Semesteroppgave

CrazyFlie trilateration network

Magnus Berdal

I AM NOT CONVINCED ABOUT USING THE NAME CRAZYFLIE

Contents

ALWAYS KEEP SPELLCHECKERS ON

1 Introduction

A first responder (FR) is a person who is among those responsible for going immediately to the scene of an accident or emergency to provide assistance [1]. They will typically be employed by the emergency services such as ~~the~~ police, fire department or health services, and take it upon themselves to secure the health of people, property and the enivornment. Often ~~times~~ this includes sacrificing their own safety in order to secure that of others. The the need for quick action gives them no other choice.

Search-and-rescue personell in particular expose themselves to dangerous environments. Burning buildings, collapsed or flooded caves etc. are just some examples of the rapid-changing hazardeous evironments in which FRs can find themselves. Often ~~times~~ they enter without any knowledge of what is waiting for them on the other side. Imagine a burning building. The local fire department has just arrived. There is no time to spare, so the first firefighters enter to save the lives of those inside before the floor plan is inspected. It might not even be valid as the roaring fire continues to eat up the walls and support beams for the roof. Still the firefighters enter in order to rescue those inside. If something is to happen to the entering firefighters there is no way to locate them, and the ones entering to save them still don't know what is waiting for them inside.

Advancements in technology can be used to limit the dangers to which FRs expose themselves. With the increasing performance and decreasing cost of mobile robots, and the ability to outfit them with whatever equipment one might think of, it is relevant to think of how this might be used to the advantage of first responders. Using disposable robots for tasks such as mapping ~~the~~ UNKNOWN environment could drastically improve the safety of FRs. In the example stated above, mobile robots could be dispatched into the burning house and continuously feed the firedighters with a real time map of the environment. Robots could also enter the building to set up an ad-hoc network [2] used for locating FRs inside GNSS denied environments, or supply information about events happening, such as changes in the environment or the presence of poisenous gasses etc.

The challenge in creating such ad-hoc networks lay in enabling the robots to quickly set up the network as desired without supervision. This is the matter that will be studied in this report.

VERY NICE :)

CONTROL

2 Background

Micro Areal Vehicles (MAVs) have great potential in contributing to indoor search and rescue missions. Their small size and weight make them easy to transport and allows for rapid field deployment. Furthermore they are agile, allowing them to operate in complex environments and accessing hard-to-get-to places.

Limitations of current technologies, prohibiting the existance of fully autonomous MAVs, are examined in [3]. They define the following requirements for a fully autonomous MAV:

- Inference: The ability to infer situational awareness from sensory measurements.
- Reasoning: The ability to define a mission based on abstract human-defined goals.
- Unsupervised learning: The ability to adapt and learn its own controll strategies without human supervision.

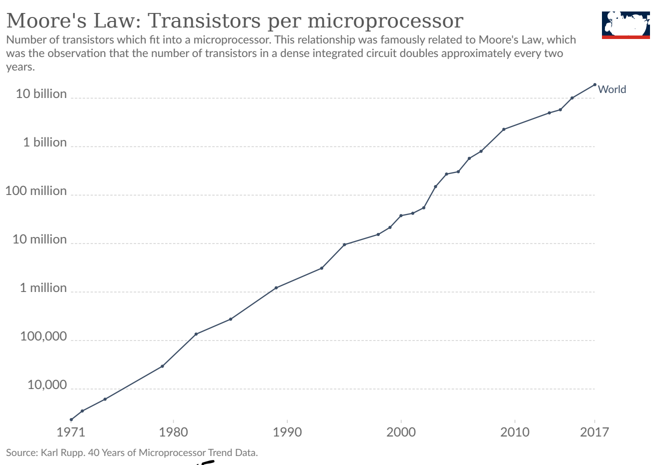
is

Due to the constraints on the size of MAVs, the exis e tance of fully autonomous MAVs ~~are~~ mainly dependent on the exis e tance of small and efficient enough hardware components such as power supplies, sensors and processors allowing them to quickly process ~~the~~ incoming data and convert them ~~to~~ actions. In [3] ~~they~~ conclude that as of March 2017 no fully autonomous MAV system exist s

INTO AUTHORS THE

Although a fully autonomous MAV system is yet to be realised, the advancement in transistor density shown in Figure 1, and continously dimishing price of microprocessors continue to allow for faster, and thus more complex, on-board computations. This allows for increased level s of autonomy of the MAVs as they can perform ever more complex computations locally without the aid of other computation units or human interference.

BIGGER



THE

Figure 1: Evolution of number of transistors per microprocessor.

Most MAVs utilize electrical propulsion systems as they are easier to miniaturize, and have faster response time than combustion based systems. Energy is typically stored on LiPo batteries. Figure 2 shows how the energy density and price of Lithium-ion batteries have evolved over the last decades. As the energy density of Lithium-ion batteries already have and is projected to

Bigger

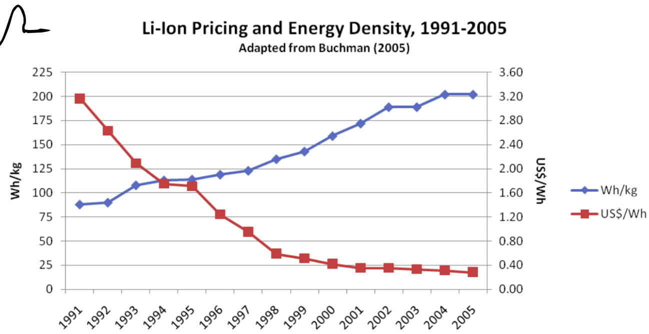


Figure 2: Evolution of energy density and cost of Lithium-ion batteries. Source: [4].

continue to increase, this will allow for fitting more power consuming equipment on the MAVs, and increasing their endurance. Thus enabling MAVs to take on more complex and long lasting missions, resulting in an increasing autonomy.

Not only hardware components have made leaps over the last decades. The advancement in autonomy of MAV systems has been a field of great scientific interest. The Robotics & Perception Group at the University of Zurich have performed excessive research into the field of visual-inertial based real-time navigation and mapping by MAVs in GNSS denied environments [5, 6]. The goal of which is enabling MAVs to localize themselves in and map unknown environments without external aid.

In [5] a fast and robust visual-odometry algorithm (SVO) is presented. The method is faster than previously presented methods, and is therefore especially usefull for MAV operations as it should be able to run on a onboard computer with limited processing power. It is concluded that the method is especially usefull for state estimation onboard MAVs as the algorithm runs at a high enough rate to provide accurate state estimates. Furthermore the use of depth-filters yields an accurate map of the environment with few outliers.

In [6] Scaramuzza et al. present a system for autonomous mapping of unknown indoor and outdoor environments performed by a single MAV using SVO. The MAV is supplied with a trajectory it is to follow, and using only on-board processing and sensing follows said trajectory and provides at the same time a real-time map of the environment.

A lot of research has also been put into distributed swarm control: the task of making a set of MAVs work together to fulfill some collective goal, and doing so with only local information. In [7] De Gennaro and Jadbabaie present a distributed method for optimizing the connectivity of a multi-agent network. In their approach each agent knows only the state of its neighbours, where another agent is defined as a neighbour if it fulfills some proximity condition. Their approach shows through simulations that the the swarm reaches a configuration that corresponds to a maxima of the second smallest eigenvalue of the Laplacian matrix. ASSOCIATED TO THE COMMUNICATION GRAPH

Cassandras and Zhong present in [8] a distributed method for a multi-agent network in which the goal is maximizing the joint detection probability of randomly occuring events in an unknown environment. They also present an algorithm that preserves connecticity of the network of agents. They conclude that using their method, the swarm of agents reach a configuration corresponding to a local maxima of the joint detection probability function. Furthermore they find that when connecticity preservation is imposed, the swarm settles at a configuration in which the joint detection probability is smaller than when connectivity of the network is not taken into account.

Note to self:

- CrazyFlie (and other small scale drones)
- Increase in computational power
- State of the art drones

THE LIT. REV. SEEMS MORE A COLLECTION OF PAPERS THAN A FLOW OF RESULTS CONNECTED WITH THIS REPORT

3 Notation

- Ω: Mission space.
- $\mathcal{O}$: Set of obstacles inside the mission space.
- $\mathcal{F}$: Feasible space.
- $\mathbf{x}_a$: Position of agent a.
- r_a: Maximum communication range of agent a.
- $\hat{p}(\mathbf{x}_a, \mathbf{y})$: Probability of agent a being able to communicate with entity at position $\mathbf{y}$.
- V_a: Visible set of agent a.
- U_a: Invisible set of agent a.
- $\mathbf{X}_\mathcal{S}$: Positions of all agents in the swarm $\mathcal{S}$.
- $\Phi^n(\mathbf{X}_\mathcal{S}, \mathbf{y})$: Probability of n members in the swarm $\mathcal{S}$ being able to communicate with entity at position $\mathbf{y}$.
- $\Phi^{n^+}(\mathbf{X}_\mathcal{S}, \mathbf{y})$: Probability of n or more members in the swarm $\mathcal{S}$ being able to communicate with entity at position $\mathbf{y}$.

3.1 Mathematical operators

- For a set $\mathcal{S}$ defined by $\mathcal{S} = \{s_0 \dots s_{N-1}\},\ N < \infty$ we define:
  - $|\mathcal{S}| = N$: The size of $\mathcal{S}$.
  - $\mathrm{Comb}(\mathcal{S}, n) = \{\mathcal{A} : \mathcal{A} \subseteq \mathcal{S}, |\mathcal{A}| = n\}$: the set of all subsets of $\mathcal{S}$ of size n.
- For a set $\mathcal{S}$ of points $\mathbf{x} \in \mathbb{R}^N$ defined by $\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^N : f(\mathbf{x}) \leq \mathbf{0}\}$ we define:
  - $\delta\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^N : f(\mathbf{x}) = \mathbf{0}\}$: The boundary of $\mathcal{S}$.
  - $\mathrm{int}(\mathcal{S}) = \{\mathbf{x} \in \mathbb{R}^N : f(\mathbf{x}) < \mathbf{0}\}$: The interior of $\mathcal{S}$.

4 System Description

4.1 Feasible space

As in [9], a *mission space*, Ω, is defined as a simple polygon [10]. Within the mission space there exists $N_o \geq 0$ obstacles, each one of which is defined as a simple polygon. The set of all obstacles, $\mathcal{O}$, is defined according to (1).

$$\mathcal{O} = \begin{cases} \{o_0 \dots o_{N_o-1}\} & , N_o > 0 \\ \emptyset & , N_o = 0 \end{cases} \tag{1}$$

The obstacles in $\mathcal{O}$ constrains the movement of entities within the mission space, as it is not possible to penetrate the boundary of an obstacle. Due to this, once an entity is inside Ω, it is constrained to be positioned within Ω and outside $\mathrm{int}(o)\ \forall\ o \in \mathcal{O}$. From this we define the *feasible space*, $\mathcal{F}$, as all points where it is possible to place an entity:

$$\mathcal{F} = \{\mathbf{y} \in \mathbb{R}^2 : \mathbf{y} \in \Omega,\ \mathbf{y} \notin \mathrm{int}(o)\ \forall\ o \in \mathcal{O}\} = \Omega \setminus \bigcup_{o \in \mathcal{O}} \mathrm{int}(o) \tag{2}$$

4.2 Agent

An *agent*, denoted by an index a, is defined by its position $\mathbf{x}_a \in \mathbb{R}^2$ and its maximum range of communication r_a. From this we define the communication disk of agent a:

$$D_a = \{\mathbf{y} \in \mathbb{R}^2 : \|\mathbf{x}_a - \mathbf{y}\| \leq r_a\} \tag{3}$$

Assuming line-of-sight (LoS) communication, meaning an agent cannot communicate with an entity if there is an obstacle or a mission space wall between them, we define the *visible set* of agent a:

$$V_a = \{\mathbf{y} \in \mathbb{R}^2 : \mathbf{y} \in D_a, \lambda\mathbf{y} + (1-\lambda)\mathbf{x}_a \in \mathcal{F}\ \forall\ 0 \leq \lambda \leq 1\} \tag{4}$$

The counterpart to the visible set, called the invisible set of agent a, is simply defined as:

$$U_a = \mathcal{F} \setminus V_a \tag{5}$$

An example of the visible set for an agent is show in Figure 3.

The probability of an agent a being able to communicate with another entity positioned at a point $\mathbf{y}$, from now on called the local probability of agent a, is defined according to:

Find notation for $\mathbb{R}^5$ with non-negative values in last dimension

$$\hat{p} : \mathbb{R}^2 \to [0,1] \quad \hat{p}(\mathbf{x}_a, \mathbf{y}) = \begin{cases} p(\|\mathbf{x}_a - \mathbf{y}\|) & , \mathbf{y} \in V_a \\ 0 & , \mathbf{y} \in U_a \end{cases} \tag{6}$$

Where $p(\cdot)$ is a non-increasing function of its argument.

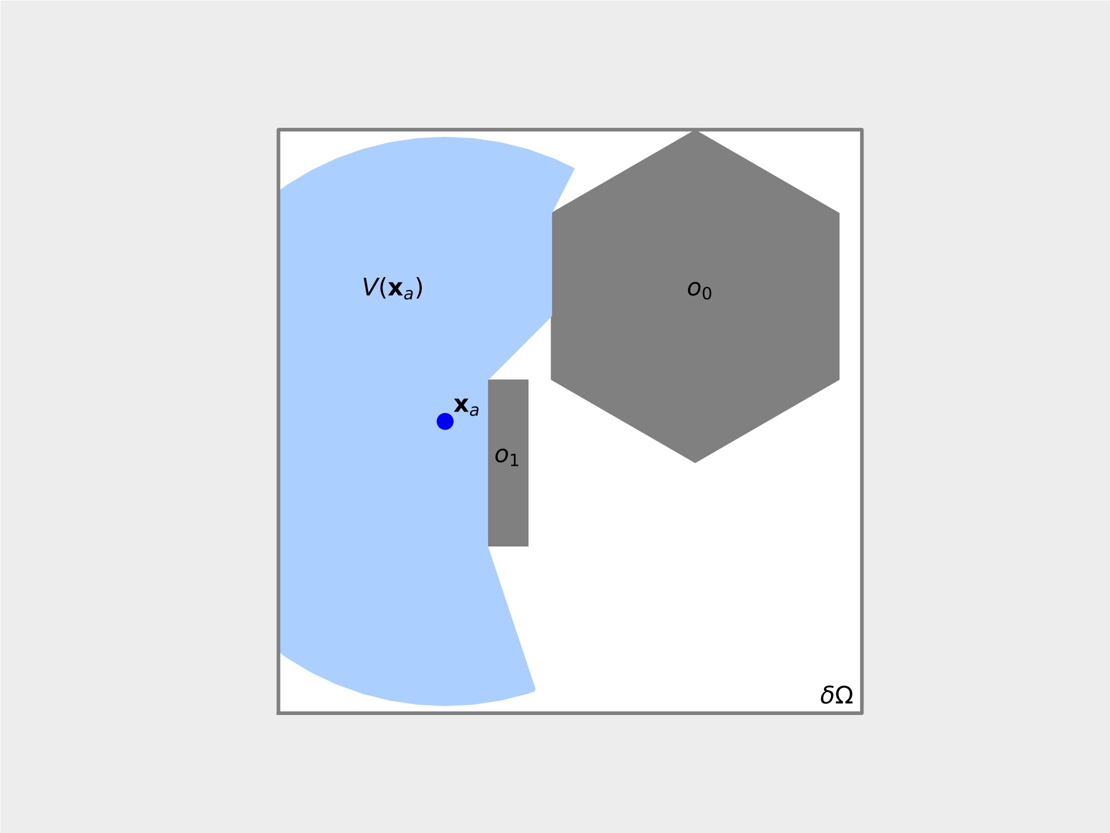


Figure 3: Visible set (light blue) for the agent placed at $\mathbf{x}_a$ in a rectangular mission space Ω with two obstacles ($\mathcal{O} = \{o_0, o_1\}$)

Update figure: $\mathbf{s}_i$ should be $\mathbf{x}_a$

4.3 Swarm

A *swarm*, $\mathcal{S}$ of size N is a set of agents $\{a_0 \dots a_{N-1}\}$. The state of the swarm is described by the state of its participants , and is expressed as vector $\mathbf{X}_\mathcal{S} \in \mathbb{R}^{2N}$ as shown in (7).

$$\mathbf{X}_\mathcal{S} = \begin{bmatrix} \mathbf{x}_{a_0} \\ \vdots \\ \mathbf{x}_{a_{N-1}} \end{bmatrix}, \quad \mathbf{X}_{\mathcal{S},i} = \mathbf{x}_{a_i} \tag{7}$$

The maximum radii of communication of the swarm are represented as the vector $\mathbf{r} \in \mathbb{R}^N$ as shown in (8).

$$\mathbf{r}_\mathcal{S} = \begin{bmatrix} r_{a_0} & \dots & r_{a_{N-1}} \end{bmatrix}^T \tag{8}$$

Assuming that the distributions for all agents in the swarm are independent lets us use (7) in [11] to express the probability of n members in the swarm, $\mathcal{S}$, being able to communicate with an entity at a point $\mathbf{y}$:

$$\Phi^n(\mathbf{X}_\mathcal{S}, \mathbf{y}) = \sum_{A \in Comb(\mathcal{S}, n)} \prod_{a \in \mathcal{A}} \hat{p}(\mathbf{x}_a, \mathbf{y}) \prod_{a \in \mathcal{S} \setminus \mathcal{A}} \big(1 - \hat{p}(\mathbf{x}_a, \mathbf{y})\big) \tag{9}$$

Later we will use the probability of *at least* n members in a swarm beign able to communicate with an entity placed at $\mathbf{y}$, which is defined as:

$$\Phi^{n^+}(\mathbf{X}_\mathcal{S}, \mathbf{y}) = 1 - \sum_{i=0}^{n-1} \Phi^i(\mathbf{X}_\mathcal{S}, \mathbf{y}) \tag{10}$$

5 Problem formulation

Using a swarm of N CrazyFlie drones we want to set up a network in a mission space where the drones work as beacons in order to deliver precise positional data to entities entering the mission space.

skriv om at det er nskelig gjre dette distribuert

5.1 Multilateration

Multilateration is the process of determining the positions of unknown points in space by measurements of distances from known points [12]. In order to perform this task in two-dimensional space, at least three known points are needed.

Given $n \geq 3$ beacons located at positions $\mathbf{x}_a \in \mathbb{R}^2,\ 0 \leq a < n$ where not all points lie on a single line, i.e.

$$\text{Rank}(\mathbf{V}) = 2, \quad \mathbf{V} = \begin{bmatrix} \mathbf{x}_0 - \mathbf{x}_1 & \dots & \mathbf{x}_0 - \mathbf{x}_{n-1} \end{bmatrix} \in \mathbb{R}^{2\times(n-1)} \tag{11}$$

the location of an entity, denoted by $\mathbf{y} \in \mathbb{R}^2$, can be determined as follows:

1. The entity broadcasts signal and starts a timer at t_0.
2. Beacons at $\mathbf{x}_a,\ 0 \leq a < n$ receive broadcasted signal and immediately responds with a packet containing $\mathbf{x}_a$.
3. When receiving the packet from beacon at $\mathbf{x}_a$, the entity stores the time of reception in a variable $t_{1,a}$.
4. When at least 3 beacons have responded, the entity calculates the distance from itself to beacon at $\mathbf{x}_a$: $d_a = \frac{1}{2}s(t_{1,a} - t_0)$, where s is the propagation speed of the signal. The factor $\frac{1}{2}$ is due to the signal traveling two times the distance between the entity and the beacon placed at $\mathbf{x}_a$ (the ping travels from the entity to the agent, and the packet sent by the agent travels back again).
5. Based on the distances, d_a, and the positions of the beacons the entity can determine its position by calculating the point where circles centered at $\mathbf{x}_a$ with radii d_a intersect.

If sufficiently many beacons respond an ML (Maximum Likelihood) estimator of the position of the entity can be computed [13]. Defining the error function:

$$e_a(\mathbf{y}) = s(t_{1,a} - t_{0,a}) - \|\mathbf{x}_a - \mathbf{y}\| = d_a - \|\mathbf{x}_a - \mathbf{y}\| \tag{12}$$

We obtain the estimate for the position of the entity by solving:

$$\mathbf{y}_{\text{MMSE}} = \min_{\mathbf{y}} \mathbf{E}^T\mathbf{E}, \quad \mathbf{E} = \begin{bmatrix} e_0(\mathbf{y}) \\ \vdots \\ e_{n-1}(\mathbf{y}) \end{bmatrix} \tag{13}$$

As the pings sent by the entity that is to be located travel at large velocities, sufficient spread of the beacons is necessary to ensure accurate locating. This is due to the resolution of the internal clock of the entity setting a bound on accuracy of time measurements, and thus distance measurements. Increasing the distance between beacons causes the distance traveled by the pings to differ more, and thus decreasing the probability of mutliple pings returning to the entity within the same clock cycle. Figure 4 shows how the position of an entity can be determined from the known positions of 3 agents.

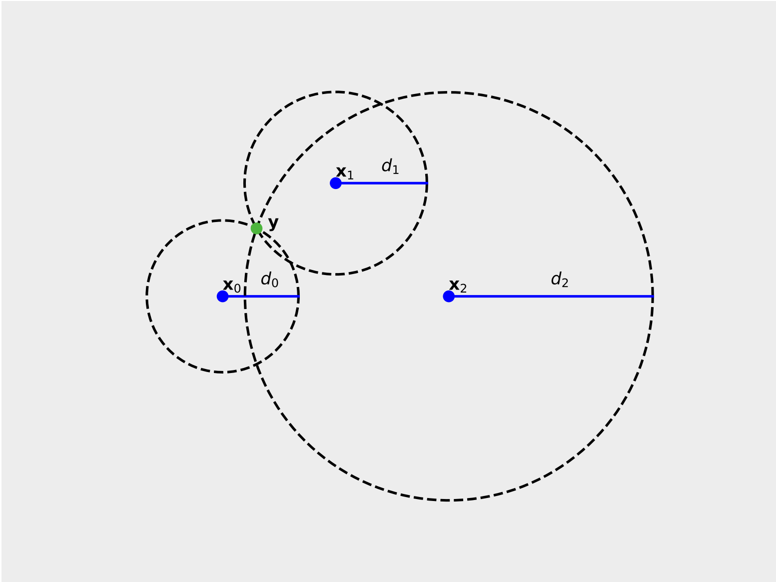


Figure 4: Position, $\mathbf{y}$, of entity determined by trilateration using known positions of $n = 3$ beacons.

mention how spread of agents is benefitial and motivate it using litterature

5.2 Coverage

It is clear from subsection 5.1 that three or more agents are needed to perform the task of multilateration. Hence a point $\mathbf{y} \in \mathcal{F}$ is said to be *covered* iff. it is within communication range of at least three agents, i.e. it is possible to determine the position of an entity placed at $\mathbf{y}$ through multilateration.

As discussed in subsection 5.1 it is beneficial that agents used for multilateration spread out to some extent in order to ensure sufficient accuracy of multilateration.

5.3 Constraints

It is clear from subsection 5.1 that agents used as beacons in a multilateration scheme cannot be positioned on a straight line, as this would make it impossible to uniquely determine the unknown position of an entity. Due to the fact that only agents close to the entity whose position is to be determined will take part in the multilateration scheme, we impose only that agents that together can be used for multulateration must satisfy the non-linear position requirement in (11).

We also impose that any two agents must be some minimum distance apart at any given time. This is due to the fact that agents colliding could cause damage to the hardware, and possibly render them unusable.

5.4 Objective function derivation

The objective function presented here is inspired by [9], but differs in that for the purpose of multilateration, it is required that at least three agents must be within range of a point in order for the point to be covered.

We assume we have a total of N agents in a swarm $\mathcal{N}$ at our disposal. Furthermore we assume that all agents have the same maximum radius of communication:

$$r_a = r \; \forall \; a \in \mathcal{N} \tag{14}$$

Using (10) we can express the probability of a point $\mathbf{y}$ being covered by the swarm as:

$$\Phi^{3^+}(\mathbf{X}_{\mathcal{N}}, \mathbf{y}) = 1 - \Phi^0(\mathbf{X}_{\mathcal{N}}, \mathbf{y}) - \Phi^1(\mathbf{X}_{\mathcal{N}}, \mathbf{y}) - \Phi^2(\mathbf{X}_{\mathcal{N}}, \mathbf{y}) \tag{15}$$

In order to formulate a distributed optimization algorithm, we rewrite the probability of coverage in (15) with focus on a single drone a. We partition the swarm, $\mathcal{N}$, into two disjoint sets: $\{a\}$ and $\mathcal{N} \setminus \{a\}$. Using this we can rewrite (15) as:

$$\begin{aligned}
\Phi^{3^+}(\mathbf{X}_{\mathcal{N}}, \mathbf{y}) = 1 & \\
& - \big(1 - \hat{p}(\mathbf{x}_a, \mathbf{y})\big) \prod_{k \in \mathcal{N} \setminus \{a\}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big) \\
& - \hat{p}(\mathbf{x}_a, \mathbf{y}) \prod_{k \in \mathcal{N} \setminus \{a\}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big) \\
& - \big(1 - \hat{p}(\mathbf{x}_a, \mathbf{y})\big) \sum_{j \in \mathcal{N} \setminus \{a\}} \hat{p}(\mathbf{x}_j, \mathbf{y}) \prod_{k \in \mathcal{N} \setminus \{a\} \setminus \{j\}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big) \\
& - \hat{p}(\mathbf{x}_a, \mathbf{y}) \sum_{j \in \mathcal{N} \setminus \{a\}} \hat{p}(\mathbf{x}_j, \mathbf{y}) \prod_{k \in \mathcal{N} \setminus \{a\} \setminus \{j\}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big) \\
& - \big(1 - \hat{p}(\mathbf{x}_a, \mathbf{y})\big) \sum_{\mathcal{A} \in Comb(\mathcal{N} \setminus \{a\}, 2)} \prod_{j \in \mathcal{A}} \hat{p}(\mathbf{x}_j, \mathbf{y}) \prod_{k \in \mathcal{N} \setminus \{a\} \setminus \mathcal{A}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big) \\
& = 1 \\
& - \prod_{k \in \mathcal{N} \setminus \{a\}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big) \\
& - \sum_{j \in \mathcal{N} \setminus \{a\}} \hat{p}(\mathbf{x}_j, \mathbf{y}) \prod_{k \in \mathcal{N} \setminus \{a\} \setminus \{j\}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big) \\
& - \big(1 - \hat{p}(\mathbf{x}_a, \mathbf{y})\big) \sum_{\mathcal{A} \in Comb(\mathcal{N} \setminus \{a\}, 2)} \prod_{j \in \mathcal{A}} \hat{p}(\mathbf{x}_j, \mathbf{y}) \prod_{k \in \mathcal{N} \setminus \{a\} \setminus \mathcal{A}} \big(1 - \hat{p}(\mathbf{x}_k, \mathbf{y})\big)
\end{aligned} \tag{16}$$

Applying (9) to (16) yields:

$$\begin{aligned}
\Phi^{3^+}(\mathbf{X}_{\mathcal{N}}, \mathbf{y}) &= 1 - \Phi^0(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) - \Phi^1(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) - \Phi^2(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) \big(1 - \hat{p}(\mathbf{x}_a, \mathbf{y})\big) \\
&= \Phi^{3^+}(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) + \Phi^2(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) \hat{p}(\mathbf{x}_a, \mathbf{y})
\end{aligned} \tag{17}$$

Rewriting (17) yields:

$$\begin{aligned}
\Phi^{3^+}(\mathbf{X}_{\mathcal{N}}, \mathbf{y}) &= \Phi^{3^+}(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) - \hat{p}(\mathbf{x}_a, \mathbf{y}) \Phi^{3^+}(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) + \hat{p}(\mathbf{x}_a, \mathbf{y}) \Phi^{2^+}(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) \\
&= (1 - \hat{p}(\mathbf{x}_a, \mathbf{y})) \Phi^{3^+}(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y}) + \hat{p}(\mathbf{x}_a, \mathbf{y}) \Phi^{2^+}(\mathbf{X}_{\mathcal{N} \setminus \{a\}}, \mathbf{y})
\end{aligned} \tag{18}$$

It is clear that the probability of the point $\mathbf{y}$ being covered can be seen on as an interpolation between two probability measures with the local probability of agent a as the interpolation variable. Low local probability of agent a means that the probability of coverage supplied the swarm as a whole depends more on the coverage supplied by the swarm excluding agent a. In the extreme case where the local probability of agent a is zero, the probability of covering $\mathbf{y}$ depends only on the coverage supplied by the swarm excluding agent a.

Higher local probability of agent a means that the contribution of a towards covering the point $\mathbf{y}$ is greater, thus less weight is put on the the probability of the swarm excluding agent a covering the point. Instead more weight is put on the probability of at least *two* other agents being able to communicate with an entity at $\mathbf{y}$. This is due to the fact that the probability of agent a being able to communicate with said entity is higher, and we need only two or more other agents to be able to communicate with the entity at $\mathbf{y}$ to make the total number of agents covering $\mathbf{y}$ three or more.

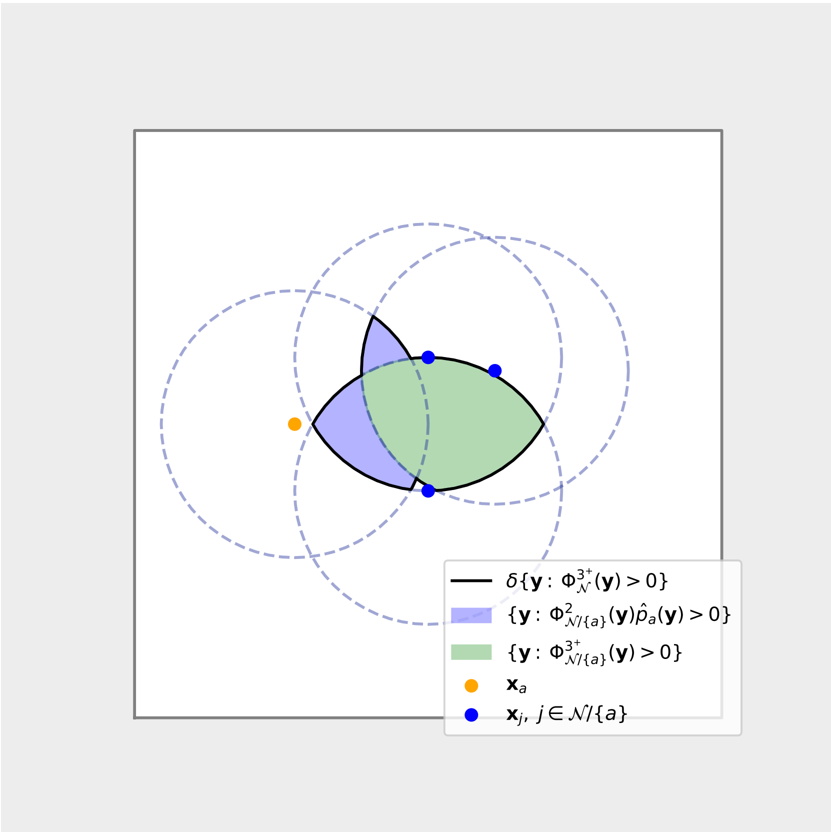


Figure 5: Non-zero regions of integrads in (19). Note that perturbing the orange circle (position of agent a) does not affect the green region, as it is defined only by the intersections of the disks surrounding the blue points (other agents in the swarm).

We note that the overall probability of coverage over the feasible space can be written as:

$$
\begin{aligned}
P(\mathbf{X}_{\mathcal{N}}) &= \int_{\mathcal{F}} \Phi^{3^+}(\mathbf{X}_{\mathcal{N}}, \mathbf{y})d\mathbf{y} = \int_{\mathcal{F}} \Phi^{3^+}(\mathbf{X}_{\mathcal{N}\setminus\{a\}}, \mathbf{y}) + \Phi^2(\mathbf{X}_{\mathcal{N}\setminus\{a\}}, \mathbf{y})\hat{p}(\mathbf{x}_a, \mathbf{y})d\mathbf{y} \\
&= \int_{\mathcal{F}} \Phi^{3^+}(\mathbf{X}_{\mathcal{N}\setminus\{a\}}, \mathbf{y})d\mathbf{y} + \int_{\mathcal{F}} \Phi^2(\mathbf{X}_{\mathcal{N}\setminus\{a\}}, \mathbf{y})\hat{p}(\mathbf{x}_a, \mathbf{y})d\mathbf{y}
\end{aligned} \tag{19}
$$

We note that the first term in (19) is independent of the position of a in both its domain and integrand. This independence is visualized in Figure 12. Thus we can rewrite the overall coverage probability over $\mathcal{F}$ as:

$$
P(\mathbf{X}_{\mathcal{N}}) = P(\mathbf{X}_{\mathcal{N}\setminus\{a\}}) + P_a(\mathbf{X}_{\mathcal{N}}) \tag{20}
$$

Where the *local* probability of coverage for agent a over the feasible space is defined as:

$$
P_a(\mathbf{X}_{\mathcal{N}}) = \int_{\mathcal{F}} \Phi^2(\mathbf{X}_{\mathcal{N}\setminus\{a\}}, \mathbf{y})\hat{p}(\mathbf{x}_a, \mathbf{y})d\mathbf{y} \tag{21}
$$

As in [9] we note that from the viewpoint of agent a, the swarm can be partitioned into three disjoint sets: $\{a\}$, $\mathcal{B}_a$ and $\mathcal{C}_a$. The latter sets are defined as:

$$
\mathcal{B}_a = \{j \in \mathcal{N} \setminus \{a\} : \|\mathbf{x}_a - \mathbf{x}_j\| \leq 2r\} \tag{22a}
$$

$$
\mathcal{C}_a = \{j \in \mathcal{N} \setminus \{a\} : \|\mathbf{x}_a - \mathbf{x}_j\| > 2r\} \tag{22b}
$$

The set $\mathcal{B}_a$, from now on called the neighbours of a, contains all agents in the swarm, $\mathcal{N}$, whose communication disks form a non-empty intersection with that of a. $\mathcal{C}_a$ contains all agents whose communication disks do not intersect with that of a.

Applying (22) to (21) yields:

$$
\begin{aligned}
P_a(\mathbf{X}_{\mathcal{N}}) &= \int_{\mathcal{F}} \Phi^2(\mathbf{X}_{\mathcal{N}\setminus\{a\}}, \mathbf{y})\hat{p}(\mathbf{x}_a, \mathbf{y})d\mathbf{y} \\
&= \int_{\mathcal{F}} \Big(\Phi^2(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y}) + \Phi^2(\mathbf{X}_{\mathcal{C}_a}, \mathbf{y}) + \Phi^1(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y})\Phi^1(\mathbf{X}_{\mathcal{C}_a}, \mathbf{y})\Big)\hat{p}(\mathbf{x}_a, \mathbf{y})d\mathbf{y}
\end{aligned} \tag{23}
$$

Partitioning the domain of integration into the visible set and invisible set of agent a, and noting that $\hat{p}(\mathbf{x}_j, \mathbf{y}) = 0 \ \forall \ j \in \mathcal{C}_a, \ \mathbf{y} \in V_a$ such that $\Phi^n(\mathbf{X}_{\mathcal{C}_a}, \mathbf{y}) = 0 \ \forall \ n \in \mathbb{Z}^+, \ \mathbf{y} \in V_a$, and $\hat{p}(\mathbf{x}_a, \mathbf{y}) = 0 \ \forall \ \mathbf{y} \in U_a$ yields:

$$
\begin{aligned}
P_a(\mathbf{X}_{\mathcal{N}}) &= \int_{V_a} \Big(\Phi^2(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y}) + \Phi^2(\mathbf{X}_{\mathcal{C}_a}, \mathbf{y}) + \Phi^1(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y})\Phi^1(\mathbf{X}_{\mathcal{C}_a}, \mathbf{y})\Big)\hat{p}(\mathbf{x}_a, \mathbf{y})d\mathbf{y} \\
&+ \int_{U_a} \Big(\Phi^2(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y}) + \Phi^2(\mathbf{X}_{\mathcal{C}_a}, \mathbf{y}) + \Phi^1(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y})\Phi^1(\mathbf{X}_{\mathcal{C}_a}, \mathbf{y})\Big)\hat{p}(\mathbf{x}_a, \mathbf{y})d\mathbf{y} \\
&= \int_{V_a} \Phi^2(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y})p(\|\mathbf{x}_a - \mathbf{y}\|)d\mathbf{y} = L(\mathbf{X}_{\mathcal{B}_a \cup \{a\}})
\end{aligned} \tag{24}
$$

Thus the local probability of coverage for an agent a is dependent on the position, $\mathbf{x}_a$, of agent a in both domain and integrand, and the positions of the neighbours of agent a. We call the swarm consisting of a and its neighbours agent a's local swarm.

We measure the proximity of two agents, a and j, by:

$$
D(\mathbf{x}_a, \mathbf{x}_j) = k_1 e^{-k_2\|\mathbf{x}_a - \mathbf{x}_j\|} \tag{25}
$$

The function has two tunable parameters, k_1 and k_2, whose effect is show in Figure 6.

Using (24) and (25) We now state the *local* objective for an agent a as:

$$H(\mathbf{X}_{\mathcal{B}_a \cup \{a\}}) = L(\mathbf{X}_{\mathcal{B}_a \cup \{a\}}) - \sum_{j \in \mathcal{B}_a} D(\mathbf{x}_a, \mathbf{x}_j) \tag{26}$$

where the second term is added to ensure sufficient spread of agents.

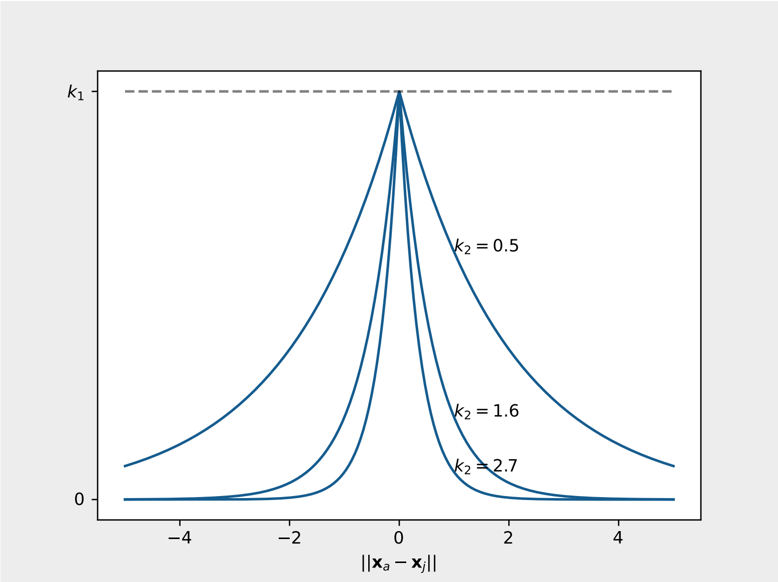


Figure 6: Proximity term in (27) for two agents.

5.5 Optimization problem formulation

Using (26) and the constraints mentioned in 5.3, we define the optimization problem as:

$$\begin{aligned} \max_{\mathbf{x}_a} \; & H(\mathbf{X}_{\mathcal{B}_a \cup \{a\}}) \\ \text{s.t. } & \mathbf{x}_a \in \mathcal{F} \\ & |\mathcal{B}_a| \geq 2 \\ & \|\mathbf{x}_a - \mathbf{x}_j\| \geq r_{min} \; \forall \, j \in \mathcal{B}_a \\ & \text{Rank}(\mathbf{V}) = 2 \\ & \mathbf{V} = \begin{bmatrix} \mathbf{x}_a - \mathbf{X}_{\mathcal{B}_a,0} & \dots & \mathbf{x}_a - \mathbf{X}_{\mathcal{B}_a,|\mathcal{B}_a|-1} \end{bmatrix} \end{aligned} \tag{27}$$

The second term of the objective in (27) penalizes the clustering of agents. In essence it works as anti-gravity between neighbours, as it will cause agent a to move away from it's neighbouring agents.

The equality constraints in (27) impose that agent a is not placed such that it lies on a straight line through all it's neighbours. Furthermore they implicitly impose that agent a has at least two neighbours at all times, as $\text{Rank}(\mathbf{V}) \leq \min(n, m)$ for a matrix $\mathbf{V} \in \mathbb{R}^{n \times m}$.

6 Implementation

6.1 Local probability

We assume that the agents have perfect communication capabilities within their maximum range. Thus we set

$$\hat{p}(\mathbf{x}_a, \mathbf{y}) = \begin{cases} 1, & \mathbf{y} \in V_a \\ 0, & \mathbf{y} \in U_a \end{cases} = 1_{\left\{\mathbf{y} \in V_a\right\}} \tag{28}$$

where $1_{\{\cdot\}}$ is the indicator function, which is simply equal to one if the clause in the subscript is true and zero otherwise. This implies that the overall probability of coverage over the feasible space in (19) is simply the area of all intersctions of three or more visible sets.

6.2 Computing the local objective function

We partition the neighbours of a into two sets:

$$\mathcal{B}_{aV} = \{j \in \mathcal{B}_a : \mathbf{y} \in V_j\} \tag{29a}$$

$$\mathcal{B}_{aU} = \{j \in \mathcal{B}_a : \mathbf{y} \in U_j\} \tag{29b}$$

Thus for a given point $\mathbf{y}$, the set $\mathcal{B}_{aV}$ contains all neighbours of a whose visible set contains $\mathbf{y}$, and $\mathcal{B}_{aU}$ contains all neighbours of agent a whose visible set does not contain $\mathbf{y}$.

Now the local probability of coverage for agent a can be written as:

$$\begin{aligned} L(\mathbf{X}_{\mathcal{B}_a \cup \{a\}}) &= \int_{V_a} \Phi^2(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y}) 1_{\{\mathbf{y} \in V_a\}} d\mathbf{y} = \int_{V_a} \Phi^2(\mathbf{X}_{\mathcal{B}_a}, \mathbf{y}) d\mathbf{y} \\ &= \int_{V_a} \sum_{n=0}^{2} \Phi^n(\mathbf{X}_{\mathcal{B}_{aV}}, \mathbf{y}) \Phi^{2-n}(\mathbf{X}_{\mathcal{B}_{aU}}, \mathbf{y}) d\mathbf{y} \\ &= \int_{V_a} \sum_{n=0}^{2} 1_{\{|B_{aV}|=n\}} 1_{\{2-n=0\}} d\mathbf{y} \\ &= \int_{V_a} 1_{\{|B_{aV}|=2\}} d\mathbf{y} \end{aligned} \tag{30}$$

Thus the value of the local objective function is simply the area where the visible set of a overlaps with those of excatly two neighbouring agents.

7 Simulation Results

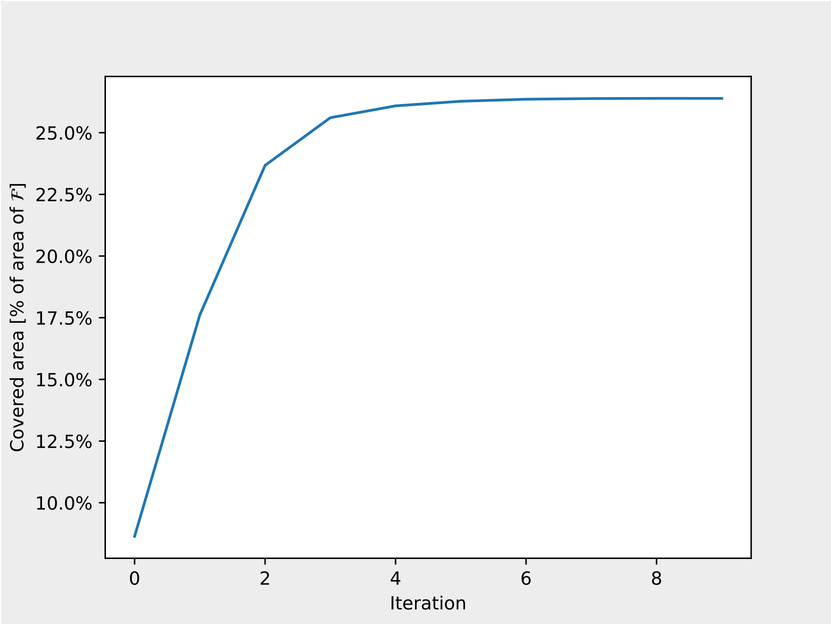


Figure 7: Percentage covered area of $\mathcal{F}$ vs. iteration count for rectangular obstacle-less environment. Area of $\mathcal{F} = 100\,\mathrm{m}^2$

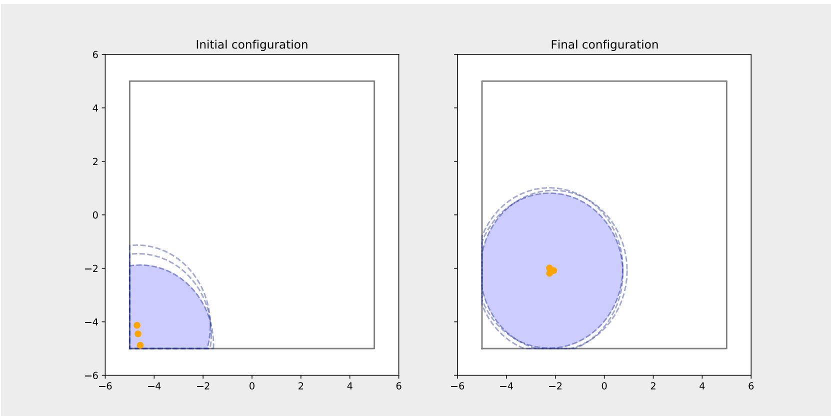


Figure 8: Inital and final position of agents in rectangular obstacle-less environment. Area of $\mathcal{F} = 100\,\mathrm{m}^2$, maximum communication range $r = 3\,\mathrm{m}$.

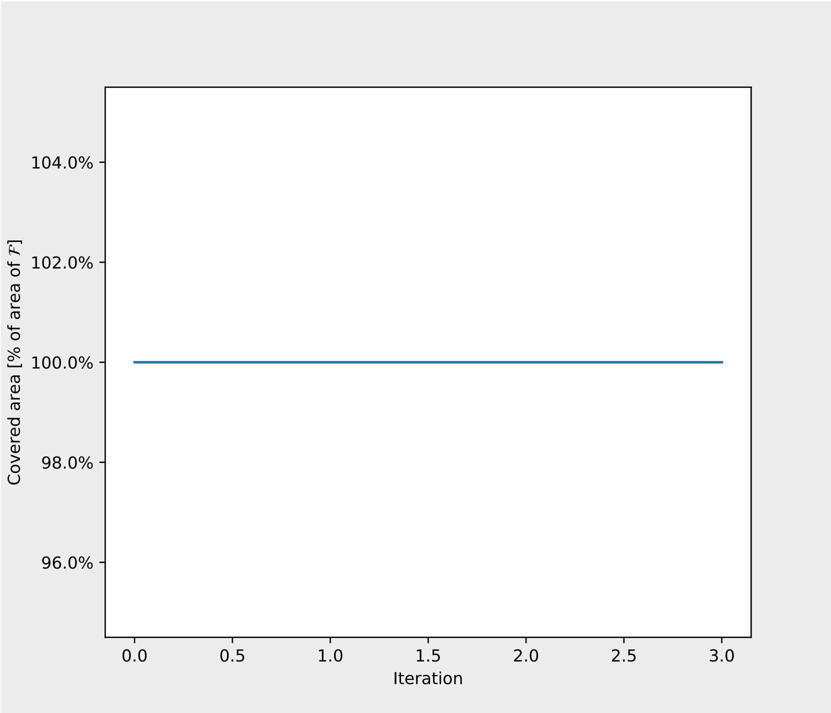


Figure 9: Percentage covered area of $\mathcal{F}$ vs. iteration count for rectangular obstacle-less environment. Area of $\mathcal{F} = 4\,\mathrm{m}^2$

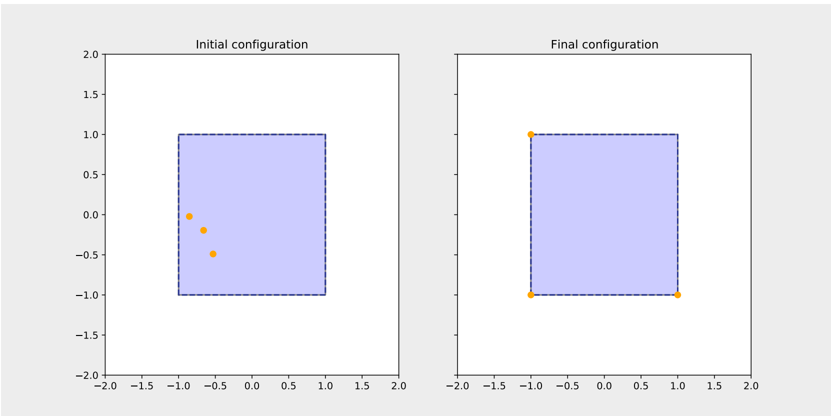


Figure 10: Inital and final position of agents in rectangular obstacle-less environment. Area of $\mathcal{F} = 4\,\mathrm{m}^2$, maximum communication range $r = 9\,\mathrm{m}$.

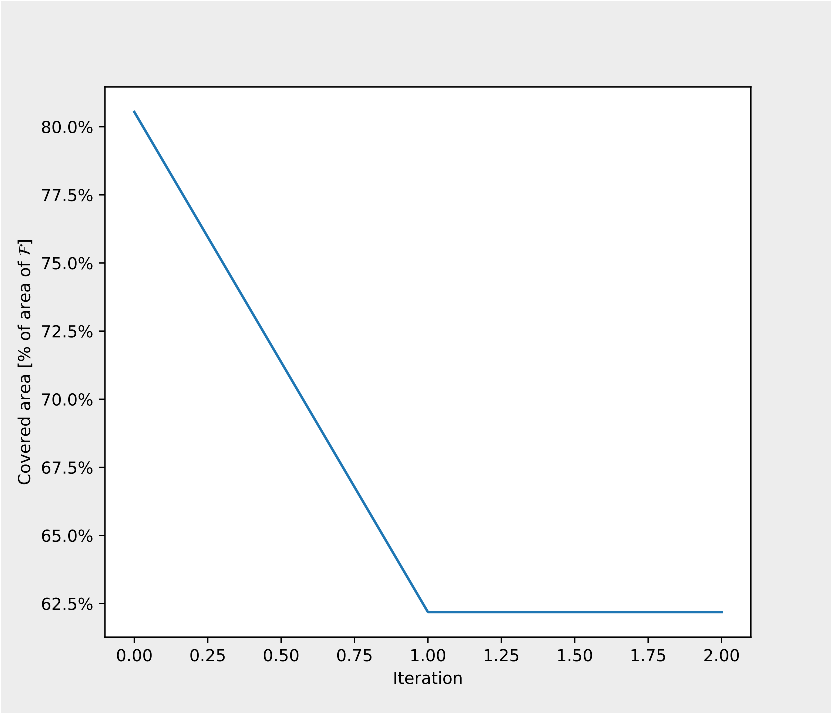


Figure 11: Percentage covered area of $\mathcal{F}$ vs. iteration count for rectangular environment with rectangular central obstacle. Area of $\mathcal{F} = 4\,\mathrm{m}^2$

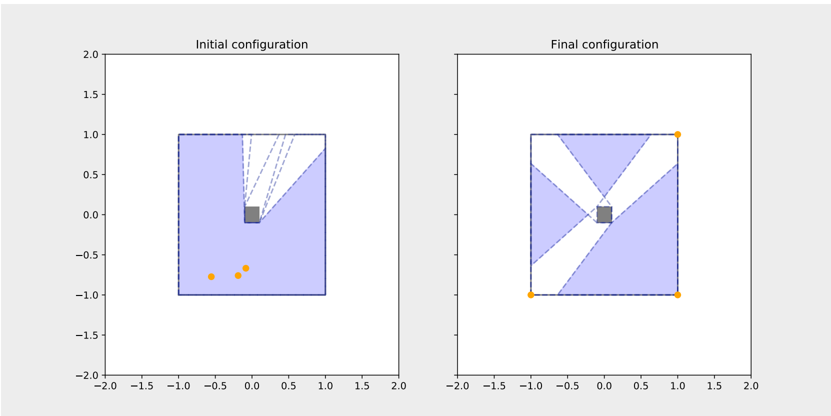


Figure 12: Inital and final position of agents in rectangular environment with rectangular central obstacle. Area of $\mathcal{F} = 4\,\mathrm{m}^2$, maximum communication range $r = 9\,\mathrm{m}$.

8 Discussion

9 Future Work

References

[1] M.-W. Dictionary, "First responder," accessed 19 Nov. 2020. [Online]. Available: https://www.merriam-webster.com/dictionary/first%20responder

[2] M. Gavhale and P. D. Saraf, "Survey on algorithms for efficient cluster formation and cluster head selection in manet," *Procedia Computer Science*, vol. 78, pp. 477 – 482, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050916000934

[3] M. Elbanhawi, A. Mohamed, R. Clothier, J. Palmer, M. Simic, and S. Watkins, "Enabling technologies for autonomous mav operations," *Progress in Aerospace Sciences*, vol. 91, pp. 27 – 52, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0376042116300367

[4] D. Anderson, "An evaluation of current and future costs for lithium-ion batteries for use in electrified vehicle powertrains," Master's project, Duke University, 2009. [Online]. Available: https://hdl.handle.net/10161/1007

[5] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 15–22.

[6] M. Faessler, F. Fontana, C. Forster, E. Mueggler, M. Pizzoli, and D. Scaramuzza, "Autonomous, vision-based flight and live dense 3d mapping with a quadrotor micro aerial vehicle," *Journal of Field Robotics*, vol. 33, no. 4, pp. 431–450, 2016. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21581

[7] M. C. De Gennaro and A. Jadbabaie, "Decentralized control of connectivity for multi-agent systems," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006, pp. 3628–3633.

[8] M. Zhong and C. G. Cassandras, "Distributed coverage control and data collection with mobile sensor networks," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 5604–5609.

[9] X. Sun, C. G. Cassandras, and K. Gokbayrak, "Escaping local optima in a class of multi-agent distributed optimization problems: A boosting function approach," 2014.

[10] E. W. Weisstein, "Simple polygon. From MathWorld–A Wolfram Web Resource," last visited on 10/11/2020. [Online]. Available: https://mathworld.wolfram.com/SimplePolygon.html

[11] Y. H. Wang, "On the number of successes in independent trials," *Statistica Sinica*, vol. 3, no. 2, pp. 295–312, 1993. [Online]. Available: http://www.jstor.org/stable/24304959

[12] C. Wolff, "Multilateration," last visited on 13.11.2020. [Online]. Available: https://www.radartutorial.eu/02.basics/rp52.en.html

[13] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '01. New York, NY, USA: Association for Computing Machinery, 2001, p. 166179. [Online]. Available: https://doi.org/10.1145/381677.381693