Simple, Efficient and Robust Strategies to Traverse Streets*

Alejandro López-Ortiz[†] Sven Schuierer[‡]

Abstract

We present a family of strategies for the problem of searching in an unknown street for a target of unknown location. 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. Although this ratio is worse than the ratio of the best previously known strategy, which achieves a detour of at most $2\sqrt{2} \sim 2.8284$ times the length of a shortest path, the simplicity of the analysis is interesting in its own. We 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 $\frac{1}{2}\sqrt{\pi^2 + 4\pi + 8} \sim 2.75844$. The $\pi + 1$-competitive strategy is very similar in spirit to the first published strategy of which the best analysis is very involved and gives a bound of ~ 4.44. More importantly, we show that the $\pi + 1$ strategy is robust under small navigational errors.

1 Introduction

One of the main problems in robotics is to find a path from the current location of the robot to a given goal of unknown location, particularly in those cases where the robot has only a partial knowledge of its surroundings.

In this paper we assume that the robot is equipped with a vision system that provides a visibility map of its *local* environment. Based on this information the robot has to find a path to a visually identifiable given goal that is located somewhere within the scene. The search of the robot can be viewed as an on-line problem in which the amount of information available to the robot increases as it discovers its surroundings in its travels. A natural measure of the quality of a search strategy is to use the framework of competitive analysis as introduced by Sleator and Tarjan [12]. A search strategy is called *c-competitive* if the path traveled by the robot to find the goal is a most c times longer than a shortest path. The parameter c is called the *competitive ratio* of the strategy.

Since there is no strategy with a competitive ratio of $o(n)$ for scenes with arbitrary obstacles having a total of n vertices [2], the on-line search problem has been studied previously in various contexts where the geometry of the obstacles is restricted [1, 2, 3, 4, 10, 11].

Klein introduced the notion of a *street* which allowed for the first time a search strategy with a constant competitive ratio [7]. In a street, the starting points s and the goal g are located on the boundary of the polygon and the two polygonal chains from s to g are mutually weakly visible. Klein presents a strategy for searching in streets and gives an upper bound on its competitive ratio of $1 + 3/2\pi$ (~ 5.71). The analysis was recently improved to $\pi/2 + \sqrt{1 + \pi^2/4}$ ($\sim$ 4.44) by Icking [6]. Though Klein's strategy performs well in practice—he reports that no example had been found for which his strategy performs worse than 1.8—the strategy and its analysis are both quite involved and no better competitive ratio could be shown until, recently, Kleinberg presented a new approach.

In this paper we present a $\pi + 1$ (~ 4.14) analysis of a strategy similar to Klein's. This analysis is significantly simpler than other published work [7, 9]. It also has the advantage that the strategy proposed is robust under small navigational errors. The simplicity of the strategy and analysis points naturally to possible improvements in the strategy. To illustrate this we present a hybrid method which

*This research is partially supported by the DFG-Project "Diskrete Probleme", No. Ot 64/8-1.

[†]Department of Computer Science, University of Waterloo, Waterloo, Ontario CANADA N2L 3G1, e-mail: alopez-o@neumann.UWaterloo.ca

[‡]Department of Computer Science, University of Western Ontario, London, Ont., Canada N6A 5B7, and Institut für Informatik, Universität Freiburg, Am Flughafen 17, D-79110 Freiburg, FRG, e-mail: schuiere@informatik.uni-freiburg.de

uses the $\pi + 1$-competitive strategy and results in a $\frac{1}{2}\sqrt{\pi^2 + 4\pi + 8} \sim 2.75844$ that betters the best previously known of $2\sqrt{2}$-competitive ratio [8].

The rest of this paper is organized as follows. In the next section we introduce some notation and definitions. Then the family of strategies is described in Section 3, and in Section 4 we present its analysis. In the next to last section we present and analyze the hybrid strategy and then we conclude with some observations and directions of further research.

2 Definitions and Assumptions

We consider a simple polygon P in the plane with n vertices and a robot inside P which is located at a start point s on the boundary of P. The robot has to find a path from s to the goal g. The search of the robot is aided by simple vision (i.e. we assume that the robot knows the visibility polygon of its current location). Furthermore, the robot retains all the information seen so far (in memory) and knows its starting and current position. We are, in particular, concerned with a special class of polygons called *streets* first introduced by Klein [7].

Definition 2.1 [7] *Let P be a simple polygon with two distinguished vertices, s and g, and let L and R denote the clockwise and counterclockwise, resp., oriented boundary chains leading from s to g. 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, g) is called a* street.

We denote the L_2-distance between two points p_1 and p_2 by $d(p_1, p_2)$ and the L_2-norm of a point p by $||p||$.

Definition 2.2 *Let P be a street with start point s and goal g. If p is a point of P, then the* visibility polygon *of p is the set of all points in P that are seen by p. It is denoted by $V(p)$.*

Definition 2.3 *A* window *of $V(p)$ is an edge of $V(p)$ that does not belong to the boundary of P (see Figure 1).*

A window w splits P into a number of subpolygons $P_1, \ldots, P_k$ one of which contains $V(p)$. We denote the union of the subpolygons that do not contain $V(p)$ by P_w.

All windows are collinear with p. The end point of a window w that is closer to p is called the *entrance*

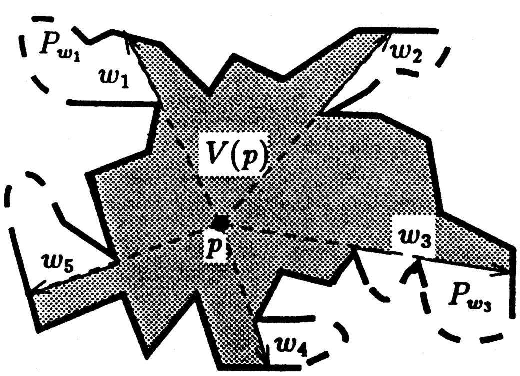

Figure 1: The the visibility polygon $V(p)$ of p with windows $w_1, \ldots, w_5$.

point of w. We assume that a window w has the orientation of the ray from p to the entrance point of w. We say a window w is a *left window* if the part P_w of P that does not contain $V(p)$ is locally to the left of w w.r.t. the given orientation of w. A *right window* is defined similarly.

Definition 2.4 *Two windows w_1 and w_2 are* clockwise consecutive *if the clockwise oriented polygonal chain of $V(p)$ between w_1 and w_2 does not contain a window different from w_1 and w_2.* Counterclockwise consecutive *is defined analogously.*

3 A Family of Strategies

As observed by Kleinberg the shortest path $\mathcal{P}$ from s to g consists of a number of line segments that touch reflex vertices of P. The general strategy we follow is to start at a reflex vertex v of P that belongs to $\mathcal{P}$ and to identify another reflex vertex v' of $\mathcal{P}$ that is closer to g by traveling further on. If the robot has identified v', then it moves to it and starts the search anew. A move from one reflex vertex of P on $\mathcal{P}$ to another closer to g is called a *step*.

If the robot has traveled along the path $\mathcal{P}$, then we assume that the robot knows the part of P that can be seen from $\mathcal{P}$, i.e. the robot maintains the polygon $V(\mathcal{P}) = \bigcup_{p \in \mathcal{P}} V(p)$. We say a window w of $V(p)$ is a *true window* w.r.t. $\mathcal{P}$ if it is also a window of $V(\mathcal{P})$.

In the following we present the relevant results about true windows from [9].

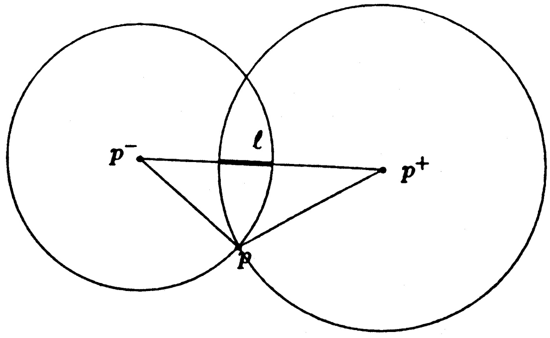

Figure 2: The subsegment of target points.

Lemma 3.1 *If w is a right (left) window of $V(p)$ and the boundary of P_w belongs to L (R), then w is not a true window.*

Lemma 3.2 *All windows that belong to L (R) are clockwise (counterclockwise) consecutive in $V(p)$.*

True windows are called *consecutive* if there is no true window that is between them. An immediate corollary of Lemmas 3.1 and 3.2 is that true left and true right windows are consecutive.

Corollary 3.3 *If w_0 is the window that is intersected by $\mathcal{P}$ the first time, then all true left (right) windows are clockwise (counterclockwise) consecutive from w_0 in $V(p)$.*

Because of Corollary 3.3 there is a clockwise-most true left entrance point from w_0 which we denote by p^+ and a counterclockwise-most true right entrance point of $V(p)$ which we denote by p^- provided that $V(p)$ contains both true left and right windows. The point p^+ is called the *left extreme entrance point* and p^- the *right extreme entrance point* of $V(p)$. As observed by Klein, it is only when $V(p)$ contains both true left and right windows when the optimal strategy is unclear. Thus, our strategy mimics Klein's strategy for the cases without two true windows [7, 9]. Namely, the cases are:

Case 1. The goal g is visible to the robot. The robot moves to g on a straight line.

Case 2. There is no true left (right) window. The robot moves to p^- (p^+).

Case 3. The points p, p^+, and p^- are collinear. The robot moves along the line $\overline{pp^+}$ to the closer point of p^+ and p^-.

In the case of two true windows, the robot has to determine the trajectory to follow. As proposed by Klein, the robot selects a target point t_i in the line $\overline{p_i^+p_i^-}$. The robot moves then in a straight line towards t_i until either of the entrance points has changed or the goal has been identified and reached. There are several possible criteria to select t_i. Klein studied the case where t_i balances the current absolute detour, as compared to the possible optimal trajectory, which is either the line joining the chain of left points p_i^- or joining the right points p_i^+, depending on the actual location of the goal. Different criteria may be used to select the target point t_i. In particular, this leads to the main family of stategies:

Strategy Walk-in-Circles. Let ℓ be the subsegment of $\overline{p_i^-p_i^+}$ which consists of the points t such that $d(p_i^-,t) \le d(p_i^-,p_i)$ and $d(p_i^+,t) \le d(p_i^+,p_i)$ (see Figure 2). The algorithm chooses a point t_i in the target segment ℓ and moves in a straight line towards it. If a new window appears, the robot recomputes ℓ according to the updated points p_{i+1}^+ and p_{i+1}^-, and the new position p_{i+1}, until the goal is found. (see Figure 3).

4 Analysis

We 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 trajectory traversed by the robot is determined by the sum of the length of all segments $\overline{p_ip_{i+1}}$, i.e. $\sum_{i=0}^{n-1} d(p_i,p_{i+1})$, where n is the number of extreme entrance points seen by the robot in a step. Note that $p_0 = s$ and $p_n^+ = g$. The length of each of these segments can be bounded by using the triangle inequality; viz. with notation as in Figure 3, we have that $d(p_i,p_{i+1}) \le d(p_i,q) + d(q,p_{i+1})$, where q is the point determined by the intersection of the line $p_{i+1}p_i^+$ and the circle centered at p_i^+ passing through p_i.

In turn the length of $d(p_i,q)$ is bounded by the length of the circular arc p_iq. Let $\alpha_i = \angle qp_i^+p_i$ measured in radians. The length of the circular arc p_iq is given by $\alpha_i \cdot d(p_i,p_i^+)$. Thus,

$$\begin{aligned}\sum_i d(p_i,p_{i+1}) &\le \sum_i (d(p_i,q) + d(q,p_{i+1})) \\ &\le \sum_i \left(\alpha_i \cdot d(p_i,p_i^+) + d(q,p_{i+1})\right),\end{aligned}$$

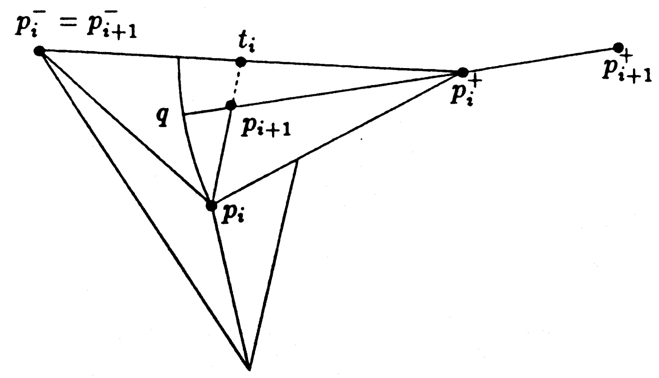

Figure 3: A single step in the strategy.

and the competitive ratio is determined by

$$\frac{\sum_i d(p_i, p_{i+1})}{Opt} \leq \frac{\sum_i \alpha_i \cdot d(p_i, p_i^+) + \sum_i d(q, p_{i+1})}{Opt};$$

where Opt is the length of the optimal walk from the starting point to the goal. Note that, if the final target is on the right chain as assumed, then $Opt = d(p_0, p_0^+) + \sum_{i=0}^{n-1} d(p_i^+, p_{i+1})$.

The following two lemmas allows us to simplify the expression above. These lemmas follow quite naturally from the diagrams, and we provide a formal proof only for completeness.

Lemma 4.1 *Let D_i denote the length of the optimal walk from point p_i to the final target p_n^+. Then $D_{i+1} = D_i - d(q, p_{i+1})$.*

Proof: If the target is located on the right side, the optimum trajectory from any given point is to move on a straight line to the uppermost visible point on the right chain, and follow the chain of points $p_i^+ p_{i+1}^+$ from then onwards. From point p_i, the length of the optimum trajectory is then $D_i = d(p_i, p_i^+) + \sum_{j=i}^{n-1} d(p_j^+, p_{j+1}^+)$, and after moving to p_{i+1} is $D_{i+1} = d(p_{i+1}, p_i^+) + \sum_{j=i}^{n-1} d(p_j^+, p_{j+1}^+) = d(p_i, p_i^+) - d(q, p_{i+1}) + \sum_{j=i}^{n-1} d(p_j^+, p_{j+1}^+) = D_i - d(q, p_{i+1})$ as required, since q is located on the circle centered at p^+ and passing through p_i. □

At the starting position the distance D_0 is precisely the length of the right chain walk $d(p_0, p_0^+) + \sum_{i=0}^{n-1} d(p_i^+, p_{i+1}^+)$, and at the end of the walk the robot finds itself at a zero distance from the target point (i.e. the robot is at the target point). Then the sum of the actual gains overall must be $\sum_i d(q, p_{i+1}) = \sum_i d(p_i^+, p_{i+1}^+)$. Thus

$$\begin{aligned}\frac{\sum_i d(p_i, p_{i+1})}{Opt} &\leq \frac{\sum_i \alpha_i \cdot d(p_i, p_i^+) + \sum_i d(q, p_{i+1})}{Opt} \\ &= \frac{\sum_i \alpha_i \cdot d(p_i, p_i^+)}{Opt} + 1.\end{aligned}$$

The term $\sum_i \alpha_i \cdot d(p_i, p_i^+)$ can be seen as a weighted sum, where the α_is are the weights. Let $\beta = \sum_i \alpha_i$, and let k be such that p_k is the point in the robot's trajectory such that $d(p_k, p_k^+) \geq d(p_i, p_i^+)$. In other words, p_k denotes the largest term in the unweighted sum. Then we have $\sum_i \alpha_i \cdot d(p_i, p_i^+) \leq \sum_i \alpha_i \cdot d(p_k, p_k^+) = \beta \cdot d(p_k, p_k^+)$.

Lemma 4.2 *The distance $d(p_k, p_k^+)$ is no larger than the length of the polygonal chain $Opt = d(p_0, p_0^+) + \sum_{i=0}^{n-1} d(p_i^+, p_{i+1}^+)$.*

We actually prove a stronger result, namely that $d(p_i, p_i^+) \leq d(p_0, p_0^+) + \sum_{j=0}^{i-1} d(p_j^+, p_{j+1}^+)$.

Proof: By induction on the number of steps i. When $i = 0$, the two terms are equal and thus the inequality holds. For $i+1$, we have that $d(p_{i+1}, p_{i+1}^+) = d(p_i^+, p_{i+1}^+) + d(p_i, p_i^+) - d(p_{i+1}, q)$ (see Figure 3); and by induction hypothesis, $d(p_{i+1}, p_{i+1}^+) \leq d(p_i^+, p_{i+1}^+) + d(p_0, p_0^+) + \sum_{j=0}^{i-1} d(p_j^+, p_{j+1}^+) - d(p_{i+1}, q) \leq d(p_0, p_0^+) + \sum_{j=0}^{i} d(p_j^+, p_{j+1}^+)$. □

This implies

$$\frac{\sum_i d(p_i, p_{i+1})}{Opt} \leq \frac{\beta \cdot d(p_k, p_k^+)}{Opt} + 1 \leq \beta + 1.$$

Lastly, as it was noted by Klein (see proof of lemma 2.7 in [7]), if the angle $\angle p_i^- p_i p_i^+$ ever exceeds π then at the point where the angle was π –or possibly even before– there must have been no true left window. In this case, the robot moves to the current p^+ with competitive ratio bounded by $\pi + 1$.

As a consequence, the trajectory can be analyzed in two parts. First, until the robot moves to the point p^+ as a "temporal target", and second, the search afterwards, in which we start anew from a point on the right chain onwards towards the goal. The robot then recurses in the second search, and the total competitive ratio is bounded by the maximum of the competitive ratio on both parts.

Notice that $\angle p_{i+1}^- p_{i+1} p_{i+1}^+ \geq \angle p_i^- p_i p_i^+ + \alpha_i$, and thus $\pi \geq \angle p_n^- p_n p_n^+ \geq \angle p_0^- p_0 p_0^+ + \sum_i \alpha_i \geq \beta$. From which follows that the competitive ratio for each part of the algorithm is, at worst,

$$\frac{\sum_i d(p_i, p_{i+1})}{Opt} \leq \pi + 1.$$

Theorem 4.3 *A robot moving traveling under the strategy* Walk-in-Circles *has a* $\pi+1$ *competitive ratio.*

As the target in each step is selected from the interval ℓ 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 smallest vs largest edges encountered and the smallest distinguished angle between left or right extreme entrance points.

5 A Hybrid Method

From the analysis above is clear that the competitive ratio of strategy *Walk-in-Circles* is directly dependent on the total "turn" angle β. As it was pointed out, β is smaller than π minus the initial angle $\angle p_0^- p_0 p_0^+$. 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 [8] is followed for initial angles $\angle p_0^- p_0 p_0^+$ smaller than $\pi/2$ and the strategy of Section 3 is used for angles larger than $\pi/2$.

Hybrid Strategy.

Cases 1-3 are as in Section 3.

Case 4 If $\angle p_0^- p_0 p_0^+ \leq \pi/2$ then the robot moves on the line perpendicular to $\overline{p_0^- p_0^+}$. As the robot advances it updates the vertices p_i^- and p_i^+ as the windows seen change. When either of $\angle p_i^- p\, p_0$ or $\angle p_i^+ p\, p_0 = \pi/2$, where p is the current position of the robot, it switches to strategy *Walk-in-Circles*, with p as starting point.

Case 5 If $\angle p_0^- p_0 p_0^+ \geq \pi/2$ then the robot uses strategy *Walk-in-Circles*.

From the discussion in Section 4, 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 $d(p_0, p_0^+) + \sum_i d(p_i^+, p_{i+1}^+)$. Let j be the index of the reflex vertex in which the robot switched strategies. Notice that $\angle p_j^+ p_0 p_j^-$ is now bigger equal to $\pi/2$.

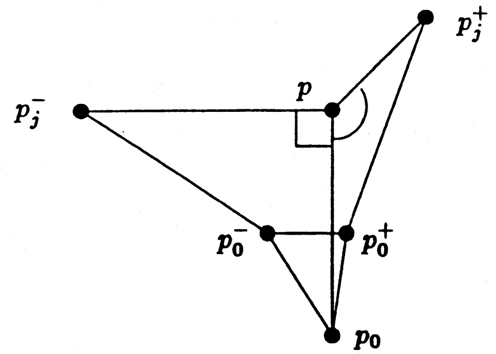

Figure 4: A hybrid strategy.

Lemma 5.1 *The distance traversed by the robot up to the point where it switches strategy is bounded by* $d(p_0, p) \leq \sqrt{d(p_0, p_j^{\pm})^2 - d(p, p_j^{\pm})^2}$ *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 4, the law of the cosines states $d(p_0, p_j^+)^2 = d(p_0, p)^2 + d(p, p_j^+)^2 - 2\, d(p_0, p)\, d(p, p_j^+) \cos(\angle p_0 p p_j^+)$; which implies $d(p_0, p_j^+)^2 \geq d(p_0, p)^2 + d(p, p_j^+)^2$ as $\angle p_0 p p_j^+ \geq \pi/2$, from which the lemma follows. □

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 $(\pi/2+1)\left(d(p, p_j^+) + \sum_{i=j}^{n-1} d(p_i^+, p_{i+1}^+)\right)$. Thus the competitive ratio is given by $\mathcal{R}/Opt$ where,

$$\begin{aligned} \mathcal{R} &= \sqrt{d(p_0, p_j^+)^2 - d(p, p_j^+)^2} + \\ &\quad (\pi/2+1)\left(d(p, p_j^+) + \sum_{i=j}^{n-1} d(p_i^+, p_{i+1}^+)\right) \\ Opt &= d(p_0, p_0^+) + \sum_{i=0}^{n-1} d(p_i^+, p_{i+1}^+). \end{aligned}$$

Let $Opt' = d(p_0, p_j^+) + \sum_{i=j}^{n-1} d(p_i^+, p_{i+1}^+)$. Since $Opt \geq Opt'$ then $\mathcal{R}/Opt \leq \mathcal{R}/Opt'$. Without loss of generality, we can assume that $d(p_0, p_j^+) = 1$. If $\mathcal{R}/Opt' \leq (\pi/2+1+k)$ for some $k \geq 0$, then $\mathcal{R} \leq (\pi/2+1+k)\, Opt'$, which implies

$$\begin{aligned} &\sqrt{1 - d(p, p_j^+)^2} + (\pi/2+1)\, d(p, p_j^+) \\ &\leq (\pi/2+1+k) + k \sum_{i=j}^{n-1} d(p_i^+, p_{i+1}^+). \end{aligned}$$

Since $k \cdot \sum_{i=j}^{n-1} d(p_i^+, p_{i+1}^+)$ can be arbitrarily small, for the expression above to be satisfied we need $\pi/2 + 1 - \sqrt{1 - d(p, p_j^+)^2} - (\pi/2 + 1)\, d(p, p_j^+) \geq -k$. 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 $k \geq \frac{1}{2}\sqrt{\pi^2 + 4\pi + 8} - \pi/2 - 1$ follows. Since the competitive ratio $\mathcal{R}/Opt$ is bounded by $\pi/2 + 1 + k$, we have the following theorem.

Theorem 5.2 *A robot using the* Hybrid Strategy *has a* $\frac{1}{2}\sqrt{\pi^2 + 4\pi + 8}$ *competitive ratio.*

The value $\frac{1}{2}\sqrt{\pi^2 + 4\pi + 8}$ is approximately 2.758...

6 Conclusions and Open Problems

We introduced and analyzed the first family of strategies for the street navigation problem. Because of this approach, the resulting algorithm is more robust under navigational error. It remains to be shown if it is possible for a robot to traverse a scene with a predetermined maximal navigational error per unit traversed at a predetermined competitive ratio. We also introduced a hybrid strategy which has a better competitive ratio than either of the original two strategies that define it. As the hybrid strategy shows, the $\pi + 1$ analysis not tight for small initial angles. In principle, it may be possible to improve on the $\pi + 1$ ratio by analyzing the *Walk-in-Circles* strategy differently for each case. AS well, the best lower bound known for traversing streets is $\sqrt{2}$. The gap between this lower bound and the best upper bound is still significant, and remains to be improved

7 Acknowledgement

We would like to thank Prabahakar Ragde for his valuable comments on an earlier version of this paper.

References

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

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

[3] 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.

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

[5] 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. A preliminary version appeared in *Proc. 32nd IEEE Symp. on Foundations of Computer Science*, (1991), pp. 298-303.

[6] Ch. Icking. Ph. D. Thesis, Fernuniversität Hagen, 1994.

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

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

[9] A. López-Ortiz and S. Schuierer. "Going Home Through and Unknown Street", *Technical Report 445*, Department of Computer Science, The University of Western Ontario, 1994.

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

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

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