

# Evacuating Two Robots from Multiple Unknown Exits in a Circle

Jurek Czyzowicz\*  
Département d'informatique  
Université du Québec en  
Outaouais  
Gatineau, Canada

Stefan Dobrev  
Institute of Mathematics  
Slovak Academy of Sciences  
Bratislava, Slovak Republic

Konstantinos Georgiou  
Department of Combinatorics  
& Optimization  
University of Waterloo  
Waterloo, Canada

Evangelos Kranakis\*  
School of Computer Science  
Carleton University  
Ottawa, Canada

Fraser MacQuarrie  
School of Computer Science  
Carleton University  
Ottawa, Canada

## ABSTRACT

Distributed on a unit circle are  $k$  exits. Two autonomous mobile robots are placed on the circle. Each robot has a maximum speed of 1 and the robots can communicate wirelessly. The robots have a map of the domain, including exits, but do not have knowledge of their own initial locations on the domain, rather they only know their relative distance. The goal of the *evacuation problem* is to give an algorithm for the robots which minimizes the time required for both robots to reach an exit, in the worst case.

We consider two variations of the problem depending on whether the two robots have control over their initial distance. When the initial distance of the robots is part of the input (i.e. no control), we show that simple algorithms exist which achieve optimal worst case evacuation times for the cases where: the robots begin collocated with an arbitrary distribution of the exits; and when the exits are either collocated or evenly spaced, with arbitrary starting positions of the robots. We also give upper and lower bounds on the problem with arbitrary exit distribution and starting positions of the robots. For the problem where robots can choose their initial distance (with knowledge of, but not control over the distribution of exits), we propose a natural family of algorithms parameterized by the maximum distance between any two exits.

## Categories and Subject Descriptors

F.2.2 [Computing methodologies]: Analysis of Algorithms and Problem Complexity

\*Research supported in part by NSERC (Natural Sciences and Engineering Research Council) grant.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ICDCN '16 Jan. 4-7, Singapore

Copyright 2016 ACM X-XXXXX-XX-X/XX/XX ...\$15.00.

## General Terms

Algorithms, Theory

## Keywords

Algorithm, circle, evacuation, mobile robots, searching

## 1. INTRODUCTION

Search algorithms have a long and extensive history of study in mathematical and theoretical computer science literature. Examples of variants of search problems and models include: probabilistic search [23], game theoretic applications [4], cops and robbers [10], classical pursuit and evasion [21], search problems and group testing [1], etc.

The search problem we study in this paper involves the evacuation of robots from a domain. Whereas the goal of traditional search problems is to minimize the time required for searchers to find one or more targets hidden in a domain, the goal of the problem studied in this paper is to minimize the time required for every searcher to reach a target (referred to as an *exit*). We can think of many instances where this problem is natural, beginning with the problem of evacuating people from a smoke-filled room. Other potential problem instances include: cows trying to escape from a pasture, robots needing to find a charging station, and boats trying to find safe harbour during a storm. In all these cases, it is not sufficient for the group of searchers to simply find a target; they must all reach a target in order to achieve the goal of the problem.

While *searching* for exits is a core part of the problem, the cost of a solution is measured not by the time required for an exit to be found, but by the time required for both robots to reach an exit. Therefore, we want to take into consideration both the time it takes to find an exit, and the additional time required for evacuation after an exit has been found. This means we may wish to sacrifice some efficiency in terms of finding an exit so that the robots are always relatively close to some exit. Because of the interplay between these two goals, this paper yields interesting results even though the domain is rather simple. We believe these results provide a solid foundation for the investigation of this problem on more complex domains.

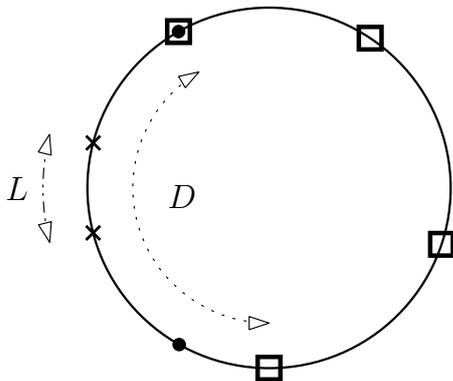


Figure 1: Example of an evacuation problem with 4 exits. Robot is represented by  $\bullet$ , an exit by  $\square$ , and a starting position by  $\times$ .

## 1.1 Preliminaries

Consider a circle of unit length, i.e. of radius  $1/2\pi$ . Distributed on the circle are  $k$  exits. Two autonomous mobile robots are placed on the circle and can move in both directions along the circle. The goal of the *evacuation problem* is to give an algorithm for the robots which minimizes the time required for both robots to reach any exit, in the worst case. Note that the robots do not need to reach the same exit.

We assume that the robots move with unit speed; know at all times the distances (length of the circular arcs) between each other in each direction (and are therefore able to agree on a sense of direction); and are able to communicate wirelessly. Therefore, when one robot discovers an exit, it is able to instantaneously alert the other robot to the location of the exit. We assume that the robots have a map of the environment – knowledge of the length of the circle as well as the relative distance between any two consecutive exits which we assume are indistinguishable. Similarly, we assume that robots do not know the exact exit positions on the circle.

We will use the following notation throughout this paper (see Figure 1). The shortest arc between the starting positions of the two robots will be denoted by  $L$ . For a placement of  $k \geq 2$  exits, we let  $D$  denote the length of the *longest arc* defined by any two consecutive exits. In other words, when  $k = 2$ ,  $D$  is not the distance of the two exits, rather its complement with respect to the disc perimeter (which is 1).

Formally, the problems we study are:

**PROBLEM 1.** (*L given*) Two robots are placed at a given arc distance  $L$  on a circle of perimeter length 1 containing  $k$  exits, where the maximum arc length between any two consecutive exits is  $D$ . Find an algorithm which takes  $D$  and  $L$  as input and minimizes the worst case time for the all robots to evacuate the circle.

**PROBLEM 2.** (*L chosen*) Two robots are placed at a chosen arc distance  $L$  on a circle of perimeter length 1 containing  $k$  exits, where the maximum arc length between any two consecutive exits is  $D$ . Find an algorithm which takes  $D$  as input, but may choose  $L$ , which minimizes the worst case time for the all robots to evacuate the circle.

## 1.2 Related Work

Table 1: Results for Problem 1, where  $D$  and  $L$  are inputs to the problem (i.e. fixed). All results are optimal (i.e. optimal cost,  $C_{opt} = \dots$ ) except for where bounds are given. The upper and lower bounds are parametrized, and should be interpreted as bounding only the specific instance of the problem for a given input of  $D$  and  $L$ . Note that when  $D = 1$ , all exits are colocated.

$D \backslash L$	0	$(0, \frac{1}{2}]$
1	$\frac{3}{4}$	$\frac{3}{4} + \frac{L}{2}$
$(1, \frac{1}{k})$	$\frac{3}{4} \cdot D$	$\frac{3}{4} \cdot D - \frac{L}{2} \leq C_{opt}(D, L) \leq \frac{3}{4} \cdot D + \frac{L}{2}$
$\frac{1}{k}$	$\frac{3}{4} \cdot \frac{1}{k}$	$\frac{3}{4} \cdot \frac{1}{k} + \frac{\delta}{2}$

Search problems involving mobile agents are well studied in the literature. In the context of searching in a geometric domain, the agents are commonly referred to as robots and are often tasked with coordinating their efforts to perform a search.

When the map of their environment is unknown to the robots at the beginning of the search, a common problem is that of *exploring* the environment [2, 3, 14, 18]. However, it is common that the goal of the robots is different, and exploration is only necessary insofar as it facilitates the achievement of that goal [20, 22].

When a map of the environment is known to the robots, problems often involve searching for a target located somewhere in the environment. That target might be stationary, as in the cow path problem [8, 9] or the plane searching problem [5, 6]. The target might also be mobile, where the problems are often presented as search games, commonly called cops and robbers or pursuit-evasion games [10, 12, 16, 21].

Problems relating to evacuation have been studied in a centralized setting, with the goal of creating evacuation plans [7, 15]. Evacuation in a distributed setting, as is the case in this paper, has been studied on a disc in [13] (and the follow-up work [19]), which is similar to the *lost at sea* problem [17]; and on a line in [11]. In [13] the authors introduced and studied the evacuation problem of  $k$  robots through one exit located on the perimeter of a disk when the robots have wireless communication, and when they do not. The major difference between this work and ours is that we only allow robots to move on the perimeter of the disk, whereas robots in [13, 19] can move anywhere on the perimeter or in the interior of the disk. As a result, our model allows us to distinguish the cases when the robots do not have a free choice of their initial arc-distance on the perimeter of the disk (which we denote by  $L$ ), as well as when they do. Moreover, under this topology, the only communication model we are motivated to consider is wireless, since if restricted to face-to-face communication our problem trivializes. Additionally, where [13] studies the problem for  $k$  robots and one exit, we consider the problem with two robots and  $k$  exits.

## 1.3 Outline and Results of the Paper

In Section 2, we consider the problem for a single exit, and give an algorithm which achieves an optimal worst case

Table 2: Costs of Algorithm  $\mathcal{A}$  for Problem 2, where  $D$  is an input to the problem and  $L$  is chosen by the algorithm.

$D$	$L$	$Cost$
$(\frac{6}{7}, 1]$	$1 - D$	$\frac{5}{4} \cdot D - \frac{1}{2}$
$(\frac{4}{5}, \frac{6}{7}]$	$\frac{5}{2} \cdot D - 2$	$1 - \frac{D}{2}$
$[\frac{1}{k}, \frac{4}{5}]$	$0$	$\frac{3}{4} \cdot D$

evacuation cost of  $\frac{3}{4} + \frac{L}{2}$ , where  $L$  is the initial distance between the robots.

In Section 3, we explore the problem with  $k$  exits. We begin by showing an optimal algorithm when the robots begin colocated which achieves a worst case evacuation cost of  $\frac{3}{4} \cdot D$ , where  $D$  is the longest distance between any two exits. We then give, for the case when the exits are evenly spaced, an optimal algorithm which achieves a worst case evacuation cost of  $\frac{3}{4} \cdot \frac{1}{k} + \frac{\delta}{2}$ , where

$$\delta = \begin{cases} L \bmod \frac{1}{k} & : L \bmod \frac{1}{k} \leq \frac{1}{2k} \\ \frac{1}{k} - L \bmod \frac{1}{k} & : L \bmod \frac{1}{k} > \frac{1}{2k} \end{cases}$$

Finally, in Section 4 we further explore the problem with  $k$  exits, but consider the case with arbitrary  $D$  and  $L$ . We begin by giving upper and lower bounds of  $\frac{3}{4} \cdot D + \frac{L}{2}$  and  $\frac{3}{4} \cdot D - \frac{L}{2}$ , respectively, for Problem 1, when  $D$  and  $L$  are inputs. We then finish the section by giving an algorithm for Problem 2 which takes  $D$  as input and chooses  $L$ . This algorithm is able to match the lower bound of Problem 1 for some values of  $D$  and  $L$ . We conjecture that our choice of  $L$  is optimal for these values of  $D$ , which would imply a lower bound for these values of  $D$ .

The results of this paper for Problem 1 (given  $L$ ) are summarized in Table 1. The results of this paper for Problem 2 (chosen  $L$ ) are summarized in Table 2.

## 2. EVACUATION WITH A SINGLE EXIT

We begin by considering the problem with only one exit. To further simplify things, we begin this section by considering the simplest of cases: when there is only one exit, and where the robots begin colocated.

LEMMA 1. *The evacuation problem with a single exit and where the two robots begin colocated ( $L = 0$ ), can be solved with an optimal worst case cost of  $\frac{3}{4}$ .*

PROOF. We first show that the problem can be solved with worst case cost  $\frac{3}{4}$ . Consider the following simple algorithm: The robots begin searching in opposite directions until one of them finds the exit, at which point it will broadcast the location of the exit to the other robot. The other robot then calculates and moves along the shortest route to the exit.

We analyse this algorithm as follows: suppose the robots each move distance  $x$  before one of them finds the exit (see Figure 2). The cost of the algorithm is determined by the time taken by the second robot to exit, and therefore  $C = x + \min(2x, 1 - 2x)$ , or  $C = \min(3x, 1 - x)$ . We note that  $3x$  is monotone increasing and  $1 - x$  is monotone decreasing with respect to  $x$ , and therefore the choice of  $x$  which maximizes

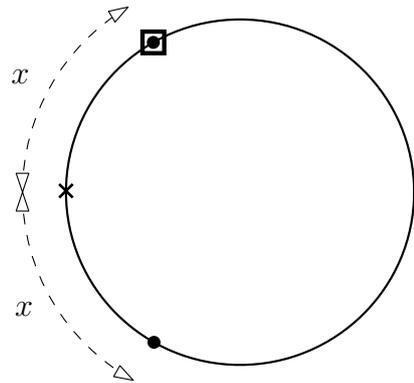


Figure 2: Single exit evacuation of colocated robots.

$\min(3x, 1 - x)$ , and therefore the total cost, will be when  $3x = 1 - x$ , or when  $x = 1/4$ . The worst case cost for our algorithm will therefore be  $3/4$ .

We now show that the algorithm described above is optimal. We define the term *unsearched antipodal pair* to mean a pair of antipodal points on the circle, neither of which has been searched. For any algorithm which solves the problem described in the lemma, during the execution of its worst case there is a point in time  $t$  where one of the points of the last remaining unsearched antipodal pair becomes searched by one of the robots, which we will call  $r_s$ . The exit is then “placed” at the point opposite the one just searched. Clearly the time required for  $r_s$  to exit is at least  $t + 1/2$ . At time  $t$ , the points searched by both robots are of length at most  $2t$ . Since no other unsearched antipodal pair exists at time  $t$ , it must be the case that  $2t \geq 1/2$ . Therefore  $t \geq 1/4$ , which means that the worst case cost of any algorithm is at least  $3/4$ . ■

We now consider the general problem with one exit: where the robots are forced to start at an arbitrary distance  $L$ .

THEOREM 1. *The evacuation problem with a single exit and where the robots begin at distinct starting points at distance  $L \in (0, \frac{1}{2}]$ , can be solved with an optimal worst case cost of  $\frac{3}{4} + \frac{L}{2}$ .*

PROOF. It is easy to show that a worst case cost of  $\frac{3}{4} + \frac{L}{2}$  is feasible. The algorithm achieving this is simply to have the robots walk towards each other until they meet – which takes time  $L/2$  – and then to execute the algorithm from Lemma 1. Therefore the worst case cost is  $\frac{3}{4} + \frac{L}{2}$ .

Note that if there is an unsearched antipodal pair at time  $t = \frac{1}{4} + \frac{L}{2}$ , the theorem follows using the same arguments as in the proof of Lemma 1. Hence, assume that there is no unsearched antipodal pair at time  $t$ . Observe also that since  $L \leq \frac{1}{2}$ ,  $t \leq \frac{1}{2}$ . Since by time  $t$  each robot can search only an area of size at most  $t$ , then if at time  $t$  a robot is located in an area searched by the other robot, its antipodal point is unsearched and so the adversary can place the exit there to force the evacuation time of  $\frac{3}{4} + \frac{L}{2}$ . Therefore, it is sufficient to consider the situation depicted in Figure 3, and due to symmetry, we may assume  $x \leq y$  w.l.o.g.

There are four important points to consider in the segment searched by each robot after time  $t = \frac{1}{4} + \frac{L}{2}$ : The starting location of the robot, the position of the robot at time  $t$ , and the two endpoints of the search segment. The

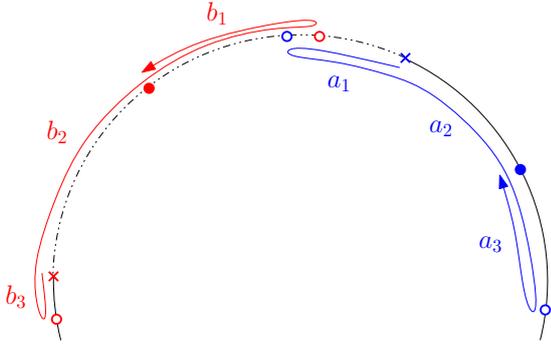


Figure 3: Two robots starting at distance  $L$  (denoted by the dashed line of the circle) exploring the circle up to time  $t = \frac{1}{4} + \frac{L}{2}$ . The crosses represent the starting locations of the robots; the solid points represent the location of each robot at time  $t$ ; and the hollow points represent the endpoints of the segment searched by each robot at time  $t$ .

line segments (and, to simplify notation, also their lengths) between these relevant points of the searched interval are labelled with the following convention:  $a_2$  (respectively  $b_2$ ) is the segment defined by the starting and current position of the robot,  $a_1$  (resp.  $b_1$ ) is the line segment defined by one of the endpoints and the boundary of  $a_2$ , where additionally  $a_1$  belongs to  $L$ , and finally,  $a_3$  (resp.  $b_3$ ) is the remaining segment between the other end point and the other boundary of  $a_2$ , where  $a_3$  does not belong to  $L$ . Note that none of  $a_1, a_2, a_3$  (resp.  $b_1, b_2, b_3$ ) intersect with the possible exception of their endpoints.

The values of  $x$  and  $y$  are defined dependant on whether the segments  $a_2$  and  $b_2$ , respectively, belong to the segment of length  $L$  connecting the starting locations of the robots. They have value 1 if the corresponding segment belongs to  $L$ , and value 0 otherwise. In this case,  $x = 0$  (since  $a_2$  does not belong to  $L$ ) and  $y = 1$  (since  $b_2$  does). Note that  $a_1$  and  $b_1$  may overlap, however, as explained above, we may assume that at time  $t$  no robot is within the area explored by the other robot. Observe also that all the values are non-negative, but many of them can be zero.

From the fact that the robots travel at unit speed we get

$$2a_1 + a_2 + 2a_3 \leq \frac{1}{4} + \frac{L}{2} \quad (1)$$

$$2b_1 + b_2 + 2b_3 \leq \frac{1}{4} + \frac{L}{2} \quad (2)$$

The definition of  $x$  and  $y$  yields

$$yb_2 + b_1 + a_1 + xa_2 \geq L \quad (3)$$

Summing (1) with (2) and subtracting (3) produces

$$2b_3 + (1-y)b_2 + b_1 + a_1 + (1-x)a_2 + 2a_3 \leq \frac{1}{2} \quad (4)$$

At time  $t$ , if the points opposite to the robots are unsearched or are being searched (at time  $t$ ), the adversary can place the exit at such an opposite point, forcing evacuation time of  $\frac{3}{4} + \frac{L}{2}$ .

Hence, it is sufficient to consider only situations in which

$$(1-x)a_2 + a_1 + b_1 + b_2 + b_3 \geq \frac{1}{2} \quad (5)$$

and

$$(1-y)b_2 + b_1 + a_1 + a_2 + a_3 \geq \frac{1}{2} \quad (6)$$

hold. Now, it is sufficient to consider two cases

- Case  $x = 0$ : Since all the values are non-negative, (4) contradicts (6)
- Case  $x = y = 1$ :

$$\begin{aligned} L + \frac{1}{2} &\geq (1) + (2) \\ &= 2b_3 + b_2 + 2b_1 + 2a_1 + a_2 + 2a_3 \\ &\geq b_3 + b_2 + 2b_1 + 2a_1 + a_2 + a_3 \\ &= (6) + (5) \\ &\geq 1 \end{aligned}$$

which is a contradiction since  $L \leq \frac{1}{2}$ . ■

### 3. EVACUATION WITH $K$ EXITS

In this section, we study the more general problem with  $k$  exits. We begin by considering the problem when the  $k$  exits are all collocated, and make the following observation:

**OBSERVATION 1.** *The evacuation problem with  $k$  exits and where the exits are collocated ( $D = 1$ ), is equivalent to the evacuation problem with a single exit.*

Therefore, all the results we have shown for a single exit are also valid for  $k$  collocated exits. We now consider the problem for  $k$  exits with an arbitrary  $D$ , but where the robots begin collocated.

**THEOREM 2.** *The evacuation problem with  $k$  exits, where the robots begin collocated ( $L = 0$ ), can be solved with an optimal worst case cost of  $\frac{3}{4} \cdot D$ , where  $D$  is the longest distance between exits.*

**PROOF.** The solution for this more general problem is very similar to that for Lemma 1. The algorithm is almost identical: The robots begin searching in opposite directions until one of them finds the exit, at which point it will broadcast the location of the exit to the other robot. The other robot then calculates the direction which minimizes the distance which guarantees finding an exit and moves in that direction until it exits.

We analyse this algorithm in a similar manner: suppose the robots each move distance  $x$  before one of them finds an exit (see Figure 4). The second robot can move  $2x$  to evacuate through the same exit as the first robot, or it can continue moving in the same direction, where it will encounter an exit at time no more than  $D - 2x$  (since  $D$  is the maximum distance between exits). The cost of the algorithm is determined by the time taken by the second robot to exit, and therefore  $C = x + \min(2x, D - 2x)$ , or  $C = \min(3x, D - x)$ . We note that  $3x$  is monotone increasing and  $D - x$  is monotone decreasing with respect to  $x$ , and therefore the choice of  $x$  which maximizes  $\min(3x, D - x)$ , and therefore the total cost, will be when  $3x = D - x$ , or when  $x = D/4$ . The worst case cost for our algorithm will therefore be  $\frac{3}{4} \cdot D$ .

We note that the input to the above algorithm is only the longest distance between exits, and at no time are the

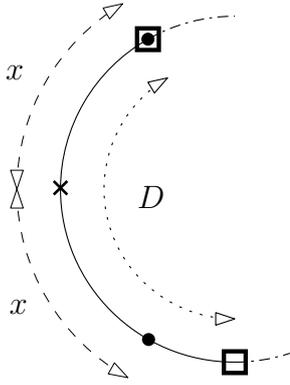


Figure 4:  $k$ -exit evacuation of collocated robots.

robots on any other arc than the one on which they start. Now suppose that the above algorithm is not optimal. Then there exists an algorithm  $\mathcal{A}$  which has cost  $C_{\mathcal{A}} < \frac{3}{4} \cdot D$  for some  $D$ . Now consider a circle of length  $D$  with a single exit. By Lemma 1, the optimal worst case cost for this circle is  $\frac{3}{4} \cdot D$ . However, if we were to execute algorithm  $\mathcal{A}$ , we can achieve a better worst case cost for the circle. This is not possible, and so algorithm  $\mathcal{A}$  cannot exist. ■

Finally, we conclude this section by considering the case when the exits are all evenly spaced.

**THEOREM 3.** *The evacuation problem with  $k$  equally spaced exits ( $D = \frac{1}{k}$ ), and where the robots begin at starting points at distance  $L$ , can be solved with an optimal worst case cost of  $\frac{3}{4} \cdot \frac{1}{k} + \frac{\delta}{2}$ , where*

$$\delta = \begin{cases} L \bmod \frac{1}{k} & : L \bmod \frac{1}{k} \leq \frac{1}{2k} \\ \frac{1}{k} - L \bmod \frac{1}{k} & : L \bmod \frac{1}{k} > \frac{1}{2k} \end{cases}$$

**PROOF.** For the sake of analysis, we assume that the coordinates of the circle are assigned so that the point 0 corresponds with one of the exits. We first observe that since the exits are evenly spaced, when a robot searches a point  $p \in [0, 1]$ , it is simultaneously searching all points  $(p + i/k) \bmod 1$ , for  $i = 1 \dots k$ . We can visualize this as there being “virtual” robots at each of these points which behave exactly as the robot at point  $p$ . Consider now what happens if we take every point of the circle modulus  $(1/k)$ . We begin by noting that every exit is now located at  $0 \bmod (1/k)$ . A robot at point  $p$  and its corresponding virtual counterparts are all located at  $p \bmod (1/k)$ . Each arc of the circle is therefore an exact mirror of the others and we can collapse the problem to the problem of finding a single exit on a circle of length  $1/k$ . An example of this, where  $k = 4$ , is visualized in Figure 5. The only issue remaining to address is the initial distance of the robots,  $L$ . Since  $L \bmod (1/k)$  can be larger than  $1/2k$ , we must adjust the initial distance in the single exit problem to be:

$$\delta = \begin{cases} L \bmod \frac{1}{k} & : L \bmod \frac{1}{k} \leq \frac{1}{2k} \\ \frac{1}{k} - L \bmod \frac{1}{k} & : L \bmod \frac{1}{k} > \frac{1}{2k} \end{cases}$$

The theorem follows immediately from Theorem 1. ■

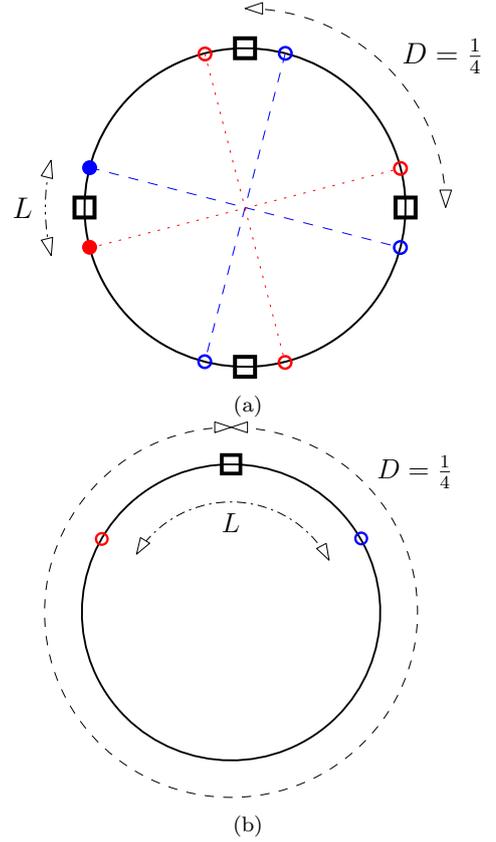


Figure 5: Example transformation to single exit problem for a 4-exit evacuation with equally spaced exits.

#### 4. $K$ -EXIT EVACUATION WITH ARBITRARY $D$ AND $L$

In the previous section, we looked at cases where either  $D$  or  $L$  (or both) were fixed at a special value. We now examine the cases for arbitrary values of  $D$  and  $L$ . We begin by bounding the worst case cost above and below for Problem 1. We then propose an algorithm which gives a solution for Problem 2, for an arbitrary  $D$  when the algorithm has a free choice of  $L$ .

**LEMMA 2.** *The evacuation problem with  $k$  exits, where the robots begin at distance  $L$ , can be solved with a worst case cost of  $\frac{3}{4} \cdot D + \frac{L}{2}$  and cannot be solved with a worst case cost better than  $\frac{3}{4} \cdot D - \frac{L}{2}$ , where  $D$  is the longest distance between exits.*

**PROOF.** Consider the algorithm where the robots walk towards each other until meeting, and then execute the algorithm from Theorem 2. It takes time  $\frac{L}{2}$  until the robots are collocated, and at most  $\frac{3}{4} \cdot D$  to execute the algorithm.

To show the lower bound, assume that there is an algorithm  $\mathcal{A}$  which achieves a worst case cost better than  $\frac{3}{4} \cdot D - \frac{L}{2}$ . We can then construct an algorithm  $\mathcal{A}'$  where two collocated robots first move away from each other until they are distance  $L$  apart. This will take time  $L/2$ . The robots then execute algorithm  $\mathcal{A}$ . Clearly algorithm  $\mathcal{A}'$  has worst case time strictly better than  $\frac{3}{4} \cdot D$ , which is impossible, since  $\frac{3}{4} \cdot D$  is the optimal worst case time for two collocated robots. ■

The upper bound is not surprising, since we know there are values of  $D$  and  $L$  for which this is optimal. What about the lower bound? To this point, every situation we have considered where the robots begin at distinct starting points has lead to higher costs than when the robots begin colocated. Is it possible that there are values of  $D$  and  $L$  for which we can find an algorithm which has a lower cost than the algorithm for the same  $D$  but when the robots are colocated? In fact, it is. We conclude this section by developing an algorithm  $\mathcal{A}$  (stated in Theorem 4) which, for small values of  $L$  and large values of  $D$ , gives a worst case cost which is better than the optimal cost for the same  $D$  when the robots begin colocated.

**DEFINITION 1 (ALGORITHM  $\mathcal{A}_L$ ).** *Let  $L \in [0, 1/2]$  be a fixed constant. For any input to the  $k$ -exit evacuation problem with known exit-distances, algorithm  $\mathcal{A}_L$  places the robots  $A, B$  at distance  $L$  on the circle and at arbitrary positions. The two robots move in opposing directions until one of them, say  $A$  finds an exit (and evacuates) at which point it transmits the information to the second robot. Then robot  $B$  moves to the closest known exit and evacuates.*

**LEMMA 3 (PERFORMANCE OF  $\mathcal{A}_L$ ).** *Let  $D \in [1/2, 1]$ , which represents the longest distance between exits, be an input to the  $k$ -exit evacuation problem. Then, for every  $L \in (0, 1 - D]$ , the performance of  $\mathcal{A}_L$  is exactly*

$$\max_{x \in [0, 1/2 - L/2]} g(x),$$

where

$$g(x) := \begin{cases} \min\{3x + L, D - L - x\} & , \text{ if } x \geq 1 - D \\ \min\{3x + L, 1 - L - x\} & , \text{ if } x < 1 - D \end{cases}$$

**PROOF.** Since  $L \in (0, 1 - D]$ , we know that not all the exits can lie in the interval between the starting positions of the robots, and so the robots will necessarily encounter at least one exit before they meet. Denote the time when the first exit is found as  $x$ . Moreover, after time  $1/2 - L/2$ , and given that no robot has located an exit until then, the two robots meet on the ring. Hence, if  $x$  denotes the time it takes until one robot locates the exit, then  $x \in [0, 1/2 - L/2]$ .

In order to evaluate the performance of  $\mathcal{A}_L$  we need to examine two cases (shown in Figure 6).

In the first case (Figure 6a), the exit is located after time  $x \geq 1 - D$ , say by robot  $A$ . We know all exits are contained in an interval of size  $1 - D$ . Since both robots have already travelled distance  $1 - D$ , it is impossible for there to be any exits in the interval between their starting positions. Therefore, the second robot knows immediately where the two closest exits are: The exit through which the first robot evacuated, at distance  $2x + L$  in the opposite direction; and an exit at distance  $D - 2x - L$  in the same direction. Since it will choose the best route, the cost in this case is  $x + \min\{2x + L, D - L - 2x\}$ .

In the second case (Figure 6a), we assume that the exit is located after time  $x < 1 - D$ , say again by robot  $A$ . If robot  $B$  continues moving in the same direction, then since in the worst case the undiscovered exits could be located in the arc between the initial positions of the two robots, robot  $B$  will find the discovered exit in time  $1 - 2x - L$ . If robot  $B$  chooses to reverse back, then the undiscovered exits might

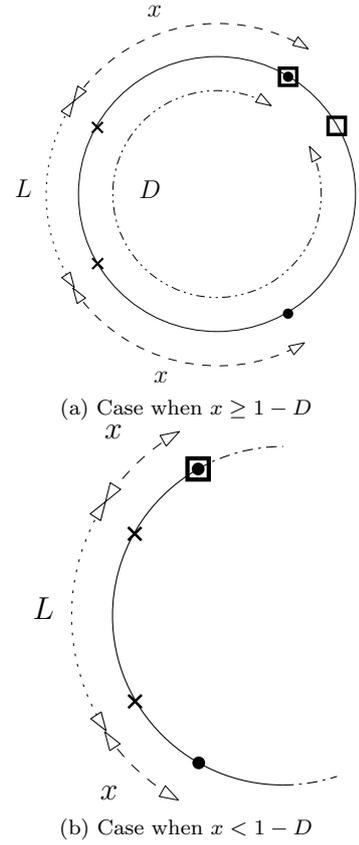


Figure 6:  $k$ -exit evacuation for algorithm  $\mathcal{A}_L$ .

not be in the arc between the initial positions of the robots, in which case robot  $B$  will find again the discovered exit in additional time  $2x + L$ . To conclude, if  $x < D$ , the cost of the algorithm in the worst case is  $x + \min\{2x + L, 1 - L - 2x\}$ . ■

Given that  $D$  will be given as input to the new evacuation problem, an algorithm may adapt its evacuation strategy accordingly. This is what we propose and explore next.

**THEOREM 4.** *Let  $D \in [1/k, 1]$ , the longest distance between exits, be an input to the  $k$ -exit evacuation problem where  $L$  is a free choice. Consider the algorithm  $\mathcal{A}$  that simulates  $\mathcal{A}_L$ , where  $L = L(D)$  is chosen according to the following rule*

$$L = \begin{cases} 1 - D & , \text{ if } 6/7 < D \leq 1 \\ 5D/2 - 2 & , \text{ if } 4/5 \leq D \leq 6/7 \\ 0 & , \text{ if } 1/k \leq D < 4/5 \end{cases}$$

Then the cost  $C(D)$  of algorithm  $\mathcal{A}$  is

$$C(D) = \begin{cases} 5D/4 - 1/2 & , \text{ if } 6/7 < D \leq 1 \\ 1 - D/2 & , \text{ if } 4/5 \leq D \leq 6/7 \\ 3D/4 & , \text{ if } 1/k \leq D < 4/5 \end{cases}$$

**PROOF.** Note that  $L = L(D)$  always satisfies the condition that  $L \leq 1 - D$ , hence Lemma 3 is applicable for all

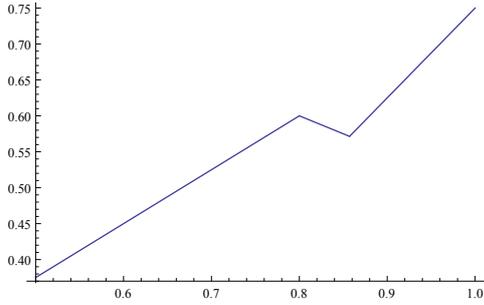


Figure 7: Performance of Algorithm  $\mathcal{A}$  for  $k = 2$  exits.

values of  $D \in [1/2, 1]$ . Then, the cost  $C(D)$  of algorithm  $\mathcal{A}$  can be evaluated by an exhaustive case analysis that depends on the relative value of  $D$  with respect to the critical values  $1, 6/7, 4/5, 1/k$ , and it is depicted in Figure 7 for  $k = 2$  exits.

We begin by observing that when  $L = 0$ , the cost of algorithm  $\mathcal{A}$  is  $3D/4$  by Theorem 2. We now evaluate the remaining cases by considering the point in time  $x$  after which the first exit has been found.

**Case:**  $x \geq 1 - D$

From Lemma 3, we know that the time to evacuate for robot  $B$  is  $\min(3x + L, D - L - x)$ . We note that  $3x + L$  is monotone increasing and  $D - L - x$  is monotone decreasing with respect to  $x$ , and therefore the choice of  $x$  which maximizes  $\min(3x + L, D - L - x)$ , and therefore the total cost, will be when  $3x + L = D - L - x$ , or when  $x = D/4 - L/2$ . In this case, the cost will therefore be  $\frac{3}{4} \cdot D - \frac{L}{2}$ .

**Case:**  $x < 1 - D$

From Lemma 3, we know that the time to evacuate for robot  $B$  is  $\min(3x + L, 1 - L - x)$ . We will focus solely on the case where the cost is  $3x + L$  (even if this is more expensive than  $1 - L - x$ ). We want to choose  $L$  to minimize

$$\max\left(3x + L, \frac{3}{4} \cdot D - \frac{L}{2}\right).$$

Since  $3x + L$  is increasing, and  $3D/4 - L/2$  is decreasing w.r.t.  $L$ , this is minimized when

$$3x + L = \frac{3}{4} \cdot D - \frac{L}{2}.$$

Since,  $x \leq 1 - D$ , this holds if

$$3 - 3D + L \leq \frac{3}{4} \cdot D - \frac{L}{2}.$$

Rearranging, we see that

$$L \leq \frac{5}{2} \cdot D - 2,$$

and since  $L \geq 0$ , this only holds if

$$\frac{4}{5} \leq D.$$

Furthermore, we choose  $L = 5D/2 - 2$  to minimize the cost of the algorithm. Finally we note that  $L < 1 - D$ , and so when

$$1 - D \leq \frac{5}{2} \cdot D - 2,$$

we must set  $L$  to a maximum value of  $L = 1 - D$ . Rearranging the above, we see that this happens when  $D > 6/7$ . Putting these  $L$  values into  $3D/4 - L/2$  give us the costs listed in the Theorem. ■

We conclude this section with a conjecture, which if true would imply that Algorithm  $\mathcal{A}$  is optimal for  $D \geq 6/7$ .

**CONJECTURE 1.** For  $D \geq 1/2$ , any choice of  $L$  such that  $L > 1 - D$  is suboptimal for Problem 2.

## 5. CONCLUSION AND OPEN PROBLEMS

In this paper, we studied the evacuation problem on a circle for 2 robots and  $k$  exits, and showed many optimal results. It is not immediately intuitive that the worst case solution costs are independent of the number of exits, but are dependent on the longest distance between exits.

With regards to future work, we would like to prove our conjecture. It remains open to further explore the arbitrary cases of  $D$  and  $L$  and provide an algorithm for this problem. It would also be interesting to utilize more knowledge of the exit distribution to tighten the bounds for values of  $D$  and chosen values of  $L$ . It would valuable to better understand the reason why some values of  $D$  allow for choices of  $L$  which beat the algorithms for colocated robots, while some do not.

Study of the problem with more than two robots is a natural extension to this work. Some results are known for the case of multiple robots with one exit on a disc (instead of a circle) [13], and even in this case optimal results do not yet exist.

The study of the problem on different topologies is completely open, as is the study of the problem under different communications models, while another interesting variant of the problem would be the case where only a subset of the robots need to be evacuated.

## 6. REFERENCES

- [1] R. Ahlswede and I. Wegener. *Search problems*. Wiley-Interscience, 1987.
- [2] S Albers and Monika R. Henzinger. Exploring unknown environments. *SIAM J. Comput.*, 29(4):1164–1188, 2000.
- [3] Susanne Albers, Klaus Kursawe, and Sven Schuierer. Exploring unknown environments with obstacles. *Algorithmica*, 32(1):123–143, 2002.
- [4] S. Alpern and S. Gal. *The theory of search games and rendezvous*, volume 55. Kluwer Academic Publishers, 2002.
- [5] R. Baeza Yates, J. Culberson, and G. Rawlins. Searching in the plane. *Information and Computation*, 106(2):234–252, 1993.
- [6] Ricardo Baeza-Yates and René Schott. Parallel searching in the plane. *Computational Geometry*, 5(3):143–154, 1995.
- [7] Nadine Baumann and Martin Skutella. Earliest arrival flows with multiple sources. *Mathematics of Operations Research*, 34(2):499–512, 2009.
- [8] Anatole Beck. on the linear search problem. *Israel Journal of Mathematics*, 2(4):221–228, 1964.
- [9] R. Bellman. An optimal search. *Siam Review*, 5(3):274–274, 1963.

- [10] A. Bonato and R. Nowakowski. *The game of cops and robbers on graphs*. American Mathematical Soc., 2011.
- [11] Marek Chrobak, Leszek Gaŝieniec, Thomas Gorry, and Russell Martin. Group search on the line. In *SOFSEM 2015: Theory and Practice of Computer Science*, pages 164–176. Springer, 2015.
- [12] Timothy H Chung, Geoffrey A Hollinger, and Volkan Isler. Search and pursuit-evasion in mobile robotics. *Autonomous Robots*, 31(4):299–316, 2011.
- [13] Jurek Czyzowicz, Leszek Gaŝieniec, Thomas Gorry, Evangelos Kranakis, Russell Martin, and Dominik Pajak. Evacuating robots via unknown exit in a disk. In *Distributed Computing*, pages 122–136. Springer, 2014.
- [14] Xiaotie Deng, Tiko Kameda, and Christos H. Papadimitriou. How to learn an unknown environment (extended abstract). In *FOCS*, pages 298–303, 1991.
- [15] Sándor Fekete, Chris Gray, and Alexander Kröller. Evacuation of rectilinear polygons. In *Combinatorial Optimization and Applications*, pages 21–30. Springer, 2010.
- [16] Fedor V. Fomin and Dimitrios M. Thilikos. An annotated bibliography on guaranteed graph searching. *Theor. Comput. Sci.*, 399(3):236–245, 2008.
- [17] Brian Gluss. An alternative solution to the “lost at sea” problem. *Naval Research Logistics Quarterly*, 8(1):117–122, 1961.
- [18] Frank Hoffmann, Christian Icking, Rolf Klein, and Klaus Kriegel. The polygon exploration problem. *SIAM Journal on Computing*, 31(2):577–600, 2001.
- [19] Czyzowicz J. and Georgiou K., Kranakis E., Narayanan L., Opatrny J., and Vogtenhuber B. Evacuating robots from a disk using face-to-face communication. In *CIAC*, volume to appear, 2015.
- [20] Jon M Kleinberg. On-line search in a simple polygon. In *Proceedings of the fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 8–15. Society for Industrial and Applied Mathematics, 1994.
- [21] P. Nahin. *Chases and Escapes: The Mathematics of Pursuit and Evasion*. Princeton University Press, 2012.
- [22] C. H. Papadimitriou and M. Yannakakis. Shortest paths without a map. *Theoretical Computer Science*, 84(1):127–150, 1991.
- [23] L. D. Stone. *Theory of optimal search*. Academic Press New York, 1975.