Evacuating an Equilateral Triangle in the Face-to-Face Model*

Huda Chuangpishit[1], Saeed Mehrabi[2], Lata Narayanan[3], and Jaroslav Opatrny[4]

1 **Department of Computer Science, Concordia University, Montreal, Canada**
hoda.chuang@gmail.com
2 **School of Computer Science, Carleton University, Ottawa, Canada**
saeed.mehrabi@carleton.ca
3 **Department of Computer Science, Concordia University, Montreal, Canada**
lata@cs.concordia.ca
4 **Department of Computer Science, Concordia University, Montreal, Canada**
opatrny@cs.concordia.ca

Abstract

Consider k robots initially located at the centroid of an equilateral triangle T of sides of length one. The goal of the robots is to *evacuate* T through an exit at an unknown location on the boundary of T. Each robot can move anywhere in T independently of other robots with maximum speed one. The objective is to minimize the *evacuation time*, which is defined as the time required for *all* k robots to reach the exit. We consider the *face-to-face* communication model for the robots: a robot can communicate with another robot only when they meet in T.

In this paper, we give upper and lower bounds for the face-to-face evacuation time by k robots. We show that for any k, any algorithm for evacuating $k \geq 1$ robots from T requires at least $\sqrt{3}$ time. This bound is asymptotically optimal, as we show that a straightforward strategy of evacuation by k robots gives an upper bound of $\sqrt{3} + 3/k$. For $k = 3, 4, 5, 6$, we show significant improvements on the obvious upper bound by giving algorithms with evacuation times of 2.0887, 1.9816, 1.876, and 1.827, respectively. For $k = 2$ robots, we give a lower bound of $1 + 2/\sqrt{3} \approx 2.154$, and an algorithm with upper bound of 2.3367 on the evacuation time.

1998 ACM Subject Classification I.1.2. Algorithms, D.1.3 Distributed programming

Keywords and phrases Distributed algorithms, Robots evacuation, Face-to-face communication, Equilateral triangle

Digital Object Identifier 10.4230/LIPIcs.OPODIS.2017.11

1 Introduction

Searching for an object at an unknown location in a specific domain in the plane is a well-studied problem in theoretical computer science [1, 4, 5, 21, 22]. The problem was initially studied when there is only one searcher, whom we refer to as a *robot*. The target is assumed to be a point in the domain, and the robot can only find the target when it visits that point. The goal then is to design a trajectory for the robot that finds the target as soon as possible. Recent work has focused more on *parallel* search by several robots, which can reduce the

* The work of LN and JO is supported in part by NSERC. This work was done when SM was visiting Concordia University.

21st International Conference on Principles of Distributed Systems (OPODIS 2017).
Editors: James Aspnes, Alysson Bessani, Pascal Felber, and João Leitão; Article No. 11; pp. 11:1–11:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

search time as the robots can distribute the search effort among themselves. The *search time* by k robots is generally defined to be the time the *first* robot reaches the target.

A natural generalization of the parallel search problem, called the *evacuation problem*, was recently proposed: consider several robots inside a region that has a single *exit* at an unknown location on its boundary. All robots need to reach the exit, i.e., *evacuate* the region, as soon as possible. This is essentially the parallel search problem where the exit is the search target, however we are interested in minimizing the time the *last* robot arrives at the exit. The problem was introduced in [8] (see also [10]), and a number of papers on this problem have been published since then.

The evacuation problem substantially depends on the way robots can communicate among themselves. Two models of communication have been proposed: in the *wireless model*, each robot can communicate wirelessly with the other robots instantaneously, regardless of their locations. In the *face-to-face model*, two robots can communicate with each other only when they meet, i.e., when they occupy the same location at the same time. Since in the wireless model robots can communicate with each other regardless of their locations, as soon as a robot finds the exit, it can announce it to other robots. This is not possible in the face-to-face communication, which makes the evacuation problem more challenging, due to the very limited communication capabilities of robots.

In this paper, we study the problem of evacuating a unit equilateral triangle in the face-to-face model with k robots, all of which are initially located at the centroid of the triangle. Our objective is to design the trajectories of the robots so as to minimize the *worst-case evacuation time*, which is defined as the time it takes for *all the robots* to reach the exit.

Related work. A classical problem related to our paper is the well-known *cow-path* problem introduced by A. Beck [4], in which a cow searches for a hole in an infinite linear fence. An optimal deterministic algorithm for this problem and for its generalization to several fences is known, e.g., Baeza-Yates et al. [2]. Since then several variants of the problem have been studied [3, 6, 7, 11, 15, 16, 18, 17, 20].

Lopez-Ortiz and Sweet [20] asked for the worst-case trajectories of a set of robots searching *in parallel* for a target point at an unknown location in the plane. Feinerman et al. [14] (see also [13]) introduced a similar problem in which a set of robots that are located at a cell of an infinite grid and being controlled by a Turing machine (with no space constraints) need to find the target at a hidden location in the grid. In these two models of multi-robot searching, the robots cannot communicate at all. By controlling each robot by asynchronous finite state machine, Emek et al. [12] studied this problem in which the robot can have a "local" communication in some sense and proved that the collaboration performance of the robots remains the same, even if they possess a constant-size memory. Lenzen et al. [19] extended this problem by introducing the *selection complexity* measure as another factor in addition to studying the time complexity of the problem.

The evacuation problem with several robots has been studied in recent years under wireless and face-to-face models of communications. For the wireless model, Czyzowicz et al. [8] studied the problem of evacuating a unit disk, starting at the center of the disk. They gave a tight bound of 4.83 for the evacuation time of $k = 2$ robots, as well as upper and lower bounds of, respectively, 4.22 and 4.159 for $k = 3$. These bounds for k robots become $3 + \pi/k + O(k^{-4/3})$ and $3 + \pi/k$, respectively [8]. Czyzowicz et al. [11] also studied the evacuation problem for k robots for unit-side squares and equilateral triangles in the wireless model. For a unit-side square, they gave optimal algorithms for evacuating $k = 2$

robots when located at the boundary of the square. Moreover, for an equilateral triangle, they gave optimal evacuation algorithms for $k = 2$ robots in any initial position on the boundary or inside the triangle. They also showed that increasing the number of robots cannot improve the evacuation time when the starting position is on the boundary, but three robots can improve the evacuation time when the starting position is the centroid of the triangle. Recently, Brandt et al. [6] considered the evacuation problem for k robots on m concurrent rays under the wireless model. Finally, the evacuation problem on a disk with three robots at most one of which is *faulty* was recently studied by Czyzowicz et al. [9] under the wireless model.

For the face-to-face model, Czyzowicz et al. [8] gave upper and lower bounds of, respectively, 5.74 and 5.199 for the evacuation time of $k = 2$ robots initially located at the center of a unit disk. Both the upper and lower bounds were improved by Czyzowicz et al. [10] to 5.628 and 5.255, respectively. Closing this gap remains open. When $k = 3$ the upper and lower bounds for the face-to-face model are 5.09 and 5.514, respectively, and $3 + 2\pi/k$ and $3 + 2\pi/k - O(k^{-2})$ for any $k > 3$ [8].

Our results and organization. In this paper, we study the evacuation of k robots from an equilateral triangle under the face-to-face model. We present the following results:

- For $k \geq 3$, we show that any algorithm for evacuating k robots from triangle T requires at least $\sqrt{3}$ time. We prove that this bound is asymptotically optimal by giving a simple algorithm that achieves an upper bound of $\sqrt{3} + 3/k$.
- We show that a significant improvement on the above upper bound can be obtained using the *Equal-Travel Early-Meeting* strategy. In this strategy the travel time of all robots is the same and they use a *meeting point* for all robots before the entire boundary is explored to share information. Applying this strategy we design algorithms for k =2,3,4,5, and 6 with evacuation times of ≈ 2.4114, 2.0887, 1.982, 1.8760 and 1.823, respectively.
- For $k = 2$ we prove a lower bound of 2.154 on the evacuation time. We improve the evacuation algorithm for $k = 2$ even further by replacing an early meeting with one or several *detours* inside the triangle which improves the evacuation time to 2.3367.

We specify some preliminaries and notation in Section 2. Then, we give the proofs of our lower bounds in Section 3, and present our evacuation algorithms in Section 4. We conclude the paper with a summary of our results and a discussion of open problems in Section 5.

2 Preliminaries

For two points p and q in the plane, we denote the line segment connecting p and q by pq and the length of pq by $|pq|$. Throughout the paper, we denote an equilateral triangle by T and denote the vertices of T by A, B, and C. Thus we sometimes write ABC to refer to T. We always assume that the sides of T have length 1 and every robot moves at maximum speed 1. Throughout the paper we use the following triangle terminology:

- By h we denote the *height* of the equilateral triangle. Observe that $h = \sqrt{3}/2$.
- We denote by O the *centroid* of T (i.e., the intersection point of the three heights of T).
- We use x, and y to denote the distance of O to a vertex, and to the side of the triangle, respectively; notice that $x = 2h/3 = \sqrt{3}/3$ and $y = h/3 = x/2 = \sqrt{3}/6$.

We define $E_{\mathcal{A}}(T, k)$ to be the worst-case evacuation time of the unit-sided equilateral triangle T by k robots using algorithm $\mathcal{A}$, assuming the robots are initially located at the centroid of the triangle, the exit is located at an unknown location on the boundary of the triangle, and the robots communicate using the face-to-face model. Also, we define $E^*(T, k)$ to be the *optimal* evacuation time of the triangle by k robots in the face-to-face model.

A deterministic algorithm for the evacuation problem by k robots takes as input the triangle and the k robots located at its centroid, and outputs for each robot a *fixed trajectory* consisting of a sequence of connected line segments or curves to be followed. We assume every robot knows the trajectories of all the robots. A robot R follows its trajectory unless:

- R sees the exit: R may then quit its trajectory and go to a point where it can intercept another robot and inform it about the exit.
- R meets another robot who has found the exit: R then quits its trajectory and proceeds directly to the exit.

Observe that the robots are initially co-located, and the initial part of their trajectories may be identical, i.e., when going to the boundary of the triangle. Later on, the trajectories of two or several robots may intersect and the intersection point may be reached by all robots at the same time. We call such a point a *meeting point*. A meeting point might be in the interior of the triangle and it can serve as a place for the robots to exchange information about the progress in the search for exit. If one of the robots has found the exit, they can proceed towards it. Otherwise, the robots can continue in the search for the exit separately or together. As shown in [11], and in Section 4, an algorithm with a meeting point in the interior of the region can improve the evacuation time in some cases.

If the trajectory of a robot leaves the boundary of the triangle and returns to the boundary without a planned meeting point, we say that the robot makes a *detour*. The robot may make such a detour to enable another robot that has found the exit to intercept it. In the absence of an interception, the robot has gained information about the *absence* of the exit in some part of the boundary. A detour in the trajectories of two robots was used in [10] to improve the evacuation time, and we use this strategy in Subsection 4.2.

3 Lower Bounds

In this section, we prove lower bounds on the evacuation time. We first show that regardless of the number of robots, $\sqrt{3}$ is a lower bound on the evacuation time. This bound holds even if the exit is known to be at one of the three vertices of the triangle.

▶ **Theorem 1.** *Consider k robots $R_1, R_2, \ldots, R_k$, initially located at the centroid of an equilateral triangle T. In the face-to-face model the evacuation time of k robots $E^*(T, k) \geq \sqrt{3} \approx 1.732$.*

Proof. Consider a deterministic and arbitrary evacuation algorithm $\mathcal{A}$ for k robots. We first run the algorithm to see which vertex is the last one visited by the robots (two or even three vertices could be visited at the same time as the last ones in which case, we choose an arbitrary one as last). Assume without loss of generality that A is the last vertex visited by any of the robots; let I_1 be the input in which the exit is at A. Consider the execution of the algorithm on input I_1, and let t be the time the second of the three vertices is visited by some robot R. Without loss of generality, let this second vertex be B; that is, R visits vertex B at time t on input I_1. Let I_2 be the input where the exit is at the remaining vertex C. We argue that the evacuation time of the algorithm must be $\geq 3x$ on one of these two inputs.

If $t \geq 3x - 1$, then it takes an additional time 1 for robot R to reach the exit at A, leading to a total evacuation time of at least $3x$ on input I_1. Therefore, assume that $t < 3x - 1$. Since R has to reach B before time $3x - 1$, we claim that it is impossible for R to meet a robot R' that has already visited A or C before R reaches B at time t. Suppose R was able to meet R' that had visited A (without loss of generality) at some meeting point M at time t_M. Then clearly $t_M \geq x + |AM|$. After meeting R', the robot R needs time at least $|MB|$ to get to B. We conclude that $t \geq t_M + |MB| \geq x + |MB| + |MA| \geq x + 1$. However, $x + 1 > 3x - 1$, a contradiction. Thus, R's trajectory to B, reaching B at time $t < 3x - 1$ cannot allow a meeting between R and any robot that has already visited A or C. Therefore, the behaviour of the robot R must be the same on inputs I_1 and I_2 until time t when R arrives at B. Observe now that $t \geq x$. At time $2x$, if the robot R is at distance $> x$ from A, the adversary puts the exit at A (input I_1), and if it is at distance $> x$ from C, it puts the exit at C. Combined with the fact that at time $2x$, the robot R can travel at most distance $2x - t \leq x$ from B, we have the desired result. ◀

The above bound is asymptotically optimal, as we will describe a simple algorithm in Section 4 that evacuates k robots in $\sqrt{3} + 3/k$ time from an exit situated anywhere on the boundary. We remark also that in the wireless communication model, $E^*(T, 6) = \frac{2\sqrt{3}}{3}$ (D. Krizanc, private communication, 2015), showing that for the equilateral triangle, evacuation even with arbitrarily many robots takes much more time in the face-to-face model, than evacuating only six robots in the wireless model.

When $k = 2$, we are able to prove a better lower bound of ≈ 2.15. The argument used for the lower bound is an adversary argument: depending on what the algorithm does, the adversary places the exit in such a way so as to force the claimed evacuation time. The key insight can be summarized as follows: if an algorithm is to do better than the claimed lower bound, either the robots cannot meet in a useful way to shorten the time to reach the exit, or they simply cannot finish the exploration. To this end, we first prove the following technical lemma. We first need some notation. For the equilateral triangle ABC, let D, E and F denote the middle point of sides AB, AC and BC, respectively, and let $\mathcal{S} = \{A, B, C, D, E, F\}$. We say two points in $\mathcal{S}$ have *opposite positions* if one point is a vertex of the triangle T and the other point is located on the opposite side of that vertex. For example, the vertex C and a point in $\{A, D, B\}$ have opposite positions.

▶ **Lemma 2** (Meeting Lemma). *Consider a deterministic algorithm $\mathcal{A}$ for evacuating two robots that are initially at the centroid of an equilateral triangle T, and let $p_1, p_2 \in \mathcal{S}$ have opposite positions. If $\mathcal{A}$ specifies a trajectory for one of the robots in which it visits p_1 at time t' satisfying $t' \geq 0.5 + y$ and a trajectory for the other robot in which it visits p_2 at time t such that $t' < t < 0.5 + h + y = 0.5 + 4y$, then the two robots cannot meet between time t and t'.*

Proof. Suppose for a contradiction that the robots meet at time $t' < t_m \leq t$ at some point z. Since p_1 and p_2 have opposite positions $|p_1 p_2| \geq h$. Therefore, $|p_1 z| + |z p_2| \geq h$. Moreover $|p_1 z| \leq t_m - t'$ and $|z p_2| \leq t - t_m$. This implies that

$$h \leq |p_1 z| + |z p_2| \leq (t_m - t') + (t - t_m) = t - t' < 0.5 + h + y - t' \leq 0.5 + h + y - 0.5 - y = h,$$

which is a contradiction. ◀

▶ **Theorem 3.** *Consider 2 robots R_1, R_2, initially located at the centroid of an equilateral triangle T. If the robots communicate using face-to-face model, then the evacuation time of two robots $E^*(T, 2) \geq 1 + 4y = 1 + 2/\sqrt{3}$.*

Proof. Suppose for the purpose of contradiction, that there is a deterministic algorithm $\mathcal{A}$ for evacuation by two robots, such that $E_{\mathcal{A}}(T, 2) < 1 + 4y$. We first focus attention on the set of points $\mathcal{S} = \{A, B, C, D, E, F\}$. There exists some input I on which the exit is the last

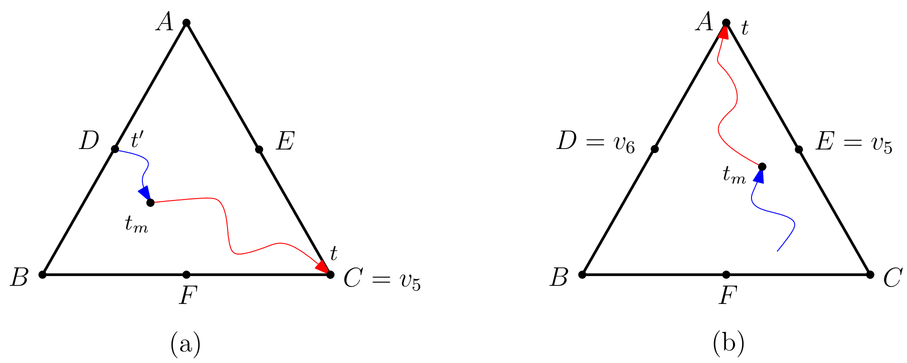

Figure 1 (a) An illustration in support of the proof of the Meeting Lemma. (b) An illustration in support of case 2 in the proof of Theorem 3.

point in $\mathcal{S}$ to be visited by either of the robots, according to the trajectories specified by $\mathcal{A}$. Let t be the time the *fifth* point of $\mathcal{S}$ is visited by a robot on input I. Let $v_1, v_2, \ldots, v_6$ be the order in which the points in M are visited by the robots, on input I; the exit is at v_6. Without loss of generality assume that v_5 is visited by robot R_1. Clearly, v_6 is not yet visited before time t; it may be visited at or after time t. First, note that since at least five points are visited at or before time t, one of the robots must have visited at least three points in M. It follows that $t \geq 1 + y$. If $t \geq 0.5 + 4y$, since the exit is at v_6, which is at least 0.5 away from R_1, we obtain $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$, a contradiction. We conclude that $1 + y \leq t < 0.5 + 4y$.

We now consider the following exhaustive cases depending on whether v_5 is a vertex of T or a midpoint of a side of T.

Case 1. v_5 is a vertex of T. Without loss of generality assume that v_5 is C. See Figure 1(a). If v_6 is any of A, D, B, then at time t, R_1 needs time at least h to arrive to v_6, which implies that $E^*(T, 2) \geq t + h \geq 1 + 4y$, a contradiction. So we conclude that v_6 is at either E or F. Since $t < 0.5 + 4y$, robot R_1 could have visited at most one of A, D, B by time t. This means that R_2 must have visited at least two of A, D, B. Let v be the second vertex of the set A, D, B to be visited by R_2, and assume it arrives there at time t'. Note that $t' \geq 0.5 + y$. By the Meeting Lemma, the two robots do not meet at any time between t' and t on input I.

Now consider an input I' in which the exit is at v. Clearly the robots behave identically on both inputs I and I' until time t'. After this time, R_2 on seeing the exit at v may behave differently; however robot R_1 must behave exactly as in I unless it meets robot R_2, which by the Meeting Lemma, cannot happen until time t. Therefore, after time t, it takes at least an additional h to reach the exit at v, giving a total evacuation time of at least $t + h \geq 1 + 4y$, a contradiction.

Case 2. v_5 is a midpoint of a side of T, and v_6 is another midpoint: Without loss of generality assume that v_5 is E, and v_6 is D; see Figure 1(b). Then, all three vertices must have been visited before or at time t. Since R_1 cannot visit two vertices *before* arriving at E at time $t < 0.5 + h + y$, we conclude that R_2 must visit two vertices by time t. Referring to Figure 1(b), consider the second vertex visited by R_2. Observe that R_2 cannot arrive there before time $1 + x$. (i) If it is B, then we put the exit at E. This way, R_2 needs time at least h to get to E from B and so $E^*(T, 2) \geq 1 + x + h \geq 1 + 4y$. (ii) If it is C, then we put the exit at D. Then, R_2 needs time at least h to get to D from C and so $E^*(T, 2) \geq 1 + x + h \geq 1 + 4y$.

We conclude that the second vertex visited by R_2 must be A. We first note that R_2 cannot have visited all F, B, A or all F, C, A by time t, as visiting either set of three points takes time at least $0.5 + x + h > 0.5 + y + h > t$. So, R_1 must have visited F and C (or F

and B) before coming to E. Let P be the second of the two points visited by R_1 before going to E. Note that P could be F, C, or B, and suppose R_1 visits P at time t'. Clearly, $t' \geq 0.5 + y$. Since R_2 must visit A at or before time $t < 0.5 + 4y$, by the Meeting Lemma, the robots cannot meet at any time $t' \leq t_m \leq t$. In other words, R_1 cannot meet R_2 after the latter has visited P and reach A on time. Now consider the input I' in which the exit is at P. On input I', the robot R_2 will have the same behavior as input I until it reaches A at time $t > 1 + x$ and then needs to get to the exit (which is at F or C or B). Therefore we have $E^*(T, 2) \geq 1 + x + h \geq 1 + 4y$, a contradiction.

Case 3. v_5 is a midpoint of a side of T, and v_6 is a vertex: Without loss of generality assume that v_5 is E. If v_6 is B, then R_1 needs time $\geq h$ to reach the exit, so on input I, the evacuation time is at least $t + h \geq 1 + h + y$. Therefore, assume without loss of generality that v_6 is A; see Figure 1(b). If a single robot visits both B and C, it takes time at least $2 + x > 1 + 4y$ to reach vertex A. Therefore, B and C must be visited by different robots. We consider separately the two cases: R_1 visits C and R_2 visits B; and R_1 visits B and R_2 visits C.

Suppose R_1 visits C before visiting E, and R_2 visits B. First observe that R_1 cannot also visit D, as visiting C, D, and E takes time at least $0.5 + 4y$, a contradiction to $t_E < 0.5 + 4y$. Therefore R_2 must visit D in addition to B. Either R_1 or R_2 must visit F. If R_2 visits F, Lemma 5 assures that $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$ and if R_2 does not visit F, Lemma 6 does the same.

Suppose instead that R_1 visits B before visiting E and R_2 visits C. Then R_1 cannot visit both D and F, as visiting D, B, F, E takes time at least $1.5 + y > 0.5 + 4y$, Lemmas 7, 8, and 9 now assure that $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$ for the cases when R_1, in addition to visiting B and E, visits F, visits D, and visits neither, respectively. ◀

For all the lemmas below, we assume that according to algorithm $\mathcal{A}$, we have $v_5 = E$ and $v_6 = A$, and robot R_1 visits E at time $1 + y \leq t_E < 0.5 + 4y$. We start with a simple observation that is used repeatedly.

▶ **Observation 4.** *Let p be a point on the boundary. If at time $1 + 4y - |Ap|$, both A and p are unvisited then $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$.*

Proof. Put the exit at whichever of the two points is visited later. Since at time $1 + 4y - |Ap|$, neither is visited, the time to evacuate is at least $1 + 4y - |Ap| + |Ap| = 1 + 4y$. ◀

▶ **Lemma 5.** *If R_2 visits B, D and F, and R_1 visits C and E, then $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$.*

Proof. First, observe that if B is not visited *first* of the three points B, D, F, then $t_B > 0.5 + y$. Since E is visited by R_1 and $t_E \geq 1 + y$, by the Meeting Lemma, R_1 and R_2 cannot meet between t_B and t_E. Thus if the exit is at B, it will take R_1 time at least $t_E + h \geq 1 + 4y$ to reach there. We conclude that B must be visited first. If R_2 visits B, D, and F in that order, then $t_F \geq 1 + x$, so by Observation 4, we have $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$. So, R_2 must visit B, F and D in this order.

Let P be the closest point from B on the BD line segment that is not visited by R_2 before it visits F. Then the time for R_2 to reach F is at least $|OP| + |PB| + |BF|$ (Figure 2, the blue trajectory). Therefore, the earliest time R_2 can reach P is $|OP| + |PB| + |BF| + |FP|$. It can be verified that for any point M on the BD line segment, this time is more than $1 + 4y - |AM|$, therefore it is true for the point P defined above. Also, R_1 cannot visit P on time: if it visits P before C (Figure 2, the green trajectory), we have $t_E \geq |OP| + |PC| + |CE| \geq |OD| + |DC| + |CE| = 0.5 + 4y$, and if it visits C before P (Figure 2, the red trajectory), we have $t_E \geq |OC| + |CP| + |PE| \geq |OC| + |CD| + |DE| = 2y + 3y + 0.5$. Thus neither robot can visit P before time $1 + 4y - |AP|$. The lemma now follows from Observation 4. ◀

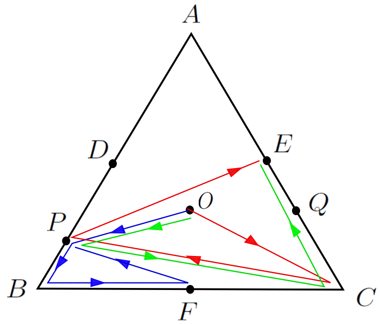

Figure 2 An illustration of possible trajectories of R_1 and R_2, in support of Lemma 5.

▶ **Lemma 6.** *If R_2 visits B and D and R_1 visits C, F, and E, then $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$.*

Proof. First observe that if R_2 visits D before B, then $t_B \geq 0.5 + y$. Since E needs to be visited by the other robot R_1 and $t_E \geq 1 + y$, by the Meeting Lemma, R_1 and R_2 cannot meet between t_B and t_E. Thus if the exit is at B, it will take R_1 time at least $t_E + h \geq 1 + 4y$ to reach there. We conclude that B must be visited first. Also, clearly E is visited last by R_1. If R_1 visits F before C, then $t_C \geq 0.5 + y$, and $t_D \leq t_E \leq 0.5 + 4y$, so by the Meeting Lemma, R_1 and R_2 cannot meet before time $0.5 + 4y$ and after R_2 visits B. Thus if the exit is at B, it will take R_1 time at least $t_E + h \geq 1 + 4y$ to reach there, and we conclude that $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$. Therefore, R_1 must visit C, F, and E in that order. Using a similar argument as in Lemma 5, we can see that there exists an unvisited point P on the CE segment at time $1 + 4y - |AP|$. It follows from Observation 4 that $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$. ◀

▶ **Lemma 7.** *If robot R_1 visits B, F and E, and R_2 visits C and D, then $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$.*

Proof. First observe that E must be visited last, and if R_1 visits F before B then $t_E \geq 0.5 + 4y$. So B must be visited before F. Now let P be a point at distance 0.3 from B on the BD segment, and Q a point at distance 0.34 from C on the CE segment. It can be verified that if R_1 visits a point on the PD segment before arriving at B, then $t_E \geq 0.5 + 4y$. Similarly, R_1 cannot visit any point in the QC line segment if it is to reach E by time $0.5 + 4y$. See Figure 3 (a). Therefore the entire PD line segment and the entire QC line segments must be visited by R_2. Now we consider the order of visiting D, Q, C, P. If D or P are visited before C, then C cannot be reached before $4y$ which means that if the exit is at A, R_2 cannot reach it before time $1 + 4y$. So C has to be visited before P or D, Figure 3 (b). Regardless of whether Q or C is visited first, it can be verified that it is impossible for R_2 to reach P before time $1 + 4y - |AP|$, yielding the desired conclusion, using Observation 4. ◀

▶ **Lemma 8.** *If robot R_1 visits B, D and E, and R_2 visits C and F, then $E_{\mathcal{A}}(T, 2) \geq 1 + 4y$.*

Proof. R_2 must visit F before C, as otherwise as shown in the proof of Lemma 5, there will be a point P on the CE segment that cannot be visited before time $1 + 4y - |AP|$. If R_1 visits D before B, then $t_E \geq 0.5 + 4y$. So R_1 must visit B before D. By Observation 4, the entire BC edge must be visited at or before time $1 + y$. Let Q be the leftmost point on the BC edge that is not visited by robot R_1. Then R_2 must visit the entire QC segment. Since R_2 must visit C before time $4y$, we see that $|BQ| > 0.236$. As a result $t_D \geq 1.13155$. Let R be the point at distance 0.05 from D on the the DA segment, and S be the point at distance

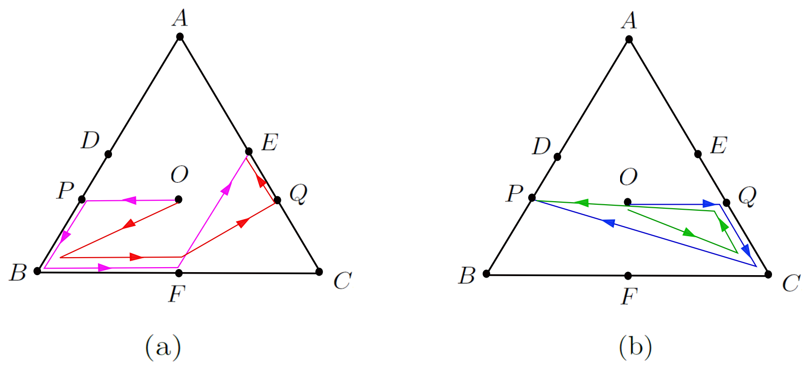

Figure 3 (a) An illustration of possible trajectories of R_1 and (b) possible trajectories of R_2, in support of Lemma 7.

0.03 from E on the EC segment. Using the assumption that it reaches E before time $0.5+4y$, it can be verified that R_1 cannot visit either R or S before time $1.657 > 0.5+4y$, and $1.661 > 0.5+4y$ respectively. Then since the SE segment must be visited by R_2 before time $0.5+4y$, R_2 cannot reach R before time $|OF|+|FC|+|CS|+|SR| > 1.75 > 1+4y-|AR|$. It follows from Observation 4 that $E_{\mathcal{A}}(T,2) \geq 1+4y$. ◀

▶ **Lemma 9.** *If robot R_1 visits B and E, and R_2 visits C, F, and D, then $E_{\mathcal{A}}(T,2) \geq 1+4y$.*

Proof. We observe that D must be visited last; if F is visited last, $t_F \geq 0.5+4y$, and if C is visited last, then $t_C \geq 1+y > 4y$. In both cases, Observation 4 gives the desired result. The rest of the proof is analogous to the case when robot R_1 visits B, F, E. ◀

4 Evacuation Algorithms and Upper Bounds

In this section, we give evacuation algorithms for k robots, $k \geq 2$, that are initially located on the centroid of an equilateral triangle, and derive upper bounds on the evacuation time by analyzing their performance.

Consider a straightforward strategy for evacuating k robots, that we call the *Equal-Exploration* strategy: divide the boundary into k equal-sized contiguous sections of length $3/k$ each, and assign each robot to explore a unique section of the boundary. Each robot goes to one endpoint of its assigned section, it explores its assigned section in time at most $3/k$, then it returns to the centroid to meet the other robots to share the result of its exploration. Since the sections can be chosen so that no section begins at a vertex, each robot takes time less than $2x$ for the total travel to and from the centroid. Thus, all robots are at the centroid at time less than $2x+3/k$. Finally, all robots travel together to the exit taking time at most x. Clearly, this algorithm has evacuation time less than $3x+3/k = \sqrt{3}+3/k$. Although very simple, by Theorem 1, this bound is asymptotically optimal. As such, we have the following:

▶ **Observation 10.** *The Equal-Exploration strategy for k robots has worst case evacuation time less than $\sqrt{3}+3/k$.*

In the rest of this section we propose several improvements of the above strategy and obtain better upper bounds for $k \leq 6$. However, our strategies can be easily used to obtain evacuation algorithms for other values of k.

4.1 Equal-Travel Early-Meeting Algorithms

First notice that the time to travel from and to the centroid can vary by almost a factor of 2 for different robots. In particular, robots that are assigned a section of the boundary starting or ending close to a midpoint have to travel a much smaller distance to or from the centroid, than robots that are assigned a section of the boundary that starts/ends close to a vertex. In the worst case the exit could be discovered by a robot that is the last to arrive at the centroid. Thus, our second strategy, which we call the *Equal-Travel* strategy, divides the boundary into sections in such a way that the total lengths of trajectories of all robots is equalized. In other words, we aim to equalize the *travel time* of all robots. Clearly, the Equal-Travel strategy should in general lower the evacuation time compared to the Equal-Exploration strategy. For a general value of k, such a division into equal length trajectories requires a solution of a system of equations of order 4, without a significant improvement for large values of k, and thus we did not do it in general.

Second notice that in both the Equal-Exploration and Equal-Travel strategies, the robots meet in the interior of the triangle *after* the entire boundary has been explored, and then travel together to the exit. In the following we show that the evacuation time of a triangle can be improved if the robots stop the exploration of the boundary *early* and go to a meeting point before the boundary is explored entirely. After this *early meeting*, either the robots go together to the exit, or they all go together to explore the rest of the boundary where the exit is now known to be.

In the sequel, we describe Equal-Travel Early-Meeting algorithms that combine the equal travel strategy with an early meeting. Such an algorithm for the evacuation of k robots selects the location of the early meeting point. It divides the boundary into $k+1$ sections, of which k require the same travel times, and are assigned to unique robots, and the last section is a *common section*, to be explored together by all robots. In the first phase each robot explores its assigned section (the last section remains unexplored) and returns to the meeting point. If the exit has been found by one of them, all robots proceed to the exit. If the exit has not been found, it must lie in the unexplored last section. The robots go together to explore the common last section and evacuate together. By optimizing the position of the meeting point, the location of the sections, and the length of the common section, we obtain Early-Meeting algorithms with better performance than those using only the Equal-Travel strategy. Next we give the details of the Early-Meeting algorithm for each $k \in \{2, 3, 4, 5, 6\}$.

▶ **Theorem 11.** *There are Equal-Travel Early-Meeting algorithms for two and three robots with evacuation times $E^*(T, 2) \leq 2.4113$, and $E^*(T, 3) \leq 2.08872$.*

Proof. First we consider the case of 2 robots. Let p be the point in the middle of side BC, and r_1, r_2 be points on sides AB, AC to be determined later, and the meeting point D be the bisector of segment r_1r_2 as on Figure 4(a). R_1 is assigned the section of the boundary from p to B to r_1, and R_2 is assigned the section from p to C to r_2, the rest of the boundary being the final common section. Now, if the exit is discovered by R_2 then R_1 travels the distance at most $t_1 = y + 0.5 + |Br_1| + |r_1D| + |DC|$. If the exit is discovered in the common section, it travels distance at most $t_2 = y + 0.5 + |Br_1| + 2|r_1D| + 2|Ar_1|$. The worst-case evacuation time is $max(t_1, t_2)$. Solving the equation $t_1 = t_2$, we obtain $|Br_1| = 0.68868$ and the evacuation time is at most 2.4113.

Next, we consider the case of 3 robots. Consider the following instance of the Early-Meeting strategy. Place points p_1, r_1, r_2 on sides BC, BA, AC as shown in Figure 4(b), the exact locations to be determined later. Centroid O is designated as the meeting point of the robots. Point p_2 is placed so that the distance $|Op_1| + |p_1p_2| = x$, the line segment p_1p_2

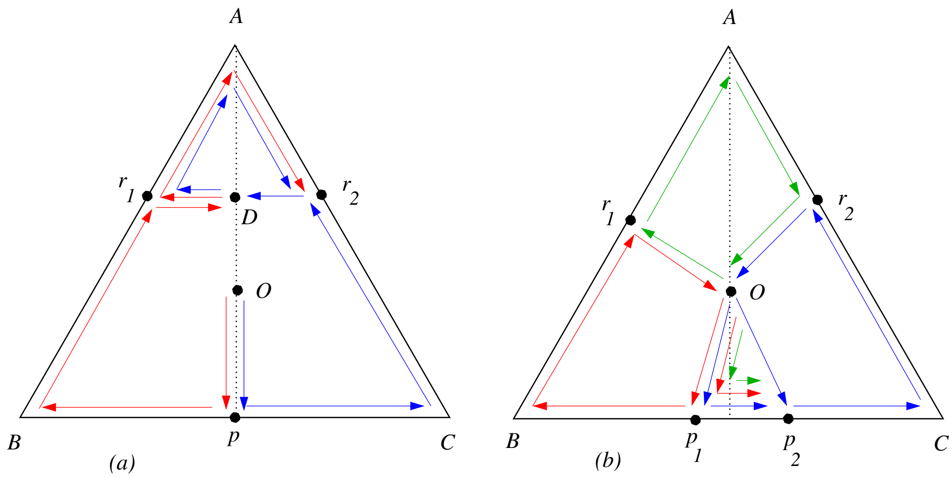

Figure 4 Equal-Travel Early-Meeting trajectories for (a) two robots, and (b) three robots.

being designated as the common section. R_1 is assigned the section of the boundary from p_1 to B and to r_1, R_2 the section from r_1 to A to r_2, and R_3 the section from r_2 to C to p_2. If one of the robots discovers the exit in its section, the other robots need to travel to it from the meeting point at the centroid, which adds distance at most x. In this setup the maximum distance robots travel are

$$t_1 = |Op_1| + |p_1B| + |Br_1| + |r_1O| + x \text{ for } R_1,$$
$$t_2 = |Op_2| + |p_2C| + |Cr_2| + |r_2O| + x \text{ for } R_2, \text{and}$$
$$t_3 = |Or_1| + |r_1A| + |Ar_2| + |r_2O| + x \text{ for } R_3.$$

Solving the set of equations $t_1 = t_2$, $t_2 = t_3$ and then optimizing the position of p_1, we get that the optimal placement of points is $|Bp_1| = .4545932809$, $|Br_1| = .4747719935$ and $|Cr_2| = .5454067191$, and the evacuation time of the algorithm at most 2.08872. All equations were solved and optimizations done using the Maplesoft [23]. ◀

It is easy to see that for $k = 2$ and $k = 3$ the Equal-Travel algorithms without any early meeting have evacuation times $y + 2.5 \approx 2.788$ and $2y + 1 + x \approx 2.155$, respectively.

We remark here that the Equal-Travel Early-Meeting algorithm for two robots given above can be further improved by additional optimization of the position of the meeting point D. We have not done it because in Section 4.2 we show a different strategy, applicable only to two robots, which gives an evacuation time that is better than that obtained with the optimized Early-Meeting strategy.

Clearly the approach used in Theorem 11 can be generalized to any $k > 2$. Start with selecting a position p_1 for the beginning of the common section and partition arbitrarily the boundary by placing points $r_1, r_2, \ldots, r_{k-1}$, see Figure 5 for $k = 4$ and 5. Point p_2 is placed so that the distance $|Op_1| + |p_1p_2| = x$. Similarly as above, we obtain a system of $k-1$ equations for the values of $r_1, r_2, \ldots, r_{k-1}$ that produce equal travel time for all robots in the first phase. Finally, by optimizing the value of p_1 we get the final assignment of the trajectories of robots. In this manner we obtained the following theorem.

▶ **Theorem 12.** *There are Equal-Travel Early-Meeting algorithms for $k = 4$, 5, and 6 robots with $E^*(T, 4) \leq 1.9816$. $E^*(T, 5) \leq 1.876$, $E^*(T, 6) \leq 1.8263$, respectively.*

Proof. Similar to the proof of Theorem 11 and thus omitted. ◀

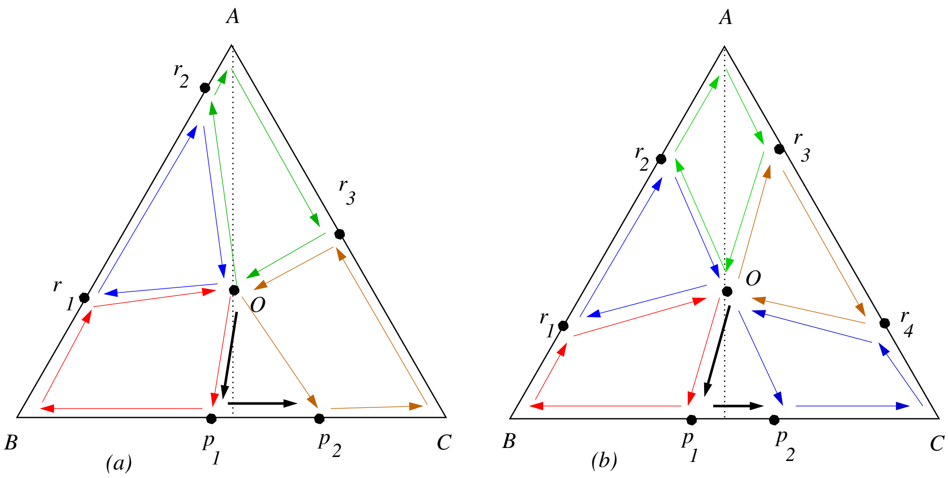

Figure 5 Establishing Equal-Travel Early-Meeting Trajectories of (a) four robots, and (b) five robots. The common part of trajectories of the robots is shown as thick black line.

As one could expect, the improvement in the evacuation time obtained by adding an early meeting to evacuation algorithms diminishes with increasing k. For 6 robots there is an Equal-Travel algorithm with the evacuation time of 1.8411, which is only 0.065 more than that of the algorithm from Theorem 12.

4.2 Equal-Travel with Detour Algorithms for Two Robots

If the evacuation problem is limited to two robots, the Equal-Travel Early-Meeting strategy can be improved further by using an *Equal-Travel with Detour* strategy in which an early meeting of two robots is replaced by a *detour*. This gives a further improvement on the evacuation time. The idea of using a detour was originally mentioned in the context of evacuating a disk with two robots in [8]. In this paper we show that a detour can also improve the evacuation time in an equilateral triangle, and multiple detours can improve it further.

As in the Equal-Travel Early-Meeting strategy, trajectories of both robots have the same length. The trajectory for each robot consists of multiple sections of the boundary, separated by detours. Each robot starts with the exploration of a section of the boundary. At some point, the robot leaves the boundary and makes a detour inside the triangle (see Figure 6 for an example). If it is not intercepted by the other robot during the detour, the robot concludes that the exit was not found by the other robot in its first section, and goes back to the boundary to explore its second section of the boundary. After the robot has finished exploring its second section of the boundary, it can make another detour, and so on. It is critical that the detours are designed so that if one of the robots finds the exit in its section, it has enough time to intercept the other robot during the corresponding detour of the other robot. We point out the salient features of the detour strategy, and its differences from the early meeting strategy:

- There are no early meeting points; the trajectories of the robots do not intersect except for the initial part to get to the boundary of the triangle, and at the very end. The detour part of trajectories of robots inside the triangle only get close to each other. This makes trajectories of the robots shorter.
- Each robot is assigned more than one section of the boundary.

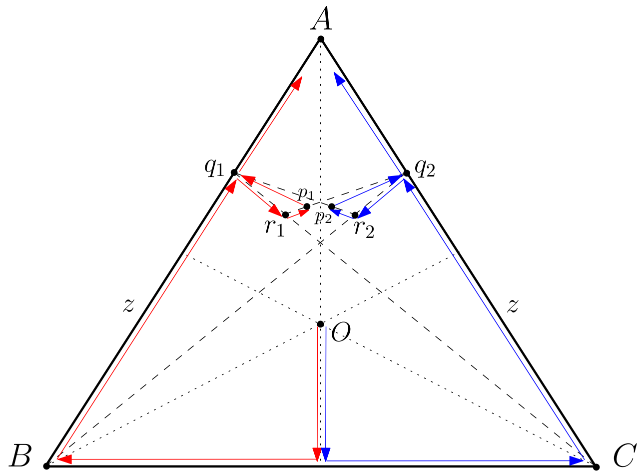

Figure 6 The trajectories of two robots with one detour for each robot.

- There is no common section of the boundary to be explored by both robots.
- A robot can do multiple detours.

We first give the details of an Equal Travel with Detour algorithm with a single detour.

▶ **Theorem 13.** *There is an Equal-Travel with Detour evacuation algorithm for two robots, using a single detour, that has evacuation time* ≤ 2.3838*.*

Proof. The Equal-Travel with Detour trajectories of the robots are shown in Figure 6. Points q_1 and q_2 are located symmetrically on the sides AB and AC at distance $z > 0.5$ (the exact value to be determined later) from B and C, respectively. Since the trajectories of robots are symmetric, we specify below the trajectory of R_2 only. R_2 is assigned the section from the midpoint of BC to C and from C to q_2. It starts a detour at point q_2 where it goes in the direction of point B. Point r_2 is chosen to be on the line segment q_2B so that $z + |q_2r_2| = |r_2B|$.
From r_2 the detour of R_2 goes in the direction of point q_1 until point p_2 which is chosen so that $|q_2r_2| + |r_2p_2| = |p_2q_1|$. The detour part of the trajectory of R_2 terminates with the line segment p_2q_2, where its trajectory continues with the section of the boundary from q_2 to A. Let $t_1 = y + 0.5 + z + |q_2B|$, and $t_2 = y + 0.5 + z + |q_2r_2| + |r_2p_2| + |p_2q_2| + 2(1 - z)$. We argue below that the worst case evacuation time of this algorithm is $max(t_1, t_2)$. We consider all possible locations for the exit on the sections of the boundary explored by R_1 (the case when the exit is discovered by R_2 is symmetric).

Clearly, if the exit is located on the line segment q_1A, then the evacuation time is at most t_2. If the exit is found by R_1 on the side BC, then it can intercept R_2 at point r_2 since $z + |q_2r_2| = |r_2B|$, and the robots reach the exit in time at most t_1. Assume now that R_1 finds the exit on the line segment q_1A at D, see Figure 7. Consider the triangle with sides y_1, y_2, y_3 obtained by drawing a line though point D parallel with the line going through q_1 and r_2. Since $z > 0.5$, it is easy to see that $|q_1r_2| < |Bq_1| < |Br_2|$ and by the similarity of triangles $y_3 < y_1 < y_2$. Let D' be the intersection point of the line going through q_1 and r_2 with the line drawn through D and parallel with Br_2. Since $y_1 + |DD'| < y_2 + |DD'| = |Br_2|$, when R_2 reaches r_2 robot R_1 is at D' to intercept R_2, and the distance traveled by R_2 from r_2 to D is $y_3 + |DD'| < |Br_2|$, and so the total time to reach D is less then t_1. Notice that if the value of y_1 is so that the parallel line through D would not intersect the line segment r_2p_2 then R_1 can intercept R_2 at p_2 and the the evacuation time remains less than t_1.

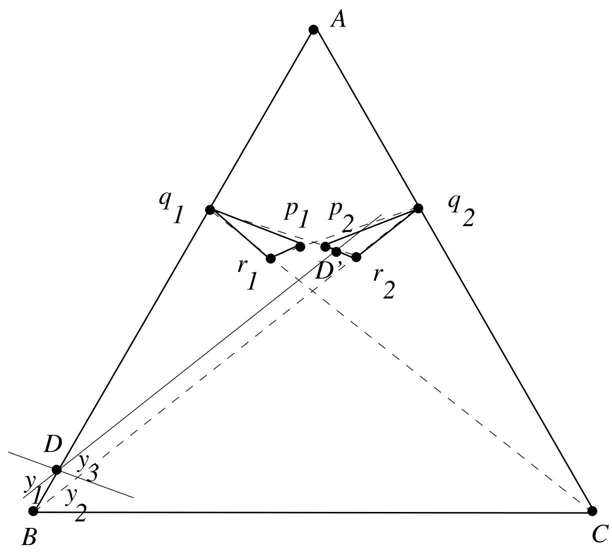

Figure 7 Evacuation time for exit at D is less that that for B.

We have shown that the worst-case evacuation time is $max(t_1, t_2)$. By equating $t_1 = t_2$, we obtain $z = 0.7151$ and $t_1 = 2.3837$. Thus the worst-case evacuation time of this algorithm is 2.3837. ◀

Consider the Equal-Travel with Detour algorithm described above and suppose that the robots do not find the exit in the first phase. That is, the robots will go back to points q_1 and q_2 to explore the rest of the boundary of T (i.e., the line segments q_1A and q_2A). Observe that for triangle $\triangle Aq_1q_2$, at this time the robots are in the same situation as they were when visiting vertices B and C for the triangle $\triangle ABC$. Furthermore, like for the triangle $\triangle ABC$, where the worst case occurs in the neighbourhood of B and C, the worst case for triangle $\triangle Aq_1q_2$ occurs when the exit is located in the neighbourhood of q_1 or q_2. Therefore, we can use an additional detour in the triangle Aq_1q_2 and improve the evacuation time in the top part as illustrated in Figure 8. However, to improve the evacuation time for the whole triangle, we need to re-balance all of the worst case evacuation times in $\triangle ABC$. In particular, we derive the expression t_1 for the maximum evacuation time if the exit is discovered by R_1 before reaching q_1, the expression t_2 for the maximum evacuation time if the exit is in the line segment q_1q_1', and the expression t_2 for the maximum evacuation time if the exit is in the line segment $q_1'A$. By solving the equations $t_1 = t_2, t_2 = t_3$ we calculated the optimized positions of q_1, q_2, q_1', q_2' using numerical calculations, obtaining $|Bq_1| = 0.666$, $Bq_1' = 0.9023$ and the evacuation time 2.3367. Thus we have the following improved evacuation time.

▶ **Theorem 14.** *There is an Equal-Travel with Detour evacuation algorithm for $k = 2$ robots that uses two detours with the evacuation time at most* 2.3367*.*

We remark that the use of an additional detour as above can be applied any number of times in the upper part of the triangle. With each additional detour in the upper part of the triangle we have one more case of maximal evacuation time and one more equation to add to the system of equations to be solved. However, with each additional detour the upper part of the triangle becomes much smaller, and thus the improvement in the evacuation time is extremely tiny for more than two detours.

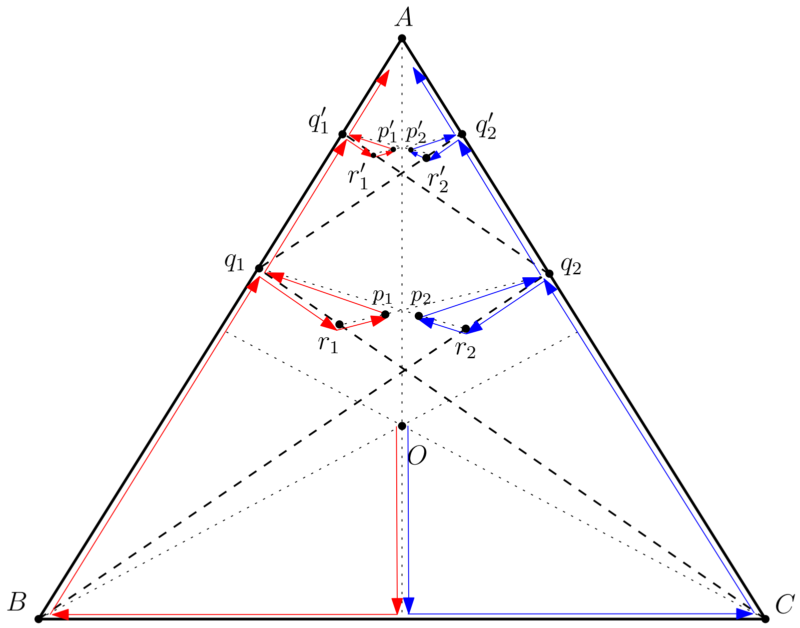

Figure 8 Trajectories of two robots with two detours.

5 Discussion

We studied the evacuation of an equilateral triangle by k robots, initially located at its centroid. The robots can communicate only if they are in the same position at the same time, i.e., they use the face-to-face communication. We showed a lower bound of $\sqrt{3}$ on the evacuation time for any number k of robots, and gave a simple strategy that achieves this bound asymptotically. We introduce the Equal-Travel Early-Meeting strategy for evacuation algorithms in which the robots meet at an early meeting point inside the triangle before the whole perimeter is examined. This strategy gave us upper bounds of 2.08872, 1.9816, 1.876 and 1.827 for $k = 3, 4, 5$, and 6 robots, respectively.

For $k = 2$ robots, we proved a lower bound of $1 + 2/\sqrt{3} \approx 2.154$. We then show that for $k = 2$ the Early-Meeting strategy can be improved by replacing the early-meeting by shorter detours in the interior of the triangle, and obtained an upper bound of 2.3367 on the evacuation time. This upper bound is achieved with two detours.

Although we limited our study to the evacuation of the equilateral triangle, the algorithmic strategies used in this paper should be applicable to other search domains. Finding tight bounds for the evacuation time in the face-to-face model remains an open problem. A clear understanding of the search domains in which early meetings or detours are *provably* useful also remains elusive.

References

1. R. Ahlswede and I. Wegener. *Search problems.* Wiley-Interscience, 1987.
2. R. A. Baeza-Yates, J. C. Culberson, and G. J. E. Rawlins. Searching with uncertainty (extended abstract). In *Proceedings of the 1st Scandinavian Workshop on Algorithm Theory (SWAT 88)*, pages 176–189, 1988.
3. R. A. Baeza-Yates and R. Schott. Parallel searching in the plane. *Computational Geometry*, 5:143–154, 1995.

4 A. Beck. On the linear search problem. *Israel Journal of Mathematics*, 2(4):221–228, 1964.

5 A. Bonato and R. Nowakowski. *The Game of Cops and Robbers on Graphs.* American Mathematical Society, 2011.

6 S. Brandt, K.-T. Forster, B. Richner, and R. Wattenhofer. Wireless evacuation on m rays with k searchers. In *Proceedings of SIROCCO 2017, to appear*, 2017.

7 M. Chrobak, L. Gasieniec, T. Gorry, and R. Martin. Group search on the line. In *Proceedings of SOFSEM 2015: 41st International Conference on Current Trends in Theory and Practice of Computer Science*, pages 164–176, 2015.

8 J. Czyzowicz, L. Gasieniec, T. Gorry, E. Kranakis, R. Martin, and D. Pajak. Evacuating robots via unknown exit in a disk. In *Proceedings of the 28th International Symposium on Distributed Computing (DISC 2014), Austin, TX, USA*, pages 122–136, 2014.

9 J. Czyzowicz, K. Georgiou, M. Godon, E. Kranakis, D. Krizanc, W. Rytter, and M. Wlodarczyk. Evacuation from a disc in the presence of a faulty robot. In *Proceedings of SIROCCO 2017, to appear*, 2017.

10 J. Czyzowicz, K. Georgiou, E. Kranakis, L. Narayanan, J. Opatrny, and B. Vogtenhuber. Evacuating robots from a disk using face-to-face communication (extended abstract). In *Proceedings of the 9th International Conference on Algorithms and Complexity (CIAC 2015), Paris, France*, pages 140–152, 2015.

11 J. Czyzowicz, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, and S. M. Shende. Wireless autonomous robot evacuation from equilateral triangles and squares. In *Proceedings of the 14th International Conference on Ad-hoc, Mobile, and Wireless Networks (ADHOC-NOW 2015), Athens, Greece*, pages 181–194, 2015.

12 Y. Emek, T. Langner, J. Uitto, and R. Wattenhofer. Solving the ANTS problem with asynchronous finite state machines. In *Proceedings of the 41st International Colloquium on Automata, Languages, and Programming, (ICALP 2014), Part II*, pages 471–482, 2014.

13 O. Feinerman and A. Korman. Theoretical distributed computing meets biology: A review. In *Proceedings of the 9th International Conference on Distributed Computing and Internet Technology, (ICDCIT 2013), Bhubaneswar, India*, pages 1–18, 2013.

14 O. Feinerman, A. Korman, Z. Lotker, and J.-S. Sereni. Collaborative search on the plane without communication. In *ACM Symposium on Principles of Distributed Computing (PODC 2012), Funchal, Madeira, Portugal*, pages 77–86, 2012.

15 P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer. Gathering of asynchronous robots with limited visibility. *Theor. Comput. Sci.*, 337(1-3):147–168, 2005.

16 S. K. Ghosh and R. Klein. Online algorithms for searching and exploration in the plane. *Computer Science Review*, 4(4):189–201, 2010.

17 L. Hua and E. K. P. Chong. Search on lines and graphs. In *Proceedings of the 48th IEEE Conference on Decision and Control, CDC 2009, China*, pages 5780–5785, 2009.

18 A. Jez and J. Lopuszanski. On the two-dimensional cow search problem. *Information Processing Letters*, 109(11):543–547, 2009.

19 C. Lenzen, N. A. Lynch, C. C. Newport, and T. Radeva. Trade-offs between selection complexity and performance when searching the plane without communication. In *ACM Symposium on Principles of Distributed Comp. (PODC 2014),*, pages 252–261, 2014.

20 A. López-Ortiz and G. Sweet. Parallel searching on a lattice. In *Proceedings of the 13th Canadian Conference on Computational Geometry (CCCG 2001)*, pages 125–128, 2001.

21 P. Nahin. *Chases and Escapes: The Mathematics of Pursuit and Evasion.* Princeton University Press, 2012.

22 L. D. Stone. *Theory of Optimal Search.* Academic Press New York, 1975.

23 I. Thompson. *Understanding Maple.* Cambridge University Press, 2016.