

# Parallel Searching on $m$ Rays<sup>★</sup>

Mikael Hammar

*Department of Computer Science, Lund University, Box 118, S-221 00 Lund,  
Sweden*

Bengt J. Nilsson

*Department of Technology and Society, Malmö University College,  
Citadellsvägen 7, 205 06 Malmö, Sweden*

Sven Schuierer

*Institut für Informatik, Georges-Köhler-Allee, Geb. 051, D-79110 Freiburg,  
Germany*

---

## Abstract

We investigate parallel searching on  $m$  concurrent rays. We assume that a target  $t$  is located somewhere on one of the rays; we are given a group of  $m$  point robots each of which has to reach  $t$ . Furthermore, we assume that the robots have no way of communicating over distance. Given a strategy  $S$  we are interested in the competitive ratio defined as the ratio of the time needed by the robots to reach  $t$  using  $S$  and the time needed to reach  $t$  if the location of  $t$  is known in advance.

If a lower bound on the distance to the target is known, then there is a simple strategy which achieves a competitive ratio of 9—independent of  $m$ . We show that 9 is a lower bound on the competitive ratio for two large classes of strategies if  $m \geq 2$ .

If the minimum distance to the target is not known in advance, we show a lower bound on the competitive ratio of  $1 + 2(k+1)^{k+1}/k^k$  where  $k = \lceil \log m \rceil$  where  $\log$  is used to denote the base 2 logarithm. We also give a strategy that obtains this ratio.

*Key words:* Computational Geometry, On-line Search Strategies

---

---

<sup>★</sup> This research is supported by the DFG-Project “Diskrete Probleme”, No. Ot 64/8-3.

## 1 Introduction

Searching for a target is an important and well studied problem in robotics. In many realistic situations the robot does not possess complete knowledge about its environment, for instance, the robot may not have a map of its surroundings, or the location of the target may be unknown [3–6,8,9,11,12,15–17].

The search of the robot can be viewed as an *on-line* problem since the robot’s decisions about the search are based only on the part of its environment that it has seen so far. We use the framework of *competitive analysis* to measure the performance of an on-line search strategy  $S$  [18]. The *competitive ratio* of  $S$  is defined as the maximum of the ratio of the traveling time needed for a robot to find the target using strategy  $S$  to the optimal distance from its starting point to the target, over all possible locations in the environment of the target. Note that we have normalized the speed so that time equals distance for a full speed strategy.

A problem with paradigmatic status in this framework is searching on  $m$  concurrent rays. Here, a point robot or—as in our case—a group of point robots is imagined to stand at the origin of  $m$  concurrent rays. One of the rays contains the target  $t$  whose distance to the origin is unknown. A robot can detect  $t$  only if it stands on top of it. It can be shown that an optimal strategy for one robot is to visit the rays in cyclic order, increasing the step length each time by a factor of  $m/(m - 1)$ . In the beginning the robots starts with a step length of 1 which is assumed to be a lower bound on the distance to  $t$  [1,7]. The competitive ratio  $C_m$  achieved by this strategy is given by

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

The lower bound for searching on  $m$  rays has proved to be a very useful tool for proving lower bounds for searching in a number of classes of simple polygons, such as star-shaped polygons [14], generalized streets [6,13], HV-streets [5], and  $\theta$ -streets [5].

In this paper we are interested in obtaining upper and lower bounds for the competitive ratio of *parallel searching* on  $m$  concurrent rays. This problem has been addressed before in two contexts.

The first context is the on-line construction of hybrid algorithms the setting of which can be described as follows [10]: We are given a problem  $Q$  and  $m$  approaches to solving it. Each approach is implemented by an algorithm which is called a *basic* algorithm. We have a computer with  $k \leq m$  disjoint memory

areas which can be used to run one basic algorithm and to store the results of its computation. Only a single basic algorithm can be run by the computer at a given time. It is not known in advance which of the algorithms solves the problem  $Q$ —although we assume that there is at least one—or how much time it takes to compute a solution. In the worst case only one algorithm solves  $Q$  whereas the others do not even halt on  $Q$ . One way to solve  $Q$  is to construct a hybrid algorithm that uses the basic algorithms in the following way. A basic algorithm is run for some time, and then the computer switches to another algorithm and so on until  $Q$  is solved. If  $k < m$ , then there is not enough memory to save all the intermediate results. So sometimes the current intermediate results have to be discarded and to be recomputed later from scratch.

A different way to look at this problem is to assume that we are given  $k$  robots that have to search on  $m$  rays for a target  $t$  with  $k < m$ . Each ray corresponds to a basic algorithm, and a robot corresponds to a memory area. At any time we are allowed to move only one robot. Discarding intermediate results of the basic algorithm  $A$  corresponds to moving the robot on the ray corresponding to  $A$  back to the origin.

Kao et al. [10,19] present an algorithm for the above problem that achieves an optimal competitive ratio of

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

which is, of course, also the competitive ratio of searching with  $k$  robots on  $m$  rays if only one robot is allowed to move at a time.

In the second context a group of  $m$  point robots searches for the target. Again neither the ray containing the target nor the distance to the target are known. Now all the robots have to reach the target and the only way two robots can communicate is if they meet, that is, they have no communication device. We are going to use this model in our paper. Baeza-Yates and Schott investigate searching on the real line, that is, the case  $m = 2$  [2]. They present two strategies both of which achieve a competitive ratio of 9. They also consider searching for a target line in the plane with multiple robots and present symmetric and asymmetric strategies. However, the question of optimality, that is, corresponding lower bounds, is not considered.

In this paper we investigate search strategies for parallel searching on  $m$  concurrent rays. If a lower bound on the distance to the target is known, then there is a simple strategy that achieves a competitive ratio of 9—independent of  $m$ . We show that even in the case  $m = 2$  there is a matching lower bound of 9 on the competitive ratio of two large classes of strategies, *monotone* and

*symmetric* strategies. Moreover, we show that a lower bound of  $C$  for  $m = 2$  implies a lower bound of  $C$  for  $m > 2$ —as is to be expected. This implies, in particular, that there is no monotone or symmetric strategy for arbitrary  $m$  that has a competitive ratio better than 9. For monotone strategies we can even strengthen this result and show that

$$C_k = 1 + 2(k + 1)^{k+1}/k^k,$$

where  $k = \lceil \log m \rceil$  is lower bound on the competitive ratio.

We also consider the case that the minimum distance to the target is not known in advance which turns out to be essentially equivalent to restricting ourselves to monotone strategies. We again show a lower bound on the competitive ratio of  $C_k$  where  $k = \lceil \log m \rceil$ . We also present a (monotone) strategy that achieves this competitive ratio.

The paper is organized as follows. In the next section we present some definitions and preliminary results. In particular, we present three strategies to search on the line ( $m = 2$ ), each with a competitive ratio of 9. In Section 3 we show a matching lower bound of 9 for two large classes of strategies. In Section 4 we extend our results to the case  $m > 2$ . Finally, in Section 5 we present an optimal algorithm to search on  $m$  rays if there is no minimum distance to the target.

## 2 Preliminaries

In the following we consider the problem of a group of  $m$  robots searching for a target of unknown location on  $m$  rays in parallel. The robots have the same maximal speed which we assume without loss of generality to be 1 distance unit per time unit. If the robots have unbounded speed, then the time to find the target (both off-line and on-line) can be made arbitrarily small. The speed of a robot may be positive (if it moves away from the origin) or negative (if it moves towards the origin).

Let  $S$  be a strategy for parallel searching on  $m$  rays and  $T_S(D)$  the maximum time the group of robots needs to find and reach a target placed at a distance of  $D$  if it uses strategy  $S$ . Since the maximum speed of a robot is one, the time needed to reach the target if the position of the target is known is  $D$  time units. The competitive ratio is now defined as the maximum of  $T_S(D)/D$ , over all  $D \geq 0$ . In some applications a lower bound  $D_{min}$  on the distance to the target may be known. If such a lower bound exists, then we assume without loss of generality that  $D_{min} = 1$ . It will turn out that the existence of  $D_{min}$  leads to a drastically lower competitive ratio if  $m > 2$ .

We define different classes of possible strategies to search on  $m$  rays in parallel. We say a strategy is *monotone* if, at all times, all the robots (that do not know the location of the target) have non-negative speed. We say a strategy is *full speed* if all the robots travel at a speed of 1 or  $-1$  at all times. We say a strategy is *symmetric* if, at all times, all the robots (that do not know the location of the target) have the same speed.

We illustrate the different types of strategies for  $m = 2$ . The optimal monotone strategy is for each robot to travel at a speed of  $1/3$  on each ray. After one robot has found the target, it runs back to fetch the other. This leads to a competitive ratio of 9. This strategy is described in [2]. In the next section we show a lower bound of 9 on the competitive ratio of monotone strategies. The optimal (full-speed) symmetric strategy is for each robot to double the distance that has been explored before and then to return to the origin. This strategy can only be applied if a lower bound on the distance to the target is known. It achieves a competitive ratio of 9. Again this strategy is described in [2] and we show a lower bound of 9 on the competitive ratio of symmetric strategies in the next section. Finally, an asymmetric strategy is for both robots to walk together and to use the optimal strategy for one robot to search on two rays. This again yields a competitive ratio of 9.

### 3 Searching on Two Rays

In this section we consider the problem of two robots searching for a target of unknown location on the real line in parallel; the robots are initially placed at the origin. We assume in the following that a lower bound on the distance from the origin to the target of  $D_{min} = 1$  is known.

For monotone strategies we have the following lower bound.

**Theorem 1** *There is no monotone strategy that achieves a better competitive ratio than 9 to search on two rays in parallel.*

**PROOF.** The proof uses an adversary to place the target point in order to maximize the competitive ratio.

Let us enumerate the robots and define  $v_i(T)$  as the *average* speed of robot  $i \in \{1, 2\}$  at time  $T$ , i.e., the distance of the robot to the origin at this time divided by the time. It is clear that a monotone search strategy is completely specified by the two average speed functions.

First of all we realize that the two robots will not both go in the positive or negative direction since then the adversary can place the target on the ray not

visited by the robots, yielding an infinite competitive ratio.

Hence, the two robots go in different directions. As time passes the robots move continuously and monotonically with some speed along the line until one of them finds the target. This robot now travels at full speed to the other robot and communicates to it the location of the target and they both return to this target point. Consider the value  $v_{min}$  specified by

$$v_{min} = \inf_{t \geq T_{min}} \min\{v_1(t), v_2(t)\},$$

where  $T_{min}$  is the first point in time such that both robots are at least a distance  $D_{min}$  from the origin. From the definition of  $v_{min}$ , we know that  $v_1(T) \geq v_{min}$  and  $v_2(T) \geq v_{min}$ , for all  $T > T_{min}$ .

For any  $\varepsilon > 0$ , there is a time  $T_\varepsilon \geq T_{min}$  such that either  $v_1(T_\varepsilon) \leq v_{min} + \varepsilon$  or  $v_2(T_\varepsilon) \leq v_{min} + \varepsilon$ . Assume without loss of generality that for some specific  $\varepsilon$  it holds for  $v_1(T_\varepsilon)$ . By the definition of monotonicity  $T_\varepsilon \geq T_{min}$  and hence  $v_1(T_\varepsilon)T_\varepsilon \geq v_1(T_{min})T_{min} = D_{min}$ , so the adversary places the target at the point  $D_\varepsilon = v_1(T_\varepsilon)T_\varepsilon$ . A lower bound on the competitive ratio  $C$  of the strategy can now be expressed as

$$C \geq \frac{T_\varepsilon + 2T_R}{D_\varepsilon},$$

where  $T_R$  denotes the time for the robot that found the target to reach the second robot at full speed. Since they both have to return to the target this adds an extra term  $T_R$ . Because of the monotonicity of the strategy we can express the time  $T_R$  by

$$T_R = D_\varepsilon + v_2(T_\varepsilon + T_R)(T_\varepsilon + T_R).$$

This is because the robot that finds the target goes towards the other robot at full speed traveling first the distance  $D_\varepsilon$  and then the distance  $v_2(T_\varepsilon + T_R)(T_\varepsilon + T_R)$  until they meet. We have that

$$T_R = \frac{D_\varepsilon + v_2(T_\varepsilon + T_R)T_\varepsilon}{1 - v_2(T_\varepsilon + T_R)}$$

where we can assume that  $v_2(T_\varepsilon + T_R) < 1$  since otherwise robot 1 never reaches robot 2. Hence,

$$C \geq \frac{T_\varepsilon + 2(v_1(T_\varepsilon)T_\varepsilon + v_2(T_\varepsilon + T_R)T_\varepsilon)/(1 - v_2(T_\varepsilon + T_R))}{v_1(T_\varepsilon)T_\varepsilon}$$

$$\begin{aligned}
&= \frac{1}{v_1(T_\varepsilon)} + \frac{2}{1 - v_2(T_\varepsilon + T_R)} + \frac{2}{v_1(T_\varepsilon)} \frac{v_2(T_\varepsilon + T_R)}{(1 - v_2(T_\varepsilon + T_R))} \\
&\geq \frac{1}{v_{min} + \varepsilon} + \frac{2}{1 - v_{min}} + \frac{2}{(v_{min} + \varepsilon)(1 - v_{min})} v_{min} \\
&\geq 9 - \frac{18\varepsilon}{1 + 3\varepsilon},
\end{aligned}$$

since the above expression is minimized for  $v_{min} = \frac{1}{3}$  and it tends to 9 as  $\varepsilon$  tends to 0.  $\square$

Next we look at symmetric strategies. The following lemma shows that only full speed strategies need to be considered.

**Lemma 2** *For all  $\varepsilon > 0$  and all strategies  $S$ , there is a full speed strategy  $S'$  such that the competitive ratio of  $S'$  is at most  $(1 + \varepsilon)$  times the competitive ratio of  $S$ .*

**PROOF.** Divide the time into small intervals of length  $\delta$ . If strategy  $S$  has average speed  $v$  in an interval  $I$ , then let  $S'$  be the full-speed strategy that first travels back for  $(1 - v)\delta/2$  time units and then forward for  $(1 + v)\delta/2$  time units in  $I$ . At the end of  $I$  the robot is at the same position as if it had used  $S$ . By letting  $\delta$  go to 0, the claim follows.  $\square$

The previous lemma tells us that we can simulate any strategy with a full speed strategy and therefore we only have to consider full speed strategies if we are given a lower bound on the distance to the target.

In the following we show a lower bound for symmetric full speed strategies. We start with the case  $m = 2$  and consider the general case later.

**Lemma 3** *Let  $S$  be a symmetric strategy. Then, there is a sequence of positive numbers  $(y_0, y_1, y_2, \dots)$  such that the competitive ratio  $C_S$  of  $S$  satisfies*

$$C_S \geq \sup_{k \geq 1} 1 + 2 \frac{\sum_{i=0}^{k+1} y_i}{y_k}.$$

**PROOF.** Since the strategy is symmetric, the two robots will use the same local strategy to search its own ray. Furthermore, by Lemma 2 we can assume that it is a full speed strategy. We can model a full speed strategy for a robot by saying that it first moves a distance  $x_0$  forward along the ray at full speed, then it moves a distance  $y_0$  backwards at full speed, then a distance  $x_1$  forward, a distance  $y_1$  backward, and so on. When one of the robots finds the target,

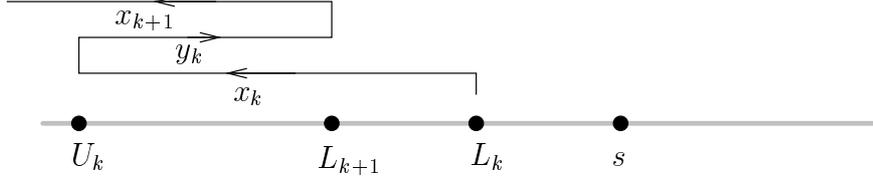


Fig. 1. The definition of  $U_k$  and  $L_k$ .

it runs back at full speed until it meets the other robot, and they both run to the target at full speed.

The proof uses an adversary to place the target point in order to maximize the competitive ratio.

We say that a robot is in *step*  $k$  when it moves forward and backward the  $k + 1^{\text{st}}$  time. Let  $L_k$  denote the distance to the origin of the turning point where the robot begins step  $k$  and let  $U_k$  denote the distance to the origin of the turning point where the robot starts to move backwards during step  $k$  (see Figure 1).

We have that

$$\begin{aligned}
 L_0 &= 0, \\
 L_k &= L_{k-1} + x_{k-1} - y_{k-1} = \sum_{i=0}^{k-1} x_i - y_i, \\
 U_k &= L_k + x_k = x_k + \sum_{i=0}^{k-1} x_i - y_i = y_k + \sum_{i=0}^k x_i - y_i.
 \end{aligned}$$

The total time that the robot has traveled when it completes step  $k$  is

$$T_k = \sum_{i=0}^k x_i + y_i.$$

We can assume that  $U_{k-1} < U_k$ , since otherwise, the strategy will not explore any new part of the ray during step  $k$ , and we can exchange the strategy for another equivalent one, where the assumption holds. In particular, we can assume that  $y_k < x_{k+1}$ , for all  $k \geq 0$ .

Furthermore we can assume that  $y_k \leq U_k$  for all  $k \geq 0$ , that is, a robot always stays on the same ray, since if this does not hold, we can exchange the strategy for an equivalent one where, if the two robots meet at the origin, they exchange places and continue on their own ray instead of the other robot's ray.

Assume that the target is placed at distance  $D$  with  $1 = D_{\min} \leq D$  and that

the two robots meet in step  $k$ , that is, during the time that one robot travels from  $U_k$  to  $L_{k+1}$ , say at point  $q \in [L_{k+1}, U_k]$ . The competitive ratio for this placement is given by

$$\begin{aligned} C_k(D) &= \frac{T_{k-1} + x_k + (U_k - q) + (q + D)}{D} = 1 + \frac{T_{k-1} + x_k + U_k}{D} \\ &= 1 + 2 \frac{\sum_{i=0}^{k-1} y_i + U_k}{D}, \end{aligned} \quad (1)$$

since  $T_{k-1} + x_k + (U_k - q)$  is the time needed to reach point  $q$  and  $q + D$  is the time needed for both robots to return to the target after they have met. It is interesting to note that, for a given  $D$ , the above analysis is completely independent of the step in which the target is *found* and only depends on the step  $k$  in which the robots *meet*.

To each step  $k$  we will associate a placement of the target point, such that the robot that finds the target finds it during a step no earlier than step  $k + 1$ . The corresponding competitive ratio of the placement is denoted  $C_k$ .

Consider placing the target at distance  $D$  after  $y_k$ . By Lemma 4 we can assume that  $y_k \geq U_{k-1}$ , so the robot that finds the target, say robot 1, discovers it earliest during step  $k$ . We claim that the robots do not meet before step  $k + 1$ . The total distance between the robots is  $2D$  at the time when the target is found. The distance is only reduced during the times when robot 2 travels back towards the origin. During such a phase the distance is reduced by two distance units per time unit since the robots travel towards each other. Now, if the target is found in step  $k$ , then the robots meet in the first step  $k^*$  such that  $\sum_{i=k}^{k^*} y_i \geq D$ . Since  $D > y_k$ , this implies that  $k^* \geq k + 1$ . If the target is not found in step  $k$ , then the robots meet earliest in step  $k + 1$  anyway. Hence, using Equation 1 the competitive ratio  $C_k$  of placing the target after  $y_k$  satisfies

$$\begin{aligned} C_k &\geq \sup_{D \in ]y_k, \infty[} \left\{ 1 + 2 \frac{\sum_{i=0}^k y_i + U_{k+1}}{D} \right\} = 1 + 2 \frac{\sum_{i=0}^k y_i + U_{k+1}}{y_k} \\ &\geq 1 + 2 \frac{\sum_{i=0}^{k+1} y_i}{y_k}, \end{aligned}$$

since  $y_{k+1} \leq U_{k+1}$ .

As a result the competitive ratio for any symmetric strategy is bounded below by

$$C = \sup_{k \geq 0} C_k \geq \sup_{k \geq 0} 1 + 2 \frac{\sum_{i=0}^{k+1} y_i}{y_k}$$

as claimed.  $\square$

**Lemma 4** *Using the terminology of the proof of Lemma 3 there is an optimal symmetric strategy with  $y_k \geq U_{k-1}$ , for all  $k \geq 1$ .*

**PROOF.** Let  $S$  be any strategy such that  $y_k < U_{k-1}$  for some  $k \geq 1$ . We show that we can exchange  $S$  for another strategy  $S^*$  such that its specifying parameters  $y_i^*$  and  $U_i^*$  have the property  $y_i^* = y_i$ , for  $0 \leq i < k$ ,  $U_i^* = U_i$ , for  $0 \leq i \leq k$  and  $y_k^* \geq U_{k-1}$ .

Let  $k^*$  be the smallest index with  $\sum_{i=k}^{k^*} y_i \geq U_{k-1}$ . The index  $k^*$  must exist and be finite since otherwise we can place the target at distance  $D > U_{k-1}$  and get an infinite competitive ratio for the strategy.

We now define two new specifying sequences  $y_i^*$  and  $U_i^*$  of  $S^*$  which will replace the sequences  $y_i$  and  $U_i$ .

For  $0 \leq i < k$ , nothing changes, that is, we set  $y_i^* = y_i$  and  $U_i^* = U_i$ . For  $k$  we define  $U_k^* = U_k$  and  $y_k^* = \sum_{l=k}^{k^*} y_l$ . Finally, we drop all indices from  $k+1$  to  $k^*$  and define  $U_{k+j}^* = U_{k^*+j}$  and  $y_{k+j}^* = y_{k^*+j}$ , for  $j \geq 1$ .

Let  $C_n(D)$  and  $C_n^*(D)$  be the corresponding competitive ratios for strategies  $S$  and  $S^*$  respectively, when the target is placed at distance  $D$  from the origin, and the robots meet at step  $n$ . Recall that

$$C_n(D) = 1 + 2 \frac{\sum_{i=0}^{n-1} y_i + U_n}{D},$$

where  $n$  is the step in which the two robots meet. For  $C_n^*(D)$  we have a similar formula.

We will compare  $C_n(D)$  and  $C_n^*(D)$  for all possible meeting steps from one onwards.

If the robots meet in step  $i$  where  $0 \leq i \leq k$ , then obviously  $C_i(D) = C_i^*(D)$ .

If in strategy  $S$  the robots meet in step  $k^* + j$ , for  $j \geq 1$ , then they will meet in step  $k + j$  in strategy  $S^*$  and it holds that  $C_{k^*+j}(D) = C_{k+j}^*(D)$ . Hence it only remains to consider the steps  $k+1, \dots, k^*$  of strategy  $S$ .

If the robots meet in step  $l$  of strategy  $S$  where  $k+1 \leq l \leq k^*$ , then they meet in step  $k$  in strategy  $S^*$ . To see this, we prove that if robot 2 (which does not know about the target) reaches the point  $U_k^* = U_k$ , then the distance between the two robots is at most  $2y_k^*$ —and they meet when robot 2 travels back in step  $k$ .

We first note that when robot 2 reaches the point  $U_l$  of  $S$ , then robot 1 is at most a distance  $2y_l$  from it (since they meet in step  $l$ ). Similarly, if robot 2 reaches point  $U_{l-1}$ , then the distance is at most  $2(y_l + y_{l-1})$ . By induction we have that when robot 2 is at point  $U_k$ , then the distance between them is at most  $2 \sum_{i=k}^l y_i \leq 2 \sum_{i=k}^{k^*} y_i = 2y_k^*$  as claimed.

The competitive ratio of strategy  $S$  if the robots meet during step  $l > k$  is

$$C_l(D) = 1 + 2 \frac{\sum_{i=0}^{l-1} y_i + U_l}{D} \geq 1 + 2 \frac{\sum_{i=0}^{k-1} y_i^* + U_k^*}{D} = C_k^*(D)$$

since  $y_i^* = y_i$ , for  $0 \leq i \leq k-1$ , and  $U_k^* = U_k < U_l$ . Thus, the competitive ratio of strategy  $S^*$  is at most as large as the competitive ratio of  $S$ .

Repeating the process for all  $k$  in  $S$  such that  $y_k < U_{k-1}$  we get a strategy  $S'$  with specifying parameters  $y'_i$  and  $U'_i$  such that  $y'_i \geq U'_{i-1}$  for all  $i \geq 1$ .

If  $S$  was taken as an optimal strategy then obviously  $S'$  must also be optimal, thus concluding the proof.  $\square$

Since for any sequence of positive numbers  $(y_0, y_1, y_2, \dots)$  the value of

$$\sup_{k \geq 0} 1 + 2 \frac{\sum_{i=0}^{k+1} y_i}{y_k}$$

is bounded from below by 9 [1,7], we have shown the following theorem.

**Theorem 5** *There is no symmetric strategy that achieves a better competitive ratio than 9 to search on two rays in parallel.*

#### 4 Searching on $m$ Rays

We now turn to searching on  $m$  rays in parallel. We assume that a group of  $m$  robots is located at the origin of the rays in the beginning and that again a lower bound on the distance to the target is known. We first show that any lower bound for searching on two rays with two robots implies a lower bound for searching on  $m$  rays with  $m$  robots.

**Theorem 6** *If  $C$  is a lower bound for searching on two rays in parallel, then  $C$  is also a lower bound for searching on  $m$  rays in parallel.*

**PROOF.** It is obvious that if the  $m$  robots have more information in the beginning about the location of the target, then the competitive ratio for a strategy that exploits this information does not increase. Assume that the robots know that the target is on one of the first two rays. They can all explore these rays in common using strategy  $S$ . We now define a new strategy  $S'$  for searching in parallel with two robots on two rays which depends on  $S$ . At all times the two robots of  $S'$  follow the robots in  $S$  that are furthest from the origin on each of the two rays. Clearly, it is possible for two robots to maintain the position of the furthest of the  $m$  robots on both rays since this position changes continuously.

Now assume that one robot (of the  $m$  robots), say robot  $i$ , finds the target. Clearly, robot  $i$  is a robot which is the furthest from the origin on its ray. Hence, one of the two robots of  $S'$  finds the target at the same time. Now all of the other robots have to be notified. It is easy to show by simple induction that we can assume that robot  $i$  notifies all the other robots itself since

- (1) no other robot can travel faster to the opposite ray than robot  $i$  and
- (2) a robot can only start chasing other robots once it meets a robot that at some point after the discovery of the target met robot  $i$ .

Obviously, the last robot that is going to be notified is, at the moment of notification, furthest from the origin on the opposite ray. But this robot is at the same position as the second robot of  $S'$ . Hence, both strategies need the same time for all robots to reach the target and the competitive ratio is the same—this implies the claim.  $\square$

We have the following corollary.

**Corollary 7** *There is no monotone or symmetric strategy that achieves a better competitive ratio than 9 to search on  $m$  rays in parallel.*

Now, there is a symmetric strategy that achieves a competitive ratio of 9 to search on  $m$  rays in parallel. The strategy is known as the *doubling strategy* and goes as follows [1,2,7]. Each robot starts by going one unit at full speed on its ray and then goes back to the origin. Then they each go two units, four units, and so on, on their corresponding ray, always doubling the distance traveled and repeatedly going back to the origin. Once a robot finds the target, it goes back at full speed to the origin and waits there until the other robots reach it. It then communicates the location of the target to the other robots and they all move at full speed to that location. The competitive ratio of the doubling strategy, if the target is at distance  $D$  from the origin is

$$\begin{aligned}
C &\leq \sup_{k \geq 1} \sup_{D \in [2^{k-1}, 2^k]} \left\{ \frac{2 \sum_{i=0}^k 2^i + D}{D} \right\} = 1 + \sup_{k \geq 1} \sup_{D \in [2^{k-1}, 2^k]} \left\{ \frac{2 \sum_{i=0}^k 2^i}{D} \right\} \\
&= 1 + \sup_{k \geq 1} \frac{2 \sum_{i=0}^k 2^i}{2^{k-1}} = 1 + 2 \sup_{k \geq 1} \frac{2^{k+1} - 1}{2^{k-1}} = 9.
\end{aligned}$$

We have proved the following theorem.

**Theorem 8** *The doubling strategy achieves a competitive ratio of 9 to search on  $m$  rays in parallel given a lower bound on the distance to the target.*

Corollary 7 can be strengthened considerably for monotone strategies. This also illustrates nicely that there is a difference between monotone and symmetric strategies for  $m > 2$ .

Consider a monotone strategy. As time passes the robots move continuously and monotonically with some speed along the rays until one of them has found the target. This robot now travels at full speed to one of the other robots and communicates to him the location of the target, they both travel to other robots to communicate the location of the target, and so on. When all robots know where the target is they all go to this target point. We use this idea to show the following lower bound.

**Theorem 9** *There is no monotone strategy that achieves a better competitive ratio than*

$$C_k = 1 + 2 \frac{(k+1)^{k+1}}{k^k},$$

where  $k = \lceil \log m \rceil$ , to search on  $m$  rays in parallel given a lower bound on the distance to the target.

**PROOF.** We start similar to the proof of the case  $m = 2$ . The  $m$  robots will each go on a different ray since otherwise the competitive ratio is infinite.

Let  $v_i(T)$  be the average speed of robot  $1 \leq i \leq m$  at time  $T$  and

$$v_{min} = \inf_{t \geq T_{min}} \min\{v_1(t), v_2(t), \dots, v_m(t)\},$$

where  $T_{min}$  is the first point in time such that all robots are at least at distance  $D_{min}$  from the origin.

For every  $\varepsilon > 0$ , there is a  $T_\varepsilon > T_{min}$ , such that  $v_i(T_\varepsilon) \leq v_{min} + \varepsilon$ , for one  $1 \leq i \leq m$ . The adversary places the target at a point  $D_\varepsilon = v_i(T_\varepsilon)T_\varepsilon$  on ray  $i$ .

Once robot  $i$  has found the target, we can designate the robots by two types. Robots of the first type are called the *hunters*, i.e., the robots that currently know the position of the target and are chasing after other robots in order to convey this information. The remaining robots are called the *prey*, and these continue on their respective ray using their monotone strategy. Initially at time  $T_\varepsilon$ , only robot  $i$  is a hunter, all the others are prey.

Denote by  $T_n$  the first time when at least  $n$  of the  $m$  robots are hunters.

We use induction to prove the following inequality

$$T_n \geq T_\varepsilon \left(1 + 2\frac{v_{\min}}{1 - v_{\min}}\right)^{\lceil \log n \rceil}.$$

Note that we can assume that  $v_{\min} < 1$  since otherwise the robot that finds the target will never reach any other robot.

The base case  $n = 1$  follows directly since we know that  $T_1 = T_\varepsilon$ .

For  $n > 1$ , we prove the claim as follows. The time  $T_n$  is of course non-decreasing in  $n$ . Consider now the point in time  $T_n$ , for some specific  $n$ . At some time prior to  $T_n$ , at least  $\lceil n/2 \rceil$  robots are hunters. Assume otherwise, i.e., that there are  $n' < \lceil n/2 \rceil$  hunters at any time prior to  $T_n$ . Then the largest number of robots that can become hunters at time  $T_n$  is  $2n' < n$ , since each hunter can produce at most one new hunter at time  $T_n$ . This is a contradiction since at time  $T_n$  we have at least  $n$  hunters. For the same reason, there is a hunter and a prey that meet at a time  $T$  no earlier than  $T_{\lceil n/2 \rceil}$  and then one of them, say robot  $l$ , chases some other prey, say robot  $j$ , and catches it at a time  $T'$  no later than  $T_n$ . Let  $T_R = T' - T$ . We have that

$$T_n \geq T_{\lceil n/2 \rceil} + T_R.$$

The distance of the robot on ray  $l$  to the origin is now  $v_l(T)T$  and it reaches the other robot on ray  $j$  at a distance of  $v_j(T + T_R)(T + T_R)$ . Hence,

$$\begin{aligned} T_R &\geq \min_{T_{\lceil n/2 \rceil} \leq T \leq T_n} \{v_l(T)T + v_j(T + T_R)(T + T_R)\} \\ &\geq \min_{T \geq T_{\lceil n/2 \rceil}} \{v_l(T)\} T_{\lceil n/2 \rceil} + \min_{T \geq T_{\lceil n/2 \rceil}} \{v_j(T + T_R)\} (T_{\lceil n/2 \rceil} + T_R). \end{aligned}$$

Therefore,

$$T_R \geq T_{\lceil n/2 \rceil} \frac{\min_{T \geq T_{\lceil n/2 \rceil}} \{v_l(T)\} + \min_{T \geq T_{\lceil n/2 \rceil}} \{v_j(T + T_R)\}}{1 - \min_{T \geq T_{\lceil n/2 \rceil}} \{v_j(T + T_R)\}}$$

$$\geq 2T_{\lceil n/2 \rceil} \frac{v_{min}}{1 - v_{min}}.$$

From above we have that

$$\begin{aligned} T_n &\geq T_{\lceil n/2 \rceil} + 2T_{\lceil n/2 \rceil} \frac{v_{min}}{1 - v_{min}} \geq T_{\lceil n/2 \rceil} \left(1 + 2\frac{v_{min}}{1 - v_{min}}\right) \\ &\geq T_\varepsilon \left(1 + 2\frac{v_{min}}{1 - v_{min}}\right)^{\lceil \log n \rceil}, \end{aligned}$$

by our induction hypothesis.

So, after  $T_m$  time, all robots are hunters. Assume that (one of) the last robot(s) to become a hunter is robot  $l$ . The competitive ratio of any monotone strategy can now be expressed by

$$\begin{aligned} C &\geq \frac{T_m + v_l(T_m)T_m + D}{D} = 1 + \frac{T_m + v_l(T_m)T_m}{v_l(T_\varepsilon)T_\varepsilon} \\ &\geq 1 + \left(1 + 2\frac{v_{min}}{1 - v_{min}}\right)^{\lceil \log m \rceil} \frac{1 + v_{min}}{v_{min} + \varepsilon} \\ (k = \lceil \log m \rceil) &= 1 + \left(\frac{1 + v_{min}}{1 - v_{min}}\right)^k \frac{1 + v_{min}}{v_{min} + \varepsilon} \\ &= 1 + \frac{(1 + v_{min})^{k+1}}{v_{min}(1 - v_{min})^k} - \Theta(\varepsilon) \\ &\geq 1 + \frac{\left(1 + \frac{1}{2k+1}\right)^{k+1}}{\frac{1}{2k+1}\left(1 - \frac{1}{2k+1}\right)^k} - \Theta(\varepsilon) \\ &= 1 + 2\frac{(k+1)^{k+1}}{k^k} - \Theta(\varepsilon), \end{aligned}$$

since the expression is minimized for  $v_{min} = 1/(2k+1)$  and it tends to  $C_k$  as  $\varepsilon$  tends to 0, which concludes the proof.  $\square$

In the next section we show that there is a monotone strategy that achieves this lower bound.

## 5 Without Lower Bound on the Minimum Distance

In this section we consider the problem of a group of  $m$  robots searching for a target of unknown location on  $m$  rays in parallel where no lower bound on the distance from the origin to the target is known. It should be noted that

if one allows an additive constant in the definition of the competitive ratio, then it is not necessary to know the minimum distance to the target.

However, using the stronger definition of competitive ratio this special case has been considered before for searching on the line—even with only one searcher which requires somewhat artificial assumptions on the strategies [7]. These assumptions are unnecessary in our case. Moreover, there are problems in which no lower bound on the distance to the target is known, for instance, when one wants to check whether a polygon is star-shaped or not [14].

We begin by presenting a strategy that achieves the competitive ratio

$$C_k = 1 + 2 \frac{(k+1)^{k+1}}{k^k},$$

where  $k = \lceil \log m \rceil$ . We then show that, in fact, no strategy can do better than this.

### 5.1 The Strategy

The optimal strategy is a monotone strategy where all the robots move, one on each ray, with a constant speed  $v$ . When one robot finds the target it searches for a robot at full speed to tell it where the target is located. Then they both go at full speed to search for two more robots and tell them the location of the target, and so on. After each step the number of robots that know the location of the target is doubled. Once all robots know the location, they all move to the target. Suppose the target is on some ray and at distance  $D$  from the origin. The strategy consists of  $k$  steps. Step  $i$  starts when  $2^i$  robots know the location to the target and ends when  $2^{i+1}$  robots know the location to the target; that is, in step  $i$  the  $2^i$  robots that currently know the position of the target chase  $2^i$  of those robots that do not. In the last step  $k = \lceil \log m \rceil$  only  $m - 2^{k-1}$  robots search for the remaining robots whereas the rest moves to the target.

Let  $T_i$  denote the time it takes to complete step  $i$ . It takes any of the robots  $D/v$  time to find the target, and when all robots know the location of the target, it takes them time  $T_F$  to go to the target. Hence, the competitive ratio of the strategy is

$$C = \frac{D/v + \sum_{i=0}^{k-1} T_i + T_F}{D},$$

where  $k = \lceil \log m \rceil$ .

The time that has passed in order to inform  $2^i$  robots is  $D/v + \sum_{j=0}^{i-1} T_j$ . Hence, at the end of step  $i - 1$  all the robots have a distance of  $D + v \sum_{j=0}^{i-1} T_j$  to the origin. The  $2^i$  informed robots now start chasing  $2^i$  uninformed robots. In order to reach them, they have to travel a distance of  $D + v \sum_{j=0}^{i-1} T_j$  to reach the origin, a distance of  $D + v \sum_{j=0}^{i-1} T_j$  to reach the location of the chased robot at the beginning of step  $i$ , and a distance of  $vT_i$  until the uninformed robots are finally reached. Hence,  $T_i$  is given by the equation

$$T_i = 2(D + v \sum_{j=0}^{i-1} T_j) + vT_i$$

and

$$T_i = \frac{2D + 2v \sum_{j=0}^{i-1} T_j}{1 - v} = \frac{2D + 2v \sum_{j=0}^{i-2} T_j}{1 - v} + \frac{2v}{1 - v} T_{i-1} \leq T_{i-1} \frac{1 + v}{1 - v}.$$

This is a recurrence relation with the solution

$$T_i \leq T_0 \left( \frac{1 + v}{1 - v} \right)^i = \frac{2D}{1 - v} \left( \frac{1 + v}{1 - v} \right)^i.$$

We obtain

$$\sum_{i=0}^{k-1} T_i \leq \frac{2D}{1 - v} \sum_{i=0}^{k-1} \left( \frac{1 + v}{1 - v} \right)^i = \frac{D}{v} \left( \left( \frac{1 + v}{1 - v} \right)^k - 1 \right).$$

By the above considerations the time  $T_F$  it takes for the robots to get to the target is the time of the last robot that is informed to go to the origin  $D + v \sum_{j=0}^{i-1} T_j$  plus the distance  $D$  to the target

$$T_F = D + v \sum_{i=0}^{k-1} T_i + D \leq D \left( \frac{1 + v}{1 - v} \right)^k + D.$$

So, the competitive ratio of the strategy is

$$\begin{aligned} C &\leq \frac{1}{v} + \frac{1}{v} \left( \frac{1 + v}{1 - v} \right)^k - \frac{1}{v} + \left( \frac{1 + v}{1 - v} \right)^k + 1 = 1 + \left( \frac{1}{v} + 1 \right) \left( \frac{1 + v}{1 - v} \right)^k \\ &= 1 + \left( \frac{1}{\frac{1}{2k+1}} + 1 \right) \left( \frac{1 + \frac{1}{2k+1}}{1 - \frac{1}{2k+1}} \right)^k = 1 + 2 \frac{(k + 1)^{k+1}}{k^k}, \end{aligned}$$

if we set the speed  $v = \frac{1}{2k+1}$ .

We have proved the following result.

**Theorem 10** *There is a monotone strategy that achieves a competitive ratio of  $C_k = 1 + 2(k+1)^{k+1}/k^k$ , where  $k = \lceil \log m \rceil$ , to search on  $m$  rays in parallel if no lower bound on the distance to the target is known.*

Since Theorem 9 remains valid if no lower bound on the distance to the target is known, this strategy is an optimal *monotone* strategy (with or without minimum distance to the target). But we can show an even stronger result if no minimum distance to the target is known.

## 5.2 A Lower Bound

**Theorem 11** *There is no strategy at all that achieves a better competitive ratio than  $C_k = 1 + 2(k+1)^{k+1}/k^k$ , where  $k = \lceil \log m \rceil$ , to search on  $m$  rays in parallel if no minimum distance to the target is known.*

**PROOF.** Any strategy can be specified by the average speed functions  $v_1(T), \dots, v_m(T)$  of the  $m$  robots. (In conjunction with information about whether a robot switches ray, at some point in time.) Given these average speed functions, an adversary can extract information about the time length that a robot moves monotonically along a ray. (This includes also the time that a robot stands still at the origin.) Let  $T_i$ , for  $1 \leq i \leq m$  denote the time that robot  $i$  moves monotonically along a ray. Each  $T_i$  is greater than 0 since either the robot stands still or moves along some ray. Let  $T_{min} = \min_{1 \leq i \leq m} \{T_i\}$ .

Now consider the speeds of the robots in the beginning. Assume that robot 1 is (one of) the robot(s) that starts with the least speed  $v_{min}$ , that is,  $v_{min} = \lim_{T \rightarrow 0} v_1(T)$ .<sup>1</sup> For  $\varepsilon > 0$ , let  $T_\varepsilon$  be the time such that, for all  $0 < T \leq T_\varepsilon$ ,

- (1)  $v_1(T) \leq v_{min} + \varepsilon$  and
- (2)  $v_i(T) \geq v_{min} - \varepsilon$ , for all  $1 \leq i \leq m$ .

The adversary now places the target on ray 1 at distance  $D = v_1(T_D)T_D$  where

$$T_D = \frac{\min\{T_\varepsilon, T_{min}\}}{C_k},$$

for  $k = \lceil \log m \rceil$ . If the strategy uses more than  $C_k T_D$  time, then the competitive ratio is trivially bounded from below by  $C_k$ .

<sup>1</sup> We assume that this limit exists, for all  $1 \leq i \leq m$ . Hence we disallow functions like  $1/2(\sin(1/x) + 1)$  for the average speed.

If, on the other hand, the strategy uses less than  $C_k T_D$  time, then the strategy is monotone in the interesting time interval and we can apply a proof similar to the proof of Lemma 9. Only now  $T_n$  satisfies

$$T_n \geq T_D \left( 1 + 2 \frac{v_{min} - \varepsilon}{1 - v_{min} + \varepsilon} \right)^{\lceil \log n \rceil}.$$

This proves our claim.  $\square$

## 6 Conclusions

We consider search strategies for parallel search on  $m$  concurrent rays. We show that a straight forward generalization of the so called doubling strategy, from searching on the line to searching on  $m$  concurrent rays, yields a competitive ratio of 9 if a minimum distance from the origin to the target is known in advance. Furthermore, we prove that 9 is a lower bound on the competitive ratio for both monotone and symmetric strategies in this case.

We also prove a lower bound of

$$1 + 2 \frac{(k+1)^{k+1}}{k^k}$$

on the competitive ratio, where  $k = \lceil \log m \rceil$ , which applies to monotone strategies and strategies for the case when the minimum distance from the origin to the target is not known in advance. Finally, we give a search strategy that achieves this ratio regardless of whether such a minimum distance is known or not, giving us an optimal search strategy in the latter case.

The question that remains unanswered is whether the lower bound of 9 can be generalized to hold for any strategy.

## Acknowledgments

We would like to thank three anonymous referees for the careful reading of our manuscript and the many constructive suggestions.

## References

- [1] R. Baeza-Yates, J. Culberson, and G. Rawlins. Searching in the plane. *Inform. and Comput.*, 106:234–252, 1993.
- [2] R. Baeza-Yates and R. Schott. Parallel searching in the plane. *Comput. Geom. Theory Appl.*, 5:143–154, 1995.
- [3] M. Betke, R. Rivest, and M. Singh. Piecemeal learning of an unknown environment. *Machine Learning*, 18(2–3):231–254, 1995.
- [4] K-F. Chan and T. W. Lam. An on-line algorithm for navigating in an unknown environment. *Internat. J. Comput. Geom. Appl.*, 3:227–244, 1993.
- [5] A. Datta, Ch. Hipke, and S. Schuierer. Competitive searching in polygons—beyond generalized streets. In *Proc. Sixth Symposium on Algorithms and Computation*, pages 32–41. LNCS 1004, 1995.
- [6] A. Datta and Ch. Icking. Competitive searching in a generalized street. *Comput. Geom. Theory Appl.*, 13:109–120, 1999.
- [7] S. Gal. *Search Games*. Academic Press, 1980.
- [8] Ch. Icking and R. Klein. Searching for the kernel of a polygon: A competitive strategy. In *Proc. 11th Annu. ACM Sympos. Comput. Geom.*, pages 258–266, 1995.
- [9] Ch. Icking, R. Klein, and E. Langetepe. An optimal competitive strategy for walking in streets. In *Proc. 16th Symp. on Theoretical Aspects of Computer Science*, volume 1563 of *Lecture Notes Comput. Sci.*, pages 110–119. Springer-Verlag, 1999.
- [10] M. Y. Kao, Y. Ma, M. Sipser, and Y. Yin. Optimal constructions of hybrid algorithms. *J. Algorithms*, 29:142–164, 1998.
- [11] R. Klein. Walking an unknown street with bounded detour. *Comput. Geom. Theory Appl.*, 1:325–351, 1992.
- [12] J. M. Kleinberg. On-line search in a simple polygon. In *Proc. 5th ACM-SIAM Symp. on Discrete Algorithms*, pages 8–15, 1994.
- [13] A. López-Ortiz and S. Schuierer. Generalized streets revisited. In M. Serna J. Diaz, editor, *Proc. 4th Europ. Symp. on Algorithms*, pages 546–558. LNCS 1136, 1996.
- [14] A. López-Ortiz and S. Schuierer. Searching and on-line recognition of star-shaped polygons. Manuskript, 2000.
- [15] A. Mei and Y. Igarashi. An efficient strategy for robot navigation in unknown environment. *Inform. Process. Lett.*, 52:51–56, 1994.
- [16] C. H. Papadimitriou and M. Yannakakis. Shortest paths without a map. *Theoret. Comput. Sci.*, 84(1):127–150, 1991.

- [17] S. Schuierer and I. Semrau. An optimal strategy for searching in unknown streets. In Ch. Meinel and S. Tison, editors, *Proc. 16th Symp. on Theoretical Aspects of Computer Science*, pages 121–131. LNCS 1563, 1999.
- [18] D. D. Sleator and R. E. Tarjan. Amortized efficiency of list update and paging rules. *Comm. ACM*, 28:202–208, 1985.
- [19] Y. Yin. *Teaching, Learning, and Exploration*. PhD thesis, Department of Mathematics and Laboratory for Computer Science, MIT, 1994.