Competitive Online Searching for a Ray in the Plane

Andrea Eubeler* Rudolf Fleischer† Tom Kamphans* Rolf Klein* Elmar Langetepe*
Gerhard Trippen‡

Abstract

We consider the problem of a searcher that looks for a lost flashlight in a dusty environment. The search agent finds the flashlight as soon as it crosses the ray emanating from the flashlight, and in order to pick it up, the searcher has to move to the origin of the light beam.

First, we give a search strategy for a special case of the ray search—the window shopper problem—, where the ray we are looking for is perpendicular to a known ray. Our strategy achieves a competitive factor of ≈1.059, which is optimal. Then, we consider the search for a ray with an arbitrary position in the plane. We present an online strategy that achieves a factor of ≈22.513, and give a lower bound of ≈16.079.

Keywords: Online motion planning, competitive ratio, searching, ray search

1 Introduction

Searching in an unknown environment is a basic task in robot motion planning and well-studied in many settings. For example, Gal and independently Baeza-Yates et. al. [7, 2] considered the task of finding a point on an infinite line using a searcher, that starts in the origin and neither knows the distance nor the direction towards the goal. They introduced the so called *doubling strategy* that is, the agent moves alternately to the left and to the right, doubling it's exploration depth in every iteration step. Searching on the line was generalized to searching on m concurrent rays starting from the searcher's origin, see [7, 2]. Many other variants were discussed since then, for example m-ray searching with restricted distance (Hipke et. al. [9], Langetepe [13], Schuierer [14]), m-ray searching with additional turn costs (Demaine et. al. [4]), parallel m-ray searching (Hammar et. al. [8]) or randomized searching (Kao et. al. [11]).

The quality of a strategy that deals with incomplete information —an *online strategy*—is usually measured by the cost of the online solution compared to the optimal solution. More precisely, let $|S|$ denote the cost of an online strategy, S, and $|S_{\mathrm{Opt}}|$ the cost of the optimal solution, then we call S *competitive with factor* C, iff there exists a constant A such that $|S| \leq C \cdot |S_{\mathrm{Opt}}| + A$ holds for every input to S. In our case, the costs incurred by a search strategy is given by the length of the path covered by the searcher, and the optimal solution is the length of the shortest path from the searcher's origin to the goal. The competitive framework was introduced by Sleator and Tarjan [17] and used for many settings, see e. g. the survey by Fiat and Woeginger [5]. For a general overview of online motion planning problems and its analysis see the surveys [3, 15, 16, 10]. Another measure is the *search ratio*, see Koutsoupias et. al. [12] and Fleischer et. al. [6]

In this paper, we consider the search for the origin of a ray in the plane. The searcher has no vision, but recognizes the ray and it's origin as soon as it enters it. First, we consider a simplified version of this problem: the origin of the ray, r, we are looking for is located on another ray, r', perpendicular to r. The searcher's start point and r are located on the same side of r'. Moreover, r' is known. We call this problem the *window shopper problem*, since we can imagine r' as a line of shopping windows. A buyer walks along these windows, looking e. g. for a present, and walks towards the window as soon as an appropriate item is spoted. We give a search strategy for this problem, that achieves an optimal competitive factor of 1.059 . . . Then we consider the general case, and give a search strategy that achieves a factor of 22.513 . . . and give a lower bound of 16.079 . . .

2 The window shopper problem

First, we consider the problem of finding a gift along a line of shopping windows. W. l. o. g. we assume, that the line of sight, i. e. the ray r we are looking for, is parallel to the X-axis, starting in $(1, y_r)$ for $y_r \geq 0$, and emanating to the left hand side of r'. The searcher starts in the origin $(0, 0)$, see Figure 1. The

*University of Bonn, Institute of Computer Science I, D-53117 Bonn, Germany.
†Fudan University, 220 Handan Road, Shanghai 200433
‡Hong Kong University of Science and Technology Clear Water Bay, Kowloon Hong Kong.

The work described in this paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKUST6010/01E) and by a grant from the Germany/Hong Kong Joint Research Scheme sponsored by the Research Grants Council of Hong Kong and the German Academic Exchange Service (Project No. G-HK024/02).

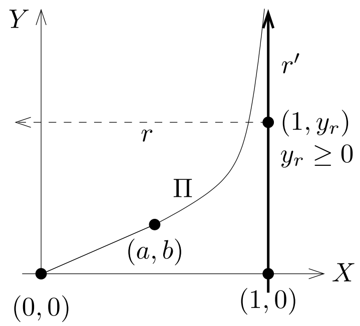

Figure 1: A strategy for the window shopper problem.

goal is discovered as soon as we reach its height, i. e. its Y-coordinate y_r. In order to reach the goal we finally have to move towards it (if we haven't reached it by coincidence).

Theorem 1 *There exists a strategy with an optimal competitive factor of* $1.059\ldots$ *for searching the origin of a ray,* r*, that emanates from a known ray* r' *perpendicular to* r*.*

Proof. Apparently a good search path moves simultaneously along and towards the wall, i. e. in positive X- and Y-direction. It is obvious that the competitive factor converges to 1 for goals with very small Y-coordinate and also for goals with a large Y-coordinate. Using this properties, we construct the following search strategy, Π: First, we follow a line segment. In the second part we follow a curve $f(x)$ that converges to the wall and maintains the value of the competitive factor given by the line segment at the beginning of the search path, see Figure 1.

By specifying the first part of the search path as the line segment from the origin to a point (a, b) we can describe the competitive factor as a monotone increasing function $\phi(t)$ with

$$\phi(t) = \frac{t\sqrt{a^2+b^2}+1-ta}{\sqrt{1+t^2b^2}},$$

and $\phi'(t) \geq 0 \ \ \forall t \in [0,1]$. Hence, $b \leq \sqrt{1-2a}$ follows.

From now on we assume $b = \sqrt{1-2a}$ and therefore $a = \frac{1-b^2}{2}$.

Now we are looking for a curve $f(x)$ that maintains the value

$$\delta = \frac{\sqrt{a^2+b^2}+1-a}{\sqrt{1+b^2}} = \sqrt{1+b^2}$$

for the competitive factor.

This means that the length of the whole search path, i. e. the line segment(s) and the curve, is δ times the Euclidean distance from the origin to the goal.

Solving the resulting differential equation of the form

$$\sqrt{a^2+b^2}+1-x+\int_a^x \sqrt{1+f'(t)^2}\,dt = \delta\cdot\sqrt{1+f(x)^2}$$

yields the values $b = 0.349\ldots$ and $\delta = 1.059\ldots$

To show the optimality of the given strategy, we observe an arbitrary curve $g(x)$ starting at the origin. Then we have to consider two cases.

Case 1: g reaches height b left to a.
If the goal is at height b then g has a competitive factor which is worse than our competitive factor. This also holds for goals above b, because the curve f has already reached its maximum value at (a, b).

Case 2: g reaches height b right to a.
Then the curve g intersects the curve f at some intersection point A; at the latest when the curve f reaches the wall. In this case the length of the path on g is longer than the path on f (since f is monotone increasing and convex), so the competitive factor of g is again worse than the factor of f in A. $\square$

3 Searching for a ray

Now, we suppose that we are positioned somewhere in the plane and we want to find an arbitrary ray. It seems to be a good strategy to search for the ray by moving on a logarithmic spiral. Since we have no sight the ray is only found when we cross it. Our aim is to reach the origin of the ray—simply by following the ray after we have found it.

We are interested in the worst case. This means that we want to construct a position of the ray that maximizes the competitive factor. Since we want to search the ray by moving on a spiral, we are looking for a spiral that minimizes the worst case in the plane.

The spiral is given by equation

$$f(\theta) = ae^{b\theta}, \qquad -\infty < \theta < \infty.$$

Due to the properties of the spiral, we only have to consider positions of the ray in one turn (2π).

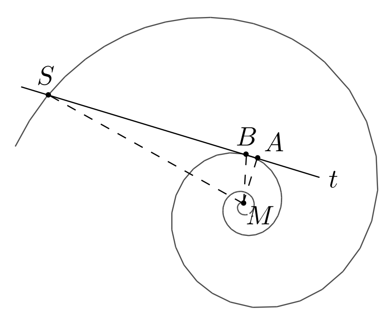

Figure 2: The tangent t to the spiral in point B.

One can easily see that the competitive factor reaches a local maximum if the ray is a tangent t to the spiral: Let B now be the tangent point of t, see Figure 2. The ray is missed in point B and detected in point S. We examine different positions of the ray's origin on the tangent. The starting point can not be on the left side of B, because otherwise the ray would no longer be a tangent. The competitive factor reaches a higher value for A than for B, see Figure 2, because A is farther away from S than B and $|\overline{MA}|$ is shorter than $|\overline{MB}|$.

First, we will find the best spiral for a point A, so that $|\overline{MA}|$ is perpendicular to tangent t.

Lemma 2 *The competitive factor δ_A for the point A, so that $|\overline{MA}|$ is perpendicular to the tangent t, only depends on the spiral parameter b and is given by*

$$\delta_A(b) = \frac{e^{b(2\pi+\beta(b))}}{\sin\alpha \cdot \cos\alpha} + \frac{e^{b(2\pi+\beta(b))} \cdot \sin(\beta(b))}{\sin^2\alpha} + b\,.$$

Its minimum value is $22.4908\ldots$ for $b = 0.1137\ldots$, where α is the tangent angle and β the angle $\angle BMS$.

Now we will show, that this b also gives us the optimal spiral for all tangents and all origins. This can be shown as follows. The adversary can move the point A to A' which gives a factor of $\delta_{A'}(b,\gamma) = \cos\gamma \cdot \delta_A(b) + \sin\gamma$ by simple geometry (see Figure 3).

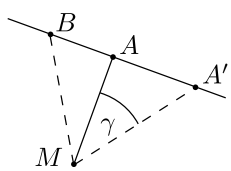

Figure 3: The triangle MAA'.

Therefore in general we have to minimize $\delta_A(b)$ whereas the adversary can choose the worst case γ. By simple analysis we have:

Theorem 3 *The best worst case competitive factor is $22.5130\ldots$ This value is reached by point A' which is specified by $\gamma = 0.4443\ldots$ and $b = 0.1137\ldots$*

Due to space limitations we omit the proofs of Lemma 2 and Theorem 3. See the full version of the paper.

4 A lower bound for searching a ray

We discuss a subproblem and consider a subset of rays, such that the extension of every ray goes through the starting point s.

If we consider the full bundle of lines passing through s, the given problem is equivalent to the problem of searching for a point in the plane as presented by Alpern and Gal [1]. We assume that the goal is detected, if it is swept by the radius vector of the trajectory. Alpern and Gal [1] showed that among all monotone and periodic strategies, a logarithmic spiral represented by polar coordinates $(\gamma, e^{b\gamma})$ gives the best search strategy in this setting. A strategy S represented by its radius vector $X(\gamma)$ is called periodic and monotone, if γ is always increasing and X also satisfies $X(\gamma + 2\pi) \geq X(\gamma)$.

The factor of the best monotone and periodic strategy is given by $\min_b e^{2\pi}b\sqrt{1+\frac{1}{b^2}} = 17.289\ldots$ and achieves its minimum for $b = 0.15540\ldots$, see Alpern and Gal [1]. Note, that the task does not include that the origin of the ray has to be visited.

Unfortunately, it was not shown that a periodic and monotone strategy is the best strategy for this problem. Alpern and Gal state, that it *seems to be a complicated task* to show that the spiral optimizes the competitive factor. Thus, the given factor cannot be adapted to be a lower bound to our problem.

Therefore we consider a discrete bundle of n rays that emanate from the start and which are seperated by an angle $\alpha = \frac{2\pi}{n}$, see Figure 4. We are searching for a goal on one of the n rays. Again the goal is detected if it is swept by the radius vector of the trajectory. Note, that if n goes to infinity we are back to the original problem. But we can neither assume that we have to visit the rays in a periodic order nor that the depth of the visit increases in every step.

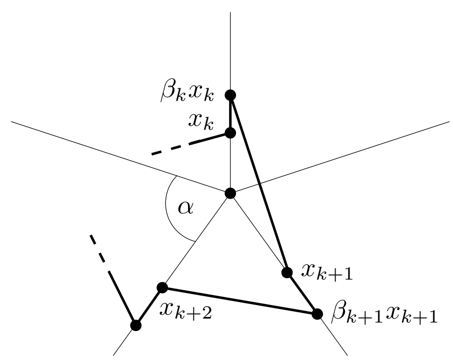

Figure 4: A bundle of n rays and the representation of a strategy.

Therefore we would like to make an approximation and represent a strategy as follows. At the k-th step, we hit a ray, say ray i, at distance x_k and leave the ray at distance $\beta_k x_k$ with $\beta_k \geq 1$. Therefore we move a distance $\beta_k x_k - x_k$ along the ray i and then we move to the next ray within a distance $\sqrt{(\beta_k x_k)^2 - 2\cos(\alpha)\beta_k x_k x_{k+1} + (x_{k+1})^2}$, see Figure 4. Let us assume that the ray i for x_k and $\beta_k x_k$ is visited the next time at index J_k. The worst-case occurs if we did not see the goal at the ray i up to distance x_k and find the goal at step x_{J_k} on i arbitrarily close behind $\beta_k x_k$. The competitive factor is bigger than

$$\frac{1}{\beta_k x_k}\Bigg(x_{J_k} - \beta_k x_k + \sum_{i=1}^{J_k-1} \beta_i x_i - x_i + \sqrt{(\beta_i x_i)^2 - 2\cos(\alpha)\beta_i x_i x_{i+1} + (x_{i+1})^2}\Bigg),$$

where $x_{J_k} - \beta_k x_k$ denotes the movement to goal.

By simple trigonometry the shortest distance from $\beta_k x_k$ to a neighboring ray is given by $\beta_k x_k \sin\left(\frac{2\pi}{n}\right)$. Fortunately, this distance is smaller than the distance $\sqrt{(\beta_k x_k)^2 - 2\cos(\alpha)\beta_k x_k x_{k+1} + (x_{k+1})^2}$ to any other ray. Therefore a lower bound on the above worst-case factor is given by

$$-1 + \frac{\sum_{i=1}^{J_k-1} \beta_i x_i \sin\left(\frac{2\pi}{n}\right)}{\beta_k x_k}$$

Altogether, we have to find a lower bound for $\frac{\sum_{i=1}^{J_k-1} \beta_i x_i}{\beta_k x_k}$ where J_k always denotes the next visit of the ray of x_k. Fortunately, this problem also represents the competitive analysis for the m ray problem where we can only move along the rays. It was shown by [7] and [2] that for this problem there is an optimal strategy that visits the rays with increasing depth and in a periodic order, that is $J_k = k + n$ and $i = k$. The best strategy is given by $f_i = (n/(n-1))^i$. Altogether, this results in a function

$$(n-1)\sin\left(\frac{2\pi}{n}\right)\left(\frac{n}{n-1}\right)^n$$

for n rays. We can make n arbitrarily big because our construction is valid for every n. Note, that we also have a lower bound for the problem of searching a point in the plane, the lower bound is close to the factor of the spiral.

Theorem 4 *For the ray search problem there is no strategy that achieves a better factor than*

$$-1 + \lim_{n\to\infty} (n-1)\sin\left(\frac{2\pi}{n}\right)\left(\frac{n}{n-1}\right)^n = -1 + 17.079\ldots$$

Additionally, every strategy for searching a point in the plane achieves a competitive factor bigger than $17.079\ldots$ *and the optimal spiral achieves a factor of* $17.289\ldots$

5 Conclusion

We considered the problem of searching a ray and its origin under the competitive framework.

If the ray starts on a known ray r' and is also perpendicular to r' we will find the origin within a path length of $1.059\ldots$ times the shortest path to the origin. This factor is optimal.

In general a logarithmic spiral solves the task within a competitive factor of $22.51\ldots$ whereas a lower bound of $16.079\ldots$ is given.

The lower bound construction can also be used if it is not necessary to visit the origin and if the corresponding line of every ray goes through the starting point. For this sub-problem a competitive strategy with factor $17.289\ldots$ was already known. We can proof that there is no strategy with a factor better than $17.079\ldots$ in this setting.

There are still some gaps between the lower and upper bounds of the factors which have to be closed.

References

[1] S. Alpern and S. Gal. *The Theory of Search Games and Rendezvous.* Kluwer Academic Publications, 2003.

[2] R. Baeza-Yates, J. Culberson, and G. Rawlins. Searching in the plane. *Inform. Comput.*, 106:234–252, 1993.

[3] P. Berman. On-line searching and navigation. In A. Fiat and G. Woeginger, editors, *Competitive Analysis of Algorithms.* Springer-Verlag, 1998.

[4] E. D. Demaine, S. P. Fekete, and S. Gal. Online searching with turn cost. Submitted to Theor. Comput. Sci.

[5] A. Fiat and G. Woeginger, editors. *On-line Algorithms: The State of the Art*, volume 1442 of *Lecture Notes Comput. Sci.* Springer-Verlag, 1998.

[6] R. Fleischer, T. Kamphans, R. Klein, E. Langetepe, and G. Trippen. Competitive online approximation of the optimal search ratio. In *Proc. 12th Annu. European Sympos. Algorithms*, volume 3221 of *Lecture Notes Comput. Sci.*, pages 335–346. Springer-Verlag, 2004.

[7] S. Gal. *Search Games*, volume 149 of *Mathematics in Science and Engeneering.* Academic Press, New York, 1980.

[8] M. Hammar, B. J. Nilsson, and S. Schuierer. Parallel searching on m rays. *Comput. Geom. Theory Appl.*, 18:125–139, 2001.

[9] C. Hipke, C. Icking, R. Klein, and E. Langetepe. How to find a point on a line within a fixed distance. *Discrete Appl. Math.*, 93:67–73, 1999.

[10] C. Icking, T. Kamphans, R. Klein, and E. Langetepe. On the competitive complexity of navigation tasks. In H. Bunke, H. I. Christensen, G. D. Hager, and R. Klein, editors, *Sensor Based Intelligent Robots*, volume 2238 of *Lecture Notes Comput. Sci.*, pages 245–258, Berlin, 2002. Springer.

[11] M.-Y. Kao, J. H. Reif, and S. R. Tate. Searching in an unknown environment: An optimal randomized algorithm for the cow-path problem. *Inform. Comput.*, 133(1):63–79, 1996.

[12] E. Koutsoupias, C. H. Papadimitriou, and M. Yannakakis. Searching a fixed graph. In *Proc. 23th Internat. Colloq. Automata Lang. Program.*, volume 1099 of *Lecture Notes Comput. Sci.*, pages 280–289. Springer, 1996.

[13] E. Langetepe. *Design and Analysis of Strategies for Autonomous Systems in Motion Planning.* PhD thesis, Department of Computer Science, FernUniversität Hagen, 2000.

[14] A. López-Ortiz and S. Schuierer. The ultimate strategy to search on m rays? *Theor. Comput. Sci.*, 261(2):267–295, 2001.

[15] J. S. B. Mitchell. Geometric shortest paths and network optimization. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 633–701. Elsevier Science Publishers B.V. North-Holland, Amsterdam, 2000.

[16] N. S. V. Rao, S. Kareti, W. Shi, and S. S. Iyengar. Robot navigation in unknown terrains: introductory survey of non-heuristic algorithms. Technical Report ORNL/TM-12410, Oak Ridge National Laboratory, 1993.

[17] D. D. Sleator and R. E. Tarjan. Amortized efficiency of list update and paging rules. *Commun. ACM*, 28:202–208, 1985.