# Optimal Multi-Sensor Deployment via Sample-Based Quality-of-Service Distribution Matching

Donipolo Ghimire, and Solmaz S. Kia, *Senior Member, IEEE*

***Abstract*— This paper considers a multi-sensor service matching deployment problem over a set of discrete target points that populate a finite flat surface. The service can be event detection among targets using a vision sensor or an acoustic receiver, video surveillance for target monitoring, or providing wireless coverage to the targets. The quality-of-service (QoS) of the sensors is spatially nonuniform and can be anisotropic. The sensors are heterogeneous in the sense that their QoS distribution over their sensing footprint is not the same. The objective is to determine the sensor's best deployment position and orientation such that the collective multi-sensor QoS distribution matches the spread of the targets in the environment as closely as possible. To solve this problem, we propose a two-stage deployment strategy. First, we partition the environment using the computationally efficient $K$-means clustering algorithm. Then, we sample points from the QoS distribution over the sensing footprint. Then, for each sensor-cluster pair, we use an iterative closest-point approach inspired by the point cloud registration algorithms used in computer vision to determine the best deployment position and orientation for the sensor. Finally, we use a linear assignment problem framework to assign the clusters to the sensors. Numerical examples demonstrate our results.**

***Keywords*: multi-sensor deployment; service-matching; iterative closest point algorithm**

## I. Introduction

In recent years due to rapid progress in wireless communication and embedded micro-sensing technologies, we have witnessed increasingly more use of groups of sensors or agents to 'cover' a region of interest to achieve goals like monitoring, data collection, and enhancing wireless coverage. However, due to the cost, deploying unlimited agents to attain full coverage is not feasible. Thus, the main problem of interest in multi-agent deployment for coverage is optimal coverage given a limited number of agents. In this paper, we consider a multi-agent deployment problem for coverage over a dense set of discrete targets with known locations in which the coverage service provided by each agent is modeled as an anisotropic spatial distribution, see Fig. 1. In this paper, we seek an effective deployment strategy to determine each agent's deployment position and orientation such that the topological distribution of the set of agents' coverage service footprint is close to the spatial distribution of the targets on the ground.

The authors are with the Department of Mechanical and Aerospace Engineering, University of California Irvine, Irvine, CA 92697, {dghimire,solmaz}@uci.edu. The work of the second author was supported by the US national science foundation (NSF) ECCS 1653838 award.

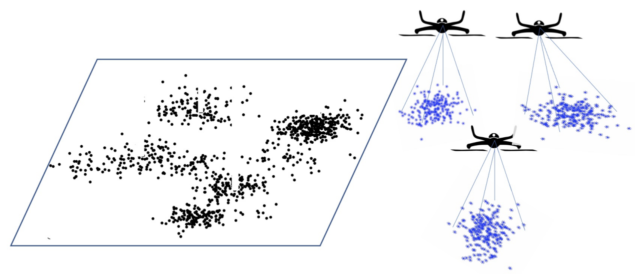


Fig. 1 – Example of a set of geographical points of interest (picture on the left). The finite number of agents with different coverage service footprints, with a limited sensing zone; blue points represent samples from the spatial distribution of the QoS of the agents (picture on the right).

*Literature Review*: The problem addressed in this paper falls in the general area of sensor deployment for coverage [1]–[5]. One of the popular techniques for area coverage is the Voronoi-based deployment strategies [1], [6], [7]. This approach assumes a disc-shaped sensing footprint with a uniform quality-of-service (QoS) distribution over the sensing footprint. However, most sensors in practice are anisotropic, see Fig. 2, and have a non-uniform QoS distribution as shown in Fig. 3. Anisotropic Voronoi partitioning-based algorithms are proposed in the literature, however, they often consider simple footprints such as wedges and ellipsoids with a uniform QoS distribution over the sensing footprint [8]–[10]. Other Voronoi-based approaches are also studied in [11]–[14]. Voronoi-based algorithms are intended for coverage over continuous space and rely on geometric relations which may not be easily invoked for intricate sensor footprints shown in Fig. 2. Coverage over discrete points can be transformed into coverage over continuous space by estimating and using target distributions as a priority function. For example, [3], [15] use the expectation-maximization algorithm to estimate the density of the target as a Gaussian Mixture Model (GMM). Deployment algorithms over areas endowed by position priority functions are studied in the literature [1], [9], [10], [16], [17]. But these algorithms have limitations in considering intricate sensor footprints and often do not consider the non-uniform probabilistic QoS. On the other hand, recent work such as [3], [15] consider a probabilistic service distribution. Their method attempts a distribution matching deployment but is limited to sensors with Gaussian QoS distribution.

*Statement of Contributions:* This paper considers a multi-sensor service matching deployment problem over a set of discrete target points that populate a finite flat surface. The QoS of the sensors is spatially nonuniform and can be anisotropic. The sensors are heterogeneous in the sense that their QoS distributions over the sensing footprint are different from one another. We make no assumptions about the sensing footprint shape or the probability distribution of the QoS of the sensors. The objective is to determine the best position and orientation for the sensors to be deployed to such that the collective multi-sensor QoS distribution is as similar as possible to the spread of targets in the environment. To solve this problem, we proposed a two-stage deployment strategy. First, we partition the environment using the computationally efficient $K$-means clustering algorithm for discrete and non-convex domains. Then, we sample points from the QoS distribution over the sensing footprint. Then, we use these samples to determine the similarity between the quality of the service of each sensor around its sensing footprint and the targets in each cluster. We determine the similarity based on the earth mover distance metric [18]. In computing the similarity, we incorporate the freedom to transform (rotate and translate) the sensors in the earth mover distance minimization problem formulation. The resulting optimization problem can be formulated as a non-linear mixed integer programming. To solve this problem, we draw inspiration from the iterative closest point approach [19] used in computer vision for point cloud matching. Once the similarity between each sensor's QoS and the target points in the clusters are obtained, we cast the sensor deployment problem as a linear assignment problem which can be solved using existing algorithms in the literature [20]. The process determines what cluster each sensor should be deployed to and their location and orientation in the cluster. Two simulation studies inspired by anisotropic vision and acoustic sensors problem settings demonstrate our results.

*Notation*: We let $\mathbb{R}$, $\mathbb{Z}$, $\mathbb{Z}_{>0}$ and $\mathbb{Z}_{\geq 0}$ denote the set of reals, integers, positive integers, and non-negative integers, respectively. Given a set $\mathcal{Z}$ its cardinality is $|\mathcal{Z}|$. We denote the Euclidean norm of a vector $\mathbf{x} \in \mathbb{R}^2$ by $\|\mathbf{x}\|$. For a matrix $\mathbf{R}$, its transpose is $\mathbf{R}^\top$.

## II. Problem definition

We consider a multi-sensor deployment problem over a finite set of points of interest (referred hereafter as targets) that are densely scattered over a finite 2D planar space $\mathcal{W} \subset \mathbb{R}^2$. $m$ represents the number of discrete targets scattered in this plane $\mathcal{W}$. The discrete targets may consist of either real-world entities, such as humans or animals, information sources, or points sampled from a distribution of an event of interest, e.g., pollution spills. Targets' position $\mathbf{t}_i = (x_i, y_i) \in \mathcal{W}$, $i \in \{1, \cdots, m\}$ are known. We want to deploy a set of $N$ sensors, denoted by $\mathcal{V}_s = \{1, \cdots, N\}$, with heterogeneous sensing to provide a 'service' in accordance with the 'spread' of the targets, see Fig. 1. The service can be event detection among targets using a vision sensor or an

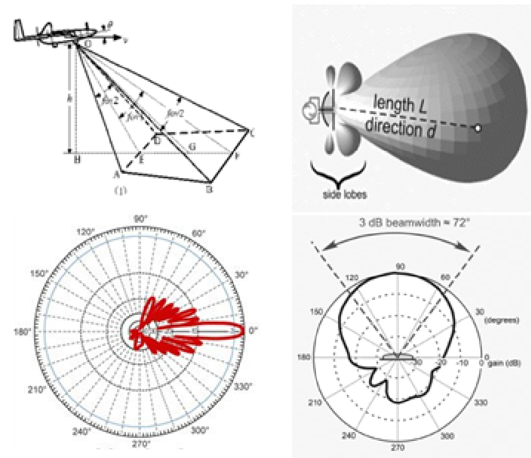



Fig. 2 – Anisotropic Sensor footprint of different sensors like a camera (top-left) [21], directional RFID (top right, bottom-right), and directional antennas (bottom-left).

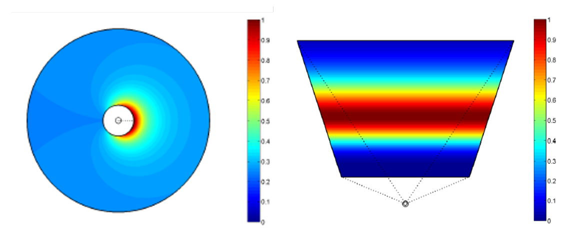


Fig. 3 – Probability of detection of an acoustic receiver (left) and a camera (right) [22]. The image shows that the probability is anisotropic and detection probability is not uniform across the sensing zone.

acoustic receiver, video surveillance for target monitoring, or providing wireless coverage to the targets. As discussed in the introduction, the sensing footprint is often anisotropic, non-symmetric, and intricate; see Fig. (2). The spatial QoS over the sensing footprint is also nonuniform and often conforms to a pattern induced by their physics, see Fig. (3).

Due to cost, there are a limited number of sensors to deploy that may not be enough to provide service for all the targets. The objective of this paper is to propose a multi-sensor deployment strategy that determines the deployment pose (position and orientation) of the sensors by taking into account the sensor footprint, the quality of the service distribution of the sensors over that footprint, and the heterogeneity of the sensors to achieve 'optimal' coverage. In what follows, we refer to the mobile vehicles (e.g., UAV, ground vehicle, mobile robot) that a sensor is mounted on, in general, as a *robot*.

## III. The proposed deployment method

To present our optimal deployment strategy, we first, introduce some functions which we will use in our design. We assume that the sensor is mounted in a fixed configuration on the deployment robot. A finite deterministic (possibly approximate) sensor footprint with a known geometric

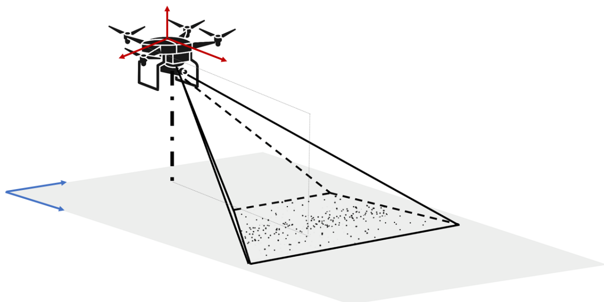


Fig. 4 – The sensor footprint of a camera mounted on a UAV. We sample a finite number of points in the footprint according to the quality of the service distribution.

boundary is considered. For illustration purposes, we take a camera sensor mounted on a UAV at a fixed height, $h$, from the ground as shown in Fig. 4. We assume that there is a mapping function $f$ which given the position of $B \in \mathbb{Z}_{\geq 1}$ finite number of samples on the sensor footprint, denoted by $\{\mathbf{c}_l\}_{l=1}^{B}$, returns the position of the center of the robot denoted by $\mathbf{r} = (x^r, y^r, h) \in \mathbb{R}^3$. That is for each sensor $i \in \{1, \cdots, N\}$ there exists $f_i$ such that

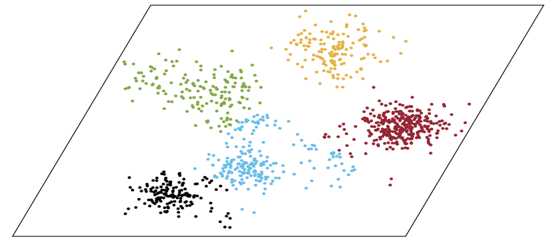
$$\mathbf{r}_i = f_i(\{\mathbf{c}_l\}_{l=1}^{B_i}). \tag{1}$$

We also assume that there exists a sampling function $S(n, i)$ which draws $n$ finite samples with known locations inside the sensor $i \in \{1, \cdots, m\}$ footprint according to the spatial distribution of the sensor's QoS.

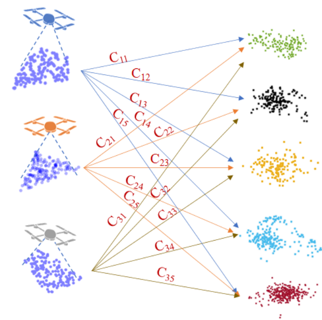
Our deployment strategy is illustrated in Fig. 5. In step 1 of our design, we cluster the target points using the $K$-means clustering algorithm. $K$-mean clustering [23] partitions the points to $K \in \mathbb{Z}_{>1}$ clusters in a way that each point belongs to the cluster with the nearest mean (cluster centers or cluster centroid). The number of clusters $K$ is determined offline using the silhouette analysis [24]. Other clustering algorithms such as GMM-based clustering or Fuzzy C-Means Clustering [23] can also be used. We chose K-mean clustering due to its low computational complexity and hard clustering approach that produces non-overlapping clusters.

Step 2 of our deployment strategy is to solve an optimal assignment problem that assigns sensors to their respective best clusters and finds the best position, i.e. location and orientation, in the cluster based on the similarity between the QoS distribution and the distribution of the points in the clusters.

Let the number of points in each cluster $j \in \mathcal{K} = \{1, \cdots, K\}$ be $m_j^{\mathrm{c}}$. Let the position of the target points in cluster $j \in \mathcal{K}$ be denoted by $\mathbf{t}_1^j, \cdots, \mathbf{t}_{m_j^{\mathrm{c}}}^j$. For each sensor $i \in \{1, \cdots, N\}$, given an initial pose, let the position vector of the sample points drawn according to the QoS distribution in the sensor footprint be $\mathbf{s}_1^i, \cdots, \mathbf{s}_{n_i^{\mathrm{s}}}^i$, where $n_i^{\mathrm{s}}$ is the total number of the samples. To compute the *similarity match* between the QoS distribution and the target distribution in a



Step 1: clustering



Step 2: Service-matching assignment

Fig. 5 – Deployment strategy: Step 1 uses K-means clustering to group the targets into $K = 5$ clusters $\mathcal{K}_1, \cdots, \mathcal{K}_5$ (clusters are distinguished by color); Step 2 is an assignment problem that matches the sensors to the clusters based on their similarity.

cluster, we use the *earth mover distance* metric [18] which is defined as follows. In computing this similarity measure denoted by, $C_{i,j}^{\star}$, we assume that we draw the same number of samples as the number of target points in cluster $j$, i.e., $n_i^{\mathrm{s}} = m_j^{\mathrm{c}} = n_{i,j}$.

$$C_{i,j}^{\star} = \min_{\mathbf{R}_{i,j}, \mathbf{d}_{i,j}, \mathbf{T}} \sum_{l=1}^{n_{i,j}} \sum_{k=1}^{n_{i,j}} T_{l,k} \|\mathbf{R}_{i,j}\mathbf{s}_l^i + \mathbf{d}_{i,j} - \mathbf{t}_k^j\|^2, \text{ s.t.} \tag{2a}$$

$$T_{l,k} \in \{0, 1\}, \qquad 0 < l, k \leq n_{i,j}, \tag{2b}$$

$$\sum\nolimits_{l=1}^{n_i^{\mathrm{s}}} T_{l,k} = 1, \quad l = 1, \cdots, n_{i,j}, \tag{2c}$$

$$\sum\nolimits_{l=1}^{n_i^{\mathrm{s}}} T_{l,k} = 1, \quad k = 1, \cdots, n_{i,j}, \tag{2d}$$

$$\mathbf{R}_{i,j}^{\top}\mathbf{R}_{i,j} = \mathbf{R}_{i,j}\mathbf{R}_{i,j}^{\top} = \mathbf{I}_2. \tag{2e}$$

where $\mathbf{T} = [T_{l,k}]$ is the assignment matrix. Here, $\mathbf{R}_{i,j}\mathbf{s}_1^i + \mathbf{d}_{i,j}, \cdots, \mathbf{R}_{i,j}\mathbf{s}_{n_{i,j}}^i + \mathbf{d}_{i,j}$ are the rigid-body rotation (via rotation matrix $\mathbf{R}$) and translation (via shift vector $\mathbf{d}$) of the sample points from the initial condition $\mathbf{s}_1^i, \cdots, \mathbf{s}_{n_{i,j}}^i$ to the final deployment pose. Notice that in computing $C_{i,j}^{\star}$, the optimization problem (2) is searching for best $\mathbf{R}_{i,j}$ and $\mathbf{d}_{i,j}$ in a way that the representative samples from the sensor $i$'s QoS distribution can be 'moved' with the minimum displacement in the euclidean sense such that sensor $i$ covers all the target points in the cluster $j$. In what follows, we denote these best rotation matrices and the translation vectors obtained from solving (2) as $\mathbf{R}_{i,j}^{\star}$ and $\mathbf{d}_{i,j}^{\star}$ respectively.

Once $C_{i,j}^{\star}$ for every sensor $i \in \mathcal{V}_s$ and every cluster $j \in \mathcal{K}$

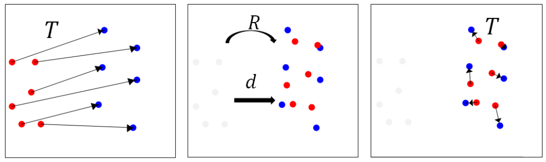



Fig. 6 – A sequence of correspondences by solving an assignment problem (left plot), optimal transportation given the correspondence (middle plot), a new correspondence after transformation (right plot). As one can see the correspondence in the left and right plots are not identical.

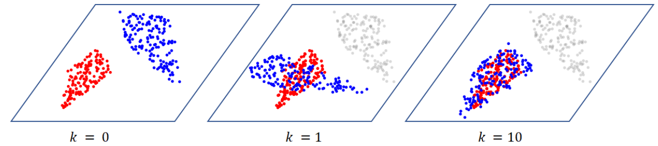



Fig. 7 – Two points clouds with arbitrary distribution (left), where one is a rotated and translated version of the other while also adding Gaussian noise. Result after first iteration, $k = 1$ (middle), and iteration, $k = 10$ (right).

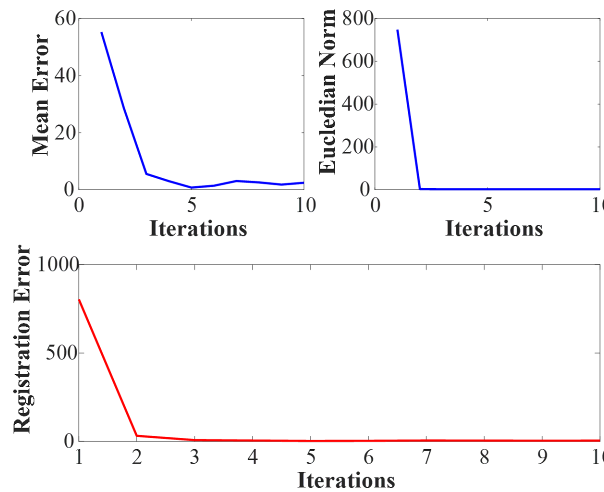



Fig. 8 – Graphs representing the convergence of Mean Error (top-left), euclidean norm (top-right), and registration error(bottom) for the example depicted in Fig. 7, where the initial placement of the two points sets are shown in red and blue in the left plot. agent-target pair.

is computed, we cast the sensor deployment as the linear assignment problem,

$$\mathbf{Z}^\star = \arg\min \sum\nolimits_{i\in\mathcal{V}_\mathrm{s}} \sum\nolimits_{j\in\mathcal{K}} Z_{i,j} C^\star_{i,j}, \tag{3a}$$

$$Z_{i,j} \in \{0,1\}, \quad i \in \mathcal{V}_\mathrm{s},\ j \in \mathcal{K}, \tag{3b}$$

$$\sum\nolimits_{j\in\mathcal{K}} Z_{i,j} = 1,\ \forall i \in \mathcal{V}_\mathrm{s}, \tag{3c}$$

$$\sum\nolimits_{i\in\mathcal{V}_\mathrm{s}} Z_{i,j} \le 1,\ \forall j \in \mathcal{K}, \tag{3d}$$

where $\mathbf{Z}^\star$ is the final assignment matrix. $Z_{i,j} = 1$ means that sensor $i$ is assigned to cluster $j$. The constraint (3c) ensures that every sensor is assigned to a cluster, while constraint (3d) ensures that each cluster is assigned to at most one sensor. Optimization problem (3) is a mixed linear integer programming that is well-known in the literature. This problem can be solved using continuous relaxation via the existing numerical solvers or using algorithms such as the Hungarian algorithm, see [20]. Here, we assume that $|\mathcal{K}| \ge |\mathcal{V}_s|$, which is often the case in practice, i.e., we have fewer sensors than areas to cover. However, the assignment problem can be easily reformulated to address the scenarios where we have more sensors than the areas to cover, and we want to choose the best sensors to deploy.

Once the assignment problem is solved and $\mathbf{Z}^\star$ is obtained, the sensor's assignment is completed. The final deployment position of the robot carrying the sensor $i$ assigned to cluster $j$ ($Z_{i,j} = 1$) is then extracted using the mapping function (1) using $\mathbf{c}^\star_l = \mathbf{R}^\star_{i,j}\mathbf{c}_l + \mathbf{d}^\star_{i,j}$. The final deployment orientation is calculated by rotating the body coordinate system of the robot using the rotation matrix $\mathbf{R}^\star_{i,j}$. This completes the deployment strategy design.

Even though we outlined a deployment strategy that leads to a service-matching deployment, a critical component is left to address, which is solving the optimization problem (2) in an efficient manner. This optimization problem is a mixed-integer nonlinear program. In the following section, we proposed a method inspired by the Iterative Closest Point (ICP) algorithm used in point-set registration in computer vision [19] to solve the optimization problem.

### *A. Iterative closest point approach to solve similarity match problem*

Optimization problem (2) consists of two actions. For the first action, we seek to find the correspondence matrix $\mathbf{T} = [T_{l,k}]$ which determines what sample $l$ should be 'moved' to the target point $k$ so that the overall distance 'traveled' by the samples to cover the target points is minimized. Whereas, for the later action, we treat the sensor's QoS as a rigid body and find the best transformation, $\mathbf{R}_{i,j}$ and $\mathbf{d}_{i,j}$, given the correspondences. Based on this observation, one way to solve (2) is to decouple the correspondence-based registration and the optimal transformation steps. To achieve this, we start with $\mathbf{R}_{i,j} = \mathbf{I}$ and $\mathbf{d}_{i,j} = \mathbf{0}$ and solve (2), which becomes an instance of a linear assignment problem whose global optimal solution can be found using existing algorithms. After obtaining the correspondence-based registration, we solve (2) for the given $\mathbf{T}$ to obtain the optimal transformation. This second problem is an instance of the general problem of least-squares fitting of two point sets [25]: given a set of points $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^{\bar{n}}$ and $\mathcal{Q} = \{\mathbf{q}\}_{i=1}^{\bar{n}}$, find a transformation (rotation $\mathbf{R}$ and translation $\mathbf{d}$) to transform

$\mathcal{P}$ that matches $\mathcal{Q}$ as close as possible,

$$(\hat{\mathbf{R}}, \hat{\mathbf{d}}) = \underset{\mathbf{R},\mathbf{d}}{\text{argmin}} \sum_{i=1}^{\bar{n}} \|\mathbf{R}\,\mathbf{p}_i + \mathbf{d} - \mathbf{q}_i\|^2, \quad s.t. \tag{4a}$$

$$\mathbf{R}^\top \mathbf{R} = \mathbf{R}\mathbf{R}^\top = \mathbf{I}. \tag{4b}$$

Problem (4) can be solved using a singular value decomposition (SVD)-based method described in [25]. However, since the two actions, correspondence-based registration and optimal transformation, are not decoupled in (2), this two-step sequential approach does not lead to the optimal solution. See the example in Fig. 6, which shows that the correspondence after the transformation changes. Therefore we need to iterate.

The iterations however increase the computational cost, which is mainly because of solving the assignment problem to compute the correspondence. Well-know optimal solutions such as the Hungarian method or simplex-based solution using continuous relaxation have computation complexity, $\mathcal{O}(\bar{n}^3)$ and $\mathcal{O}(\bar{n}^4)$, respectively [20], where $\bar{n}$ is the number of the sample points in each set. Even the sub-optimal sequential Greedy Algorithm, which theoretically ensures an approximation factor of 63%, has a complexity of $\mathcal{O}(\bar{n}^2 \log\bar{n})$ [20]. To reduce this computational cost, we follow the ICP algorithm approach [19], which uses a nearest-neighbor approach for correspondence-based registration. Using the kd-tree data structure this correspondence method comes with a computational complexity of $O(\bar{n} \log\bar{n})$ [26]. We chose ICP because it is easy to implement, can be customized to meet the need of specific applications, and is computationally efficient. Fig. 7 shows the result of using this ICP-based iterative algorithm for an example scenario where we want to overlay the blue samples over the red samples in the tightest possible way in the sense of the earth mover measure (2). Fig. 8 shows the quantitative analysis of the improvements achieved through the iteration process (results are shown for 10 iterations).

## IV. Numerical Demonstration

In this section, we demonstrate the effectiveness of our proposed service-matching deployment algorithm via two academic numerical examples. The environment and discrete targets over which agents are deployed are shown in the top plot of Fig. 5. The environment $\mathcal{W}$ is a rectangle of $[-350, 350] \times [-300, 300]$ meters. Furthermore, the position of the targets in this spatial domain is given a priori. The number of targets is $m = 1000$. We use the $K$-mean algorithm to cluster the targets into $K = 5$ clusters as shown in the top plot in Fig. 5; each identical colored set of points represents a cluster. The number of clusters was calculated using the silhouette analysis. In the following section, we consider two multi-sensor deployment scenarios. Our numerical examples are inspired by vision-based detection and acoustic monitoring problems. For the first scenario, we consider a group of $N = 4$ UAVs mounted with cameras, and for the second, we consider a group of

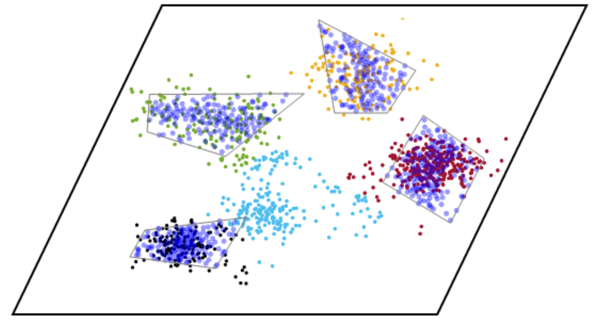


Fig. 9 – Placement of multiple UAVs with camera for maximal coverage taking into account the best rotation and translation.

$N = 5$ holonomic mobile robots with acoustic receivers. Both of these examples use anisotropic sensors whose QoS distribution looks like the ones shown in Fig. 3; for more details see [22]. The goal of this demonstration is to place these anisotropic sensors to achieve the maximum possible coverage in the target domain $\mathcal{W}$.

**Unmanned Aerial Vehicles with cameras**: In this example, there are four UAVs with visual sensors where every sensor has different resolutions and depth from a camera's point of view. The altitude at which these UAVs are flying is different, $h = \{15, 30, 45, 60\}$ meter. Thus, the sensor footprint for each agent is different, making their coverage service problem heterogeneous. The sensor footprint of the camera can be calculated by using the sensor model from [22] which is also used to sample points from the footprint of the camera. This example illustrates a scenario where the number of sensors is less than the number of partitions in the environment $\mathcal{W}$, i.e., we have limited resources. However, the algorithm works for any number of sensor-to-cluster ratios. The sensor footprints and the corresponding samples based on the probabilistic sensor model are shown in Fig. 9 as trapezoidal shapes with blue points overlaid on the targets. The placement is the result of using our proposed deployment strategy described in Section III, which shows the effectiveness of our algorithm in providing a service-matching deployment.

**Holonomic Robot with Acoustic Receiver:** Next, we present simulation results for holonomic robots with acoustic receivers. In this example, $N = 5$ mobile robots with directional acoustic receivers are assigned to maximize the probability of detection over the environment $\mathcal{W}$. Since these sensors are anisotropic we seek to determine the best position and rotation for each sensor. The boundaries of the footprint of an acoustic sensor are limited by the maximum, $D_{max}$, and minimum, $D_{min}$, distance from the sensor. In addition, the intensity of received sound is dependent on the angle and distance from the sensor which is modeled as a cardioid as shown in Fig. 10. Samples are generated from the cardioid model as illustrated in [27]. The sensor footprints and the corresponding samples based on the probabilistic sensor model are shown in Fig. 10 as cardioid shapes with blue points overlaid on the targets. The placement is the

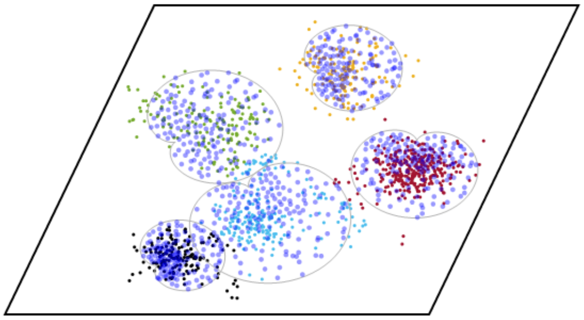


Fig. 10 – Placement of multiple robots with acoustic receivers with cardioid shaped footprints for maximal coverage taking into account the best rotation and translation.

result of using our proposed deployment strategy described in Section III, which shows the effectiveness of our algorithm in accomplishing a service-matching deployment.

## V. Conclusion

In this paper, a deployment framework for heterogeneous mobile sensors to efficiently cover a group of dense targets was proposed. Since most sensors are anisotropic, the paper focused on modeling a solution approach that worked for anisotropic sensors like cameras, directional antennas, and directional acoustic receivers for which the deployment strategy should not only determine the position for deployment but also the direction to deploy. The proposed method was also designed to consider the nonuniform QoS of the sensors over their sensing footprint. To solve this problem, we proposed a two-stage deployment strategy. First, we partitioned the environment using the computationally efficient $K$-means clustering algorithm. We made no assumptions such as Gaussianess for the spatial distribution of the targets or the QoS of the sensors. In the second step, we used an iterative closet point approach inspired by the point cloud registration algorithms used in the computer vision problems along with a linear assignment problem to determine which cluster each robot should be deployed to and at what location and orientation. Case studies inspired by anisotropic sensors like a camera and acoustic receivers were studied where the goal was to maximize the collective overlap between the sensor footprint of each agent and target distribution. For future work, we will extend our approach to deployment over targets populating a three-dimensional surface.

## References


[1] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[2] A. Pierson and D. Rus, "Distributed target tracking in cluttered environments with guaranteed collision avoidance," in *2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pp. 83–89, 2017.

[3] Y. Chung and S. S. Kia, "A distributed service-matching coverage via heterogeneous agents," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4400–4407, 2022.

[4] J. W. Durham, R. Carli, P. Frasca, and F. Bullo, "Discrete partitioning and coverage control for gossiping robots," *IEEE Transactions on Robotics*, vol. 28, no. 2, pp. 364–378, 2012.

[5] S. Miah, A. Y. Panah, M. M. H. Fallah, and D. Spinello, "Generalized non-autonomous metric optimization for area coverage problems with mobile autonomous agents," *Automatica*, vol. 80, pp. 295–299, 2017.

[6] J. Cortes, "Coverage optimization and spatial load balancing by robotic sensor networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 3, pp. 749–754, 2010.

[7] S. Kim and M. Egerstedt, "Heterogeneous coverage control with mobility-based operating regions," in *American Control Conference*, pp. 2148–2153, 2022.

[8] A. Gusrialdi, T. Hatanaka, and M. Fujita, "Coverage control for mobile networks with limited-range anisotropic sensors," in *IEEE Int. Conf. on Decision and Control*, pp. 4263–4268, 2008.

[9] K. Laventall and J. Cortés, "Coverage control by multi-robot networks with limited-range anisotropic sensory," *International Journal of Control*, vol. 82, no. 6, pp. 1113–1121, 2009.

[10] F. Farzadpour, X. Zhang, X. Chen, and T. Zhang, "On performance measurement for a heterogeneous planar field sensor network," in *IEEE International Conference on Advanced Intelligent Mechatronics*, pp. 166–171, 2017.

[11] M. Santos, Y. Diaz-Mercado, and M. Egerstedt, "Coverage control for multirobot teams with heterogeneous sensing capabilities," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 919–925, 2018.

[12] M. Santos and M. Egerstedt, "Coverage control for multi-robot teams with heterogeneous sensing capabilities using limited communications," in *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, pp. 5313–5319, 2018.

[13] S. Kim, M. Santos, L. Guerrero-Bonilla, A. Yezzi, and M. Egerstedt, "Coverage control of mobile robots with different maximum speeds for time-sensitive applications," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3001–3007, 2022.

[14] O. Arslan, H. Min, and D. E. Koditschek, "Voronoi-based coverage control of pan/tilt/zoom camera networks," in *IEEE Int. Conf. on Robotics and Automation*, pp. 5062–5069, 2018.

[15] A. Z. Abdulghafoor and E. Bakolas, "Two-level control of multiagent networks for dynamic coverage problems," *IEEE Transactions on Cybernetics*, pp. 1–12, 2021.

[16] A. Pierson, L. C. Figueiredo, L. C. Pimenta, and M. Schwager, "Adapting to performance variations in multi-robot coverage," in *IEEE Int. Conf. on Robotics and Automation*, pp. 415–420, 2015.

[17] O. Arslan and D. E. Koditschek, "Voronoi-based coverage control of heterogeneous disk-shaped robots," in *IEEE International Conference on Robotics and Automation*, pp. 4259–4266, 2016.

[18] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.

[19] J. Yang, H. Li, D. Campbell, and Y. Jia, "Go-ICP: A globally optimal solution to 3d ICP point-set registration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, pp. 2241–2254, nov 2016.

[20] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems*. Society for Industrial and Applied Mathematics, 2012.

[21] H. Chen, X.-M. Wang, Y.-S. Jiao, and Y. Li, "Research on search probability and camera footprint of region coverage for uavs," in *2009 IEEE International Conference on Control and Automation*, pp. 1920–1924, IEEE, 2009.

[22] B. Hexsel, N. Chakraborty, and K. Sycara, "Distributed coverage control for mobile anisotropic sensor networks. robotics institute, pittsburgh," tech. rep., PA, Tech. Rep. CMU-RI-TR-13-01, 2013.

[23] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[24] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[25] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 698–700, 1987.

[26] P. Ram and K. Sinha, "Revisiting kd-tree for nearest neighbor search," in *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining*, pp. 1378–1388, 2019.

[27] B. Hexsel, N. Chakraborty, and K. Sycara, "Coverage control for mobile anisotropic sensor networks," in *IEEE Int. Conf. on Robotics and Automation*, pp. 2878–2885, 2011.