Obstacle Discovery in Distributed Actuator and Sensor Networks*

Rong Zheng and Amit Pendharkar

Department of Computer Science
University of Houston
Houston, TX, 77204, USA
http://www.cs.uh.edu

Technical Report Number UH-CS-08-16

December 20, 2008

Keywords: Active Sensing, Segment Voronoi Diagram, Obstacle Detection.

Abstract

Distributed active sensing is a new sensing paradigm, where active sensors (a.k.a actuators) as illuminating sources and passive sensors as receivers are distributed in a field, and collaboratively detect interested events. In this paper, we study the fundamental properties of distributed actuator and sensor networks (DASNs) in detecting and localizing obstacles. A novel notion of "exposure" is defined, which quantifies the dimension limitations in detectability. Using simple geometric constructs, we propose polynomial-time algorithms to compute the exposure and bounding regions where the center of the obstacles may lie.

*This work was in part supported by US National Science Foundation under contract NSF CNS-0832089.

Obstacle Discovery in Distributed Actuator and Sensor Networks*

Rong Zheng and Amit Pendharkar

Abstract

Distributed active sensing is a new sensing paradigm, where active sensors (a.k.a actuators) as illuminating sources and passive sensors as receivers are distributed in a field, and collaboratively detect interested events. In this paper, we study the fundamental properties of distributed actuator and sensor networks (DASNs) in detecting and localizing obstacles. A novel notion of "exposure" is defined, which quantifies the dimension limitations in detectability. Using simple geometric constructs, we propose polynomial-time algorithms to compute the exposure and bounding regions where the center of the obstacles may lie.

Index Terms

Active Sensing, Segment Voronoi Diagram, Obstacle Detection.

I. Introduction

Obstacle discovery concerns with the problem of detecting the presence and determining the location of obstacles. Many applications can be found in robot navigation, object tracking, and surface and/or structure fatigue testing etc. Traditional obstacle detection systems roughly fall into two categories [6], namely, passive sensing and active sensing systems. Passive sensing relies on the existence of energy emitting/reflecting sources for detection. Examples are cameras operating in visible or infrared spectrums. Active sensing systems on the other hand, provide their own energy for illumination. Most existing active sensing systems are *stand-alone* systems with co-located transmitters and receivers. Energy emitted by the transmitter is reflected by the objects and received by the receiver. Active sensing can not only provide ranging information but also speed of target obstacles directly (e.g., via the Doppler effect). Examples of active sensing systems include Radar, Lidar (light detection and ranging sensor), or Ladar (laser detection and range sensor). Compared with passive sensing, *stand-alone* active systems have the advantage of being not restricted by environmental illumination conditions. However, they face difficulties in interpreting the output signal returned and high acquisition costs. Furthermore, much interference will be incurred to the environment in order to reach further distance using high transmission power, since the reflection waves are subject to substantial diffraction and attenuation.

With the advancement of MEMS technologies, dense deployment of cheap sensors becomes economically viable. Such sensors usually feature on-board processing and wireless communication capability. This gives rise to a new paradigm of obstacle discovery, namely, distributed active sensing. A distributed active sensing network (DASN) consists of active and passive sensors distributed in a field. In a DASN, active sensors (also called actuators) emit waveforms (acoustic or electro-magnetic), which propagate through the medium. Passive sensors are distributed and perform detection and signal processing. Through the exchange of processed data, presence and location of interested obstacles can be assessed. Compared to *stand-alone* active sensing systems, DASNs can observe events at close proximity, and have the potential to reduce energy consumption (at the actuators) and reduce signal processing overhead (at the sensors), and thus lead to lower deployment costs.

To make our discussions more concrete, let us consider one specific application in the area of structural health monitoring. Piezoceramics-based transducers (PZTs) have the nice property that they can function both as *sensors* and *actuators*. As an actuator, a PZT excites elastic stress waves that propagate through concrete structures. As a sensor, a PZT transforms mechanical waves into electric current and can be used to detect the wave response of the structure. A hole or a damage inside a concrete structure acts as a stress relief in the wave propagation path, and thus higher degree of attenuation can be observed (Figure 1). This is one of the basic principles in structural health monitoring using PZT sensors. In this application, holes or damages inside the structure have

*This work was in part supported by US National Science Foundation under contract NSF CNS-0832089.

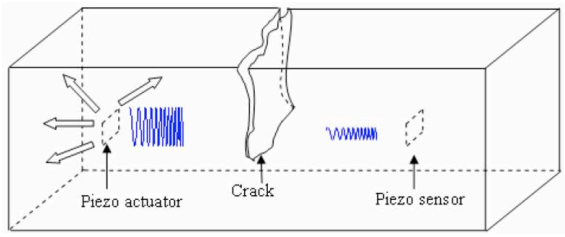
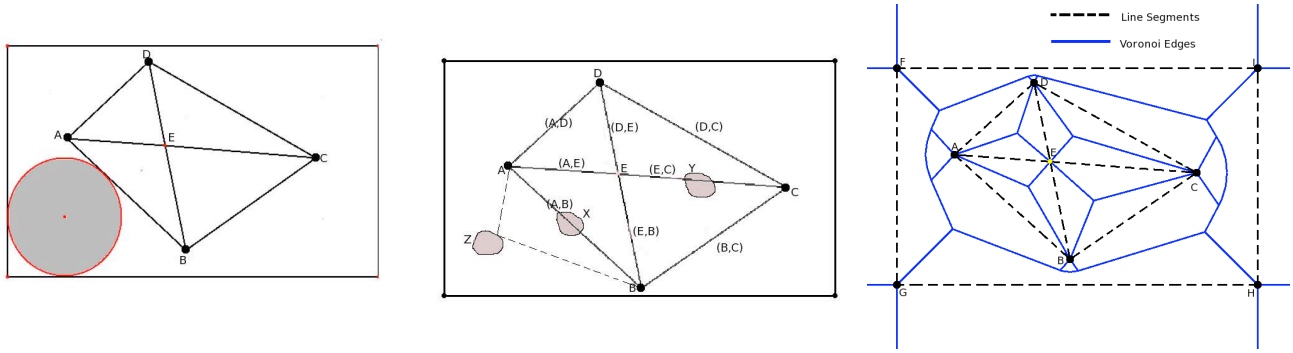


Fig. 1: Crack detection using active sensing in structural health monitoring.



(a) Line segments associated with propagation paths between visible sensors and actuators and area exterior boundary. Exposure as the radius of the largest disk (shaded) not intersecting with any path connecting sensors and actuators and the bounding segments of the deployment area

(b) Obstacle X and Y intersect with line segments AB and EC respectively. Obstacle Z cannot be detected since it is not on the propagation path of any actuator and sensor pair

(c) Segment Voronoi diagram computed on the point and line segment sites. Voronoi edges are bisectors of two incident sites. Voronoi vertices are intersection of edges, and are at equal distance to three sites.

Fig. 2: Line segments, exposure and segment Voronoi diagram in distributed active sensing. A, B, C, D are both sensors and actuators.

different propagation properties from their surrounding medium and thus can treated as "obstacles". Determining the presence and location of holes or faults provides necessary information to assess the serviceability of civil infrastructure in a timely manner.

The thesis of this paper is to study the fundamental properties of distributed active sensing by establishing a set of geometric models to describe such systems. The first question is *detectability*, namely, under what conditions is an obstacle detectable by an array of actuators and sensors? We introduce the notion of "exposure", which quantifies the dimension limitations in detectability. Using computational geometry techniques, we propose an algorithm that computes the maximum exposure of an arrangement of actuators and sensors. The second question is *decidability*, namely, to what degree can the locations of obstacles be determined, and multiple obstacles be distinguished? We first prove by examples a negative result of undecidability. However, when additional constraints (from domain-specific knowledge) are given, such uncertainty can be reduced. We propose a polynomial-time algorithm that characterizes the regions where the centers of obstacles may lie. Albeit based on simplified geometric constructs, to our knowledge, ours is the first work that investigates DASNs, and provides a quantitative and qualitative characterization of objective discovery in such networks, namely, *detectability* and *decidability*.

The rest of the paper is organized as follows. In Section II, the detection and obstacle models are introduced. We formally characterize detectability of DASNs and propose an algorithm to compute the exposure metric in Section III. Decidability of obstacles and location algorithms are investigated in Section IV. In Section V, we present some numerical results to evaluate the complexity and effectiveness of the proposed algorithms. Finally, we conclude the paper with related work in Section VI and conclusion in Section VII.

II. Models and Preliminaries

Consider a set of actuators and sensors placed in a bounded 2-D region. A sensor can detect signals from an actuator if there exists an *unobstructed* path in-between, subject to distance constraints. In this case, the actuator is *visible* to the sensor. Reflections on exterior or interior boundaries may create non-line-of-sight paths. Such non-line-of-sight paths can be broken into a collection of line segments and treated individually in a similar fashion as line-of-sight paths between sensors and actuators (by introducing artificial actuators and sensors). Distance constraints are required due to attenuation of signal in the medium. This further limits the number of visible sensor-actuator pairs. Known exterior and interior boundaries can also be approximated by a collection of line segments. Given nodal location, layout and propagation properties of the medium, one can model a DASN as an arrangement of line segments. The interiors of line segments and their endpoints (corresponding to sensors and actuators) are called *sites* $s \in \mathcal{S}$. Note that two line segments can intersect at an interior point under general configurations. These are called *strongly intersecting* line segments. Otherwise, the segments are weakly or non-intersecting. An example is given in Figure 2(a), where four sensor/actuator A, B, C, D are visible to each other. Line segments AC and BD are strongly intersecting. AB and AD are weakly intersecting.

Obstacle model: Obstacles can have several effects on a DASN. First, an obstacle can destroy existing sensors and actuators. This can be detected by unavailability of the sensors and actuators e.g., by sending *keep-alive* messages to management entities or periodic polling. Second, it obstructs the propagation paths between sensors and actuators, making the actuators "invisible" to some sensors. In this paper, a simple thresholding mechanism is assumed, where if the wave strength received from an actuator is below a threshold ψ, the sensor determines that the path is obstructed. Though the magnitude or waveform of the received signal may provide additional information, such information is not always reliable, and thus not considered. Third, obstacles can create new propagation paths between sensors and actuators that are previously unreachable. In experiments carried out in beam structures [9], we do not observe much reflective waves. In this paper, only the first two effects are considered. Both cases can be modeled as *removal* of sites from $\mathcal{S}$.

An obstacle can be of different shapes, both convex and non-convex. To avoid the complexity of orientation, we approximate obstacles by their maximal *inner* circles. This results in a somewhat conservative estimation of the detectability of obstacles (i.e., if its inner circle is detectable, the obstacle must be detectable). However, as becomes evident in later sections, a circular model lends to elegant geometric constructs. Enlongated obstacles such as cracks, rectangular with narrow width, are modeled a collection of smaller inner circles.

Quality of obstacle discovery: The ability of a DASN in discovering obstacles can be characterized in two aspects:

- **Detectability:** Given the set of sites $\mathcal{S}$, what is the condition for an obstacle to be detectable? Clearly, the detectability of an obstacle depends on its dimension and location as well as the arrangement of sites. We are primarily interested in finding out how the arrangement of sites affect the DASN's ability to detect obstacles. Therefore, a *deployment-centric metric* is needed independent of location and dimension of obstacles.
- **Decidability:** Given the set of sites $\mathcal{S}$ and obstacles $\mathcal{O}$, can the number and location of obstacles be decided? at what accuracy?

When localizing obstacles, both computation complexity and message complexity are of interest. In this paper, we consider a centralized scheme where each sensor reports a binary value regarding the visibility of each actuator. A server upon gathering the visibility information, computes the location of the obstacles based on sensor input and knowledge of sensor locations. Sensor locations are assumed to be known using GPS or other localization mechanisms [19], [20], [22]. At the bootstrap phrase of a DASN, measurements are taken with no obstacle presented, which establishes the threshold value for detection. During runtime of the system, communication cost is constant irrespective of the presence of obstacles. Therefore, we are primarily interested in algorithms with low computation complexity.

III. Detection of obstacles in DASNs

In this section, we formally characterize the detectability of DASNs by introducing a novel deployment-centric metric, namely, "exposure". A polynomial-time algorithm is then developed to compute the exposure given arrangement of sensors and actuators.

A. Exposure

Detectability of DASNs is loosely analogous to the notion of coverage in passive sensor networks. In the coverage problem of passive sensor networks, the main goal is to eliminate the coverage holes in the network, where presence of events cannot be detected. Typically, disk or area sensing coverage models are assumed for passive sensors. As a result, asymptotic bounds can be derived for the critical density to ensure full coverage [26]. Many coverage algorithms [16], [23], [26] have been developed to select the minimum set of sensors to ensure certain coverage requirements (e.g., k-coverage). Coverage in DASNs is fundamentally different in that the ability to detect obstacles is determined by both the placement of visible actuators and sensors, as well physical medium and the environment layouts. The necessary condition for an obstacle to be detected is that it obstructs the propagation path between at least a pair of actuator and sensor. As discussed in more details in Section IV, strict "full" coverage is unattainable in DASNs as there can always be a small enough obstacle that can not be detected (or "slip through the cracks", figuratively speaking).

This observation motivates us to introduce the concept of "exposure" as *the maximum dimension of obstacles that* **cannot** *be detected by an arrangement of sites*. Though several metrics are possible, we choose to measure dimension by the radius of the maximum undetectable disks (Figure 2(a)). The exposure metric has several nice properties, i) it is *deployment-centric* and is *independent* of detailed characteristics of obstacles ii) it is readily extensible to higher spatial dimensions, and ii) it is rotation invariant. Recall that as mentioned in Section II, an obstacle can be detected if i) it destroys a sensor or actuator, and ii) it intersects with the propagation path between a pair of sensor and actuator. Therefore, exposure can be formally defined as follows.

Definition 1 (Exposure): Given an arrangement of sites $\mathcal{S}$ consisting of interiors of line segments connecting visible sensors and actuators, their endpoints and boundaries of the area, exposure is measured as the radius of the largest disk that does not intersect with any site in $\mathcal{S}$.

In the above definition, intersection of a disk and linear sites implies that the disk cuts the interior of these segments. On the other hand, intersection of the disk and point sites (i.e., the end points) implies that the point sites are inside the disk. The following claim establishes the physical meaning of exposure.

Claim 1: Given an arrangement of sensors and actuators, if an obstacle of arbitrary shape has an in-circle with radius larger than the exposure, such an obstacle is always detectable. However, the converse is not true.

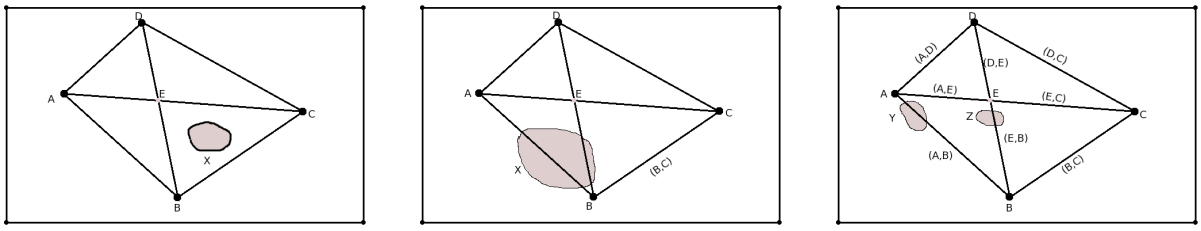
B. Algorithm

Before introducing the algorithm for determining exposure, we first explain the notion of segment Voronoi diagram.

Definition 2 (Segment Voronoi Diagram [3]): The Segment Voronoi Diagram (SVD) is defined over a set of non-intersecting sites, which can either be points or linear segments. The SVD is a subdivision of the plane into connected regions, called *cells*, associated with the sites. The cell of a site s_i is the locus of points on the plane that are closer to s_i than any other site s_j, $j \neq i$. The distance $\delta(x, s_i)$ of a point x in the plane to a site s_i is defined as the minimum of the Euclidean distances of x from the points in s_i.

The above definition can be easily extended to strongly intersecting line segments, in which, a single segment is broken into two by the intersection resulting in two interiors and three point sites. Our proposed algorithm to compute exposure is based on the observation that the Voronoi vertex associated with three sites is the center of the largest circle crossing all three sites and containing no sites in its interior. Let the maximum of the radii of all such circles be r. Then, exposure is given by $r - \epsilon$ for any arbitrary small positive ϵ.

Several algorithms have been proposed to compute SVDs in d-space [11]–[14]. We adopt the randomized incremental algorithm developed by Karavelas [11]. The main advantages of the algorithm are, it is stable and can handle both weakly intersecting and strongly intersecting sites. Now, suppose sites $\mathcal{S}$ are inserted in the digram and s is the site to be inserted. The basic step of the algorithm is as follows. First, find the first site that conflicts with the new site using the nearest neighbor query on a hierarchical data structure. Second, find all the sites that are conflicting with it by a depth-first search. Lastly, construct the new Voronoi diagram of $V(\mathcal{S} \cup s)$, and update the data structure reflecting the changes made. By the end of the procedure, SVD is created containing the information of vertices, edges, faces. To determine the exposure, distances from each Voronoi vertex to one of its incident sites are computed. Exposure is thus the maximum radius among all Voronoi vertices (minus an arbitrary small positive value).



(a) Obstacles smaller than exposure may not be detected

(b) A single obstacle intersecting with two line segments

(c) Two obstacles intersecting with two line segments separately.

Fig. 3: Detectability of DASNs: (a) cannot be detected; (b) and (c) are not distinguishable

Complexity analysis: Let the number of line segments be n and the number of intersections be $m = O(n^2)$. Computing SVD takes $O((m+n)\log^2 n)$ time. Each intersection adds two sites, therefore, the total number of sites is $n+3m$. Since the number of vertices is $O(n+3m)$ [3], it takes $O(n+3m)$ time to determine the exposure once the SVD is computed. Therefore, the total time complexity is $O((m+n)\log^2 n)$.

IV. Localization of obstacles in DASNs

Obstacles obstruct propagation paths between sensors and actuators, and render some sensor and actuator pairs in $\mathcal{S}$ invisible. They can also destroy sensors and actuators modeled as point sites. Among sites $\mathcal{S}$, $\mathcal{S}_f = \{s_1, s_2, \ldots, s_L\}$ are the sites that fail due to the presence of obstacles. The goal of obstacle localization is to determine i) the number of obstacles in the field, and ii) bounding regions where obstacles are most likely to be present. Before developing algorithms for obstacle localization, let us first discuss the fundamental decidability properties of DASNs.

A. Decidability

We summarize our observations in the following statements.

Claim 2 (Undecidability):

A. No matter how densely actuators and sensors are deployed, there exist obstacles that are undetectable.

B. The number of obstacles cannot be uniquely decided.

C. Location and shape of the obstacle cannot be uniquely decided.

Undecidability (A) is due to the positiveness of exposure. Since there always exist some non-empty cells in a SVD, obstacles located entirely in the interior of the cells cannot be decided (See example in Figure 3(a)). However, given an exposure requirement, one can strategically place sensors and actuators such that the size of maximum undetectable obstacles is bounded. Undecidability (A) necessarily implies (B) since if some obstacles are undetectable, it is impossible to determine whether there is no obstacle or many. Furthermore, when two sites fail, it is often impossible to decide whether it is a result of a single obstacle or multiple (See example in Figure 3(b) and (c)). Finally, if we move obstacles around or change their shape slightly, the set of $\mathcal{S}_f$ remain the same, which implies Undecidability (C).

Are DASNs ineffectual in obstacle localization given the above negative results? Our answer is *NO*. First, it is still possible to have a good estimation of the bounding regions that enclose the obstacles as long as the obstacles are sufficiently large. Second, with additional domain specific knowledge such as the minimum and maximum dimension of obstacles, minimum distance between obstacles or likelihood of presence, the uncertainty can be reduced. Furthermore, it has been shown that the frequency responses of obstacles of different sizes are different. For instance, low frequency waves can circumvent bigger obstacles better than high frequency waves, and vice versa. In this case, detection in a single wave length can be treated as a building block. By combining detections over multiple wavelengths, one can obtain more information regarding the location and dimension of obstacles.

Same principles apply to holes.

B. Conflict regions and their properties

As in the previous section, we model obstacles by their maximum inner disks. We introduce the notion of conflict region to describe the bounding regions where the center of obstacles can lie.

Definition 3 (Conflict regions): Given an arrangement of sites $\mathcal{S}$ and the failed sites $\mathcal{S}_f = \{s_1, s_2, \ldots, s_L\} \subset \mathcal{S}$, a conflict region $\mathcal{C}(A)$ w. r. t a set $A \subset \mathcal{S}_f$ is the union of the centers of disks, which intersect with all sites in A but none in S/A. We denote $\mathcal{C}_k = \bigcup_{|A_i|=k, A_i \in \mathcal{S}_f} \mathcal{C}(A_i)$, where $|\cdot|$ is the cardinality of a set. $\mathcal{C}_k$ is the set of all k-order conflict regions.

The concept of conflict region is closely related to order-k Voronoi diagram [3]. Given a set $\mathcal{S}$ of N sites, and an integer k between 1 and $N-1$, the order-k Voronoi diagram of $\mathcal{S}$, $V_k(\mathcal{S})$, partitions the space into regions (or cells) such that each point in a fixed region has the same k closest sites. Order-k Voronoi diagrams are an extension of Voronoi diagrams and are useful in answering k-nearest neighbor queries. Clearly, a SVD is an order- 1 Voronoi diagram. From the definition, a conflict region $\mathcal{C}(A)$ is a cell in the order-$|A|$ Voronoi diagram $V_{|A|}$ with closest neighbors in A.

Necessary and sufficient condition: Let $V_1(s_i, \mathcal{T})$ be the cell consisting of locus of point closer to site s_i than any other sites in $\mathcal{T}$. The necessary and sufficient condition for the existence of a conflict region $\mathcal{C}(A)$ is stated as follows.

Theorem 1: Let $A = \{s_{i_0}, s_{i_1}, \ldots, s_{i_{|A|-1}}\} \subset \mathcal{S}_f$. $\mathcal{C}(A) \neq \emptyset$ iff $\bigcap_{s_i \in A} V_1(s_i, \mathcal{S}/A) \neq \emptyset$.

Proof: We show a stronger result by proving the following equality,

$$\mathcal{C}(A) = \bigcap_{s_i \in A} V_1(s_i, \mathcal{S}/A) \tag{1}$$

If $\mathcal{C}(A) = \emptyset$, $\mathcal{C}(A) \subseteq \bigcap_{s_i \in A} V_1(s_i, \mathcal{S}/A)$ clearly holds. Now, suppose $\exists p \in \mathcal{C}(A)$. By definition, $\delta(p, s_i) \leq \delta(p, q_j)$, $\forall s_i \in A$ and $q_j \in \mathcal{S}/A$ (recall that $\delta(\cdot)$ is the Euclidean distance between a point and a site). This implies that, $p \in V_1(s_i, \mathcal{S}/A)$, $\forall i$. The converse can be proved similarly. ■

The above condition can be used to test the existence of conflict regions. To determine whether $\mathcal{C}(A) \neq \emptyset$, one can simply overlap $V_1(s_i, \mathcal{S}/A)$ for all $s_i \in A$. $V_1(s_i, \mathcal{S}/A)$ can be obtained by constructing the SVD for $\mathcal{S}/A$. However, this condition can not be directly used in characterizing the exact shape of the conflict regions. This prompts us to investigate a set of necessary conditions in the following.

Necessary condition: To determine all non-empty $\mathcal{C}(A)$ for $A \subset \mathcal{S}_f$, the key challenge is that the cardinality of valid A's is not known in advance. In this section, we study the necessary condition for the existence of k-order conflict regions to impose a bound on the maximum k that needs to be considered.

Definition 4 (Incidence): Two sites s_i and s_j are incident if the bisector $B(s_i, s_j)$ consisting of locus of points at equal distance to s_i and s_j is an edge of order - 1 SVD $V_1(\mathcal{S})$.

Definition 5 (Incidence graph): $G_I(\mathcal{S})$ is the incidence graph of a set $\mathcal{S}$, where the vertices are sites in $\mathcal{S}$, and an edge exists between two vertices s_i and s_j iff s_i and s_j are incident to each other in $V_1(\mathcal{S})$.

Lemma 1: If $\mathcal{C}(A) \neq \emptyset$ and $|A| \geq 2$, where $A = \{s_{i_0}, s_{i_1}, \ldots, s_{i_{|A|-1}}\} \subset \mathcal{S}$ then $\exists s_{i_s} \in A$ s. t. , $\mathcal{C}(A_s) \neq \emptyset$, where $A_s = A/s_{i_s}$; Furthermore, s_{i_s} is incident to at least one site in A_s.

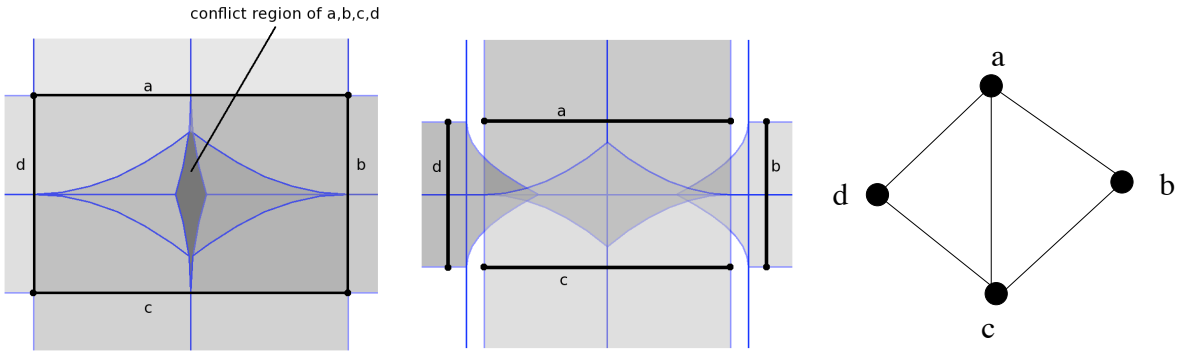
Proof: Since $\mathcal{C}(A) \neq \emptyset$, $\exists p \in \mathcal{C}(A)$, s.t., $\delta(p, s_i) \leq \delta(p, q_j)$, $\forall\ s_i \in A$ and $q_j \in \mathcal{S}/A$. Let $s_{i_s} \in A$ be the site that is *farthest* among sites in A to point p. Denote $A_s \equiv A/s_{i_s}$. Clearly, $\mathcal{C}(A_s) \neq \emptyset$ since $p \in \mathcal{C}(A_s)$.

To show the second half of the lemma, we prove by contradiction. Suppose s_{i_s} is not incident to any site in A_s. Let $V_1(s_{i_s}, \mathcal{S}/A_s)$ be the cell consisting of locus of points closest to s_{i_s} than any other point in $\mathcal{S}/A_s$. From the assumption, we have $V_1(s_{i_s}, \mathcal{S}/A_s) = V_1(s_{i_s}, \mathcal{S}/s_{i_s})$.

We first show that $\mathcal{C}(A_s) \cap V_1(s_{i_s}, \mathcal{S}/A_s) \neq \emptyset$. If $\mathcal{C}(A_s) \cap V_1(s_{i_s}, \mathcal{S}/A_s) = \emptyset$, any point that is closer to sites A/s_{i_s} are further away from i_s compared to other remaining sites in $\mathcal{S}/A$. This contradicts $\mathcal{C}(A) \neq \emptyset$. Therefore, $\mathcal{C}(A_s) \cap V_1(s_{i_s}, \mathcal{S}/s_{i_s}) \neq \emptyset$. In another word, there exists a point $q \in \mathcal{C}(A_s)$ and $q \in V_1(s_{i_s}, \mathcal{S}/s_{i_s})$. The former implies, $\delta(q, s_i) > \delta(q, s_{i_s})$, $\forall s_i \in A_s$; and the later implies that $\delta(q, s_i) < \delta(q, s_{i_s})$, $\forall s_i \in A_s$. Hence, we have the contradiction. ■

The corollary immediately follows when $|A| = 2$.

Corollary 1: If $\mathcal{C}(\{s_1, s_2\}) \neq 0$, s_1 and s_2 are incident sites. Furthermore, $\mathcal{C}(\{s_1, s_2\}) \cap B(s_1, s_2) \neq \emptyset$.



(a) By overlapping $V_1(s_i, \mathcal{S}/\{a,b,c,d\})$, where $s_i = a, b, c, d$, one can find a common 4-order conflict region $\mathcal{C}(\{a,b,c,d\})$

(b) There is no overlap among $V_1(s_i, \mathcal{S}/(\{a,b,c,d\})$, where $s_i = a, b, c, d$. 4-order conflict region $\mathcal{C}(\{a,b,c,d\}$ does not exist

(c) Incidence graph of failed sites $\{a,b,c,d\}$.

Fig. 4: Existence of 4-order conflict regions on two arrangements with the same incidence graph but different conflict regions. Sites a, b, c, d are obstructed.

The above corollary implies a one-to-one mapping between cells in an order - 2 SVD with bisectors in the order-1 SVD.

Theorem 2: If $\mathcal{C}(A) \neq \emptyset$, where $A = \{s_{i_0}, s_{i_1}, \ldots, s_{i_{|A|-1}}\} \subset \mathcal{S}$ then the subgraph induced by A in $G_I(\mathcal{S})$ is connected.

Proof: Since $\mathcal{C}(A) \neq \emptyset$, $\exists p \in \mathcal{C}(A)$, s.t., $\delta(p, s_i) \leq \delta(p, q_j), \forall\ s_i \in A$ and $s_j \in \mathcal{S}/A$. Without loss of generally, we assume that the following order holds (otherwise, we can always sort sites in A accordingly),

$$\delta(p, s_{i_0}) \leq \delta(p, s_{i_1}) \leq \ldots \leq \delta(p, s_{i_{|A|-1}})$$

We prove the statement by induction. From Corollary 1, s_{i_0} and s_{i_1} are incident. Suppose the subgraph of G_I induced by sites $s_{i_0}, s_{i_1}, \ldots, s_{i_l}$ are connected. Then, from Lemma 1, $s_{i_{l+1}}$ is incident to at least one site in $s_{i_0}, s_{i_1}, \ldots, s_{i_l}$. Therefore, s_{i_0} up to $s_{i_{l+1}}$ are connected in G_I. ■

Comment: The above theorem gives the necessary condition for the existence of high-order conflict regions based on the incidence graph. However, incident relationship is not sufficient to determine such existence. One example is given in Figure 4, where two arrangements of sites have the same incidence graph but different conflict regions (Figure 4(c)). The first arrangement, which consists of four segments and their intersections forming a rectangle, admits a 4-order conflict region. In the second arrangement, which is the result of moving the two side edges of the rectangle slightly away from each other, one can only find up to 3-order conflict regions. In another word, there exists a disk that cuts all four sites in the first arrangement but not in the second one. In Figure 4(a) and (b), existence of conflict regions is verified using overlaps among order-1 Voronoi cells by Theorem 1.

Next, we examine the property of vertices of order-k conflict regions. In a general configuration, no four sites are co-circular (i.e., having a circle touching all four sites). Therefore, vertices in order-k segment Voronoi are at equal distance to at most three sites.

Let the operand $\oplus$ between two sites A and B denotes $A \oplus B = (A - B) \cup (B - A)$, also known as the "symmetric difference" between two sets.

Lemma 2: The pair of sites $\{s_i, s_j\}$ associated with an edge, which borders two order-k conflict regions $\mathcal{C}(A)$ and $\mathcal{C}(B)$, satisfies $\{s_i, s_j\} \oplus A \oplus B = \emptyset$. In another word, if one goes from one k-order conflict region to its adjacent region, only the k-th nearest neighbor is changed.

Proof: This result can be proved by adapting the proof of Lemma 5 [15] on point Voronoi diagrams to SVDs, and is omitted due to space limit. ■

Theorem 3: Let $\mathcal{C}(A_1)$ be a valid k-order conflict region. The vertices of $\mathcal{C}(A_1)$ are at equal distance to three sites that satisfy i) one is an element of A_1, and ii) the sites are connected in the incidence graph.

Proof: Consider a vertex at the intersection of three k-order conflict regions, $\mathcal{C}(A_1)$, $\mathcal{C}(A_2)$ and $\mathcal{C}(A_3)$. Let the respective sites associated with the vertex be $s_{i_1}, s_{i_2}, s_{i_3}$. Clearly, $s_{i_1} \in A_1$ and thus the first property holds.

From Lemma 2, we have $A_1/s_{i_1} = A_2/s_{i_2} = A_3/s_{i_3}$. From Theorem 2, s_{i_1}, s_{i_2} and s_{i_3} are incident to sites in A_1/s_{i_1}. Therefore, the second property holds. ■

C. Algorithm

Several algorithms have been developed to construct order-k point Voronoi diagrams, where all sites are points in d-space. Lee [15] showed that V_k has $O(k(n-k))$ regions and also proposed an algorithm to construct V_k in $O(k^2 nlogn)$ times. In [5] and [2], randomized algorithms of complexity $O(kn^{1+\epsilon})$ and (roughly) $O(k(n-k)logn)$ were proposed, respectively. Aurenhammer and Klein [4] developed an randomized $O(k^2(n-k)logn)$ incremental algorithm. To the best of our knowledge, construction of order-k segment Voronoi diagram has not been addressed in the literature. Though conceptually, many properties of order-k point Voronoi diagrams remain valid for order-k SVD, techniques for the former do not always carry over to the later for several reasons. First, bisectors in SVDs are not all linear. Second, line segments can intersect in their interiors. Third, d-space line segment Voronoi cannot be readily mapped from intersecting hyperplanes and convex hulls in $d+1$-space [3].

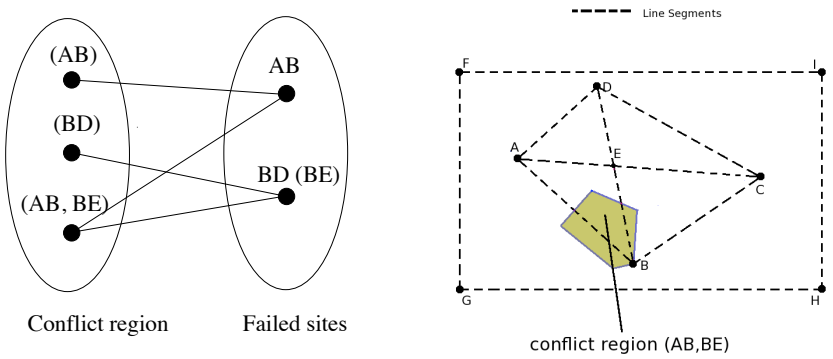
1) Determining conflict regions: Let $G_I(\mathcal{S}_f)$ be the induced subgraph of the incident graph $G_I(\mathcal{S})$. In this section, we propose an algorithm to compute all conflict regions *up to* an order K, where K is the size of the largest connected component in $G_I(\mathcal{S}_f)$. By Theorem 2, no conflict regions of higher order exists.

The basic idea of the algorithm goes as follows. For each vertex that satisfies the conditions in Theorem 3, we sort its distances to a set of sites in non-decreasing order. If the first k sites are failed sites in $\mathcal{S}_f$, then the vertex is on the edge of the k-order conflict region due to these sites. To limit the set of sites to be considered, we utilize the incidence graph and connectivity constraint in Theorem 2. Next, we present the details of the algorithm.

- **Step 1.** Construct the order-1 SVD of all sites in $\mathcal{S}$. Determine the incident graph $G_I(\mathcal{S})$ and its induced subgraph $G_I(\mathcal{S}_f)$.
- **Step 2.** Starting from any site s_i in $\mathcal{S}_f$, a depth-first search is performed to determine the maximal connected component containing s_i on $G_I(\mathcal{S}_f)$. Remove all sites contained in the connected component from $\mathcal{S}_f$ and repeat the same procedure until no more sites left in $\mathcal{S}_f$. This gives all the connected components in the subgraph $G_I(\mathcal{S}_f)$, denoted by $G_I^1, G_I^2, \ldots, G_I^M$.
- **Step 3.** Each connect component G_I^i, $i = 1, 2, \ldots, M$ is augmented by including sites in $\mathcal{S}/\mathcal{S}_f$, which are incident to at least one of the site in G_I^i, denoted by $\overline{G_I^i}$.
- **Step 4.** For each $\overline{G_I^i}$, three sites are selected with at least one site from $\mathcal{S}_f$. The vertex at equal distance to all three sites is computed.
- **Step 5.** For each vertex v determined in the previous step, the distance to all sites in $\overline{G_I^i}$ is computed and sorted in a non-decreasing order. Let the $l+1$th site be the first one not in $\mathcal{S}_f$ in the ordered list. The first l sites are then stored along with the distance to the $l+1$th site. The l sites form an l-order conflict region. The distance to the $l+1$th site gives the radius of a disk centering on the vertex containing the l sites and touching the $l+1$th site.
- **Step 6.** All vertices associated with the same set of sites are found, and stored in a link list.

At the end of the procedure, we have a collection of conflict regions and their vertices in high-order SVDs up to order K. Consider a l-order conflict region $A = \{s_1, s_2, \ldots, s_l\}$. The radius $\mathcal{R}_{max}(A)$ of the maximum disk that intersects sites in A but none other sites is given by the maximum of the radii associated with A's surrounding vertices. Let s_{i_s} in A s.t., $A_s = A/s_{i_s}$ is a conflict region. Similarly, the radius $\mathcal{R}_{min}(A)$ of the smallest disk that intersects sites in A is estimated. These two parameters are used in determining the likely locations of obstacles. To this end, we obtain for each non-empty conflict region $\mathcal{C}(A)$, its associated sites, $\mathcal{R}_{max}(A)$ and $\mathcal{R}_{min}(A)$. Clearly, $\mathcal{R}_{min}(A)$ of order-1 conflict region is zero.

Complexity analysis: Let the number of line segments be n and the number of intersections be $m = O(n^2)$. End points of line segments are considered to be separate sites from their interiors. Therefore, the total number of sites is $n+3m$. Since each Voronoi vertex has at least three edges, it has been shown that the number of vertices v, edges e and faces f in the SVD are all $O(n+3m)$ [3]. The number of edges c bounding a face is thus constant. In Step 1, the time complexity to construct the order-1 SVD is $O((n+m)log^2(n))$ using the randomized incremental algorithm in [11]. The incidence graph has the same number of vertices as the sites, and the same number of edges as the edges of the SVD. The complexity of depth-first search to traverse the incidence graph is thus $O(n+3m)$



(a) Bipartite graph of two failed sites AB and BD in Figure 3(b)(c)

(b) A 2-order conflict region (AB, BE).

Fig. 5: An illustrative example to decide obstacle location using minimum set cover.

in Step 2. For a component G_I^i of size k, its augmented graph $\overline{G_I^i}$ is at most of size $(c+1)k$ and thus $O(k)$. The number of three-tuples selected in Step 4 is $O(k^3)$. Computing the circumcenter takes constant time. In Step 5, $O(k \log k)$ time is required to sort the distance to all sites in $\overline{G_I^i}$. Let $\overline{k}$ be the average size of connected components. It should be noted that in general one cannot distinguish failures of an entire segment or its many partitions (due to strong intersections). Therefore, distance to a line segment is only computed once. The total number of components is $\frac{n}{\overline{k}}$. Therefore, the total time complexity to determine the conflict regions is $O(n\overline{k}^3 \log \overline{k} + (n+m)log^2(n))$.

2) Localizing obstacles: The conflict regions characterize bounding areas where the center of the obstacles may lie. From Lemma 1, existence of a conflict region $\mathcal{C}(A)$ of order not less than 2 necessarily implies the existence of conflict regions of lower order. For example, consider $A_1 = \{s_1, s_2\}$, $A_2 = \{s_1\}$, $A_3 = \{s_2\}$. Furthermore, $\mathcal{C}(A_i) \neq \emptyset$, $i = 1, 2, 3$. Two possible scenarios can occur. In the first case, there is a (large) obstacle intersects with both sites, s_1, s_2. In the second case, there are two (smaller) obstacles intersecting with only s_1 and s_2, respectively. Both scenarios are feasible since the sites in A are covered. To reduce the uncertainty in obstacle localization, further domain specific knowledge is required. Identifying obstacle locations can then be formulated as a set cover problem.

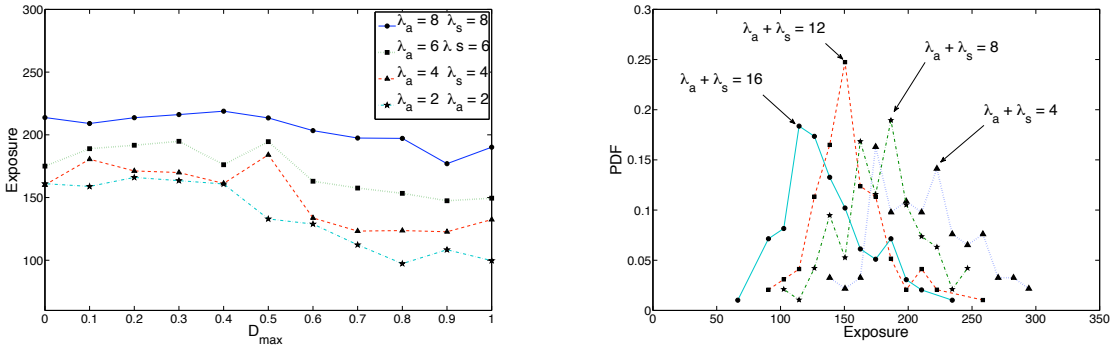
A weighted bipartite graph is constructed consisting of two disjoint partitions X and Y. The first partition X consists of sets of sites in $\mathcal{S}_f$ that form conflict regions. Elements in the second partition Y are sites in $\mathcal{S}_f$. Different partitions of the same segments due to intersection are merged to a single vertex in Y. An edge exists if a site $s_i \in Y$ is an element of a set A in X. The weight of a vertex $A \in X$ in the conflict region is a function f of $\mathcal{R}_{min}(A)$, $\mathcal{R}_{max}(A_s)$ and likelihood of the presence of obstacles in $\mathcal{C}(A)$. The function is determined by domain specific knowledge. Localization of obstacles is then to find a minimum weight set cover using the sets in X to cover elements in Y. Finding minimum weight set cover is an NP hard problem. It is however $1 + ln|Y|$ approximable [25]. In the localization problem, failed sites belonging to different connected components do not form conflict regions. Each connected component can be treated separately in the bipartite graph. Thus, $|Y|$ is limited to the size of the maximum connected component in $G_I(\mathcal{S}_f)$. If f is set to be an identity function, i.e., $f(\cdot) \equiv 1$, then the algorithm finds a minimum set cover.

Consider the example in Figure 3(b)(c). The failed sites are AB and BD. Since AB and BE are incident, there exists a 2-order conflict region $\{AB, BE\}$. The resulting bipartite graph is shown in Figure 5(a). The minimum set cover is then (AB, BE). The corresponding 2-order conflict region is shown in Figure 5(b).

The algorithm used to compute the minimum weight set cover of sites in a connected component G_I^i, $i = 1, 2, \ldots, M$ is given in Algorithm 1. It is known to be best possible unless $P \equiv NP$. For each element A of the set cover C, $R_{max}(A)$ gives the bounding area of the corresponding obstacles.

Algorithm 1: Minimum weighted set cover algorithm

input : $G_I = \{s_{i_1}, s_{i_2}, \ldots, s_{i_l}\}$, A_j, $w(A_j)$, where $\mathcal{C}(A_j) \neq \emptyset$ and $\bigcup A_j = G_I$
output: set cover C
begin
  $C \leftarrow \emptyset$
  while $C \subset G_I$ **do**
    choose an A_j with minimal $\frac{w(A_j)}{|A_j - C|}$
    $C \leftarrow C \cup A_j$
  end
  return C
end



(a) Exposure with respect to sensing range

(b) Histogram of exposure

Fig. 6: Impact of arrangement of sites on exposure, $D_{max} = 300$

V. Evaluation

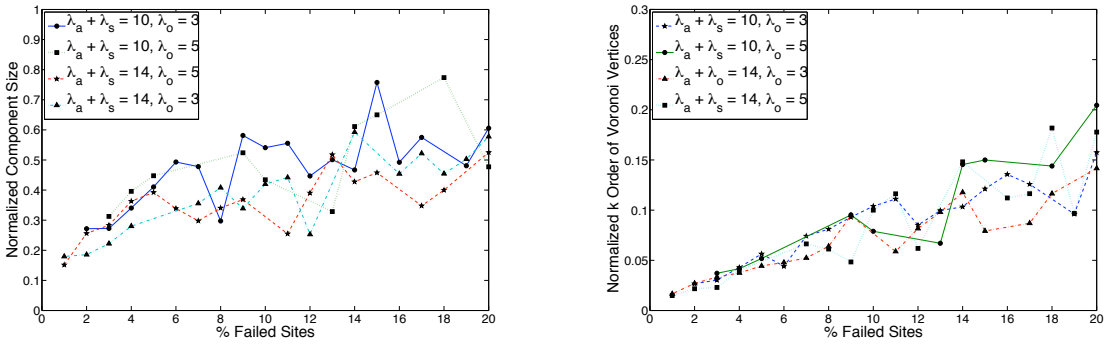
In this section, we present numerical results to evaluate the quality of obstacle discovery in DASNs. We have extended the CGAL library [1] with the proposed algorithms. In the experiments, no *a prior* knowledge regarding obstacle location is assumed. Sensors, actuators and obstacles are placed randomly with density λ_s, λ_a, and λ_o, respectively, in a square area of dimension $600 \times 600m^2$ following Poisson point processes.

Impact of node density and sensing range on exposure: Figure 6 gives the average exposure under different node density and sensing range, and its distribution with fixed sensing range ($D_{max} = 300$). As expected, as the actuator/sensor density increases or with larger D_{max} (thus more line segments exist in the field), the exposure decreases. However, compared to the field dimension, the exposure value is still quite significant. This is mainly due to the fact that random placement tends to leave large holes in the networks with no sensors, actuators and/or line segments.

Impact of percentage of failed sites: Next, we vary radius of obstacles, node densities but limit the percentage of failed sites below 20%. In this and remaining sets of experiments, the radius of obstacles is chosen from 0.1 to 0.4 of the exposure. Figure 7(a)(b) give the size of connected components and maximum k among all high-order conflict regions. Both are scaled by the total number of sites. As the percentage of failed sites grows, both two quantities increase. Interestingly, from the figures, we observe that the density sensor/actuator/obstacle has little effect on the quantities of interest compared to the percentage of failed sites.

Obstacle localization: For obstacle localization, we consider two choices in the minimum set cover. The first one (unweighted) sets $f(\cdot) \equiv 1$, i.e., all sets have equal weights. The second one (weighted) sets $f(A_j) = 0$, if $R_{max}(A_j) < \epsilon_{min}$ or $R_{max}(A_j) > \epsilon_{max}$, where A_j is a conflict region, ϵ_{min} and ϵ_{max} are the minimum and maximum obstacle size in the field; and $f(A_j) = 1$ otherwise.

Figure 8 gives the mean number of obstacles and the standard deviation with respect to the number of non-overlapping obstacles in the field. We see that the proposed algorithms can detect and localize obstacles much smaller than exposure, however they tend to under-estimate the number of obstacles for two reasons i) some obstacles are not detectable due to their small size and location, and ii) obstacles tend to cluster together when the



(a) Normalized connected component size (b) Normalized max k-order

Fig. 7: Impact of percentage of failed site on obstacle detection, $D_{max} = 300$

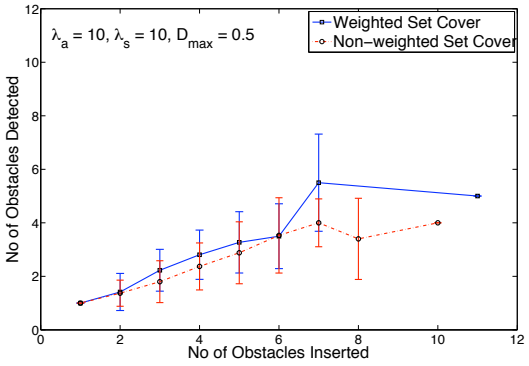


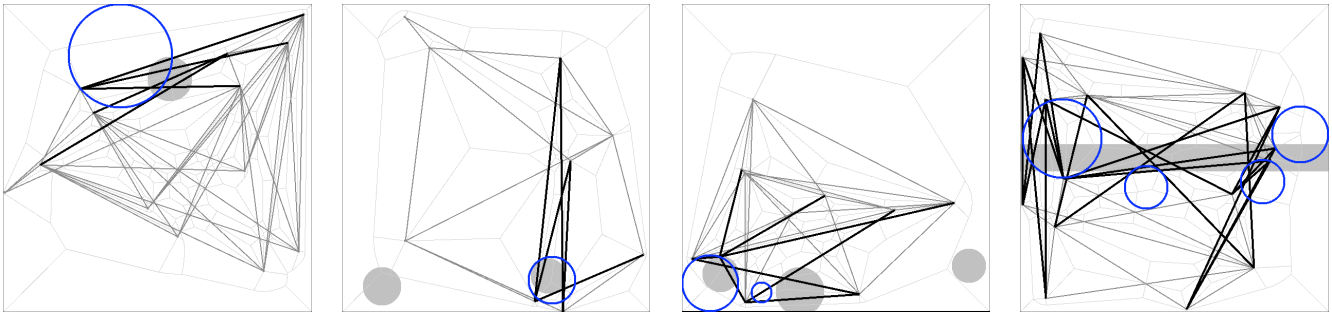
Fig. 8: Number of obstacles detected with respect to the number of actual obstacles

number gets large. The weighted set cover algorithm removes obstacles exceeding radius ϵ_{max} and thus has better accuracy. To see the obstacles detected by the proposed set cover for obstacle localization, we plot in Figure 9 the actual location of obstacles, and the maximal circles cutting failed sites but none others.

VI. Related Work

To the best of our knowledge, this is the first work that addresses obstacle discovery in DASNs. Much work has been done to detect coverage and communication holes in wireless sensor networks, which can be broadly categorized into two groups, namely i) those utilizing location information and ii) those utilizing connectivity information. Fang *et al*. [7] proposed a simple algorithm that greedily sweeps along hole boundaries and discover boundary cycles. Location information is assumed to be known and unit disk connectivity is required. Ghrist and Muhammad [10] proposed a centralized method to detect coverage holes by means of homology utilizing network communication graphs. Assuming that the connectivity is determined by the unit-disk graph model, Funke and Klein [8] proved that using linear-time algorithm, one can identify nodes on the boundaries of holes of the network. Wang *et al*. [24] developed a practical distributed algorithm for boundary detection in sensor networks, using only the communication graph. The basic idea is to identify irregularity in the communication graph. To form a hole, the diameter of the holes is larger than communication range. In DASNs, if we can view the connectivity between actuators and sensors as a graph, diameters of the obstacles can be smaller than the actuation "range".

In [21], Reichenbach *et al*. proposed a distributed obstacle localization algorithm. All sensors' location are known. The basic idea is to use a set of beacons with long communication range to partition a region into areas with different visibility to the beacons. Boundaries and corners among the areas yield a rough estimation of the obstacle location. Only one obstacle is considered in [21]. It is well known that connectivity does not imply existence of line-of-sight paths; and communication ranges change in different environments and over time. Therefore, obstacle detection based on connectivity is not reliable.



(a) One actual obstacle (b) Two actual obstacles (c) Three actual obstacles (d) Rectagular obstacle

Fig. 9: Detection of obstacles. Obstacles are shown as shaded. Solid dark lines are failued sites. Light lines are non-failed sites. Dashes lines are Voronoi edges. Blue cirlces give the maximal cicles cutting failed sites but none other. Obstacles are generated with average size 0.3 of the exposure

The concept of exposure is used in [17], [18] though in a different context. Meguerdichian *et al*. consider barrier coverage to minimize the probability of undetected penetration through the barrier. Exposure is defined as as an integral of a sensing function that depends on distance from sensors on a path from a starting point p_S to destination point p_D. In [17], [18], sensors are assumed to have a probabilistic coverage around its location. In contrast, in DASNs, detection of moving objects depends on the location of sensors and actuators.

VII. Conclusion

In this paper, we investigate feasibility of DASNs in obstacle discovery. Polynomial-time algorithms have been proposed to determine the exposure of DASNs and identify the bounding regions where centers of obstacle may lie. DASNs are advantageous compared to passive sensing networks or stand-alone active sensor systems in their low complexity and ability to observe obstacles at close proximity. This paper answers fundamental questions regarding the detectability and decidability of obstacle discovery in DASNs and establishes their use as an alternative sensing paradigm in applications such as robotics and structural health monitoring. This opens up to many interesting problems in both algorithm design and system engineering. As an ongoing work, we are devising algorithms that reduce the computation complexity and achieve better localization results.

References

[1] CGAL - computational geometry algorithms library.

[2] P. K. Agarwal, M. de Berg, J. Matousek, and O. Schwarzkopf. Constructing levels in arrangements and higher order voronoi diagrams. *SIAM J. Comput*., 27(3):654–667, 1998.

[3] F. Aurenhammer and R. Klein. *Voronoi Diagrams*. Elsevier Science Publishers, 1998.

[4] F. Aurenhammer and O. Schwarzkopf. A simple on-line randomized incremental algorithm for computing higher order voronoi diagrams. In *SCG '91: Proceedings of the seventh annual symposium on Computational geometry*, pages 142–151, New York, NY, USA, 1991. ACM.

[5] K. L. Clarkson. Applications of random sampling in computational geometry, ii. In *SCG '88: Proceedings of the fourth annual symposium on Computational geometry*, pages 1–11, 1988.

[6] A. Discant, A. Rogozan, C. Rusu, and A. Bensrhair. Sensors for obstacle detection - a survey. In *30th International Spring Seminar on Electronics Technology*, 2007.

[7] Q. Fang, J. Gao, and L. J. Guibas. Locating and bypassing holes in sensor networks. *Mob. Netw. Appl*., 11(2), 2006.

[8] S. Funke and C. Klein. Hole detection or: "how much geometry hides in connectivity?". In *Proceedings of the twenty-second annual symposium on Computational geometry*, pages 377–385, 2006.

[9] S. G., G. H., M. Y. L., H. T., D. H., and Z. R. R. H. Health monitoring of a concrete structure using piezoceramic materials. *Proceedings of the SPIE - The International Society for Optical Engineering*, 5765:108–119, 2005.

[10] R. Ghrist and A. Muhammad. Coverage and hole-detection in sensor networks via homology. In *Proceedings of the 4th international symposium on Information Processing in Sensor Networks (IPSN)*, 2004.

[11] M. I. Karavelas. A robust and efficient implementation for the segment voronoi diagram. In *International Symposium on Voronoi Diagrams in Science and Engineering*, pages 51–62, 2004.

[12] D. G. Kirkpatrick. Efficient computation of continuous skeletons. In *20th Annual Symposium on Foundations of Computer Science (SFCS)*, pages 18–27, 1979.

[13] R. Klein, K. Mehlhorn, and S. Meiser. Randomized incremental construction of abstract voronoi diagrams. *Comput. Geom. Theory Appl*., 3(3):157–184, 1993.

[14] D. T. Lee. Medial axis transformation of a planar shape. *IEEE transactions on pattern analysis and machine intelligence*, PAMI-4(4):363–369, 1982.

[15] D. T. Lee. On k-nearest neighbor voronoi diagrams in the plane. *IEEE Trans. on Computer*, C-31(6), 1982.

[16] X. Li, P. Wan, , and O. Frieder. Coverage in wireless ad hoc sensor networks. *IEEE Transactions on Computers*, 52(6), 2003.

[17] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *INFOCOM*, 2001.

[18] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak. Exposure in wireless ad hoc sensor networks. In *MobiCom*, 2001.

[19] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 50–61, 2004.

[20] D. Niculescu and B. Nath. Error characteristics of ad hoc positioning systems (aps). In *MobiHoc '04: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pages 20–30, 2004.

[21] F. Reichenbach, R. Salomon, and D. Timmermann. Distributed obstacle localization in large wireless sensor networks. In *IWCMC '06: Proceedings of the 2006 international conference on Wireless communications and mobile computing*, pages 1317–1322, 2006.

[22] A. Savvides, C.-C. Han, and M. B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 166–179, 2001.

[23] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration in wireless sensor networks. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 28–39, New York, NY, USA, 2003. ACM Press.

[24] Y. Wang, J. Gao, and J. S. Mitchell. Boundary recognition in sensor networks by topological methods. In *MobiCom '06: Proceedings of the 12th annual international conference on Mobile computing and networking*, pages 122–133, 2006.

[25] D. B. West. *Introduction to Graph Theory (2nd ed.)*, chapter 3. Prentice Hall, 1999.

[26] H. Zhang and J. C. Hou. Maintaining sensor coverage in large scale sensor networks. In *International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad hoc Wireless and Peer-to-Peer Networks*, Feb. 2004.