Unlabeled Multi-Robot Motion Planning with Improved Separation Trade-Offs

Tsuri Farhana
The Stein Faculty of Computer and Information Science, Ben-Gurion University of the Negev, Beer Sheva, Israel

Omrit Filtser
Department of Mathematics and Computer Science, The Open University of Israel, Ra'anana, Israel

Shalev Goldshtein
Department of Mathematics and Computer Science, The Open University of Israel, Ra'anana, Israel

Abstract

We study unlabeled multi-robot motion planning for unit-disk robots in a polygonal environment. Although the problem is hard in general, polynomial-time solutions exist under appropriate separation assumptions on start and target positions. Solovey et al. (RSS'15) provide a near-optimal solution assuming that start/target positions must have pairwise distance at least 4, and at least $\sqrt{5} \approx 2.236$ from obstacles. This raises the question of whether polynomial-time algorithms can be obtained in even more densely packed environments.

In this paper we present a generalized algorithm that achieve different trade-offs on the robots-separation and obstacles-separation bounds, all significantly improving upon the state of the art. Specifically, we obtain polynomial-time constant-approximation algorithms to minimize the total path length when (i) the robots-separation is $2\frac{2}{3}$ and the obstacles-separation is $1\frac{2}{3}$, or (ii) the robots-separation is ≈ 3.291 and the obstacles-separation ≈ 1.354. Additionally, we introduce a different strategy yielding a polynomial-time solution when the robots-separation is only 2, and the obstacles-separation is 3. Finally, we show that without any robots-separation assumption, obstacles-separation of at least 1.5 may be necessary for a solution to exist.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases multi-robot motion planning

Digital Object Identifier 10.4230/LIPIcs.SoCG.2026.43

Related Version *Full Version*: `https://arxiv.org/abs/2603.19502` [10]

Funding *Omrit Filtser*: This research was supported by the ISRAEL SCIENCE FOUNDATION (grant No. 2135/24).
Shalev Goldshtein: This research was supported by the ISRAEL SCIENCE FOUNDATION (grant No. 2135/24).

Acknowledgements We thank Dan Halperin for introducing us to the problem and for helpful discussions. We are also grateful to the anonymous reviewers for their insightful comments, which helped improve the structure and presentation of the paper.

1 Introduction

In the classic multi-robot motion-planning (MRMP) problem, a set of robots operate in a common workspace, and need to reach a set of target positions. More precisely, given a set of m starting positions and a set of m target positions, the goal is to compute a motion strategy for the robots to reach the targets while avoiding collisions with the boundary of the workspace and with other robots. In this paper we consider unit-disk robots that are moving in a polygonal workspace in the plane. If each robot is assigned to a specific target, the problem is referred to as *labeled* MRMP, and otherwise it is *unlabeled* MRMP.

The MRMP problem and its variants have been widely investigated, both from theoretical and practical perspectives (see, e.g., [3, 4, 5, 6, 9, 11] for some of the more recent works on the problem). Very recently, both the labeled and unlabeled variants of MRMP for unit disks in polygons with holes were shown to be PSPACE-hard [1]. Many other variants of the problem are known to be hard as well, see e.g. [8, 12, 14, 16].

Interestingly, the separation between the robots plays a key role in the difficulty of the problem. Adler, de Berg, Halperin, and Solovey [2] show that in a simple polygonal workspace with n vertices and m unit-disk robots, when the ***robots-separation***, denoted ρ, is 4 (i.e., all the start and target positions are at least distance 4 apart), a solution for unlabeled MRMP always exists, and can be found in $\tilde{O}(mn + m^2)$ time. In [7] this result is further improved by distinguishing between ***monochromatic*** robots-separation (i.e. the distance between two start or two target positions), and ***bichromatic*** robots-separation (i.e. the distance between a start position and a target position). The algorithmic approach in both papers [7, 2] is similar and does not provide any reasonable approximation on the length of the solution, i.e., the total length of the paths in the motion plan.

Solovey, Yu, Zamir, and Halperin [15] present a different approach for unlabeled MRMP, with optimality guarantee even in polygons with holes, but at the cost of adding an additional assumption on the distance between the obstacles to any start and target position. Specifically, they consider the case when the robots-separation is $\rho = 4$, and the ***obstacles-separation***, denoted ω, is $\sqrt{5} \approx 2.236$. Under these assumptions, they present an algorithm that runs in $\tilde{O}(m^4 + n^2m^2)$ time and returns a solution of total length at most $\mathrm{OPT} + 4m$. Here and throughout the paper, OPT denotes the minimum possible sum of lengths of the paths in a collision-free motion plan.

Notice that when assuming a large robots-separation ($\rho \geq 4$), the overall "density" of the robots inside the workspace becomes small, which may require the workspace to be much larger than the number of robots operating in it. Also, assuming a large obstacles-separation requires adding an additional buffering "layer" around the obstacles. This motivates the question of whether we can find weaker assumptions on either the robots-separation or the obstacles-separation, that are sufficient for having a polynomial time algorithm for unlabeled MRMP. Surprisingly, in this paper we show that both the robots-separation and obstacles-separation assumptions can be significantly relaxed, while still enabling a polynomial-time algorithm that achieves a constant-factor approximation of the solution length.

In previous works, namely [2, 7, 15], the algorithmic approach uses a ***monotone*** strategy, where robots move to their target position one by one according to some order, while other robots must remain at their start/target positions. The main result in our paper uses a weakly-monotone strategy. In a ***weakly-monotone*** motion plan, each start/target position p is associated with a revolving area - a disk of radius $r > 1$ that contains the unit disk centered at p and is empty of obstacles and other unit disks centered at other start/target positions. The robots move to their target position one by one according to some order, while other robots must remain inside the revolving area associated with their current position. Such a weakly-monotone strategy was used, e.g., in [4, 13] for a variant of labeled MRMP with unit-disk robots.

Our contribution. In this paper we present two different algorithms for unlabeled MRMP, that provide tighter trade-offs between robot-separation and obstacles-separation assumptions. In particular, for a parameter $0 \leq \varepsilon \leq 1$, a polygonal environment of complexity n, and m unit-disc robots, our first algorithm computes in $\tilde{O}(m^4 + n^2m^2)$ time a collision-free motion plan of total length $O(\mathrm{OPT})$, assuming that $\rho = \max\{4 - 2\varepsilon, 2 + \varepsilon\}$ and $\omega =$

$\max\{1+\varepsilon, \sqrt{(3-\sqrt{3}-\varepsilon)^2+1}\}$ (see Theorem 9). In Table 1 we highlight three interesting values of ε that minimizes either ρ, ω or the guaranteed length of the solution. This algorithm can be viewed as a generalization of the algorithm in [15], and indeed, as a consequence of our analysis we positively resolve a conjecture made in [15], by showing that $\omega = 1.614$ is sufficient when $\rho = 4$. Moreover, in the full version of our paper [10] we show that a monotone motion plan (such as the one produced by the algorithm in [15]) may not always exist when $\omega < 1.614$, and therefore to achieve smaller separation bounds one must use a different strategy. Indeed, as mentioned above, our motion plan is weakly-monotone. Interestingly, we also show that a weakly-monotone motion plan may not always exist when $\omega < 1.354$.

Note that our first algorithm does not cover the case of $\rho = 2$, which is the optimal (monochromatic) robots-separation. We therefore present a second algorithm using a completely different approach, which given a simple polygon with n vertices and m unit-disk robots, computes in $O(m^3 + mn^2)$ time a collision-free motion plan of total length $\mathrm{OPT} + O(m^2)$, assuming that $\rho = 2$ and $\omega = 3$ (see Theorem 10). Finally, we show that $\omega = 1.5$ may be necessary for a solution to exist.

Table 1 Summary of algorithms and their separation assumptions.

ω	ρ	Solution length	Running time	Notes	Reference
1	4	no guarantee	$\tilde{O}(mn+m^2)$	Simple polygons	[2, 7]
$\sqrt{5}$	4	$OPT + 4m$	$\tilde{O}(m^4+n^2m^2)$		[15]
≈ 1.614	4	$OPT + 4m$	$\tilde{O}(m^4+n^2m^2)$		Theorem 9
≈ 1.354	≈ 3.291	$O(OPT)$	$\tilde{O}(m^4+n^2m^2)$		Theorem 9
$1\frac{2}{3}$	$2\frac{2}{3}$	$O(OPT)$	$\tilde{O}(m^4+n^2m^2)$		Theorem 9
3	2	$OPT + O(m^2)$	$\tilde{O}(m^3+mn^2)$	Simple polygons	Theorem 10

Technical ideas and overview of the paper

Our approach in the first algorithm generalizes the strategy of Solovey et al. [15] for unlabeled MRMP, which is briefly described as follows. Given a set S of start positions, a set T of target positions, and a polygonal workspace $\mathcal{W}$, first compute an optimal set Γ of paths assigning each starting point to a target point (a perfect matching). This set of paths minimizes the sum of geodesic distances, when ignoring the other robots in the workspace. Then, one can show that when assuming $\rho = 4$ and $\omega = \sqrt{5}$, there is always at least one target which is a so-called standalone goal, that is, a target t that does not block any other path in Γ. Such a target is useful because we can place a robot there and then treat it as an obstacle. However, there might be some other robot that blocks the path that leads to t, in which case we can make a switch: we push the last robot that blocks this path to t. This switching process is possible because of the assumption $\omega = \sqrt{5}$. Then, run the algorithm recursively, but after removing the start and target positions that where satisfied in the previous recursive step, and updating the workspace to treat the robot at t as an obstacle.

Our weakly-monotone motion plan is obtained by inserting a very important modification to the algorithm of [15]: instead of choosing a standalone target in each iteration, we relax the requirement and require only an “almost standalone” target – a target that may interrupt other paths, but only by a small amount. In our strategy, we distinguish between "blocking" and "interrupting" positions. Instead of always switching paths when another robot is standing in the way, we allow robots that interrupt a path by only a small amount to move away from the path. A robot that interrupts a given path may move slightly in its close neighborhood (a disk of radius equal to the amount of interruption plus one) in order to clear the way. In contrast, a robot that blocks a path may not have enough wiggle room, and we will switch the paths.

As suggested above, this combined strategy can be viewed as a generalization of [15]: we introduce another parameter, called the "overlap parameter", which is used to determine whether a position is blocking a path or only interrupting it. This parameter allows us to define a set of constraints on ρ and ω, that allows (i) the existence of an interrupting target, (ii) the option to switch paths in the case when a robot is blocking a path, and (iii) the existence of a sufficiently large free neighborhood for an interrupting robot to clear the way. It turns out that there is an interesting trade-off between these constraints, leading to the main result of our paper (Theorem 9).

Note that the concept of interrupting positions raises some issues that were not present in the algorithm of [15]. For example, in each iteration of the recursive algorithm we need to take into account that the robot placed on a target may still interrupt other paths. In [15] this is solved simply by removing a unit disk around the interrupting target from the workspace. However, in our case, such a robot may still need to move in order to clear the way for other robots. Moreover, if we simply treat it as an obstacle, the assignment of paths in the next iteration may be too large. We solve this issue by observing that it is actually enough to remove a smaller disk around a settled target when computing the paths, while still allowing it to move within its small neighborhood when planning the motion in the following iterations. Another issue is that there may be two or more robots that interrupt a path and may need to move simultaneously. Here we describe a simple motion with respect to the ray connecting the robot moving on the path with the interrupting robots.

In Section 3, we systematically define and analyze the constraints that are required for each part of this strategy, each constraint depends on the overlap parameter. In Section 4 we present the complete generalized algorithm and prove its correctness. Then, we show how to choose an overlap parameter and adjust constraints to get the minimum possible obstacles-separation and the minimum possible robots-separation.

In our first algorithm, to clear a path, we only move robots that intersect that path. Such a strategy cannot work when the robots-separation is exactly 2, because a robot that intersect a path may be block by other robots that do not intersect that path, and thus will not be able to move and clear the path. Therefore, to achieve a separation bound of 2 between the robots, we present in Section 5 a completely different strategy, which we call the Exodus algorithm. In this strategy, we also iteratively pick a path between start and target positions that were not yet satisfied, and move a robot along that path, however, in contrast to the first algorithm, we clear the path completely by also moving robots that do not intersect the path. In simple polygons, this strategy allows us to assume robots-separation of only 2, with obstacles-separation of 3, and it applies also for labeled MRMP (see Theorem 10).

Note that the monochromatic separation must be at least 2, so the robots-separation is almost optimal (the bichromatic separation can be 0). In Section 6 we preset lower bounds on the obstacles-separation, and we leave open the question of what is the upper bound on ω when robots-separation is optimal.

2 Preliminaries

We consider a set of m unit-disk robots, moving in a polygonal ***workspace*** $\mathcal{W} \subseteq \mathbb{R}^2$ (which may be simple or cluttered with polygonal obstacles), whose overall number of edges is n. Below, we mostly follow the notations in [7] and [15].

The ***obstacle space*** $\mathcal{O}$ is the complement of the workspace $\mathcal{W}$, that is, $\mathcal{O} = \mathbb{R}^2 \setminus \mathcal{W}$. We refer to the points in $\mathcal{W}$ as positions, and say that a robot is at position $x \in \mathcal{W}$ when its center is positioned at the point x. For a point $x \in \mathbb{R}^2$ and a radius $r \in \mathbb{R}_+$, we define $D_r(x)$ to be the open disk of radius r centered at x. The unit-disk robots are defined to be open sets, and therefore two robots collide if and only if the distance between their positions is strictly less than 2. In addition, a robot collides with the obstacle space $\mathcal{O}$ if and only if its center is at distance strictly less than 1 from $\mathcal{O}$. The set of all positions in $\mathcal{W}$ where a unit-disk robot does not collide with the obstacle space is called the ***free space***, denoted $\mathcal{F} = \{x \in \mathbb{R}^2 \mid D_1(x) \cap \mathcal{O} = \emptyset\}$. Note that the free space is a closed set, and denote by $\partial\mathcal{F}$ the boundary of $\mathcal{F}$.

In the multi-robot motion-planning problem, we are given a workspace $\mathcal{W}$ with n edges and two sets $S = \{s_1, \ldots, s_m\}$ and $T = \{t_1, \ldots, t_m\}$ of points in the free space $\mathcal{F}$. The goal is to plan a collision-free motion for m robots, each positioned on a different starting point in S, such that by the end of the motion every target position in T is occupied by some robot. When the robots are distinguishable (i.e. labeled), the robot that starts at position s_i is required to end up at the target t_i. Otherwise, when the robots are indistinguishable (i.e., unlabeled), it does not matter which robot ends up at which target position, as long as all targets are occupied at the end of the motion.

Formally, we wish to find a set Γ of continuous paths $\gamma_1, \ldots, \gamma_m$, where $\gamma_i : [0, 1] \to \mathcal{F}$ for every $1 \le i \le m$, and such that $\bigcup_{i=1}^{m} \gamma_i(0) = S$ and $\bigcup_{i=1}^{m} \gamma_i(1) = T$. In addition, the paths need to be collision-free, i.e., at any point in time $t \in [0, 1]$, there are no i, j such that $\gamma_i(t)$ and $\gamma_j(t)$ are at distance strictly less than 2. Denote the size of a solution Γ by $|\Gamma| = \sum_{\gamma_i \in \Gamma} |\gamma_i|$, where $|\gamma|$ is the length of a path γ in the L_2 norm. In the optimization version of the problem, we are interested in finding a solution Γ which minimizes $|\Gamma|$, i.e., minimizing the sum of lengths of the paths.

Separation assumptions. We denote by ω the ***obstacles-separation*** assumption on an instance of MRMP, i.e., we assume that for every $p \in S \cup T$ and every point $x \in \mathcal{O}$, it holds that $\|x - p\| \ge \omega$. We denote by ρ the ***robots-separation*** assumption, i.e., we assume that for every $p_1, p_2 \in S \cup T$ it holds that $\|p_1 - p_2\| \ge \rho$.

Overlapping, blocking and interrupting positions

Let $p, q \in \mathcal{F}$ be two points in the free space. We define the ***overlap*** of p and q to be $\max\{0, 2 - \|p - q\|\}$ (see Figure 1). If $\mathcal{D}_1(p) \cap \mathcal{D}_1(q) = \emptyset$, then their overlap is 0. Otherwise, we say that p, q are ***ε-overlapping*** for $\varepsilon = 2 - \|p - q\|$, and note that $\|p - q\| = 2 - \varepsilon$.

We say that a position $p \in \mathcal{F}$ is ***ε-blocking*** a path γ if there exists $0 \le t \le 1$ such that the overlap of p and $\gamma(t)$ is **strictly larger than ε**. Blocking robots (i.e. robots that are placed on blocking positions) may be impossible to clear from the path, and we will have to make a "switch". We say that a point $p \in \mathcal{F}$ is ***ε-interrupting*** a path γ if for every $0 \le t \le 1$, the overlap of p and $\gamma(t)$ is **at most ε** (that is, it is not ε-blocking). Interrupting robots (i.e. robots that are placed on interrupting positions) will have enough wiggle room to move away from γ and "clear" the path. Notice that if for every $0 \le t \le 1$ the overlap of p

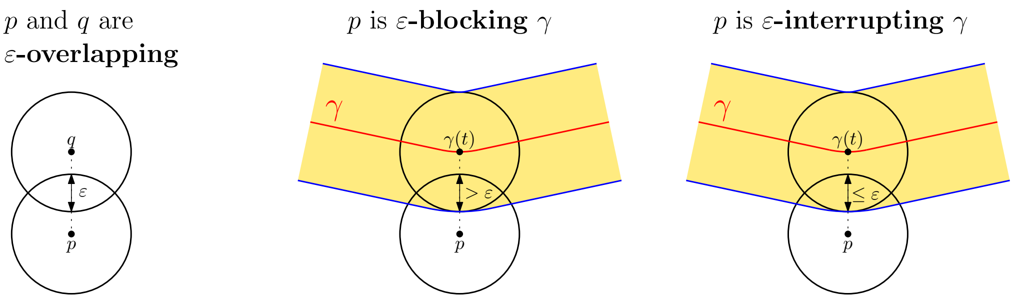

Figure 1 Overlapping, blocking, and interrupting positions.

and $\gamma(t)$ is 0, then for every $0 \leq t \leq 1$ it holds that $\|\gamma(t) - p\| > 2$, and therefore in this case we say that p is 0-interrupting γ. We say that a path γ is ***ε-blocked*** if there exists $s \in S$ that ε-blocks it, and that γ is ***ε-interrupted*** if no $s \in S$ is ε-blocking it.

The overlap parameter. For the analysis in Section 3, we fix a parameter $0 \leq \varepsilon \leq 1$ which we call the ***overlap parameter***. In these sections, for simplicity of the presentation, we sometimes omit ε from the above notions and just say "blocking" or "interrupting". In our analysis, we introduce a set of constrains described as functions of ε, which are required for the different parts of our algorithm to work. In Section 4 we show how to choose ε in order to optimize different separation bounds.

3 Constraints for tighter separation bounds

To give an intuition for the required constraints, we first provide a high-level overview of our strategy. The algorithm is recursive, and its input is the overlap parameter ε, a set S of starting positions, a set T of target positions, and a workspace $\mathcal{W}$. In each recursive iteration, one robot is moving to a target position, then S,T and $\mathcal{W}$ are updated accordingly.

The first step is to compute an ***optimal-assignment path set***, as it is defined in [15]:

▶ **Definition 1.** *Let Γ be a set of m geodesic paths $\gamma_1, \dots, \gamma_m$, where $\gamma_i : [0,1] \to \mathcal{F}$ for every $1 \leq i \leq m$, and such that $\bigcup_{i=1}^{m} \gamma_i(0) = S$, $\bigcup_{i=1}^{m} \gamma_i(1) = T$. We say that Γ is an optimal-assignment path set for $S, T, \mathcal{W}$ if it minimizes $\sum_{i=1}^{m} |\gamma_i|$ over all such path sets.*

Note that an optimal assignment path set Γ is not necessarily a feasible solution, because it ignores possible collisions between robots. This also means that the sum of paths in Γ is at most the length of an optimal solution to our MRMP problem. Let $\Gamma = \{\gamma_1, \dots, \gamma_m\}$ be an optimal-assignment path set for $S, T, \mathcal{W}$.

In Section 3.1 we present a useful lemma showing that a blocking robot can enter the geodesic path that it blocks. This lemma will be useful throughout the analysis, and it requires one constraint on the obstacles-separation, and another on the robots-separation.

The next step is to find a target $t \in T$ which does not block any path in Γ (other than the path γ from $s \in S$ to t). Then we can move a robot from s to t along γ, and it will not block other paths in the following iterations. Note that t may still interrupt other paths. In Section 3.2 we show that such a target always exists, when assuming an additional constraint on the robots-separation.

If there is a position $s' \in S$, which blocks the path γ, we need to make a switch, i.e., we construct a new path in $\mathcal{F}$ from s' to t, which is not blocked by any other robot. In Section 3.3 we show how to construct such a switch path γ', under all the constraints defined earlier. If there is no such blocking position s', we simply set $\gamma' \leftarrow \gamma$.

We now have a path γ' from some $s'' \in S$ (which may be either s or s') to the target t, which in not blocked by any robots, but may be interrupted. In Section 3.4 we show how to move a robot from s'' to t along γ', while all interrupting robots (either on start or target positions) move slightly in order to clear the path. For this we introduce two additional constraints, one on the obstacles-separation, and one on the robots-separation.

Finally, we prepare the input for the next recursive iteration, by removing s'' from S, t from T, and $D_{1-\varepsilon}(t)$ from $\mathcal{W}$. Removing $D_{1-\varepsilon}(t)$ from $\mathcal{W}$ ensures that the robots placed on t does not block any path that will be computed in any of the next iterations.

In Section 4 we present the complete algorithm based on the building blocks presented in this section, prove its correctness and analyze its running time and approximation factor.

3.1 Geodesics and blocking positions in the free space

We begin by introducing the constraints under which a blocking robot positioned on some $p \in S \cup T$ can enter a geodesic path $\gamma : [0, 1] \to \mathcal{F}$ between some start and target positions, via a straight line segment connecting it to the path.

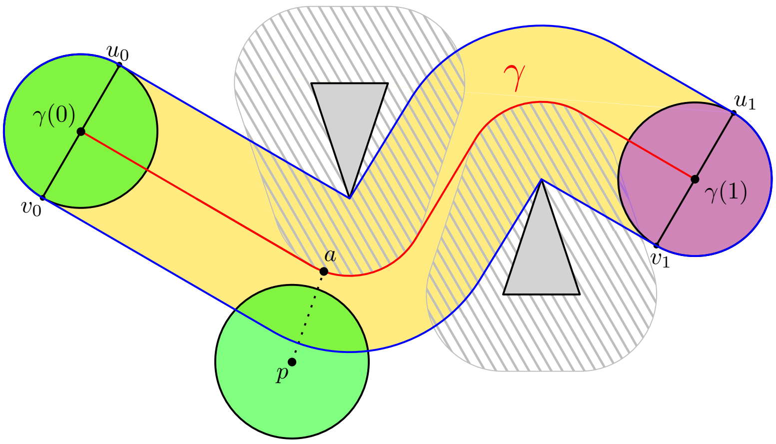

Figure 2 A geodesic path γ in $\mathcal{F}$ from a start position $\gamma(0) \in S$ to a target position $\gamma(1) \in T$ is illustrated in red. The obstacles are in gray, and the trace of γ is highlighted in yellow. A position p is blocking/interrupting γ. The point a is the closest point to p on γ.

A geodesic path between two positions $x, y \in \mathcal{F}$ is a shortest length path $\gamma : [0, 1] \to \mathcal{F}$ where $\gamma(0) = x$ and $\gamma(1) = y$. Denote by $\mathcal{D}_1(\gamma) = \bigcup_{0 \le t \le 1} \mathcal{D}_1(\gamma(t))$ the ***trace*** of γ (see Figure 2). For $0 \le t_1 \le t_2 \le 1$, denote by $\gamma[t_1, t_2]$ the subpath of γ between $\gamma(t_1)$ and $\gamma(t_2)$.

In the full version of the paper [10], we provide some very useful geometric properties of geodesic paths in the free space, which result in the following constraints. Note that we have one constraint on the obstacles-separation, and another on the robots-separation.

▶ **Constraint 1.** $\omega \ge \sqrt{(3 - \sqrt{3} - \varepsilon)^2 + 1}$

▶ **Constraint 2.**[1] $\rho \ge \sqrt{\frac{1}{2}^2 + \left(3 - \frac{\sqrt{3}}{2} - \varepsilon\right)^2}$

[1] Note that later we introduce two more constraints on ρ that together make Constraint 2 redundant, however, Lemma 2 holds already for this weaker constraint and may be of independent interest.

Let $\gamma : [0,1] \rightarrow \mathcal{F}$ be a geodesic path. We say that a point a on γ is ***locally closest*** to p if there exists $\delta > 0$ such that $D_{\|p-a\|}(p) \cap \gamma[a-\delta, a+\delta] = a$.

▶ **Lemma 2.** *Let $\gamma : [0,1] \rightarrow \mathcal{F}$ be a geodesic path such that $\gamma(0), \gamma(1) \in S \cup T$, and let $p \in S \cup T$ be a position which is ε-blocking γ. Let $a = \gamma(t)$ be a point on γ which is locally closest to p and such that $\|p-a\| < 2-\varepsilon$. If Constraints 1 and 2 hold, then $\overline{pa} \subset \mathcal{F}$.*

Intuitively, Lemma 2 states that a robot blocking γ and positioned on p can move to a along the path $\overline{pa}$, as long as no other robots are standing in its way (see Figure 2).

3.2 Existence of an ε-interrupting target

We now define formally a generalization of the standalone goal defined in [15] (Definition 3). Our definition depends on the overlap parameter ε, and it is equivalent to a standalone goal for $\varepsilon = 0$. Let $\Gamma = \{\gamma_1, \ldots, \gamma_m\}$ be an optimal-assignment path set for $S, T, \mathcal{W}$.

▶ **Definition 3.** *A target position $t \in T$ is an* **ε-interrupting target***, if for every path $\gamma \in \Gamma$ such that $\gamma(1) \neq t$, it holds that t is not ε-blocking γ.*

To show that an ε-interrupting target always exists, we need an additional constraint.

▶ **Constraint 3.** $\rho \geq 4 - 2\varepsilon$

The following theorem is a generalization of Theorem 4 in [15]. The proof follows a similar logic, however, we need to take into account some issues that arise from the generalized definition and Constraints 1–3. See the full version [10] for a complete proof.

▶ **Theorem 4.** *Let Γ be an optimal-assignment path set for $S, T, \mathcal{W}$. If Constraints 1–3 hold, then there exists an ε-interrupting target.*

3.3 Dealing with a blocked geodesic path

Let $\Gamma = \{\gamma_1, \ldots, \gamma_m\}$ be an optimal-assignment path set for $S, T, \mathcal{W}$, and assume w.l.o.g. that $\gamma_i(0) = s_i$ for every $1 \leq i \leq m$. Consider a geodesic path γ_i from s_i to t_j, and assume that is it ε-blocked. Our goal in this section is to describe the switching process: we find another start point $s_k \in S$ and a path $\gamma'_k \subset \mathcal{F}$ from s_k to t_j such that γ'_k is not ε-blocked. Moreover, we describe a path in $\mathcal{F}$ from s_i to t_ℓ, where $t_\ell = \gamma_k(1)$ (see Figure 3). The sum of lengths of these new paths will not be much longer than the original.

For a position $s_k \in S$ that ε-blocks γ_i, denote by $0 \leq w_{i,k} \leq 1$ the largest value such that $\gamma_i(w_{i,k})$ is the point on γ_i locally closest to s_k. We say that $s_k \in S$ is the ***last to ε-block γ_i*** if for any $s_x \in S \setminus \{s_k\}$ that ε-blocks γ_i, we have $w_{i,k} \geq w_{i,x}$ (i.e. the last locally closest point to s_k on γ_i is the closest to $\gamma_i(1)$ along γ_i).

Let s_k be the last to ε-block γ_i, and set $w = w_{i,k}$. Consider the paths $\gamma'_k = \overline{s_k\gamma_i(w)} \circ \gamma_i[w,1]$ and $\gamma'_i = \gamma_i[0,w] \circ \overline{\gamma_i(w)s_k} \circ \gamma_k$ (where γ_k is the geodesic path from s_k to t_ℓ, see Figure 3). We call γ'_i and γ'_k the ***switch paths*** of i and k, respectively. Notice that $\|s_k - \gamma_i(w)\| < 2-\varepsilon$, and therefore $|\gamma'_k| + |\gamma'_i| < |\gamma_k| + |\gamma_i| + 4 - 2\varepsilon$. The following theorem will help us in showing that the switch paths are not blocked (see [10] for a full proof).

▶ **Theorem 5.** *Assume that Constraints 1–3 hold. Then (i) both γ'_i and γ'_k are in $\mathcal{F}$, (ii) $\overline{s_k\gamma_i(w)}$ is not ε-blocked by any $p \in T \cup S \setminus \{s_k\}$, (iii) γ'_k is not ε-blocked by any $s_j \in S \setminus \{s_k\}$, and (iv) if t_j is not ε-blocking γ_k then it does not ε-block γ'_i.*

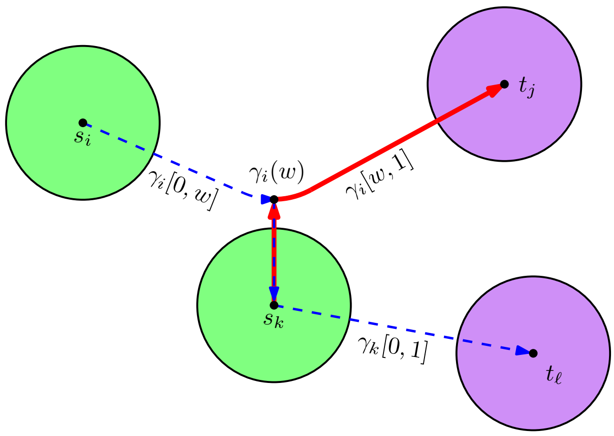

Figure 3 s_k is the last to ε-block γ_i. The switch paths are γ'_i (dashed blue) and γ'_k (red).

3.4 Dealing with robots interrupting a path

In this subsection, we discuss how to move a robot along an ε-interrupted path. We describe simple motion paths for the interrupting robots to clear the way, and we analyze the extra distance traveled by the interrupting robots. We note that shorter paths can be shown for the interrupting robots, however, as this does not effect the correctness of the algorithm, we chose to keep this part simple and describe a simple (yet rather short) motion path.

As mentioned at the beginning of this section, our motion plan consists of a weakly-monotone collision free path set, that is, robots moves to targets according some ordering one by one; when one robot travels from its starting position to some target, the other robots must stay in a small disk centered at their current position (which is either a start or a target position). To ensure that an interrupting robot is able to clear the way while remaining in a disk of radius $1+\varepsilon$ around a start/target position, we introduce two additional constraints.

▶ **Constraint 4.** $\omega \geq 1+\varepsilon$

▶ **Constraint 5.** $\rho \geq 2+\varepsilon$

Consider a robot A traveling in constant speed along a path γ_A from start position $s \in S$ to some target position $t \in T$. Let $p \in S \cup T$ be either a start or a target position, occupied by another robot B, which is ε-interrupting γ_A. We describe a motion plan for B by constructing a ***clearance path*** $\tilde{\gamma}_B : [0,1] \to \mathbb{R}^2$, along which the robot B will travel with constant speed, as follows.

Let $\gamma_A[w_1, w_2]$ be a maximal subpath where $\|\gamma_A(w) - p\| \leq 2$ for every $w \in [w_1, w_2]$, that is, when A travels along $\gamma[w_1, w_2]$, the robot B must clear the way. Note that there could be more than one such maximal subpath for B. We define the corresponding subpath $\tilde{\gamma}_B : [w_1, w_2] \to \mathbb{R}^2$ for B using a polar coordinate system, where p is the pole. Denote by r_x the radius and by θ_x the angle of a point x, and let $\tilde{\gamma}_B(w) := \big(2 - r_{\gamma_A(w)}, \theta_{\gamma_A(w)} + \pi\big)$.

The next lemma shows that the clearance path of B is feasible and relatively short. See [10] for the full proof.

▶ **Lemma 6.** *$\tilde{\gamma}_B(w)[w_1, w_2]$ is a continuous path that starts and ends at p, such that (i) $\tilde{\gamma}_B[w_1, w_2] \subset D_\varepsilon(p) \subset \mathcal{F}$, and (ii) $|\tilde{\gamma}_B[w_1, w_2]| \leq |\gamma_A[w_1, w_2]|$ for $0 \leq \varepsilon \leq 1$.*

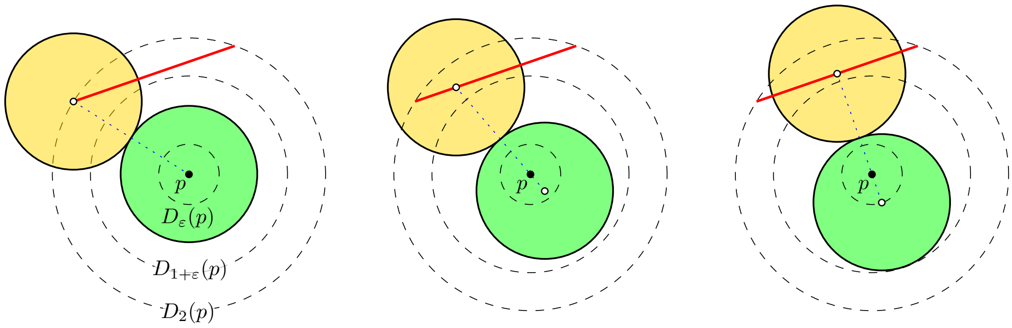

Figure 4 Left: the robot B is initially positioned at p. In red is a maximal subpath of γ_A where B needs to clear the way. Middle, right: the robot B move inside $\mathcal{D}_\varepsilon(p)$ while maintaining distance exactly 2 from the robot A.

Recall that B may interrupt γ_A in more than one maximal subpath, so overall we define the clearance path $\tilde{\gamma}_B : [0,1] \to \mathbb{R}^2$ as

$$\tilde{\gamma}_B(w) = \begin{cases} p, & \text{if } \|p - \gamma_A(w)\| \geq 2, \\ \left(2 - r_{\gamma_A(w)}, \theta_{\gamma_A(w)} + \pi\right), & \text{if } \|p - \gamma_A(w)\| < 2. \end{cases}$$

In addition, we construct such a clearance path for any other robot that interrupts γ_A. In the full version [10] we prove the following theorem, which shows that clearance paths are collision-free, assuming Constraints 1–5.

▶ **Theorem 7.** *Let A be a robot traveling in constant speed along a path γ_A from start position $s \in S$ to some target position $t \in T$, and let $\{\tilde{\gamma}_B \mid B \neq A\}$ be the set of clearance paths defined for the other robots. If Constraints 1–5 hold for $0 \leq \varepsilon \leq 1$, then the motion plan that corresponds to the paths $\gamma_A \cup \{\tilde{\gamma}_B \mid B \neq A\}$ is collision-free.*

4 Unlabeled MRMP with improved separation trade-offs

In this section we present and analyze the algorithm based on the tools developed in the previous section. The input for the algorithm is an overlap parameter ε, a workspace $\mathcal{W}$, a set S of starting positions and a set T of target positions, such that Constraints 1–5 hold. In addition, we assume that each connected component of $\mathcal{F}$ contains an equal number of start and target positions (as otherwise, there is no solution).

For the simplicity of the analysis, in this section we provide an iterative version of the algorithm. In the i'th iteration, for $1 \leq i \leq m$, the algorithm constructs a collision-free motion plan for all the robots over the time interval $[i-1, i]$, in which a single robot is moving from a start position to a target position along an ε-interrupted path (Section 3.3), and the other robots may move in their small neighborhood to clear the way (Section 3.4). The output path for a robot A will then be the concatenation $\gamma_A^* : [0, m] \to \mathcal{F}$ of the paths $\gamma_A : [i-1, i] \to \mathcal{F}$ constructed for it in all iterations. We then update S and T according to the path that was chosen, and remove from $\mathcal{W}$ a disk of radius $1 - \varepsilon$ around the target.

At the beginning of this section we gave a high level description of our algorithm, and a more detailed pseudo code is provided in Algorithm 1. Below, we prove that this algorithm returns a collision-free motion plan, with total length $O(\mathrm{OPT})$, where OPT is the optimal size of a solution to the problem.

Algorithm 1 Motion Plan for Unlabeled Disks.

Input: $\mathcal{W}$, S, T, and $0 \le \varepsilon \le 1$ such that Constraints 1–5 hold
Output: A collision-free motion plan

- For each robot $A \in [m]$, initialize an empty path γ_A^*.
- While $S \neq \emptyset$:
  1. Compute an optimal-assignment path set Γ for $S, T, \mathcal{W}$.
  2. Find an ε-interrupting target t_j, and let $\gamma_i \in \Gamma$ be the path from s_i to t_j.
  3. If γ_i is not ε-blocked, set $A \leftarrow i$, $s_A \leftarrow s_i$, $\gamma_A \leftarrow \gamma_i$.
  4. Else, let s_k be the last to ε-block γ_i, and set $A \leftarrow k$, $s_A \leftarrow s_k$, and $\gamma_A \leftarrow \gamma'_k$, where γ'_k is the switch path (Section 3.3).
  5. For all $B \neq A$, set $\gamma_B \leftarrow \tilde{\gamma}_B$, where $\tilde{\gamma}_B$ is the clearance path (Section 3.4).
  6. For each robot $B \in [m]$ set $\gamma_B^* \leftarrow \gamma_B^* \circ \gamma_B$.
  7. Set $\mathcal{W} \leftarrow \mathcal{W} \setminus \mathcal{D}_{1-\varepsilon}(t_j)$, $S \leftarrow S \setminus \{s_A\}$, $T \leftarrow T \setminus \{t_j\}$.
- Return the set of paths $(\gamma_1^*, \ldots, \gamma_m^*)$.

▶ **Theorem 8.** *Consider an input $\mathcal{W}$, S, T, and $0 \le \varepsilon \le 1$ such that Constraints 1–5 hold. If each connected component of $\mathcal{F}$ contains an equal number of start and target positions, then Algorithm 1 is guaranteed to return a collision-free motion plan Γ^*.*

Proof. First, observe that the solution returned by the algorithm is a path set that matches every start position in S to a unique target in T, because in each iteration we find a path between some $s \in S$ and $t \in T$ and then remove them from S and T for the next iteration.

Consider the first iteration of the algorithm. By Theorem 4, an ε-interrupting target t_j exists, and let γ_i be the path from s_i to t_j. If γ_i is ε-blocked, let $s_k \in S$ be the last to block γ_i, and by Theorem 5(iii), the switch path γ'_k is in $\mathcal{F}$, and it is not blocked by any other robot. Otherwise, γ_i is not blocked. In any case we choose an ε-interrupted path that leads to t_j, and by Theorem 7 the motion plan computed in this iteration is collision-free.

In the following iteration, after removing $D_{1-\varepsilon}(t_j)$ from $\mathcal{W}$, Constraints 1–5 still hold for the updated sets S and T: the robots-separation does not change, so Constraints 2, 3, and 5 still hold, and specifically by Constraint 3 for every $p_1, p_2 \in S \cup T$ we have $\|p_1 - p_2\| \ge 4 - 2\varepsilon$. Therefore, for every $p \in S \cup T \setminus \{t_j\}$ and $x \in D_{1-\varepsilon}(t_j)$ we have $\|p - x\| \ge 3 - \varepsilon$, which satisfies both Constraint 1 ($\|p - x\| \ge \rho_\varepsilon$) and Constraint 4 ($\|p - x\| \ge 1 + \varepsilon$), for any $0 \le \varepsilon \le 1$.

However, we also need to take into account the robot that is now positioned on t_j. The main observation that we will use is that t_j does not block any path γ in the free space of $\mathcal{W} \setminus D_{1-\varepsilon}(t_j)$; the reason is that the distance between any point on $a \in \gamma$ and a point $p \in D_{1-\varepsilon}(t_j)$ is at least 1, and therefore the $\|t_j - a\| \ge 2 - \varepsilon$ which means that the overlap between t_j and a is at most ε.

Consider an iteration $a > 1$ of the algorithm, and denote by $\mathcal{W}_a, S_a, T_a$ the sets $\mathcal{W}, S, T$ at the beginning of iteration a. The first step is to compute an optimal-assignment path set Γ_a. Let Γ_{a-1} be the optimal-assignment path set for $\mathcal{W}_{a-1}, S_{a-1}, T_{a-1}$. At the beginning of iteration a, we have $\mathcal{W}_a = \mathcal{W}_{a-1} \setminus \mathcal{D}_{1-\varepsilon}(t_j)$ and $T_a = T_{a-1} \setminus \{t_j\}$, where t_j is an ε-interrupting target in Γ_{a-1}. Since t_j does not block any path in Γ_{a-1}, the disk $\mathcal{D}_{1-\varepsilon}(t_j)$ does not overlap with the trace of any path in $\Gamma_{a-1} \setminus \{\gamma_i\}$. Therefore, any path $\gamma \in \Gamma_a \setminus \{\gamma_i\}$ is in the free space of $\mathcal{W}_a$. If γ_i was not blocked in iteration $a - 1$, then $S_a = S_{a-1} \setminus \{s_i\}$, and thus $\Gamma_a \setminus \{\gamma_i\}$ corresponds to a matching of S_a and T_a, which implies that Γ_a exists. Otherwise, $S_a = S_{a-1} \setminus \{s_k\}$, where $s_k \in S_{a-1}$ is the last to ε-block γ_i. Let t_ℓ be the target that was matched to it in Γ_{a-1}, and let $\gamma_k \in \Gamma_{a-1}$ be the path from s_k to t_ℓ. Observe that $\Gamma_{a-1} \setminus \{\gamma_i, \gamma_k\}$ corresponds to a matching between $S_a \setminus \{s_i, s_k\}$ and $T_{a-1} \setminus \{t_k, t_\ell\}$, and thus to have a matching between S_a and T_a we need to find a path in the free space of $\mathcal{W}_a$ between s_i and t_ℓ. We show that the switch path γ'_i from s_i to t_ℓ as defined in Section 3.3, has this property. Recall that γ'_i is a concatenation of $\gamma[1, w]$, $\overline{\gamma_i(w)s_k}$, and γ_k. The paths γ_i, γ_k are in Γ_{a-1} and therefore lie in the free space of $\mathcal{W}_{a-1}$. By Theorem 5(i), the entire path γ'_i is in the free space of $\mathcal{W}_{a-1}$. We thus need to show that the disk $\mathcal{D}_{1-\varepsilon}(t_j)$ does not overlap with the trace of γ'_i. Indeed, since t_j is an ε-interrupting target, it does not block γ_k, and therefore by Theorem 5(iv) it is also not blocking γ'_i.

The next step is to find an ε-interrupting target t_j. For this, notice that Theorem 4 still holds, because it considers only start and target positions in S_a and T_a. Let $\mathcal{A}$ be the set of robots that are already lying on target positions. Notice that we can also apply Theorem 5 on $\mathcal{W}_a, S_a, T_a$ and Γ_a, and we just need to make sure that no robot in $\mathcal{A}$ can block the switch path γ'_k. As mentioned above, this is true in general, that is, for any path $\gamma \in \Gamma_a$ from $s \in S_a$ to $t \in T_a$, no robot in $\mathcal{A}$ can block γ, because the overlap between any such robot and any point on γ is at most ε. In other words, robots in $\mathcal{A}$ can only interrupt paths in Γ_a.

The final step is using Theorem 7 to construct the clearance paths. Notice that this construction applies for any $p \in S \cup T$, and therefore it can be applied for the ε-interrupted path, even if there were robots in all the start and target positions, except for the endpoints of the path.

Lastly, notice that the concatenation of the paths that correspond to a robot A from all the iterations of the algorithm results in a continuous path. This is because each clearance path starts and ends at the same start/target position. ◀

In the full version of the paper [10] we prove the following theorem.

▶ **Theorem 9.** *Given a polygonal workspace $\mathcal{W}$ with n edges, a set S of m starting positions and a set T of m target positions, there is an algorithm that computes in $\tilde{O}(m^4+n^2m^2)$ time a collision-free motion plan of length $O(OPT)$, where OPT is the length of an optimal solution, or reports that such a motion plan does not exist, assuming that $\rho = \max\{4-2\varepsilon, 2+\varepsilon\}$ and $\omega = \max\{1+\varepsilon, \sqrt{(3-\sqrt{3}-\varepsilon)^2+1}\}$.*

The length of our solution. Note that in fact we bound the size of the solution with the term $(1+c)(\mathrm{OPT}+(4-2\varepsilon)m)$ where c is the maximum number of start/target positions that may collide a unit disk under our robots-separation constraints. For example, when the robots-separation is 2 then $c = 6$. We discuss this further in the full version of the paper [10].

Optimizing the separation bounds. We now optimize the overlap parameter over Constraints 1–5, to achieve either the smallest obstacles-separation or the smallest robots-separation possible. Recall that Constraints 1 and 4 specify the required obstacles-separation bounds, while Constraints 2, 3, and 5 specify the required robots-separation bounds. We highlight three interesting values of ε:

- For $\varepsilon = 0$ the motion plan is monotone, and the length of the solution is at most $\mathrm{OPT}+4m$. In this case, we obtain $\omega = \sqrt{13-6\sqrt{3}} \approx 1.614$ and $\rho = 4$.
- For $\varepsilon > 0$, the motion plan is weakly-monotone.
- For $\varepsilon = \frac{15-6\sqrt{3}}{13} \approx 0.354$, the obstacles-separation is minimized, and we obtain $\omega = 1 + \frac{15-6\sqrt{3}}{13} \approx 1.354$ and $\rho = 4 - 2\frac{15-6\sqrt{3}}{13} \approx 3.291$.
- For $\varepsilon = \frac{2}{3}$, the robots-separation is minimized, and we obtain $\omega = 1\frac{2}{3}$ and $\rho = 2\frac{2}{3}$.

Lower bounds. In the full version of the paper [10] we present lower bounds on the obstacles-separation when the motion plan is required to be (weakly) monotone. We provide an instances of the problem in which (i) $\omega \approx 1.614$ and no monotone motion plan exists, and (ii) $\omega \approx 1.354$ and no weakly-monotone motion plan exists. This shows that to obtain lower obstacles-separation assumptions, a completely different approach is required.

5 The Exodus algorithm: reducing the robots-separation bound

In this section we introduce a different algorithmic strategy for MRMP which we call the *Exodus Algorithm*. This strategy is more global in the sense that it also moves robots that do not directly block a path, and therefore it allows for robots-separation of 2, while requiring obstacles-separation of only 3. Whereas the previous algorithm relies on local path-clearing and larger robots-separation, Exodus replaces these local adjustments with a synchronized outward movement of all robots in order to clear the trace of a geodesic path, so that the selected robot can reach its target via this path. Note that in this section, we assume that the workspace $\mathcal{W}$ is a simple polygon. Due to space considerations, we give an overview of the algorithm and the main result here, see [10] for a formal description and proofs.

Algorithm overview. Given the sets $S, T \subset \mathcal{F}$ as before, the first step of the algorithm is to compute an optimal-assignment path set $\Gamma = \{\gamma_1, \ldots, \gamma_m\}$ for $S, T, \mathcal{W}$. Assume w.l.o.g. that $\gamma_i(0) = s_i$ and $\gamma_i(1) = t_i$ for every $1 \le i \le m$. The algorithm then proceeds iteratively. In each iteration, we select a target position $t_i \in T$ which is not yet occupied by a robot. We first extend the path γ_i so that its start and end points are on the boundary of $\mathcal{F}$. The extended path partitions $\mathcal{F}$ into maximally connected sub-regions of $\mathcal{F}$, which we call pockets. We then split each pocket using the angle bisectors corresponding to the reflex subchain of γ_i that defines the pocket. We call the cells in this refined partition Exodus cells. Each Exodus cell contains a single straight-line edge of γ_i. All the other robots are then moved outwards from the extended path by two units (which is possible because $\omega = 3$), and the movement direction for each robot is determined by the Exodus cell in which it lies. This process clears the trace of γ_i, allowing a robot positioned on s_i to move along γ_i until reaching t_i, while all the other robots remain fixed at their offset locations. After the robot reaches its target t_i, all the other robots return to their original positions by applying the inverse movement (which is possible because $\rho = 2$, i.e., no start position is too close to t_i). This iterative "corridor-opening" process motivates the name *Exodus Algorithm*. The procedure repeats until all robots reach their assigned targets. In each iterative step, $m - 1$ robots are moving $4m$ units each, while the selected robot moves along a shortest path to its assigned target (as before, $|\Gamma| \le \mathrm{OPT}$). Since there are exactly m iterations, we obtain the following theorem.

▶ **Theorem 10.** *Given a simple polygonal workspace $\mathcal{W}$ with n edges, a set S of m starting positions and a set T of m target positions, such that $\rho = 2$ and $\omega = 3$, the Exodus algorithm computes in $\tilde{O}(m^3 + mn^2)$ time a collision-free motion plan of size at most $OPT + 4m^2 - 4m$, where OPT is the length of an optimal solution.*

▶ Remark 11 (Labeled MRMP). In the labeled setting, where each start position is already associated with a specific target, we do not compute an optimal-assignment path set. Instead, we simply compute the geodesic paths between each start and its corresponding target using Wang's Euclidean shortest-path algorithm [17], and then apply the same iterative procedure described above. In this case, the running time becomes $O(m^2 \log n + mn^2)$, as no matching step is required. The motion of the robots during the corridor-opening iterations remains unchanged, and the size of the solution remains at most $\mathrm{OPT} + 4m^2 - 4m$.

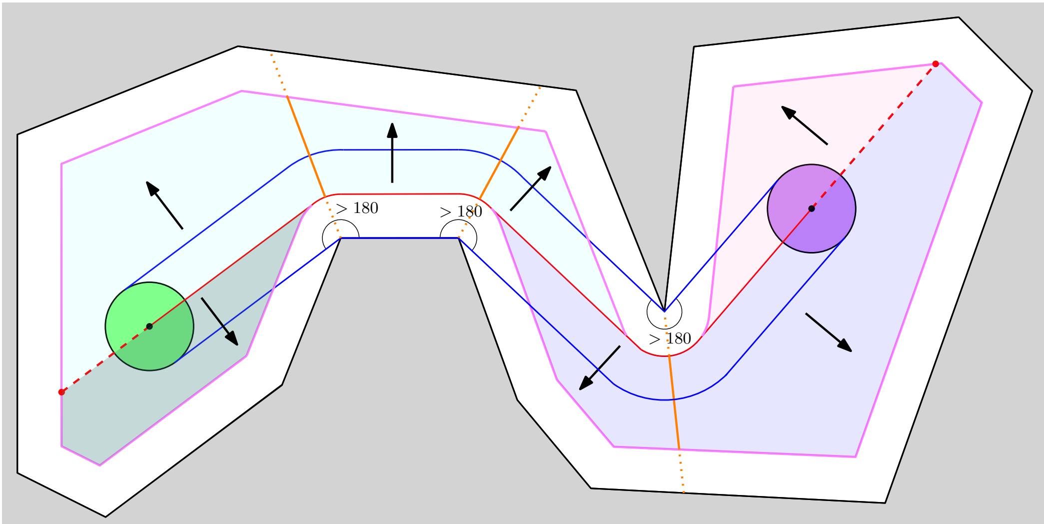

Figure 5 The extended geodesic $\hat{\gamma}$ is in red, and its trace is in blue. In pink is $\partial\mathcal{F}$. Each colored region represents a distinct pocket induced by $\hat{\gamma}$, and the orange rays are the angle bisectors generated at the corresponding reflex vertices of $\partial\mathcal{F}$. These bisectors subdivide each pocket into Exodus cells, within which a single direction vector is illustrated by the black arrows.

6 Discussion and future work

In this paper we provide multi-robot motion plans that require tighter assumptions on the density of the input setting. Note that for the Exodus algorithm we require $\rho = 2$. This separation bound is tight for monochromatic robots-separation, however, the bichromatic robots-separation can potentially be 0. This raises the question of what is the minimum obstacles-separation for which we can remove the robots-separation assumption completely.

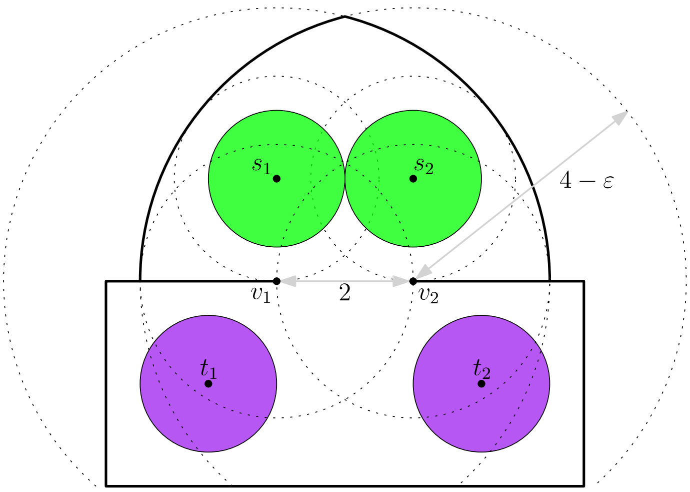

Figure 6 Unlabeled MRMP with obstacles-separation of $1.5 - \varepsilon$, for which no solution exists.

In the full version of the paper [10] we also present lower bounds for the amount of obstacles-separation needed so that a solution to the MRMP problem always exists, without any assumptions on the separation between robots. In the unlabeled case, we prove that obstacles-separation of at least 1.5 is sometimes needed (see Figure 6), and in the labeled case, we observe that obstacles-separation of at least 2 is sometimes necessary (see Figure 7).

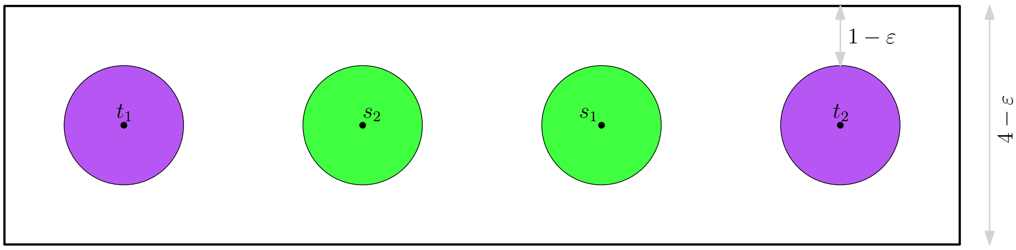

Figure 7 Labeled MRMP with obstacles-separation of $2-\varepsilon$, for which no solution exists.

References

1 Mikkel Abrahamsen, Kevin Buchin, Maike Buchin, Linda Kleist, Maarten Löffler, Lena Schlipf, André Schulz, and Jack Stade. Reconfiguration of unit squares and disks: Pspace-hardness in simple settings. In Oswin Aichholzer and Haitao Wang, editors, *41st International Symposium on Computational Geometry, SoCG 2025, Kanazawa, Japan, June 23-27, 2025*, volume 332 of *LIPIcs*, pages 1:1–1:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.SOCG.2025.1.

2 Aviv Adler, Mark de Berg, Dan Halperin, and Kiril Solovey. Efficient multi-robot motion planning for unlabeled discs in simple polygons. *IEEE Transactions on Automation Science and Engineering*, 12(4):1309–1317, 2015. doi:10.1109/TASE.2015.2470096.

3 Pankaj K. Agarwal, Mark de Berg, Benjamin Holmgren, Alex Steiger, and Martijn Struijs. Optimal motion planning for two square robots in a rectilinear environment. In Oswin Aichholzer and Haitao Wang, editors, *41st International Symposium on Computational Geometry, SoCG 2025, Kanazawa, Japan, June 23-27, 2025*, volume 332 of *LIPIcs*, pages 5:1–5:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.SOCG.2025.5.

4 Pankaj K. Agarwal, Tzvika Geft, Dan Halperin, and Erin Taylor. Multi-robot motion planning for unit discs with revolving areas. *Computational Geometry*, 114:102019, 2023. doi:10.1016/j.comgeo.2023.102019.

5 Pankaj K. Agarwal, Dan Halperin, Micha Sharir, and Alex Steiger. Near-optimal min-sum motion planning for two square robots in a polygonal environment. In David P. Woodruff, editor, *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024, Alexandria, VA, USA, January 7-10, 2024*, pages 4942–4962. SIAM, 2024. doi:10.1137/1.9781611977912.176.

6 Pankaj K. Agarwal, Benjamin Holmgren, and Alex Steiger. Near-optimal min-sum motion planning in a planar polygonal environment. *CoRR*, abs/2510.21639, 2025. doi:10.48550/arXiv.2510.21639.

7 Bahareh Banyassady, Mark de Berg, Karl Bringmann, Kevin Buchin, Henning Fernau, Dan Halperin, Irina Kostitsyna, Yoshio Okamoto, and Stijn Slot. Unlabeled Multi-Robot Motion Planning with Tighter Separation Bounds. In Xavier Goaoc and Michael Kerber, editors, *38th International Symposium on Computational Geometry (SoCG 2022)*, volume 224 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 12:1–12:16, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.SoCG.2022.12.

8 Thomas Brocken, G. Wessel van der Heijden, Irina Kostitsyna, Lloyd E. Lo-Wong, and Remco J. A. Surtel. Multi-robot motion planning of k-colored discs is pspace-hard. In Martin Farach-Colton, Giuseppe Prencipe, and Ryuhei Uehara, editors, *10th International Conference on Fun with Algorithms, FUN 2021, Favignana Island, Sicily, Italy, May 30 - June 1, 2021*, volume 157 of *LIPIcs*, pages 15:1–15:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.FUN.2021.15.

9 Guillermo Esteban, Dan Halperin, Víctor Ruíz, Vera Sacristán, and Rodrigo I. Silveira. Shortest coordinated motion for square robots. In Pat Morin and Subhash Suri, editors, *Algorithms and Data Structures - 18th International Symposium, WADS 2023, Montreal, QC, Canada, July 31 - August 2, 2023, Proceedings*, volume 14079 of *Lecture Notes in Computer Science*, pages 430–443. Springer, 2023. doi:10.1007/978-3-031-38906-1_28.

10 Tsuri Farhana, Omrit Filtser, and Shalev Goldshtein. Unlabeled multi-robot motion planning with improved separation trade-offs. *CoRR*, abs/2603.19502, 2026. arXiv:2603.19502.

11 Sándor P. Fekete, Ramin Kosfeld, Peter Kramer, Jonas Neutzner, Christian Rieck, and Christian Scheffer. Coordinated motion planning: Multi-agent path finding in a densely packed, bounded domain. In Julián Mestre and Anthony Wirth, editors, *35th International Symposium on Algorithms and Computation, ISAAC 2024, Sydney, Australia, December 8-11, 2024*, volume 322 of *LIPIcs*, pages 29:1–29:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.ISAAC.2024.29.

12 John E Hopcroft, Jacob Theodore Schwartz, and Micha Sharir. On the complexity of motion planning for multiple independent objects; pspace-hardness of the" warehouseman's problem". *The international journal of robotics research*, 3(4):76–88, 1984.

13 Israela Solomon and Dan Halperin. Motion planning for multiple unit-ball robots in $\mathbb{R}^d$. In Marco Morales, Lydia Tapia, Gildardo Sánchez-Ante, and Seth Hutchinson, editors, *Algorithmic Foundations of Robotics XIII, Proceedings of the 13th Workshop on the Algorithmic Foundations of Robotics, WAFR 2018, Mérida, Mexico, December 9-11, 2018*, volume 14 of *Springer Proceedings in Advanced Robotics*, pages 799–816. Springer, 2018. doi:10.1007/978-3-030-44051-0_46.

14 Kiril Solovey and Dan Halperin. On the hardness of unlabeled multi-robot motion planning. In Lydia E. Kavraki, David Hsu, and Jonas Buchli, editors, *Robotics: Science and Systems XI, Sapienza University of Rome, Rome, Italy, July 13-17, 2015*, 2015. doi:10.15607/RSS.2015.XI.046.

15 Kiril Solovey, Jingjin Yu, Or Zamir, and Dan Halperin. Motion planning for unlabeled discs with optimality guarantees. In Lydia E. Kavraki, David Hsu, and Jonas Buchli, editors, *Robotics: Science and Systems XI, Sapienza University of Rome, Rome, Italy, July 13-17, 2015*, 2015. doi:10.15607/RSS.2015.XI.011.

16 Paul G. Spirakis and Chee-Keng Yap. Strong np-hardness of moving many discs. *Inf. Process. Lett.*, 19(1):55–59, 1984. doi:10.1016/0020-0190(84)90130-3.

17 Haitao Wang. A new algorithm for euclidean shortest paths in the plane. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 975–988. ACM, 2021. doi:10.1145/3406325.3451037.