2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Multi-Agent Distributed and Decentralized Geometric Task Allocation

Michael Amir, Yigal Koifman, Yakov Bloch,
Ariel Barel, and Alfred M. Bruckstein

***Abstract*— We consider the general problem of geometric task allocation, wherein a large, decentralized swarm of simple mobile agents must detect the locations of tasks in the plane and position themselves nearby. The tasks are represented by an *a priori unknown* demand profile $\Phi(x,y)$ that determines how many agents are needed in each location. The agents are autonomous, oblivious, indistinguishable, and have a finite sensing range. They must configure themselves according to Φ using only local information about Φ and about the positions of nearby agents. All agents act according to the same local sensing-based rule of motion, and cannot explicitly communicate nor share information.**

We propose an approach based on gradient descent over a simple squared error function. We formally show that this approach results in attraction-repulsion dynamics. Repulsion encourages agents to spread out and explore the region to find the tasks, and attraction causes them to accumulate at task locations. The figures in this work are snapshots of simulations that can be viewed at `https://youtu.be/1_5f0MnUJag`.

I. INTRODUCTION

This work explores the topic of deploying a robotic swarm of autonomous mobile agents over a region to locate and carry out an *a priori unknown* set of tasks. The spatial location of the tasks and the number of agents required to complete them are not given to the agents in advance, and may even change over time. The goal of the agents is to find the tasks and to position themselves in the region based on the requirements of each task. The agents must also relocate in response to changes in the set of tasks - e.g., agents that complete a given task should go on to help other agents complete their tasks. Examples of this kind of setting include search and rescue missions, where agents must find and assist an unknown number of people, or forest fires, where the spread and intensity of fire evolves over time and requires varying numbers of firefighting drones to cover.

The problems and solutions considered in this work are motivated by common assumptions made in the field of swarm robotics. The objective of swarm robotics is to coordinate a robotic task-force made up of a very large number of simple mobile agents. The agents are assumed to be disposable and redundant: there are more than enough of them to satisfy the demands of all tasks even if some should crash or become lost. Swarm robotics is uniquely positioned to handle task allocation in unknown environments, because the agents can quickly cover a very large area to locate the tasks, and because there are enough agents that we need not worry about some of the agents not finding a task to work on. The main goals of this work are:

1) To describe the general problem setting of geometric task allocation for swarms of simple autonomous agents with limited sensing range.

2) To study several different task allocation problems in this setting.

3) To make the observation that sensing range limitations can be overcome by gradient descent over a simple "error function", which results in local Attraction-Repulsion Dynamics (ARD).

The model. Assume $\mathscr{N}$ identical mobile agents are initiated at arbitrary locations within some closed subregion $\mathscr{L}$ of the plane $\mathbb{R}^2$ (such as the unit square $\mathscr{L} = [0,1] \times [0,1]$) and are able to move about $\mathbb{R}^2$ freely. The agents seek to organize themselves within $\mathbb{R}^2$ in a manner determined by a *demand profile* $\Phi(x,y)$ representing the requirements of tasks. $\Phi(x,y)$ is assumed to be positive inside $\mathscr{L}$ and 0 outside of it. Different kinds of demand profiles may be considered: for example, $\Phi(x,y)$ could indicate the required number of agents near position (x,y), or $\Phi(x,y)$ could be a probability density function representing the *proportion* of agents that should be near (x,y) (as in Cortes et al. [8]), or $\Phi(x,y)$ could be some heat map that needs to be "covered" by agents (as in the signal coverage problem described below).

Let us denote the position of the ith agent at time t as $\vec{p}_i(t) = (x_i(t), y_i(t))$, and define:

$$\vec{\mathbf{q}}(t) = (x_1(t), y_1(t), x_2(t), y_2(t), \ldots x_{\mathscr{N}}(t), y_{\mathscr{N}}(t)) \quad (1)$$

To determine how well the agents satisfy a given demand profile, we define an error function $\Psi(x,y,\vec{\mathbf{q}})$ based on Φ, which measures the degree to which the demand Φ is not satisfied at point (x,y) given the agents' current positions $\vec{\mathbf{q}}$. The agents' goal is to move to a position $\vec{\mathbf{q}}$ that minimizes the total error over all locations:

$$\min_{\vec{\mathbf{q}}} \iint_{\mathbb{R}^2} \Psi(x,y,\vec{\mathbf{q}})\, \mathrm{d}x\, \mathrm{d}y \quad (2)$$

Agents have no common frame of reference, are not aware of distant agents' positions, and do not know the entire demand profile Φ in advance. Instead, we assume agents possess local sensing capabilities, such that each agent can sense other agents within a finite distance V_A of itself and knows their location relative to itself, and each agent can sense the values of Φ within distance V_A from itself.

Time is discrete. At every time step $t = 0,1,2,\ldots$, the agents make some small discrete jump in any direction based on what they sense. The distance an agent can move in a

The authors are with the Department of Computer Science, Technion Israel Institute of Technology, Haifa, Israel. E-mails: {ammicha3, igal.k, freddy}@cs.technion.ac.il, arielba@technion.ac.il

2023 62nd IEEE Conference on Decision and Control (CDC) | 979-8-3503-0124-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/CDC49753.2023.10383740

single time step is assumed to be bounded by a parameter $\Delta > 0$. We assume the agents are *oblivious*, meaning they have no memory of actions and computations from previous time steps [5], and we assume agents cannot explicitly communicate with each other. Hence, at a given point in time t, each agent must determine its next move based only on the information that it currently senses.

While agents can (locally) sense the demand profile Φ, they cannot sense the error function Ψ. Whereas Φ determines the environmental information that agents can sense about their tasks, Ψ determines the agents' overall objective, i.e., the type of task allocation problem they must solve. The main task allocation problems considered in this work are:

1) **Signal coverage**, wherein each agent is surrounded by a radial influence profile called its "signal" that represents, e.g., its influence on nearby points, and agents must minimize the squared error between their combined influence profiles and Φ, hence "covering" Φ.

2) **Target assignment**, a highly related problem wherein a finite number of discrete targets (e.g., search and rescue tasks) are placed at unknown locations in the region, and each location requires some predefined number of agents in its vicinity.

For both of these problems, our goal is to find a corresponding error function and a local rule of motion that minimizes the total error. This problem is a basic distributed and decentralized task allocation issue, and similar problems are often discussed in the literature. However, we could not find any prior work that addresses this issue under the combined assumptions of local sensing, decentralized decision making, and no explicit inter-agent communication. While sharing information among the agents may lead to more optimal solutions, agents with communication are more complex and expensive; moreover, local communication may not always be feasible due to environmental conditions.

Our conservative assumptions about the agents' capabilities result in algorithms that are easy to implement and are resilient to sudden changes or crashes: removing agents from the system, or altering the demand profile Φ in real time (e.g., due to changes in the environment or set of tasks) does not break the algorithms, and agents correctly reconfigure their constellation. The agents can thus seamlessly react to changes in the environment or the tasks. As an example, Fig 1 shows a target assignment scenario where agents, starting in the middle of the region, search for tasks placed randomly within the region via ARD. Each task demands a different number of agents to complete. When a sufficient number of agents reach a task, the task is completed and removed from Φ, freeing agents to move to other tasks.

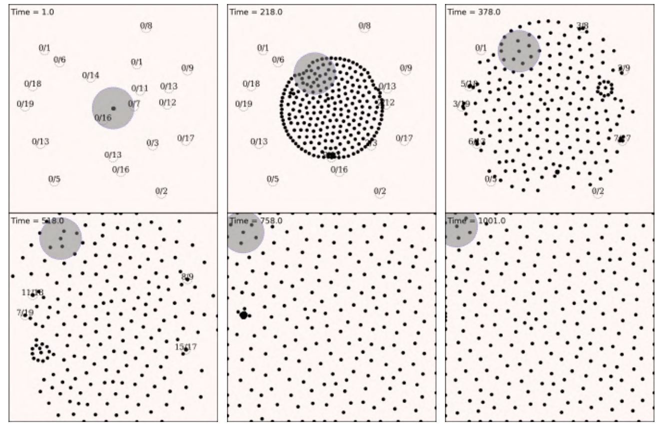

Fig. 1: Simulation of a target assignment scenario where targets disappear upon reaching the demanded number of agents. Agents move according to the electrostatic attraction-repulsion dynamics outlined in (17). The current time step is written in the top left corner of each frame. Agents' sensing range is depicted by the transparent gray disk (sensing range is depicted for a single agent to prevent visual clutter). 200 agents begin at the center of the region and expand outward, finding targets. The initial total demand of the targets is 204.

II. Related Work

The field of non-communicating, oblivious, visibility-constrained agents is referred to as *ant-like swarm robotics*. For further reading on ant-like swarm robotics, we refer the readers to the surveys [5]. Although the literature on swarm task allocation and task-based positioning is varied and deep, covering centralized [25], [17], decentralized [21], [20], [1], [15], [16], and non-geometric [29], [18], [22] models, we could not find any prior work that combined all our assumptions of local sensing, decentralized, oblivious decision making, and no explicit communication. These prior works thus take advantage of assumptions and methods that are not available to us, such as centralized trajectory planning, prior knowledge of the world, or direct inter-agent communication.

The closest works to ours that we could find are works in dynamical systems along the lines of Gazi et al. [10], [11], Cortes et al. [8] and Schwager et al. [28]. Our work differs from these due to our sensing range constraints and the different kinds of task allocation problems we explore, such as signal coverage, and in several theoretical aspects which we outline here. In works by Gazi et al. [10], [11], as well as related works such as [31], [7], various types of attraction-repulsion functions for stable swarm aggregation are considered, occasionally alongside a demand profile whose gradient guides the agents' dynamics. In contrast to Gazi et al. we do not assume Attraction Repulsion potentials but show how they arise from minimizing the squared error of a difference of demand profiles. In [27], [8] demand profile functions similar to our Φ are used to guide decentralized agents toward a desirable density distribution and deployment formation based on Voronoi diagrams. These works often represent environmental data as a multiplicative function that weighs the global importance of a given point in the region with regard to the agents' objective. In contrast, our work treats the demand profile, Φ, as a feature of the agents' sensing data on which the agents perform local computations. Our approach is applicable to limited-visibility settings, where agents have insufficient information to gauge the global importance of a point in the region.

Halftoning is a graphical technique that attempts to approximate a continuous image using sets of black dots [12]. The image is represented by a $2D$ function $\Phi(x,y)$. In electrostatic halftoning, the points are treated as particles that repel each other, and $\Phi(x,y)$ determines the local magnitude of a field of attracting forces [13], [26], [30]. The sum of these repelling and attracting forces causes the particles to position themselves so as to approximate the image expressed by Φ - a task-allocation-like problem. Such models assume that an agent can feel the forces exerted by Φ and other agents globally whereas we assume that agents are not aware of what happens beyond their visibility range V_A. However, it turns out that given a large enough number of agents, halftoning techniques can serve as effective task allocation algorithms even under such constraints - we explore this topic in Section III-B.2.

The concept of using a probability mass function to dictate the desirable density of agents at a given location has also been explored in such contexts as "Optimotaxis", Markov chain-based task allocation, and convex optimization [19], [9], [4]. However, unlike our present setting, these prior works assume either communication or global knowledge about the desired density $\Phi(x,y)$.

When $\Phi(x,y)$ is a constant function, our agents will spread uniformly inside $\mathscr{L}$, thus achieving *uniform dispersion*. Various works have been written on the uniform dispersion of agents in unknown regions and, more broadly, the coverage of unknown regions via a small number of agents. These works often share our assumptions of limited visibility and no communication [3], [2], [23], and sometimes use potential fields [24], [14]. However, these works are not directly comparable to ours; the goal of coverage and/or uniform dispersion is rather different from the task of positioning agents according to a general and non-uniform, task allocation demand profile.

III. Multi-agent task allocation

We consider two task allocation problems for limited-visibility mobile agents. The first one is *signal coverage*, in which a swarm of mobile agents surrounded by sensing profile signals must organize in the region so that their combined signal approximates an *a priori unknown* demand profile Φ. The second related problem is *target assignment*, in which there are $\mathscr{K}$ targets at a priori unknown locations, and we want to bring a certain number of agents to the location of each target.

We approach these problems through gradient descent on an appropriate squared error function. We show that this gradient descent naturally leads to *ARD*, wherein agents repulse each other and are attracted to locations with high values of Φ. Inter-agent repulsive forces cause the swarm to expand uniformly, thus covering the region of interest $\mathscr{L}$ and discovering the demand profile Φ. Attractive forces, on the other hand, cause agents to accumulate according to tasks' requirements (e.g., in target assignment, we want agents to accumulate at the locations of targets, so we have these locations exert attractive forces). The number of agents needed to cover $\mathscr{L}$ depends on the sensing range V_A. When the disk of radius V_A is small compared to $\mathscr{L}$, a large number of agents is needed to execute this strategy, making it well-suited for swarm robotics.

Section III-A formally describes the signal coverage problem and the corresponding error function Ψ. We show that gradient descent over the total error $\mathscr{G}(\vec{\mathbf{q}}) = \iint_{\mathbb{R}^2} \Psi(x,y,\vec{\mathbf{q}})\,\mathrm{d}x\,\mathrm{d}y$ leads to an attraction-repulsion-based algorithm for signal coverage. One approach to the target assignment problem is representing it as a signal coverage problem where there are highly concentrated signals at the location of each target. Hence, III-A can also be applied to the target assignment problem. An alternative ARD-based approach to target assignment can be derived from Coulomb's law, by treating targets and agents as electrically charged particles. In III-B we relate both approaches to a general form of ARD. We also discuss *scalar field coverage*, wherein agents spread across and cover a desired scalar field, as an application of our approach.

A. Signal coverage and target assignment

In the *signal coverage problem*, each agent's position is surrounded by an influence signal $\tilde{d}(x,y)$ which represents a weighted radial region of interaction (e.g., the effectiveness of a foam cannon stationed at the agent's position on a fire at (x,y), or the agent's ability to gather data at coordinate (x,y)). The agents seek to position themselves in $\mathscr{L}$ so as to minimize the squared difference between their combined signal and an "environmental demand profile" $\Phi(x,y)$ (representing, e.g., the heat map of a forest fire, or the importance of collecting data at coordinate (x,y)). Formally, given Φ and $\tilde{d}$, we define the error function:

$$\Psi(x,y,\vec{\mathbf{q}}) = \big(\Phi(x,y) - \sum_{i=1}^{\mathscr{N}} \tilde{d}(x-x_i, y-y_i)\big)^2 \tag{3}$$

and our agents' goal is to minimize

$$\mathscr{G}(\vec{\mathbf{q}}) = \iint_{\mathbb{R}^2} \big(\Phi(x,y) - \sum_{i=1}^{\mathscr{N}} \tilde{d}(x-x_i, y-y_i)\big)^2 \,\mathrm{d}x\,\mathrm{d}y \tag{4}$$

Note that this optimization goal is different from that considered by Cortes et al. [8], wherein Φ is treated as a probability density function, and agents seek to minimize the mean squared error of a randomly sampled point from Φ to the closest agent. This leads to dynamics in which agents head toward the center of their respective cell in a (possibly weighted) Voronoi tessellation. In contrast, we minimize the squared error of sums of symmetric functions (representing the signals of agents and targets respectively), which leads to ARD and different optimal configurations from [8].

Intuitively, the signal coverage model is applicable when agents' signals can usefully be combined and required to complete a task at (x,y). For example, several noisy measurements of (x,y) by different agents can be combined to cancel out noise; several foam cannons at agents' positions can be combined to better affect a fire at point (x,y).

$\tilde{d}(x,y)$ is assumed to be a symmetric, almost everywhere differentiable function determined only by $r = \sqrt{x^2+y^2}$. In other words, there is a signal function $f(r)$ such that $\tilde{d}(x,y) = f(\sqrt{x^2+y^2})$. For some parameter V, we assume that $\tilde{d}(r) = 0$ for all $r > V$ (see Fig 2).

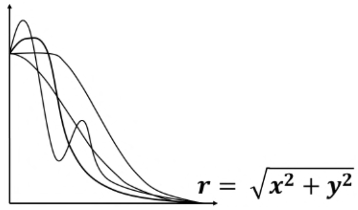

Fig. 2: Some possible choices of the signal function $f(r)$. Modifying f will alter the optimal agent signal coverage formation, hence also the agent dynamics.

For the time being, we generally restrict ourselves to the case where $\Phi(x,y)$ is a weighted sum of $\mathscr{K}$ signals of the same type as the agents'. The center of the kth signal is denoted as $\vec{c}_k = (x^S_k, y^S_k)$. Specifically, we define $\Phi(x,y) = 0$ for all $(x,y) \notin \mathscr{L}$, and otherwise define

$$\Phi(x,y) = 1 + \sum_{k=1}^{\mathscr{K}} n_k \tilde{d}(x - x^S_k, y - y^S_k) \tag{5}$$

where n_k is "the demand of the kth signal center". We assume that for all k, the disk of radius V centred at $\vec{c}_k$ is contained in $\mathscr{L}$.

The goal of this section is to show signal coverage can be attained through discrete ARD, in which at every time step the ith agent changes its position according to:

$$\begin{aligned} \vec{p}_i(t+1) &= \vec{p}_i(t) - \delta \frac{\vec{v}_i(\vec{\mathbf{q}}(t))}{\|\vec{v}_i(\vec{\mathbf{q}}(t))\|}, \text{where} \\ \vec{v}_i(\vec{\mathbf{q}}(t)) &= \sum_{j=1}^{\mathscr{N}} F(\|\vec{p}_i - \vec{p}_j\|)\overrightarrow{p_i p_j} - \sum_{k=1}^{K} n_k F(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k} \end{aligned} \tag{6}$$

Here, $F(\cdot)$ is some real-valued function derived from f and $\overrightarrow{uv}$ denotes the unit vector from u to v. Under these dynamics, every agent is attracted to the centers of the signals, but repulsed by other agents, by an amount that scales according to distance. Since our agents' sensing range is V_A, we will be interested in dynamics where $F(r) = 0$ for all $r > V_A$, since then each agent can always compute (6).

Target assignment. In the target assignment problem we assume there are K targets at locations $\vec{c}_1, \ldots \vec{c}_K \in \mathscr{L}$, and for k, $1 \le k \le K$, we want to bring $n_k \ge 1$ agents to the location $\vec{c}_k$. The targets' locations are not known to the agents in advance, and a target can only be detected by an agent at distance V_A or less. As we shall see, the degree to which $\tilde{d}(x,y)$ is concentrated near $(0,0)$ determines the diffusion of the agents around the centers of the signals. For example, when $\tilde{d}$ is heavily concentrated near $(0,0)$, we expect that in an optimal configuration there will be n_k agents very close to the center of the kth signal. Through this observation, signal coverage can be applied to the target assignment problem (see Fig 4).

In Section III-B.2 we describe a different approach to target assignment, based on Coulomb's law.

1) Gradient descent and Attraction Repulsion Dynamics: To motivate (6) as an algorithm for signal coverage, we consider a gradient descent-based rule of motion that enables the agents to minimize $\mathscr{G}(\vec{\mathbf{q}})$ while obeying the restrictions of our model. Our objective is to show that the descent dynamics can be rewritten as ARD of the form (6).

In gradient descent, the goal of each agent is to descend the gradient of $\mathscr{G}$ by a small amount. The continuous-time gradient descent dynamics for the ith agent are:

$$\frac{d\vec{p}_i}{dt} = -\vec{v}_i(\vec{\mathbf{q}}(t)), \text{where } \vec{v}_i = \begin{bmatrix} \frac{\partial \mathscr{G}}{\partial x_i} \\ \frac{\partial \mathscr{G}}{\partial y_i} \end{bmatrix} \tag{7}$$

Since we are working in discrete time dynamics, we may discretize this expression as:

$$\vec{p}_i(t+1) = \vec{p}_i(t) - \delta \frac{\vec{v}_i(\vec{\mathbf{q}}(t))}{\|\vec{v}_i(\vec{\mathbf{q}}(t))\|} \tag{8}$$

where $0 < \delta \le \Delta$ is some predefined constant. Hence, at every time step, the agents compute $\frac{\partial \mathscr{G}}{\partial x_i}$ and $\frac{\partial \mathscr{G}}{\partial y_i}$. Let us denote $\frac{\partial \tilde{d}(x - x_i, y - y_i)}{\partial x_i} = \tilde{d}_{x_i}$ and $\frac{\partial \tilde{d}(x - x_i, y - y_i)}{\partial y_i} = \tilde{d}_{y_i}$:

$$\begin{bmatrix} \frac{\partial \mathscr{G}(\vec{\mathbf{q}})}{\partial x_i} \\ \frac{\partial \mathscr{G}(\vec{\mathbf{q}})}{\partial y_i} \end{bmatrix} = 2 \iint_{\mathbb{R}^2} \left(\Phi(x,y) - \sum_{j=1}^{\mathscr{N}} \tilde{d}(x - x_j, y - y_j)\right) \begin{bmatrix} \tilde{d}_{x_i}(x,y) \\ \tilde{d}_{y_i}(x,y) \end{bmatrix} \mathrm{d}x\,\mathrm{d}y \tag{9}$$

Recalling that $\Phi(x,y) = 1 + \sum_{k=1}^{\mathscr{K}} n_k \tilde{d}(x - x^S_k, y - y^S_k)$, we split (9) into three summands:

(a) $2 \iint_{\mathbb{R}^2} \begin{bmatrix} \tilde{d}_{x_i}(x,y) \\ \tilde{d}_{y_i}(x,y) \end{bmatrix} \mathrm{d}x\,\mathrm{d}y$

(b) $2 \sum_{k=1}^{K} \iint_{\mathbb{R}^2} n_k \tilde{d}(x - x^S_k, y - y^S_k) \begin{bmatrix} \tilde{d}_{x_i}(x,y) \\ \tilde{d}_{y_i}(x,y) \end{bmatrix} \mathrm{d}x\,\mathrm{d}y$

(c) $-2 \iint_{\mathbb{R}^2} \left(\sum_{j=1}^{\mathscr{N}} \tilde{d}(x - x_j, y - y_j)\right) \begin{bmatrix} \tilde{d}_{x_i}(x,y) \\ \tilde{d}_{y_i}(x,y) \end{bmatrix} \mathrm{d}x\,\mathrm{d}y$

To compute (a), we first differentiate $\tilde{d}_{x_i}(x,y)$:

$$\begin{aligned} \tilde{d}_{x_i}(x,y) &= \frac{\partial}{\partial x_i} f(\sqrt{(x - x_i)^2 + (y - y_i)^2}) \\ &= -\dot{f}(\sqrt{(x - x_i)^2 + (y - y_i)^2}) \frac{x - x_i}{\sqrt{(x - x_i)^2 + (y - y_i)^2}} \end{aligned}$$

We infer that $\tilde{d}_{x_i}(x - x_i, y) = -\tilde{d}_{x_i}(x_i - x, y)$, and analogously $\tilde{d}_{y_i}(x, y - y_i) = -\tilde{d}_{x_i}(x, y_i - y)$. Hence (a) = $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$.

To compute the inner integral $\iint_{\mathbb{R}^2} \tilde{d}(x - x^S_k, y - y^S_k)\tilde{d}_{x_i}(x,y)\,\mathrm{d}xy$ in (b), we may assume a rotated frame of reference where $y_i = y^S_k$, and later return to the original frame of reference via the inverse rotation M_k^{-1}. We further apply the coordinate transforms $\bar{x} = x - x_i$, $\bar{y} = y - y_i$ to get:

$$\begin{aligned} &\iint_{\mathbb{R}^2} n_k \tilde{d}(x - x^S_k, y - y^S_k)\tilde{d}_{x_i}(x,y)\,\mathrm{d}x\,\mathrm{d}y \\ &= n_k \iint_{\mathbb{R}^2} \tilde{d}(\bar{x} - \|\vec{p}_i - \vec{c}_k\|, \bar{y})\tilde{d}_{x_i}(\bar{x} + x_i, \bar{y} + y_i)\,\mathrm{d}\bar{x}\,\mathrm{d}\bar{y} \\ &= -n_k \iint_{\mathbb{R}^2} f(\sqrt{(\bar{x} - \|\vec{p}_i - \vec{c}_k\|)^2 + \bar{y}^2})\dot{f}(\sqrt{\bar{x}^2 + \bar{y}^2}) \frac{\bar{x}}{\sqrt{\bar{x}^2 + \bar{y}^2}}\,\mathrm{d}\bar{x}\,\mathrm{d}\bar{y} \end{aligned} \tag{10}$$

We can therefore define a function $F(\cdot)$ such that $(10) \triangleq -n_k F(\|\vec{p}_i - \vec{c}_k\|)$. We have that $F(\|\vec{p}_i - \vec{c}_k\|) = 0$ when $\|\vec{p}_i -$

$\vec{c}_k\| \geq 2V$ (since in such cases $\tilde{d}(x-x_k^S, y-y_k^S)\tilde{d}_{x_i}(x,y)$ is 0 everywhere). We also see that:

$$\iint_{\mathbb{R}^2} n_k \tilde{d}(x-x_k^S, y-y_k^S)\tilde{d}_{y_i}(x,y)\,\mathrm{d}x\,\mathrm{d}y$$
$$= -n_k \iint_{\mathbb{R}^2} \tilde{d}(\bar{x}-(x_k^S - x_i), \bar{y}) \dot{f}(\sqrt{\bar{x}^2+\bar{y}^2}) \frac{\bar{y}}{\sqrt{\bar{x}^2+\bar{y}^2}}\,\mathrm{d}\bar{x}\,\mathrm{d}\bar{y} = 0$$

since the integrand is an odd function in $\bar{y}$. Returning to the original frame of reference, we see that $\iint_{\mathbb{R}^2} \tilde{d}(x-x_k^S, y-y_k^S)\tilde{d}_{x_i}(x,y)\,\mathrm{d}x\,\mathrm{d}y = F(\|\vec{p}_i - \vec{c}_k\|)M_k^{-1}\begin{bmatrix}1\\0\end{bmatrix} = F(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k}$, where $\overrightarrow{p_i c_k}$ is a unit vector from $\vec{p}_i$ to $\vec{c}_k$. Consequently,

$$\textbf{(b)} = -2\sum_{k=1}^{K} n_k F(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k}$$

meaning that **(b)** is a sum of vectors from the ith agent to each signal center at a distance V or less from it. By the same method, we can compute **(c)** and conclude that:

$$\vec{v}_i(\vec{\mathbf{q}}) = 2\left(\sum_{j=1}^{\mathcal{N}} F(\|\vec{p}_i - \vec{p}_j\|)\overrightarrow{p_i p_j} - \sum_{k=1}^{K} n_k F(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k}\right)$$

Since at time $t+1$ the ith agent updates its position to $\vec{p}_i(t) - \delta\frac{\vec{v}_i(\vec{\mathbf{q}})}{\|\vec{v}_i(\vec{\mathbf{q}})\|}$, we can interpret our agents' dynamics as given by a *ARD*-system, where the agents are pulled toward the signal centers and pushed away by nearby agents at a magnitude determined by $F(\cdot)$. Since $F(r)$ equals 0 whenever $r > 2V$, we see that the ith agent only needs visibility range $V_A = 2V$ to compute $\vec{v}_i(\vec{\mathbf{q}})$. Hence, setting $V = V_A/2$ guarantees that our agents can move according to the dynamics (8).

Example computation. Let us provide an explicit computation of $F(\cdot)$. Suppose $V_A = V = \infty$ and $f(r) = e^{-\Lambda r^2}$. Plugging f into (10) we see that:

$$F(r) = \iint_{\mathbb{R}^2} e^{-\Lambda((\bar{x}-r)^2+\bar{y}^2)}(2\Lambda e^{-\Lambda(\bar{x}^2+\bar{y}^2)}\bar{x})\mathrm{d}\bar{x}\mathrm{d}\bar{y}$$
$$= \sqrt{2\pi\Lambda}\cdot\frac{r\sqrt{\frac{\pi}{2}}e^{-\frac{1}{2}\Lambda r^2}}{2\sqrt{\Lambda}} = \frac{1}{2}\pi r e^{-\frac{1}{2}\Lambda r^2} \tag{11}$$

To account for agents with limited visibility, we can introduce the cut-off point $V_A/2$ to f such that $f(r) = e^{-\Lambda r^2}$ for all $r \leq V_A/2$ and $f(r) = 0$ for all $r > V_A/2$, and recompute (10). As discussed above, the resulting function will equal 0 for all $r > V_A$. When Λ is sufficiently large, $F(V_A)$ is very close to 0, so as a crude approximation we may set the cut-off point V_A directly as part of F, such that $F(r) = \frac{1}{2}\pi r e^{-\frac{1}{2}\Lambda r^2}$ for all $r \leq V_A$ and $F(r) = 0$ otherwise.

Increasing the parameter Λ results in a more concentrated signal function f, hence results in agents concentrating closer to the centers $\vec{c}_i$ - see Fig 3. As mentioned , when f is highly concentrated near 0, signal coverage can be used for target assignment. This is illustrated in Fig 4. We emphasize that this is just an example, and there are infinite possible choices of f leading to different dynamics. We also note that even when an analytic expression for (10) is not available, the agents' dynamics can be efficiently computed by caching numerical approximations of $F(r)$, since $F(r)$ is a one-dimensional function that equals 0 outside the interval $[0, V_A]$, it is non-expensive to approximate and cache.

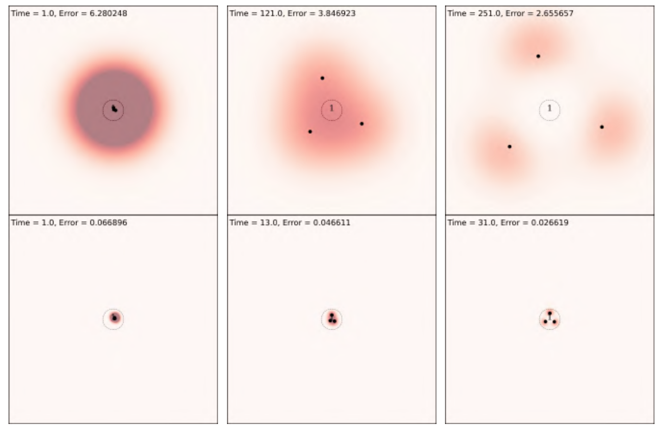

Fig. 3: The top row and bottom row depict two simulations of signal coverage attraction-repulsion dynamics with signal function $f(r) = e^{-\Lambda r^2}$. In the top row $\Lambda = 1$, and in the bottom row $\Lambda = 1000$. In both simulations, there is a single signal center with demand $n_1 = 1$. The heat map illustrates the values of the error function Ψ given the current agent positions. The rightmost (i.e., third) frame of each simulation shows the agent formation that minimizes the total error $\mathscr{G}$. Larger values of Λ cause the agents to concentrate closer to the signal center. Note that Ψ and $\mathscr{G}$ are different functions in the first and second simulation since they both depend on f.

Adding random noise. Depending on V_A and the size of the subregion $\mathscr{L}$, agents may be unable to see any other agent or target. An isolated agent acting on the dynamics outlined in (8) doesn't move. This is undesirable: we would like such agents to explore their environment and search for other tasks. To resolve this, we may add a stochastic component to the dynamics such that at every time step, the ith agent will move based on:

$$\vec{p}_i(t+1) = \vec{p}_i(t) - \delta\frac{\vec{v}_i(\vec{\mathbf{q}}(t))}{\|\vec{v}_i(\vec{\mathbf{q}}(t))\|} + \vec{r} \tag{12}$$

where $\vec{r}$ is a uniformly random vector of magnitude $\|r\| \leq \Delta - \delta$. Random noise has the effect of breaking deadlocks and causes isolated agents to keep exploring the environment. To prevent agents from wandering too far from $\mathscr{L}$ (i.e., the region of interest that contains the signals), if (12) would move an agent to a point where $\Phi(x,y) = 0$, the agent instead stays put.

2) Signal coverage for nonidentical profiles: Instead of having each signal center emit the same (weighted) signal $\tilde{d}$, we may define $\Phi(x,y)$ as a sum of different signals:

$$\Phi(x,y) = 1 + \sum_{k=1}^{\mathcal{K}} \tilde{d}_k(x-x_k^S, y-y_k^S) = 1 + \sum_{k=1}^{\mathcal{K}} f_k(\|\begin{bmatrix}x\\y\end{bmatrix} - \vec{c}_k\|)$$

where $\tilde{d}_k(x,y) = f_k(\|\begin{bmatrix}x\\y\end{bmatrix}\|)$ is a symmetric function representing the environmental signal centred at $\vec{c}_k$, and similar

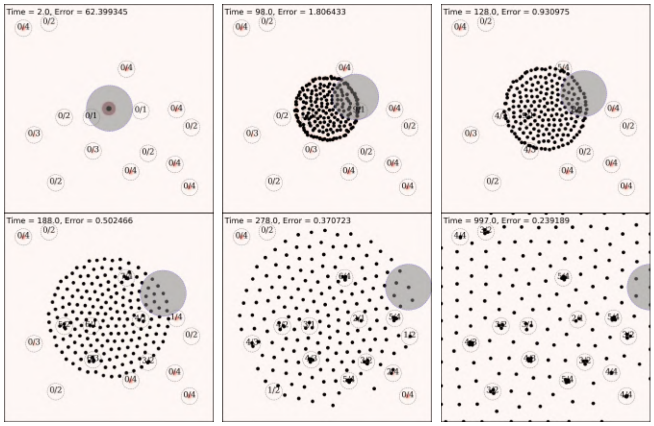

Fig. 4: Signal coverage attraction-repulsion dynamics for agents with limited sensing range. We set $f(r) = e^{-1000r^2}$ where $r < V_A/2$ and $f(r) = 0$ otherwise. The numbers on each signal denote the number of agents in its proximity, and the value of n_i. The gray disk depicts agents' visibility range. Since $f(r)$ is highly concentrated at 0, we see that n_i or more agents concentrate at the ith signal center, in effect completing a target assignment task. The heat map illustrates the values of Ψ. given the current agent positions. The time step and value of $\mathscr{G}$ are shown in the top left corner of each frame.

to before we assume $f_k(r) = 0$ for all $r \geq V$. We thus need to compute the gradient (9). Carrying out the same analysis as for the non-general case,we see that gradient descent on $\mathscr{G}$ leads to generalized ARD of the form:

$$\begin{aligned}\vec{p}_i(t+1) &= \vec{p}_i(t) - \delta \frac{\vec{v}_i(\vec{\mathbf{q}}(t))}{\|\vec{v}_i(\vec{\mathbf{q}}(t))\|} \\ \vec{v}_i(\vec{\mathbf{q}}) &= \sum_{j=1}^{\mathscr{N}} F(\|\vec{p}_i - \vec{p}_j\|)\overrightarrow{p_i p_j} - \sum_{k=1}^{K} F_k(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k}\end{aligned} \tag{13}$$

When $f_k = n_k f$ we get $F_k = n_k F$, reducing (13) to (6).

B. Generalized attraction-repulsion dynamics

In the previous subsection we gave a method for multi-agent signal coverage based on attraction-repulsion dynamics of the form (13). We showed that these dynamics arise naturally from gradient descent over an appropriate Ψ function. As an alternative way of thinking about these above dynamics, we can relate them to a general form of attraction-repulsion dynamics similar to that which appears in [11], where at every time step the ith agent moves according to the rule of motion:

$$\begin{aligned}\vec{p}_i(t+1) &= \vec{p}_i(t) - \delta \frac{\vec{v}_i(\vec{\mathbf{q}}(t))}{\|\vec{v}_i(\vec{\mathbf{q}}(t))\|} \\ \vec{v}_i(\vec{\mathbf{q}}) &= \sum_{j=1}^{\mathscr{N}} h(\|\vec{p}_i - \vec{p}_j\|)\overrightarrow{p_i p_j} - \frac{\partial}{\partial \vec{p}_i}\Phi(\vec{p}_i)\end{aligned} \tag{14}$$

in which $h(r) : \mathbb{R}_{\geq 0} \to \mathbb{R}$ is some function and $0 < \delta \leq \Delta$ is a constant. We can understand h as a repulsive force between agents, and the gradient of the demand profile Φ as an attractive force: at every time step the ith agent attempts to climb the gradient of Φ, but is repulsed by the jth agent by an amount dependent on their distance. Because the agents' sensing range is V_A, we assume that $h(r) = 0$ for all $r > V_A$.

The goal of this section is (i) to note that the dynamics (13) are a special case of (14) and (ii) to propose several alternative approaches to task allocation that can be recovered from this general form.

The dynamics of (13) are derived from gradient descent over the total error $\iint_{\mathbb{R}^2} \Psi(x,y,\vec{\mathbf{q}})\,\mathrm{d}x\,\mathrm{d}y$ where $\Psi(x,y,\vec{\mathbf{q}}) = \left(\Phi(x,y) - \sum_{i=1}^{\mathscr{N}} \tilde{d}(x-x_i, y-y_i)\right)^2$. We would be remiss if we didn't ask whether the dynamics (14) similarly minimize some error function, and whether they can similarly be justified through gradient descent. In other words, we would like to know if there exists Ψ such that the gradient $\frac{\partial \mathscr{G}}{\partial \vec{p}_i}(\vec{\mathbf{q}})$ of the total error $\mathscr{G}(\vec{\mathbf{q}}) = \iint_{\mathbb{R}^2} \Psi(x,y,\vec{\mathbf{q}})\,\mathrm{d}x\,\mathrm{d}y$ at $\vec{\mathbf{q}}$ equals the vector $\vec{v}_i(\vec{\mathbf{q}})$ in (14). Such a Ψ does exist:

$$\Psi(x,y,\vec{\mathbf{q}}) = \Big(-\frac{1}{2}\sum_{j=1}^{\mathscr{N}} H(\|\begin{bmatrix}x\\y\end{bmatrix} - \vec{p}_j\|) - \Phi(x,y)\Big)\sum_{i=1}^{\mathscr{N}} \delta(x-x_i)\delta(y-y_i) \tag{15}$$

where δ is the Dirac delta function and H is any almost everywhere differentiable function. The total error which the agents seek to minimize under this function, $\mathscr{G}(\vec{\mathbf{q}})$, equals:

$$\begin{aligned}&\iint_{\mathbb{R}^2}\Big(-\frac{1}{2}\sum_{j=1}^{\mathscr{N}} H(\|\begin{bmatrix}x\\y\end{bmatrix} - \vec{p}_j\|) - \Phi(x,y)\Big)\sum_{i=1}^{\mathscr{N}} \delta(x-x_i)\delta(y-y_i)\,\mathrm{d}x\,\mathrm{d}y \\ &= -\frac{1}{2}\sum_{1\leq i,j\leq \mathscr{N}} H(\|\vec{p}_i - \vec{p}_j\|) - \sum_{i=1}^{\mathscr{N}} \Phi(\vec{p}_i)\end{aligned}$$

hence the gradient at $\vec{\mathbf{q}}$ is

$$\frac{\partial \mathscr{G}}{\partial \vec{p}_i}(\vec{\mathbf{q}}) = \sum_{1\leq j\leq \mathscr{N}} \dot{H}(\|\vec{p}_i - \vec{p}_j\|)\overrightarrow{p_i p_j} - \frac{\partial}{\partial \vec{p}_i}\Phi(\vec{p}_i) \tag{16}$$

Defining $h(r) = \dot{H}(r)$, we see that the dynamics (14) are precisely discrete gradient descent dynamics over $\mathscr{G}$.

We now discuss several task allocation strategies that can be recovered as special cases of (14), i.e., through gradient descent over $\iint_{\mathbb{R}^2} \Psi(x,y,\vec{\mathbf{q}})\,\mathrm{d}x\,\mathrm{d}y$ for appropriate choices of H and Φ.

1) Signal coverage: We ask whether the dynamics (13) we derived for signal coverage can be recovered as a special case. Recall that (13) depends on two integrable functions: F and F_k. Let us set $H(r) = \int_0^r F(x)dx$, $H_k(r) = \int_0^r F_k(x)\,\mathrm{d}x$ and $\Phi(x,y) = -\sum_{k=1}^{K} H_k(\|\begin{bmatrix}x\\y\end{bmatrix} - \vec{c}_k\|)$. We compute that

$$\frac{\partial}{\partial \vec{p}_i}\Phi(\vec{p}_i) = -\sum_{k=1}^{K} \frac{\partial}{\partial \vec{p}_i} H_k(\|\vec{p}_i - \vec{c}_k\|)$$

$$= \sum_{k=1}^{K} F_k(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k}, \text{ which implies}$$

$$(16) = \sum_{j=1}^{\mathscr{N}} F(\|\vec{p}_i - \vec{p}_j\|)\overrightarrow{p_i p_j} - \sum_{k=1}^{K} F_k(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k}$$

hence, for these choices of H and Φ, (14) equals (13).

2) Electrostatic target assignment and Coulomb's law: In Section III-A we described the target assignment problem, and noted that it can be represented as signal coverage problems where Φ is highly concentrated at each target location. This led us to propose signal coverage as a method for target assignment. Another approach to target assignment is based on Coulomb's law, which states the attraction or repulsion forces between two point electric charges q_1, q_2 is kq_1q_2/r^2, where r is the distance between the charges, and k is a scaling factor. The idea is to assign each agent a positive charge, creating repulsion between pairs of agents, and each stationary target task a negative charge, creating attraction between agents and targets. Such an approach to particle control is widely used in image halftoning [13], [26], [30], in multi-robot collision avoidance and search and rescue [6], and more. Here we discuss this approach in the context of target assignment for large swarms of agents with limited visibility, and show that it is a special case of (13).

We begin with the case of infinite visibility where $V_A = \infty$, i.e., agents can sense other agents and targets at any distance. Let $H(r) = -\frac{1}{r}$ and $\Phi(x,y) = -\sum_{k=1}^{K} n_k H(\| \begin{bmatrix} x \\ y \end{bmatrix} - \vec{c}_k \|)$. Inserting these expressions into (16) we get

$$\begin{aligned} \frac{\partial \mathscr{G}}{\partial \vec{p}_i}(\vec{\mathbf{q}}) &= \sum_{1 \le i \le \mathscr{N}} \dot{H}(\|\vec{p}_i - \vec{p}_j\|)\overrightarrow{p_i p_j} - \sum_{k=1}^{K} n_k \dot{H}(\|\vec{p}_i - \vec{c}_k\|)\overrightarrow{p_i c_k} \\ &= \sum_{1 \le j \le \mathscr{N}} \frac{\overrightarrow{p_i p_j}}{\|\vec{p}_i - \vec{p}_j\|^2} - \sum_{k=1}^{K} n_k \frac{\overrightarrow{p_i c_k}}{\|\vec{p}_i - \vec{c}_k\|^2} \end{aligned} \tag{17}$$

This first-order dynamics (17) resembles Coulomb's force acting on targets with a positive charge of magnitude n_k and agents with a negative charge of magnitude 1.
If V_A is finite, we may define a cut-off point for H, making it so that agents are not affected by other agents and targets at distance greater than V_A, yielding $H_{V_A}(r) = \begin{cases} -\frac{1}{r} & r \le V_A \\ 0 & r > V_A \end{cases}$.

This results in dynamics that can be computed by agents with visibility range V_A.

Let us briefly relate the electrostatic target assignment approach to our previous, signal coverage-based approach: note that if we could find a signal f for which the integral (10) equalled $H_{V_A}(r)$, plugging this f into the ARD we used to solve signal coverage, (6), would precisely yield (17). However, it seems difficult to find such an f. Hence, although (17) is of a very similar form to (6), we leave open the question of whether it can be recovered as a special case of signal coverage. Fig 5 and 6 illustrate a target assignment task using electrostatic ARD.

3) Scalar field coverage: Moving away from target assignment, let us consider the problem of covering an arbitrary scalar field with agents (Figure 7). One way to do this is to let $\Phi(x,y)$ be our scalar field and model the repulsion forces of the agents the same as any of our proposed approaches to

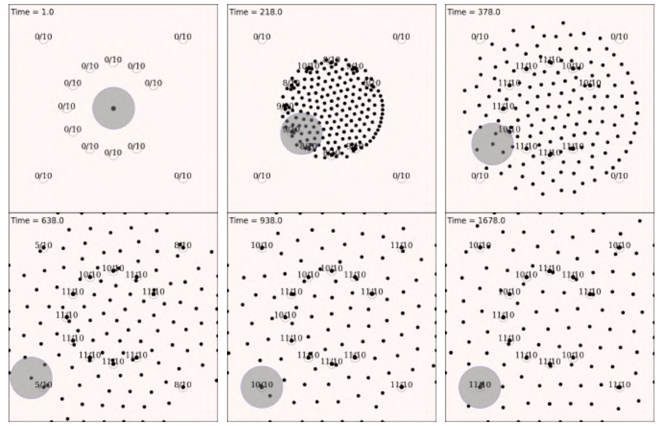

Fig. 5: Electrostatic target assignment. The number of agents demanded by each target and the number of agents in each target's proximity are depicted. Agent sensing range is illustrated by the gray disk. There are 250 agents, and the total demand of the targets is 140. Agents are all initiated at the center of the region.

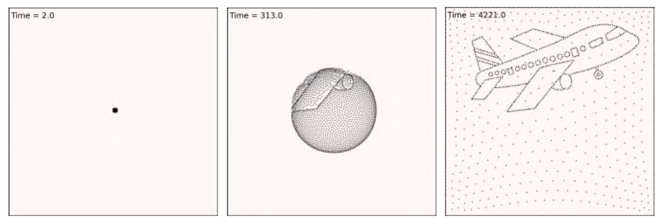

Fig. 6: Electrostatic target assignment with many targets and agents, showcasing a high degree of scalability. The total demand of the targets is 1240, each target demands 3 agents, and the number of agents is 2000. Targets not drawn for the sake of visual clarity. Agent sensing radius, not depicted in the image, is 1/10th of the bounding box.

target assignment. When using Section III-B.2's electrostatic ARD as the base model we get the following dynamics for the ith agent:

$$\frac{\partial \mathscr{G}}{\partial \vec{p}_i}(\vec{\mathbf{q}}) = \sum_{1 \le j \le \mathscr{N}} \frac{\overrightarrow{p_i p_j}}{\|\vec{p}_i - \vec{p}_j\|^2} - \frac{\partial}{\partial \vec{p}_i}\Phi(\vec{p}_i) \tag{18}$$

For example, when $\Phi(x,y) = ax + by$ is a linear scalar field, $\frac{\partial}{\partial \vec{p}_i}\Phi(\vec{p}_i) = \begin{bmatrix} a \\ b \end{bmatrix}$ and so agents will move in the direction [while spreading out due to repulsion forces. When $\Phi(x,y) = c \cdot e^{-\Lambda(x^2+y^2)}$ the agents will gather near the maximum of $\Phi(x,y)$ in a diffusive fashion (note that we also used exponential demand functions in signal coverage, but because the error function Ψ is different in signal coverage, we get different dynamics).

IV. CONCLUSION

We explored the topic of geometric task allocation for swarms of oblivious, decentralized agents with limited sensing range. We discussed two task allocation problems: signal coverage, wherein robots must "cover" an a priori unknown demand profile $\Phi(x,y)$ with influence profiles, and target assignment, wherein targets are placed in discrete, a priori unknown locations, and agents must find the targets and move to their location.

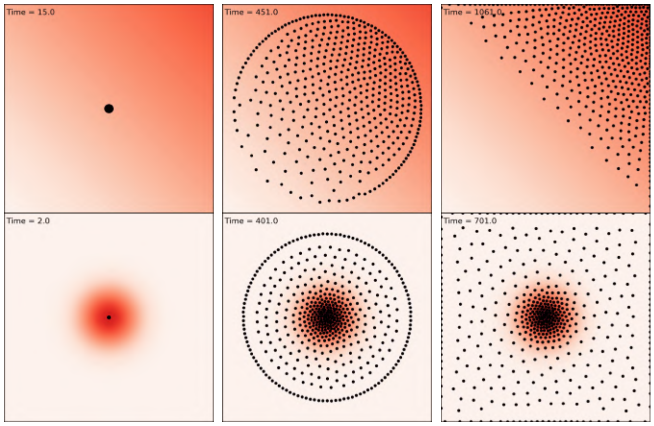

Fig. 7: Coverage of a linear demand function of the form $\Phi(x,y) = c(x+y)$ (top row), or an exponential demand function of the form $\Phi(x,y) = c \cdot e^{-\Lambda(x^2+y^2)}$ by 500 agents (bottom row). The heat map depicts $\Phi(x,y)$.

Our solutions to these problems optimize an error function and result in attraction-repulsion dynamics: agents repulse each other, encouraging exploration of the environment, and tasks attract agents, causing the agents to organize according to the tasks' requirements. In signal coverage, such dynamics are derived from gradient descent over the squared error $\Psi(x,y,\vec{\mathbf{q}}) = \left(\Phi(x,y) - \sum_{i=1}^{\mathcal{N}} \tilde{d}(x-x_i, y-y_i)\right)^2$. Despite squared errors being a natural thing to look at, surprisingly, we could not find any work in the literature obtaining ARD from the squared difference of signal functions. We proposed two different approaches to target assignment. The first is treating it as a signal coverage problem with highly concentrated environmental signals at the location of each target. The second is electrostatic target assignment via Coulomb's law. Finally, we related our solutions to a general form of attraction-repulsion dynamics, and discussed additional applications such as scalar field coverage.

References

[1] Mehdi Alighanbari. *Robust and decentralized task assignment algorithms for uavs*. PhD thesis, Massachusetts Institute of Technology, 2007.

[2] Michael Amir and Alfred M. Bruckstein. Minimizing travel in the uniform dispersal problem for robotic sensors. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2019.

[3] Michael Amir and Alfred M. Bruckstein. Fast uniform dispersion of a crash-prone swarm. In *Robotics: Science and Systems*, 2020.

[4] Behçet Açıkmeşe and David S. Bayard. Probabilistic swarm guidance for collaborative autonomous agents. In *2014 American Control Conference*, pages 477–482, 2014.

[5] Ariel Barel, Rotem Manor, and Alfred M Bruckstein. Come together: Multi-agent geometric consensus (gathering, rendezvous, clustering, aggregation). *arXiv preprint arXiv:1902.01455*, 2019.

[6] Joseph L Baxter, EK Burke, Jonathan M Garibaldi, and Mark Norman. Multi-robot search and rescue: A potential field based approach. In *Autonomous robots and agents*, pages 9–16. Springer, 2007.

[7] Liu Bo, Chu Tian-Guang, Wang Long, and Wang Zhan-Feng. Swarm dynamics of a group of mobile autonomous agents. *Chinese Physics Letters*, 22(1):254, 2005.

[8] Jorge Cortes, Sonia Martinez, Timur Karatas, and Francesco Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on robotics and Automation*, 20(2):243–255, 2004.

[9] Nazlı Demir, Behçet Açıkmeşe, and Can Pehlivantürk. Density control for decentralized autonomous agents with conflict avoidance. *IFAC Proceedings Volumes*, 47(3):11715–11721, 2014.

[10] Veysel Gazi and Kevin M Passino. A class of attractions/repulsion functions for stable swarm aggregations. *International Journal of Control*, 77(18):1567–1579, 2004.

[11] Veysel Gazi and Kevin M Passino. Stability analysis of social foraging swarms. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(1):539–557, 2004.

[12] John Gilliatt. A study of the halftone dot characteristics of rapid access halftones. 1983.

[13] Pascal Gwosdek, Christian Schmaltz, Joachim Weickert, and Tanja Teuber. Fast electrostatic halftoning. *Journal of real-time image processing*, 9(2):379–392, 2014.

[14] Andrew Howard, Maja J Matarić, and Gaurav S Sukhatme. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. In *Distributed Autonomous Robotic Systems 5*, pages 299–308. Springer, 2002.

[15] Aleksandar Jevtic, Alvaro Gutiérrez, Diego Andina, and Mo Jamshidi. Distributed bees algorithm for task allocation in swarm of robots. *IEEE Systems Journal*, 6(2):296–304, 2011.

[16] Hian Lee Kwa, Jabez Leong Kit, and Roland Bouffanais. Optimal swarm strategy for dynamic target search and tracking. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pages 672–680, 2020.

[17] Lantao Liu and Dylan A Shell. Large-scale multi-robot task allocation via dynamic partitioning and distribution. *Autonomous Robots*, 33(3):291–307, 2012.

[18] Lingzhi Luo, Nilanjan Chakraborty, and Katia Sycara. Provably-good distributed algorithm for constrained multi-robot task assignment for grouped tasks. *IEEE Transactions on Robotics*, 31(1):19–30, 2014.

[19] Alexandre R Mesquita, Joao P Hespanha, and Karl Åström. Optimotaxis: A stochastic multi-agent optimization procedure with point measurements. In *International workshop on hybrid systems: Computation and control*, pages 358–371. Springer, 2008.

[20] Nathan Michael, Michael M Zavlanos, Vijay Kumar, and George J Pappas. Distributed multi-robot task assignment and formation control. In *2008 IEEE International Conference on Robotics and Automation*, pages 128–133. IEEE, 2008.

[21] Lynne E Parker. Distributed algorithms for multi-robot observation of multiple moving targets. *Autonomous robots*, 12(3):231–255, 2002.

[22] Quoc-Viet Pham, Tuan LeAnh, Nguyen H Tran, and Choong Seon Hong. Decentralized computation offloading and resource allocation in heterogeneous networks with mobile edge computing. *arXiv preprint arXiv:1803.00683*, 2018.

[23] Ori Rappel, Michael Amir, and Alfred M Bruckstein. Stigmergy-based, dual-layer coverage of unknown regions. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pages 1439–1447, 2023.

[24] John H Reif and Hongyan Wang. Social potential fields: A distributed behavioral control for autonomous robots. *Robotics and Autonomous Systems*, 27(3):171–194, 1999.

[25] Arthur Richards, John Bellingham, Michael Tillerson, and Jonathan How. Coordination and control of multiple uavs. In *AIAA guidance, navigation, and control conference and exhibit*, page 4588, 2002.

[26] Christian Schmaltz, Pascal Gwosdek, Andrés Bruhn, and Joachim Weickert. Electrostatic halftoning. In *Computer Graphics Forum*, volume 29, pages 2313–2327. Wiley Online Library, 2010.

[27] Mac Schwager, Daniela Rus, and Jean-Jacques Slotine. Decentralized, adaptive coverage control for networked robots. *The International Journal of Robotics Research*, 28(3):357–375, 2009.

[28] Mac Schwager, Daniela Rus, and Jean-Jacques Slotine. Unifying geometric, probabilistic, and potential field approaches to multi-robot deployment. *The International Journal of Robotics Research*, 30(3):371–383, 2011.

[29] Onn Shehory and Sarit Kraus. Task allocation via coalition formation among autonomous agents. In *IJCAI (1)*, pages 655–661. Citeseer, 1995.

[30] Tanja Teuber, Gabriele Steidl, Pascal Gwosdek, Christian Schmaltz, and Joachim Weickert. Dithering by differences of convex functions. *SIAM Journal on Imaging Sciences*, 4(1):79–108, 2011.

[31] Feng WeiXing, Wang KeJun, Ye XiuFen, and Guo ShuXiang. Novel algorithms for coordination of underwater swarm robotics. In *2006 International Conference on Mechatronics and Automation*, pages 654–659. IEEE, 2006.