the initial line of movement $y = \tan(\alpha)x$. The coordinates for a (x_0, y_0) are given by

$$(x_0, y_0) = \left(\frac{4}{3+\tan(\alpha)}, \frac{4\tan(\alpha)}{3+\tan(\alpha)}\right),$$

with

$$\alpha_0 = \arctan(y_0/x_0) = \arctan\left(\frac{-3C^2 - 2\sqrt{5}\,C + 1}{C^2 - 2\sqrt{5}\,C - 3}\right).$$

By way of contradiction, let us assume that the robot achieves a $\sqrt{2}$ competitive ratio on both sides. Set then $C = \sqrt{2}$ which implies $\alpha_0 = \arctan((5-2\sqrt{10})/(1-2\sqrt{10})) \sim 0.2438146138$.

Now, if the robot moves at an angle larger than α_0 the adversary presents the robot with an almost symmetric polygon. We will show that the competitive ratio in this case is larger than $\sqrt{2}$. Notice that the robot moves on a straight line until it sees the new extreme point (see Figure 6.15). Let $\tan(\beta)$ be the slope of the line joining the current robot position and the upper extreme point at the time it sees the new extreme point. Clearly the robot must then reach the line joining the two alleys, say at a point $(2, t_y)$. The competitive ratio is minimized if the point is located in such a way that the competitive ratio is the same for both cases when the target is on the upper or lower alley. The robot traverses up to the point $(2, t_y)$ the distance $d = \sqrt{x_0^2+y_0^2}+\sqrt{(x_0-2)^2+(y_0-t_y)^2}$. Then if the target is on the lower alley, the robot must further traverse a distance $2+t_y$ as compared to the total optimal of $2\sqrt{2}$. If, however the target is on the upper alley, the robot must reach the upper corner, which is located at $(2, \tan\beta+1)$ by means of traversing a distance $\tan\beta + 1 - t_y$, as compared to the optimal trajectory of length $\sqrt{2}+1/\cos\beta$.

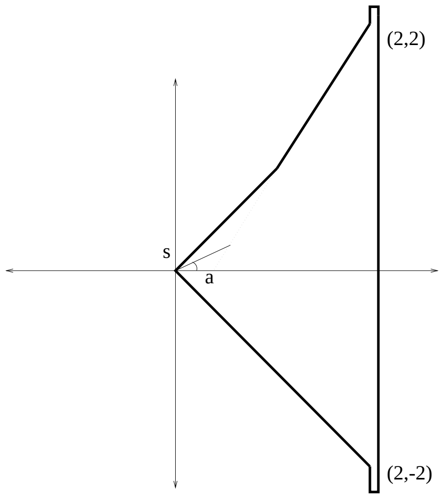

Figure 6.15: Almost symmetric polygon.

Clearly, the larger the angle α is, the worse the competitive ratio is for an almost symmetric polygon. Thus, set $\alpha = \alpha_0$ and then

$$d = \frac{\sin\beta - \cos\beta}{\sin(\beta-\alpha)} + \sqrt{\left(\frac{\cos\alpha\,(\sin\beta - \cos\beta)}{\sin(\beta-\alpha)} - 2\right)^2 + \left(\frac{\sin\alpha\,(\sin\beta - \cos\beta)}{\sin(\beta-\alpha)} - t_y\right)^2}.$$

The adversary then sets $\beta = \pi/4 + 1/14$, for which the point $t_y = -0.00281200\ldots$ The competitive ratio C is then equal to 1.4152507 for either side, which is larger than $\sqrt{2} \sim 1.414213562\ldots$.

This shows that, if the robot traverses a silent polygon at a $\sqrt{2}$ competitive ratio using a strategy that only "turns" on extreme points, then the same strategy results in a worse than $\sqrt{2}$ competitive ratio in an almost symmetric polygon.

A Note on Search Paths

The following two facts are worth noting:

Theorem 6.5 *Save for continuous bisector, all search strategies for street polygons presented in this chapter result in search paths made of at most n linear segments.*

This theorem follows from the fact that the robot changes direction only when it sees a new extreme point and there are at most n extreme points. For the case of continuous bisector we have a similar result.

Lemma 6.6 *A robot using a search path proposed by continuous bisector traverses a path made of at most n hyperbolic segments.*

Theorem 6.4 in this subsection seems to point out that an optimal search will require more than $O(n)$ snapshots, and that indeed, a $\sqrt{2}$-competitive ratio search might need continuous vision. However it is still an interesting open problem to determine if it is possible to reduce the number of "snapshots" needed during the search stage.

6.2 Searching Street Polygons with Navigational Error

6.2.1 Strategy Walk-in-Circles

In this section we present a family of strategies for the problem of searching in an unknown street. We show that a robot using a strategy from this family follows a path that is at most $\pi+1$ times longer than the shortest possible path. We then use this new strategy as part of a hybrid method to obtain an equally simple strategy of slightly more complex analysis with a competitive ratio of $\frac{1}{2}\sqrt{\pi^2+4\pi+8} \sim 2.76$.

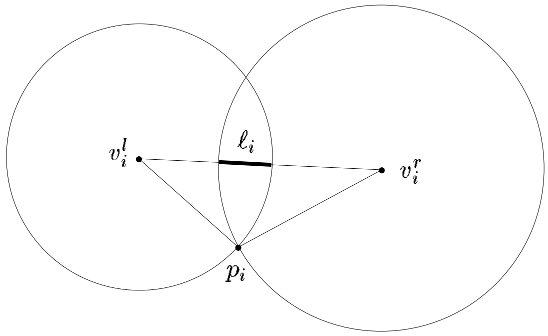

Figure 6.16: The subsegment of target points.

More importantly, we show that the $\pi+1$ strategy is robust under small navigational errors.

Let α_i^l (α_i^r) denote the angle $\angle p_{i+1}v_i^l p_i$ ($\angle p_{i+1}v_i^r p_i$). Let $\gamma_i = \angle v_i^l p_i v_i^r$ and let ℓ_i be the subsegment of $\overline{v_i^l v_i^r}$ which consists of the points q such that $d(v_i^l, q) \leq d(v_i^l, p_i)$ and $d(v_i^r, q) \leq d(v_i^r, p_i)$ (see Figure 6.16). The algorithm chooses a point t_i in the target segment ℓ_i and moves in a straight line towards it. If a new window appears, the robot recomputes ℓ_i according to the updated points v_{i+1}^r and v_{i+1}^l, and the new position p_{i+1}, until the goal is found (see Figure 6.17).

Observation 6.3 *For any strategy S we have $\gamma_{i+1} = \gamma_i + \alpha_i^l + \alpha_i^r$.*

Let $d_i^l = d(p_i, v_i^l)$ and $d_i^r = d(p_i, v_i^r)$ and let a_i^l (a_i^r) denote the actual progress towards the target if it is located on the left (right) side. That is, if the target is located on the left side, the robot should be moving towards v_i^l during step i and towards v_{i+1}^l on step $i+1$. This implies $a_i^l = d_i^l - (d_{i+1}^l - d(v_i^l, v_{i+1}^l))$ or equivalently $a_i^l = d_i^l - d(p_{i+1}, v_i^l)$ as p_{i+1}, v_i^l and v_{i+1}^l are collinear.

Observation 6.4 *The length of the shortest path, if the target is on the left is given by $sp(s,g) = \sum_{j=1}^{k} a_j^l + d_k^l$ and if on the right $sp(s,g) = \sum_{j=1}^{k-1} a_j^r + d_k^r$. In fact,*

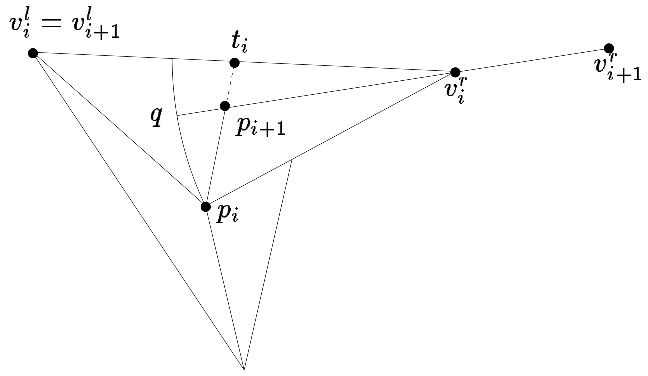

Figure 6.17: A single step in the strategy.

these equations hold in midstep as well: $sp(s, v_i^l) = \sum_{j=1}^{i} a_j^l + d_i^l$ *and* $sp(s, v_i^r) = \sum_{j=1}^{i} a_j^r + d_i^r$.

These equations follow naturally from the definition of actual progress, or alternatively can also be derived inductively.

Now we can analyse the strategy. We only consider the case where the goal turns out to be on the right side. This is without loss of generality since the local target selection strategy is invariant under reflections.

The length of the path traversed by the robot is determined by the sum of the length of all segments $\overline{p_i p_{i+1}}$, i.e. $\sum_{i=1}^{k-1} d(p_i, p_{i+1})$, where k is the number of extreme entrance points seen by the robot until one of the events of cases 1)–3) occurs (see page 58). Let q be the intersection point of the ray with origin in v_i^r going through p_{i+1} and the circle centered at v_i^r passing through p_i (see Figure 6.17). By the triangle inequality, we have $d(p_i, p_{i+1}) \leq d(p_i, q) + d(q, p_{i+1})$.

In turn, the length of $d(p_i, q)$ is bounded by the length of the circular arc $\widehat{p_i q}$. Recall that $\alpha_i^r = \angle p_{i+1} v_i^r p_i = \angle q v_i^r p_i$. The length of the circular arc $\widehat{p_i q}$ is given by

$\alpha_i^r \cdot d(p_i, v_i^r) = \alpha_i^r \cdot d_i^r$. Thus,

$$\sum_{i=1}^{k-1} d(p_i, p_{i+1}) \leq \sum_{i=1}^{k} \left(d(p_i, q) + d(q, p_{i+1})\right) \leq \sum_{i=1}^{k} \left(\alpha_i^r \cdot d_i^r + d(q, p_{i+1})\right),$$

and the competitive ratio is determined by

$$\frac{\sum_{i=1}^{k} d(p_i, p_{i+1}) + d(p_{k+1}, v_k^r)}{\lambda(\mathcal{V}_k^r)} \leq \frac{\sum_{i=1}^{k} \alpha_i^r \cdot d_i^r + \sum_{i=1}^{k} d(q, p_{i+1}) + d(p_{i+1}, v_k^r)}{\lambda(\mathcal{V}_k^r)}.$$

Recall that $\lambda(\mathcal{V}_k^r)$ is the length of the shortest path from p_1 to v_k^r. Since we assume that v_k^r is located on $sp(s, g)$, $\mathcal{V}_k^r$ is a part of $sp(s, g)$.

Note that since t_i is contained in the circle C_i with centre v_i^r and radius $d(v_i^r, p_i)$ by the definition of ℓ_i, the whole line segment $\overline{p_i t_i}$ is contained in C_i and, therefore, $d(v_i^r, p_{i+1}) \leq d(v_i^r, p_i)$. The definition of q now yields $d(q, p_{i+1}) = d(q, v_i^r) - d(p_{i+1}, v_i^r) = a_i^r$, and Observation 6.4 implies that $\sum_{j=1}^{k} a_j^r + d(p_{k+1}, v_k^r) = \sum_{j=1}^{k-1} a_j^r + d_k^r \leq \lambda(\mathcal{V}_k^r)$ and $d_i^r \leq \lambda(\mathcal{V}_k^r)$ (for all $1 \leq i \leq k$), hence, we obtain

$$\begin{aligned} \sum_{i=1}^{k} \alpha_i^r d(p_i, v_i^r) + \sum_{i=1}^{k} d(q, p_{i+1}) + d(p_{k+1}, v_k^r) &= \sum_{i=1}^{k} \alpha_i^r d_i^r + \sum_{i=1}^{k-1} a_i^r + d_k^r \\ &\leq \left(\sum_{i=1}^{k} \alpha_i^r + 1\right) \lambda(\mathcal{V}_k^r) \end{aligned}$$

As it was noted by Klein (see proof of Lemma 2.7 in [37]), the angle $\angle v_i^l p_i v_i^r$ never exceeds π. Observation 6.3 implies that $\angle v_{i+1}^l p_{i+1} v_{i+1}^r \geq \angle v_i^l p_i v_i^r + \alpha_i^r$, and thus $\pi \geq \angle v_k^l p_k v_k^r \geq \angle v_0^l p_0 v_0^r + \sum_i \alpha_i^r$, from which we obtain the following upper bound for the competitive ratio of the algorithm

$$\frac{\sum_i d(p_i, p_{i+1})}{\lambda(\mathcal{V}_k^r)} \leq \pi + 1.$$

Theorem 6.6 *A robot moving traveling under the strategy* Walk-in-Circles *has a* $\pi + 1$ *competitive ratio.*

As the target in each step i is selected from the interval ℓ_i this provides a margin of navigational error for the robot. That is, the strategy is robust under small constant bias of compass heading. The tolerance of the strategy is proportional to the aspect ratio of the shortest vs. longest edge encountered and the smallest distinguished angle between left or right extreme entrance points.

6.2.2 Hybrid Walk-in-Circles Strategies

From the analysis presented above it is clear that the competitive ratio of strategy *Walk-in-Circles* is directly dependent on the total "turn" angle $\Phi = \sum_i \alpha_i^r$. As it was pointed out, Φ is smaller than π minus the initial angle $\angle v_1^l p_1 v_1^r$. This implies that, if the initial angle is large, the strategy gives a better competitive ratio.

In this section we consider a hybrid method, in which a strategy similar to that proposed by Kleinberg [38] is followed for initial angles $\angle v_1^l p_1 v_1^r$ smaller than $\pi/2$ and the strategy *Walk-in-Circles* is used for angles larger than $\pi/2$.

Hybrid Kleinberg/Walk-in-Circles Strategy

Cases 1)–3) are as in the *High Level Strategy.*

Case 4) If $\angle v_1^l p_1 v_1^r \leq \pi/2$ then the robot moves on the line perpendicular to $\overline{v_1^l v_1^r}$ [3]. As the robot advances it updates the vertices v_i^l and v_i^r as the windows seen change. When either of $\angle v_i^l p p_1$ or $\angle v_i^r p p_1 = \pi/2$, where p is the current position of the robot, it switches to strategy *Walk-in-Circles*, with p as starting point.

[3] If the line l_1 perpendicular to the line through v_1^l and v_1^r that goes through p_1 does not intersect $\overline{v_1^l v_1^r}$, then the robot follows the side of the triangle (p_1, v_1^l, v_1^r) that is closer to l_1. This does not change the analysis.

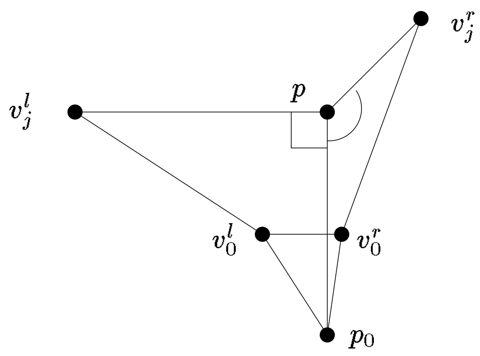

Figure 6.18: A hybrid strategy.

Case 5) If $\angle v_1^l p_1 v_1^r \geq \pi/2$ then the robot uses strategy *Walk-in-Circles*.

From the analysis of strategy *Walk-in-Circles*, it follows that Cases 1)–3) and 5) have a competitive ratio of at most $\pi/2 + 1$. Case 4) requires a more careful analysis.

If, as in the previous section, we assume that the goal lies on the right side, then the optimal trajectory is given by $d(p_1, v_1^r) + \sum_i d(v_i^r, v_{i+1}^r)$. Let j be the index of the reflex vertex in which the robot switched strategies. Notice that $\angle v_j^r p v_j^l$ is now bigger than or equal to $\pi/2$.

Lemma 6.7 *The distance traversed by the robot up to the point where it switches strategy is bounded by* $d(p_1, p) \leq \sqrt{d(p_1, v_j^{l,r})^2 - d(p, v_j^{l,r})^2}$ *on either side.*

Proof. For the vertex forming the right angle, the lemma follows trivially from the Theorem of Pythagoras. On the opposing vertex, say as in Figure 6.18, the law of cosines states $d(p_1, v_j^r)^2 = d(p_1, p)^2 + d(p, v_j^r)^2 - 2\, d(p_1, p)\, d(p, v_j^r) \cos(\angle p_1 p v_j^r)$; which implies $d(p_1, v_j^r)^2 \geq d(p_1, p)^2 + d(p, v_j^r)^2$ as $\angle p_1 p v_j^r \geq \pi/2$, from which the lemma follows. $\square$

As the robot applies strategy *Walk-in-Circles* as if p was the starting point, we have that the length of the distance traversed by it from p onwards is bounded by $(\pi/2+1)\left(d(p,v_j^r)+\sum_{i=j}^{k-1} d(v_i^r,v_{i+1}^r)\right)$. Thus the competitive ratio is given by $\mathcal{R}/\lambda(\mathcal{V}_{k-1}^r)$ where,

$$\mathcal{R} \;=\; \sqrt{d(p_1,v_j^r)^2-d(p,v_j^r)^2}+(\pi/2+1)\Big(d(p,v_j^r)+\sum_{i=j}^{k-1} d(v_i^r,v_{i+1}^r)\Big).$$

Let $\lambda(\mathcal{V}_{k-1}^r)'=d(p_1,v_j^r)+\sum_{i=j}^{k-1} d(v_i^r,v_{i+1}^r)$. Since $\lambda(\mathcal{V}_{k-1}^r)\geq\lambda(\mathcal{V}_{k-1}^r)'$, then $\mathcal{R}/\lambda(\mathcal{V}_{k-1}^r)\leq\mathcal{R}/\lambda(\mathcal{V}_{k-1}^r)'$. Without loss of generality, we can assume that $d(p_1,v_j^r)$ $=1$. If $\mathcal{R}/\lambda(\mathcal{V}_{k-1}^r)'\leq(\pi/2+1+r)$, for some $r\geq 0$, then $\mathcal{R}\leq(\pi/2+1+r)\,\lambda(\mathcal{V}_{k-1}^r)'$, which implies

$$\sqrt{1-d(p,v_j^r)^2}+(\pi/2+1)\,d(p,v_j^r)\leq(\pi/2+1+r)+r\sum_{i=j}^{k-1} d(v_i^r,v_{i+1}^r).$$

Since $\sum_{i=j}^{k-1} d(v_i^r,v_{i+1}^r)$ can be arbitrarily small, for the expression above to be satisfied we need $\pi/2+1-\sqrt{1-d(p,v_j^r)^2}-(\pi/2+1)\,d(p,v_j^r)\geq -r$. Let $f(x)=\pi/2+1-\sqrt{1-x^2}-(\pi/2+1)\,x$. This function has an absolute minimum in the domain of interest at $x_{min}=(\pi+2)/\sqrt{\pi^2+4\pi+8}$ with $f(x_{min})=\pi/2+1-\frac{1}{2}\sqrt{\pi^2+4\pi+8}$, from which the fact that $r\geq\frac{1}{2}\sqrt{\pi^2+4\pi+8}-\pi/2-1$ follows. Since the competitive ratio $\mathcal{R}/\lambda(\mathcal{V}_{k-1}^r)$ is bounded by $\pi/2+1+r$, we have the following theorem.

Theorem 6.7 *The* Hybrid Kleinberg/Walk-in-Circles Strategy *has a competitive ratio of at most* $\frac{1}{2}\sqrt{\pi^2+4\pi+8}$.

The value $\frac{1}{2}\sqrt{\pi^2+4\pi+8}$ is approximately 2.76.

Hybrid Bisector/Walk-in-Circles Strategy

It is also possible to use the bisector strategy for small initial angles and Walk-in-Circles for large values.

Theorem 6.8 *A Hybrid Bisector/Walk-in-Circles Strategy has better than* $\pi/3 + 1 \sim 2.047197$ *competitive ratio. Numerical computations bound the value to* ≈ 2.030.

Proof. From the discussion in Section 6.1.2, we obtain that for initial angles $\angle v^r p v^l < 2\pi/3$ the bisector strategy is 2-competitive. If the angle is $2\pi/3$ or larger, then the robot uses the strategy *Walk-in-Circles* at a competitive ratio of $\pi - 2\pi/3 + 1 = \pi/3 + 1 \sim 2.047197$.

Notice that it is possible to increase the value of $\mathcal{C}$ from 2 until it is such that both strategies, *Walk-in-Circles* and *Bisector* have the same competitive ratio, which happens at the value of $\mathcal{C}$ such that $\pi - 2\arccos(1/\mathcal{C}) + 1 = \mathcal{C}$. Solving numerically for $\mathcal{C}$ we obtain $\mathcal{C} = 2.0301121$, with starting angle of at most 2.1114804. $\square$

This shows that *Bisector* and *Walk-in-Circles* are relatively good strategies, even though they do not use global information. Once again, this strategy is relatively robust under navigational error.

Chapter 7

Searching $\mathcal{G}$-Streets

In previous chapters we have introduced progressively more general classes of polygons which the robot can search at a constant competitive ratio. In this chapter we focus on the class of $\mathcal{G}$-street polygons, introduced by Datta and Icking [19].

Definition 7.1 *A* **chord** *is a line segment inside a polygon with both endpoints on its boundary.*

Definition 7.2 [**19**] *Consider a simple polygon P and two distinguished points on its boundary (s, t) dividing its boundary into two chains L and R. An LR-***chord** *is a horizontal chord with one endpoint lying on L and the other on R.*

Definition 7.3 [**19**] *Consider a simple polygon P and two distinguished points on its boundary (s, t) dividing its boundary into two chains L and R. Then P is a* **generalized street** *or* $\mathcal{G}$**-street** *if for every boundary point p of P there exists an LR-chord c such that p is weakly visible from c.*

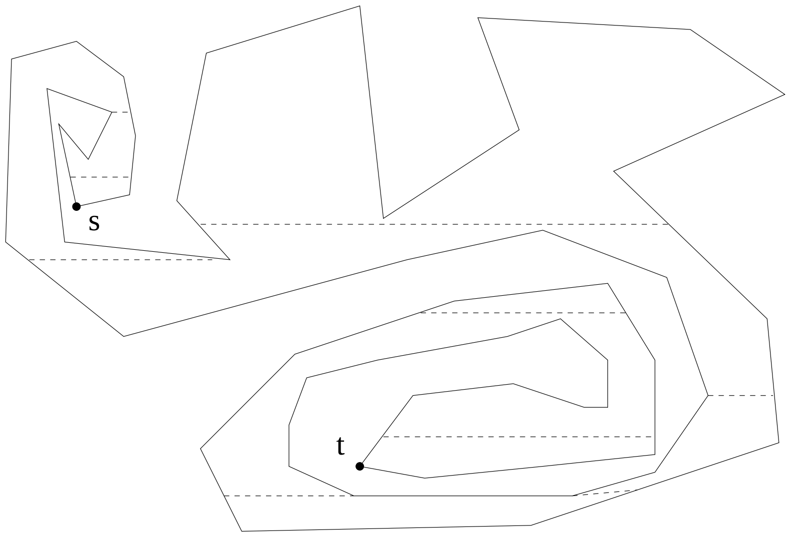

Figure 7.1: A $\mathcal{G}$-street.

For the purposes of this definition we assume that if a point p lies on a horizontal edge e of the polygon, and if e and a LR-chord share a extreme point, then p can see the LR-chord.

Street polygons, as introduced in Chapter 5, form the class of all polygons which can be "crossed", from one side of the street to the other —that is, from L to R— on a straight line. Similarly $\mathcal{G}$-streets can be crossed from a point on L to a point on R using at most two line segments, the second of which is a median divider. Figure 7.1 illustrates a $\mathcal{G}$-street. Note that the choice of median dividers for a given triple (P, s, t) is not necessarily unique.

If LR-chords could have arbitrary directions, it would follow trivially that all streets are $\mathcal{G}$-streets. However, LR-chords are horizontal line segments and thus it is not immediately obvious that all streets have such chords.

Definition 7.4 *The vertices of a polygon P are said to be in general position if no three vertices are collinear and if no two non-consecutive vertices lie on a horizontal*

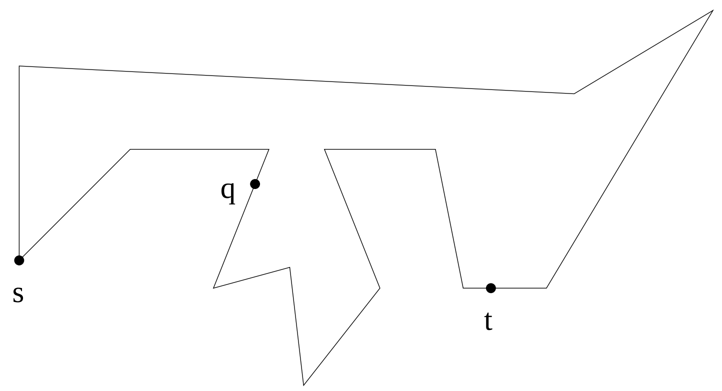

Figure 7.2: A street which is not a $\mathcal{G}$-street.

or vertical line.

Figure 7.2 shows a street polygon which is **not** in general position. Further, this polygon is not a $\mathcal{G}$-street. Notice that all horizontal chords visible from q have both endpoints lying on one of L or R but not in both.

Lemma 7.1 [19, 43] *The class of $\mathcal{G}$-streets is more general than the class of streets with vertices in general position; i.e., every street polygon with vertices in general position is a $\mathcal{G}$-streets and there are $\mathcal{G}$-streets polygons which are not streets.*

Proof. We must show that if (P, s, t) is a street then it is a $\mathcal{G}$-street. That is we must show that for all points on P there exists an LR-chord from which they are weakly visible. Let q_R be a point on the right chain R of P. We know that the left chain L is visible from q_R. That is, there exists a point q_L on the left chain of P such that the line $\overline{q_R q_L}$ is completely contained in P (see Figure 7.3).

Now, shoot horizontal rays from q_R and q_L into the interior of the polygon until they hit the boundary of the polygon. If either of the rays have endpoints on the opposite chain then they form an LR-chord for q_R. Otherwise, for every point on

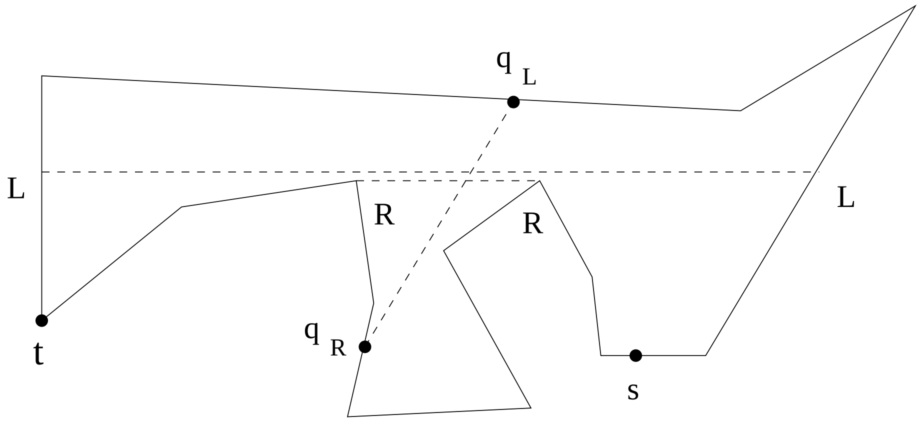

Figure 7.3: Simultaneous switch

$\overline{q_R q_L}$ consider the horizontal chord passing through it. At q_R, this chord has both endpoints on R and at q_L both endpoints are on L. It follows that at some point between q_R and q_L the endpoints must switch from R to L. Again we have two cases: either the points switch simultaneously or they do not. In the latter case, this implies that there are points in $\overline{q_R q_L}$ whose ray is an LR-chord, which is then an LR-chord for q_R as required. In the former case, we have a situation as illustrated in Figure 7.3 in which the chord switches simultaneously from being RR to LL. However this cannot occur, as it is assumed that any two non-consecutive vertices do not lie on a horizontal edge. □

The following lemma was proven by Datta et al.

Lemma 7.2 [19] *Given a $\mathcal{G}$-street polygon (P, s, t), every path from s to t intersects all LR-chords of the polygon.*

7.1 Searching $\mathcal{G}$-Streets with Unknown Destination

Definition 7.5 *An* **orthogonal $\mathcal{G}$-street polygon** *is an orthogonal polygon which is also a $\mathcal{G}$-street.*

The following theorem gives us an optimal competitive ratio for searching $\mathcal{G}$-streets. The upper bound was shown by Datta and Icking [19].

Theorem 7.1 [19, 43] *Orthogonal $\mathcal{G}$-streets can be traversed, in the worst case, at a $\sqrt{82}$-competitive ratio.*

Proof. As usual, we divide this proof into two parts. First, we show that $\mathcal{G}$-streets cannot be traversed at a better than $\sqrt{82}$-competitive ratio.

We present a family of polygons which are traversed at a $\sqrt{82} - \epsilon$ competitive ratio, where ϵ can be arbitrarily small. The family consists of the bow-tie examples shown in Figure 7.4.

Each polygon is made of a roughly triangular section of base and height n and $c \cdot n$ units long, respectively, with ragged edges, and reflected four times as to create the bow tie (see triangle composed of the dashed, dotted and bold lines in Figure 7.4).

Clearly, this polygon is a $\mathcal{G}$-street, as the line passing through the edge containing s is the divider from which one can cross the street. Furthermore, a robot searching the bow-tie polygon using the doubling strategy of Theorem 2.1 on the middle of the polygon (dashed line) can achieve a competitive ratio of $\sqrt{82}$ as the following argument demonstrates.

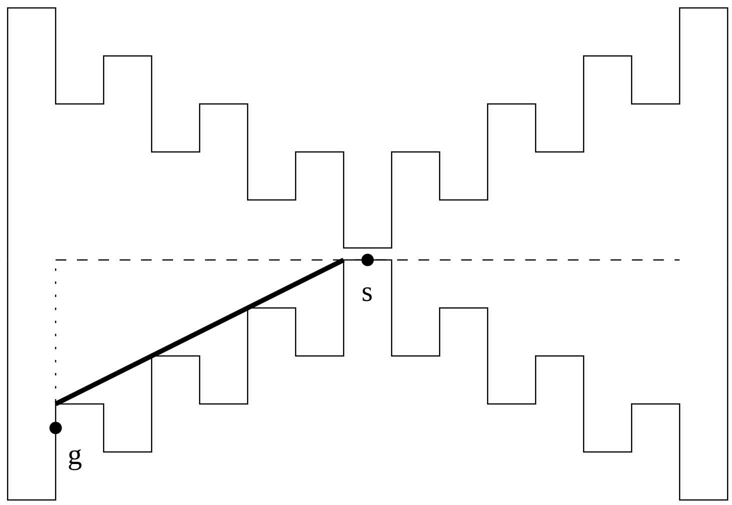

Figure 7.4: Bow-tie polygon.

As the doubling strategy is, in the limit, 9-competitive, for each bow-tie polygon there exists an $\epsilon_n \searrow 0$ as $n \to \infty$, such that the doubling strategy is at least $(9-\epsilon_n)$-competitive in a polygon of width $2n$ and height $2cn$.

The adversary then chooses the bottom of a tooth which is seen at a $(9 - \epsilon_n)$-competitive ratio from the middle line and places the target point in it. Let k be the distance from the projection of the entrance of the chosen dent to the projection of s on the dashed line. Thus the total distance traversed by the robot is $(9-\epsilon_n)k+kc$. On the other hand, the optimal trajectory (bold line) is of length $k\sqrt{1+c^2}$. The competitive ratio $\mathcal{C}$ is then given by

$$\mathcal{C} = \frac{(9-\epsilon_n)+c}{\sqrt{1+c^2}}$$

Differentiation with respect to c gives $c_0 = 1/(9 - \epsilon_n)$ as a global maximum in the interval of interest. Thus we restrict ourselves to bow-tie polygons of base n and width $(1/9)n$. For this value of c, the competitive ratio is $\sqrt{82} - (9/\sqrt{82}) \cdot \epsilon_n$ which goes to $\sqrt{82}$ as $n \to 0$.

Now, we show that the doubling strategy is optimal for bow ties.

Lemma 7.3 *Under the adversary of Theorem 2.1, a robot deviating from the doubling center line strategy has a worse competitive ratio, in the worst case, than the doubling strategy.*

Proof. This proof is remarkably simple. Assume that a robot R_1 deviates from the middle line. The adversary, of course locates the target on the kth dent as before, but on the opposite side to the current location of the robot. Now consider a second robot R_2 which moves on the projection of the R_1 on the middle line. Clearly the R_2 traverses a shorter distance than R_1 and both see the target point at the same time. Thus robot R_2 has a better competitive ratio.

Now, we know that a robot exploring on a line can do no better, in the limit, than the doubling strategy mentioned above, which concludes the proof of the lemma. □

The proof of the lemma concludes the proof of the $\sqrt{82}$ lower bound proposed by the theorem. See [19] for a strategy which is $\sqrt{82}$-competitive. □

7.2 Searching $\mathcal{G}$-Streets with Known Destination

As in the case of other searches, additional knowledge might improve the competitive ratio. We consider searches on a $\mathcal{G}$-street where the location of the target is known.

Theorem 7.2 *Known destination searches on $\mathcal{G}$-streets are at least 9-competitive and no more than $\sqrt{82}$-competitive.*

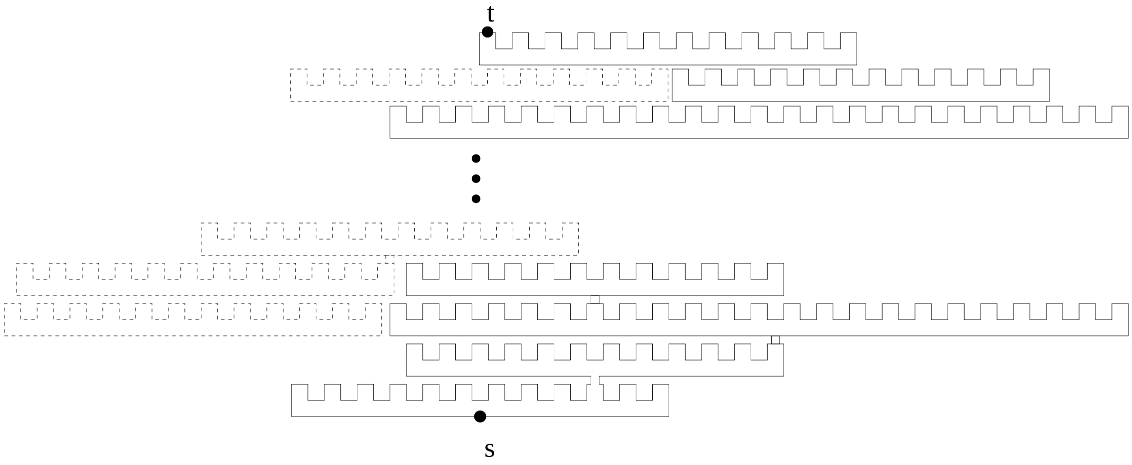

Figure 7.5: A Lego stack polygon.

Proof. The upper bound is given by Theorem 7.1 above. For the lower bound, we exhibit a family of polygons that can only be searched at a competitive ratio of 9.

Let n be the distance between s and t. Without loss of generality let the origin be the initial position of the robot and $(0, n)$ the position of the target. Figure 7.5 shows the polygon (solid lines). The dashed polygons represent other alternative polygons which can also result in polygons within this class. Each polygon in the family is made of M connected rake polygons between s and t. A connection point joins a tooth from the bottom rake to the middle of the top rake.

Rakes are numbered in the order of occurrence on the robot's path from s to t. Each rake i has height n/M or $1/(nM)$; it is symmetrically centered above the entrance point, and has length $2n$. Initially the robot sees only one rake, and searches each tooth for the opening to the next rake. (The robot knows that the target is not within a tooth, as the coordinates of the target are known to the robot, but any one given tooth might be the one leading to the target).

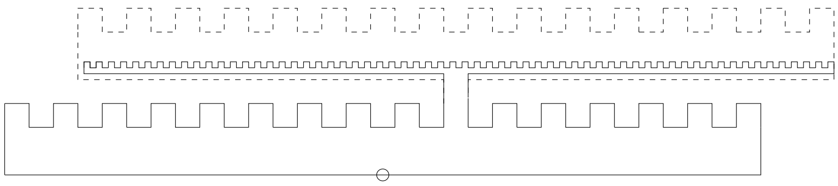

Figure 7.6: Variable height rakes.

As is usual with adversarial arguments, the adversary constructs the polygon on-line depending on the robot's moves. Let r_i be the x-coordinate of the entrance point to the ith polygon (assume that $r_i > 0$). Let D_i be the distance from the entrance point to the exit (connecting) tooth. Let C_i be the ratio of the traversed by the robot in rake i divided over D_i. A sequence $\{w_i\}$ is ϵ-increasing if $w_{i+1} \geq w_i + \epsilon$.

Adversary's Strategy

> The adversary selects a target number of rakes M that make the polygon. The height of each rake is thus, in principle, n/M. The adversary aims to create a polygon with a total optimal distance of at least $nM/2$. This gives an average of $n/2$ units per rake. To achieve this desired optimal path length, the adversary determines the height of each rake as follows:
>
> If on a given rake, the robot forces an optimum path shorter than $n/2$, the adversary makes the n subsequent rakes of height $1/M$ each (see Figure 7.6). Since on each rake the optimum path is at least a unit long, the optimum path is at least n units longer when it reaches the next regular height rake, for an average gain of $N/2$ per each $1/M$ height gain. To be more precise:

- Let $i \leftarrow 1$.

- Without loss of generality, the adversary opens a tooth on the right side, with competitive ratio $C_1 \geq 9 - \epsilon$.
  Let $b \leftarrow 1$; $\quad C_b^F \leftarrow C_1$; $\quad D_b^F \leftarrow D_1$.

- For each i from 2 to M do

  Case 1: If the robot reaches a tooth in $[0, r_i]$ with competitive ratio C_i such that $\frac{1}{2}(C_i + C_b^F) \geq 9 - \epsilon/2$, then the adversary opens that tooth.

  - If $D_b^F - D_i < n/2$ then
    Let $D^F b - 1 \leftarrow D_b^F - D_i + D_{b-1}^F$.
    Let $C_{b-1}^F \leftarrow (C_b^F(D_b^F - D_i) + C_{b-1}^F D_{b-1}^F)/D_{b-1}^F$.
    Let $b \leftarrow b - 1$.
  - Else let $D_b^F \leftarrow D_b^F - D_i$.

  Case 2: Else let $b \leftarrow b + 1$; $\quad C_b^F \leftarrow C_i$; $\quad D_b^F \leftarrow D_i$.
  Invariant: the sequence of competitive ratios C_b^F is ϵ-increasing.
  The adversary opens an alley to the right of the entrance point at a competitive ratio $C_i = C_i^R$ such that $\frac{1}{2}(C_i^L + C_i^R) \geq 9$.

- In the Mth polygon, the robot knows that its present position is horizontally aligned with the target and moves directly towards it. In this case, the adversary does not oppose the robot's move, and the robot reaches the target optimally within R_M.

For case 2, first note that, if the invariant holds, then theorem 2.5 implies that it is always possible to choose an entrance point as requested. To prove the invariant we note that if we are in case 2, then the worst-case competitive ratio for all points on the left C_i^L is such that $\frac{1}{2}(C_i^L + C_b^F) \leq 9 - \epsilon/2$ which implies $C_b^F \leq 18 - C_i^L - \epsilon$.

But we know from theorem 2.5 that $\frac{1}{2}(C_i^L + C_i^R) \geq 9$. Thus $C_i^R \geq 18 - C_i^L$ which implies $C_i^R \geq 18 - C_i^L \geq C_b^F + \epsilon$.

Thus case 2 ensures that, if the exit alley is to the right, the competitive ratio increased at least by ϵ, while case 1 ensures that if the alley is on the left, the robot traverses at least $n/2$ units which **together** with a previous right move balance out to an over 9-competitive ratio. In this case, the step is eliminated from the sequence of right moves as it has been "cancelled out" by the left move. Let $\epsilon = 1/n^2$ and $M = n^4$.

It follows that if the the robot follows a strategy which has only case 2 adversarial moves, the robot reaches the last polygon having traversed a distance of at least $(9 + M/n^2)/2 \times n^4$, and it is n^4 units away from the target, for a total competitive ratio of $(9 + n^2)/2 + 1$ which is arbitrarily large. Therefore the robot must choose a number of case 1 moves. If all of moves are case 1, once again we obtain a trivial lower bound of 9 for the competitive ratio. As we shall see, the total distance traversed by the robot is at least

$$b \cdot \frac{n}{2}\frac{(9 + b\epsilon)}{2} + b\frac{n}{2} + 9n(M - b) + 5\frac{n}{2}\frac{b}{2}.$$

The first term denotes the fact that in each of the b case 2 configurations the robot traversed at least $n/2$ units. The competitive ratio, for the first $n/2$ units is the average of all competitive ratios in $\mathcal{R}$ which comes to at least $(9+b\epsilon)/2$; this is so as each case 1 move is at least ϵ times worse in competitive ratio as the previous one. As the movement in the first term was to the right, the second term denotes the optimal trajectory all the way back (left) to the target. The third term expresses the fact that in the remaining other $M - b$ cases, the competitive ratio was at least 9, and the total distance traversed was at least n ($n/2$ units to the right and $n/2$ units back to the left). The last term accounts for the fact that the robot may traverse between $n/2$ and n units at any competitive ratio. Thus the robot may wish to maximize the distance traversed at "low" competitive ratios which occur at the beginning. The lowest competitive ratio to the right is 9, and each distance must be traversed to and fro, for a total competitive ratio of $(9+1)/2 = 5$. Such low competitive ratio can be attained in at most half of the case 2 situations.

The optimal distance is given then by

$$bn + n(M - b) + n\frac{b}{2}.$$

By differentiating we see that the competitive ratio is maximized when either $b = 0$ or $b = -2n^4 + 2\sqrt{n^8 + 19n^6}$. In the first case is easy to see that the competitive ratio is 9. For the second case, substituting we obtain

$$\begin{aligned}
2n\sqrt{n^2+19} - 2n^2 - 10 &= \frac{2n(\sqrt{n^2+19} - n)(\sqrt{n^2+19} + n)}{\sqrt{n^2+19} + n} - 10 \\
&= \frac{2 \cdot 19 \cdot n}{\sqrt{n^2+19} + n} - 10 = \frac{38}{\sqrt{1 + 19/n^2} + 1} - 10 \\
&= 9
\end{aligned}$$

as required. $\square$

Chapter 8

Conclusions

In this thesis we have made progress towards defining classes of polygons for which there exist constant competitive ratio search strategies.

In Chapter 2, we presented the advantages of a competitive ratio analysis which explains the transition from the game theoretical approach to a competitive ratio measure. We present a general proof of optimality for the 9-competitive search on the real line, and extend this result for the case of unbalanced strategies, in which it is shown that favouring searches on one side does not improve worst case searches on the average. We then extend this result to a more realistic setting of a kinetic model in which the robot has maximum speed and acceleration rates.

Some of the most relevant results in the field are described in Chapter 3. This gives a backdrop against which to place original work from latter chapters, as well as introduce some of techniques used in subsequent chapters, such as spiral searches.

In Chapter 4, we examine the feasibility of constant competitive ratio searches inside a general polygon, which naturally introduces the concept of kernel searches. In this area we present the first non-trivial lower bound for kernel searches in a star polygon. As well we exhibit a strategy which is $\sqrt{4+(2+\pi)^2}$-competitive for *on-the-other-side* target searching in star polygons; as well as a 17-competitive strategy for searching for a target in a star polygon. We provide a 9-competitive ratio lower bound for the same problem.

Then we study the problem of target searching on the class of street polygons. For orthogonal polygons we show that knowing the location of the target provides no advantage over the unknown destination case and is thus $\sqrt{2}$-competitive. To our knowledge, this is the first known destination search problem known to have the same competitive ratio as the associated unknown destination version.

In Chapter 6 we present then several strategies which combined result in a strategy for non-orthogonal streets with a 1.68 competitive ratio. This significantly improves over the previously best known strategy of ≈ 2.8. We then present two strategies, *Walk-in-Circles* and *Bisector*, the former is robust under navigational errors and oblivious, while the latter does not require depth perception mechanisms (triangulation) on the robot's vision system.

If we assume that it indeed is possible to search street polygons at a $\sqrt{2}$-competitive ratio, as it is widely believed, then one can show that such strategy must also respond to the absence of new landmarks, in contrast to most strategies proposed thus far in the literature.

We also show that a particular strategy proposed by Kleinberg, which is optimal for rectilinear street polygons, is exactly $2\sqrt{1+1/\sqrt{2}}$-competitive for general streets. For this we present a lower bound and the first proof of the upper bound in the literature.

In Chapter 7, we present a matching $\sqrt{82}$ lower bound for searches in the class of orthogonal $\mathcal{G}$-streets. For searches with known destination in $\mathcal{G}$-streets we prove a lower bound of 9, based on the unbalanced search theorem of Chapter 2, and an upper bound of $\sqrt{82}$ for searches with known destination on the same class of polygons.

Several questions remain unanswered. It is not known if the randomized cow path strategy is optimal. For the case of walking into the kernel, there remains a large gap between the upper and lower bound, which must be reduced. We conjecture that the lower bound can be improved somewhat, possibly to $\pi/2$ and the upper bound to $2 + 2\sqrt{2}$.

For street polygons, a small gap between the upper and lower bound remains to be closed. Another important area of study is to better the competitive ratio of constrained searches, such as oblivious or stroboscopic searches.

For orthogonal $\mathcal{G}$-streets, there remains a small gap between the lower and upper bounds for known destination searches, which should be closed. As well, it is an open problem if there are strategies with a constant competitive ratio for general (ie. non-orthogonal) street polygons. In fact, we have strong evidence that this is indeed possible, and we expect to have results on this problem in the near future.

Bibliography

[1] R. Baeza-Yates, J. Culberson and G. Rawlins. "Searching in the plane", *Information and Computation*, Vol. **106**, (1993), pp. 234-252.

[2] R. Baeza-Yates, R. Schott. "Parallel searching in the plane", *Computational Geometry: Theory and Applications*, Vol. **5**, (1995), pp. 143-154.

[3] E. Bar-Eli, P. Berman, A. Fiat and Peiyuan Yan. "On-line navigation in a room", *Proceedings of 7th ACM-SIAM Symposium on Discrete Algorithms*, (1992), pp. 237-249.

[4] T. Basar and P. Bernhard, Eds. *Differential Games and Applications*, Lecture Notes in Control and Information Sciences, Vol. **119**, Springer-Verlag, (1989).

[5] A. Beck. "On the linear search problem", *Israel Journal of Mathematics*, Vol. **2**, (1964), pp. 221-228.

[6] A. Beck. "More on the linear search Problem", *Israel Journal of Mathematics*, Vol. **3**, (1965), pp. 61-70.

[7] A. Beck and D. J. Newman. "Yet more on the linear search problem", *Israel Journal of Mathematics*, Vol. **8**, (1970), pp. 419-429.

[8] P. Berman, A. Blum, A. Fiat, H. Karloff, A. Rosén and M. Sacks. "Randomized robot navigation algorithms", *Proceedings of 7th ACM-SIAM Symposium on Discrete Algorithms*, (1996).

[9] P. Billingsley. *Probability and Measure*, 2nd ed., John Wiley & Sons, (1986), pp. 234-252.

[10] A. Blum and P. Chalasani. "An on-line algorithm for improving performance in navigation" *Proc. 34th IEEE Symposium on Foundations of Computer Science*, (1993), pp. 2-11.

[11] A. Blum, P. Raghavan and B. Schieber. "Navigating in unfamiliar geometric terrain", *Proceedings of 23rd ACM Symposium on Theory of Computing*, (1991), pp. 494-504.

[12] Canadian Coast Guard/Garde Cotiere Canadienne. *Merchant ship search and rescue manual*, (CANMERSAR), (1986).

[13] K-F. Chan and T. W. Lam. "An on-line algorithm for navigating in an unknown environment", *International Journal of Computational Geometry & Applications*, Vol. **3**, (1993), pp. 227-244.

[14] L. Cheng and J. D. McKendrick. "Autonomous based navigation in an unknown two dimensional environment with convex polygonal obstacles", *Proceedings of the International Society for Opt. Eng.,* Vol. **1095**, (1989), pp. 752-759.

[15] P. Colley, H. Meijer and D. Rappaport. "Motivating lazy guards", *Proceedings of the 7h Canadian Conference on Computational Geometry*, (1995), pp. 121-132.

[16] D. Crass, I. Susuki and M. Yamashita. "Searching for a mobile intruder in a corridor", *International Journal of Computational Geometry & Applications* Vol. **5**, (1995), pp. 397-412.

[17] M. Daily, J. Harris, D. Keirsey, D. Olin, D. Payton, K. Reiser, J. Rosenblatt, D. Tseng and V. Wong. "Autonomous cross-country navigation with the ALV", *Proceedings of the IEEE International Conference on Robotics and Automation,* Vol. **2**, (1988), pp. 718-726.

[18] G. Das, P. Heffernan and G. Narasimhan. "LR-Visibility in polygons", *Proceedings of the Fifth Canadian Conference on Computational Geometry,* (1993), pp. 303-308.

[19] A. Datta and Ch. Icking. "Competitive searching in a generalized street", *Proceedings 10th ACM Symposium on Computational Geometry*, (1994), pp. 175-182.

[20] X. Deng, T. Kameda and C. Papadimitriou. "How to learn an unknown environment I: the rectilinear case", *Technical Report CS-93-04*, Department of Computer Science, York University, 1993. Also as *Proceedings 32nd IEEE Symposium on Foundations of Computer Science*, (1991), pp. 298-303.

[21] J. M. Dobbie. "A survey of search theory", *Operations Research*, Vol. **16**, (1968), pp. 525-537.

[22] R. C. Fawdry. *Co-ordinate Geometry for Beginners*, G. Bell and Sons, (1962).

[23] A. Friedman. *Differential games*, Conference Board of the Mathematical Sciences by the American Mathematical Society, (1974).

[24] S. Gal. *Search Games*, Academic Press, 1980.

[25] S. Gal. "A general search game", *Israel Journal of Mathematics* Vol. **12**, (1972), pp. 32-45.

[26] S. Gal. "On the optimality of the exponential functions for some minimax problems", *SIAM Journal of Applied Mathematics* Vol. **30**, (1976), pp. 205-210.

[27] S. Gal. "A stochastic search game", *SIAM Journal of Applied Mathematics* Vol. **34**, (1978), pp. 205-210.

[28] S. Gal. "Search games with mobile and immobile hider", *SIAM Journal on Control and Optimization* Vol. **17**, (1979), pp. 99-122.

[29] O. Hájek. *Pursuit Games*, Mathematics in Science and Engineering, Vol. 120, Academic Press, 1975.

[30] A. Hemmerling. "Navigation without perception of coordinates and distances". *Technical Report TR-93-018*, International Computer Science Institute, Berkeley, CA, March 1993.

[31] Ch. Icking. "Motion and visibility in simple polygons". Ph. D. Thesis, Fernuniversität Hagen, 1994.

[32] Ch. Icking and R. Klein. "The two guards problem", *Proceedings 7th ACM Symposium on Computational Geometry*, (1991), pp. 166-175.

[33] Ch. Icking, R. Klein and L. Ma. "How to look around a corner", *Proceedings 5th Canadian Conference on Computational Geometry*, (1993), pp. 443-448.

[34] Ch. Icking and R. Klein. "Searching for the kernel of a polygon. A competitive strategy", *Proceedings 11th ACM Symposium on Computational Geometry*, (1995).

[35] B. Kalyasundaram and K. Pruhs. "A competitive analysis of algorithms for searching unknown scenes", *Computational Geometry: Theory and Applications*, Vol. **3**, (1993), pp. 139-155.

[36] M.-Y. Kao, J. H. Reif and S. R. Tate. "Searching in an unknown environment: An optimal randomized algorithm for the cow-path problem", *Proceedings of 4th ACM-SIAM Symposium on Discrete Algorithms*, (1993), pp. 441-447.

[37] R. Klein. "Walking an unknown street with bounded detour", *Computational Geometry: Theory and Applications*, Vol. **1**, (1992), pp. 325-351.

[38] J. Kleinberg. "On-line search in a simple polygon", *Proceedings of 5th ACM-SIAM Symposium on Discrete Algorithms*, (1994), pp. 8-15.

[39] J. Kleinberg. "The localization problem for mobile robots", *Proceedings 35th Symposium on Foundations of Computer Science*, (1994), pp. 520-531.

[40] B. O. Koopman, "Search and screening", Report No. 56 (ATI 64 627), Operations Evaluation Group, *Office of the Chief of Naval Operations*, Washington, D.C., 1946.

[41] B. O. Koopman, "The theory of search, I, III and III", *Operations Research*, Vols. **4, 5**, 1956-1957.

[42] D.T. Lee and F.P. Preparata. "An optimal algorithm for finding the kernel of a polygon", *Journal of the ACM*, Vol. **26**, (1979), pp. 415-421.

[43] A. Lopez-Ortiz. "Stronger bounds for on-line searches", Manuscript, 1995.

[44] A. Lopez-Ortiz and S. Schuierer. "Going home through an unknown street", *Technical Report 445*, Department of Computer Science, The University of Western Ontario, 1994.

[45] A. Lopez-Ortiz and S. Schuierer. "Going home through an unknown street", *Proceedings of 4th Workshop on Data Structures and Algorithms*, 1995, Lecture Notes in Computer Science 955, Springer-Verlag, pp. 135-146.

[46] A. Lopez-Ortiz and S. Schuierer. "Simple, efficient and robust strategies to traverse streets", *Proceedings of the 7th Canadian Conference on Computational Geometry*, 1995, pp. 217-222.

[47] A. Lopez-Ortiz and S. Schuierer. "Walking streets faster", *Technical Report CS-95-20*, Department of Computer Science, University of Waterloo, 1995.

[48] V. J. Lumelsky and A. A. Stepanov. "Path-planning strategies of a point mobile automaton moving amidst unknown obstacles of arbitrary shape", *Algorithmica,* **2**, (1987), pp. 403-430.

[49] A. Mei and Y. Igarashi. "Efficient strategies for robot navigation in unknown environment", *Proc. of 21st International Colloquium on Automata, Languages and Programming*, (1994).

[50] E. Moise. *Elementary Geometry from an Advanced Standpoint*, 2nd ed., Addison-Wesley, 1973.

[51] National Search and Rescue Secretariat/Secrétariat national Recherche et sauvetage. "CANSARP", SAR*Scene*, Vol. **4**, July 1994.

[52] M. F. Neuts. "A multistage search game", *Journal of the Society for Industrial and Applied Mathematics*, Vol. **11**, June 1963.

[53] J. O'Rourke. *Art Gallery Theorems and Algorithms*, Oxford University Press, 1987.

[54] J. O'Rourke. *Computational Geometry in C*, Cambridge University Press, 1993.

[55] C. H. Papadimitriou, P. Raghavan, M. Sudhan and H. Tamaki. "Motion planning on a graph", *Proceedings 35th Symposium on Foundations of Computer Science*, (1994), pp. 511-520.

[56] C. H. Papadimitriou and E. B. Silverberg. "Optimal piecewise linear motion of an object among obstacles", *Algorithmica,* **2**, (1987), pp. 523-539.

[57] C. H. Papadimitriou and M. Yannakakis. "Shortest paths without a map", *Theoretical Computer Science,* **84**, (1991), pp. 127-150.

[58] F. P. Preparata, M. I. Shamos. *Computational Geometry,* Springer-Verlag, New York, 1985.

[59] W. H. Ruckle. *Geometric Games and their Applications,* Pitman Publishing, 1983.

[60] I. Stewart. "The great drain robbery" in Mathematical Recreations. *Scientific American,* **264**, September 1995, pp. 206-207.

[61] V. G. Shervatov. *Hyperbolic Functions,* Topics in Mathematics, D.C. Heath, 1963.

[62] D. D. Sleator and R. E. Tarjan. "Amortized efficiency of list update and paging rules", *Communications of the ACM,* **28**, (1985), pp. 202-208.

[63] L. D. Stone. *Theory of Optimal Search,* Academic Press, (1975).

[64] I. Susuki and M. Yamashita. "Searching for a mobile intruder in a polygonal region", *SIAM Journal on Computing* Vol. **21**, (1992), pp. 863-888.

[65] US Coast Guard Fact File 1995.