Optimizing Sensor Deployment With Line-Of-Sight Constraints: Theory and Practice

Kin Sum Liu*, Brent Schiller†, Jie Gao*, Shan Lin†, Joseph S.B. Mitchell♭
*Computer Science Department, {kiliu, jgao}@cs.stonybrook.edu
†Department of Electrical and Computer Engineering, {brent.schiller, shan.x.lin}@stonybrook.edu
♭Department of Applied Mathematics and Statistics, jsbm@ams.sunysb.edu
Stony Brook University

Abstract

Various non-isotropic sensors, such as acoustic, visible light, and infrared sensors, heavily rely on the line of the sight signal propagation to achieve desired sensing and monitoring quality in a complex environment. Although researchers have tested these sensing systems in many real scenarios, there is still limited theory to guide the sensor deployment with realistic sensing characteristics. In this paper, we design deployment algorithms for robust coverage under specific angle of arrival sensing requirements. We formulate the optimal deployment problem as a robust variant of the Art Gallery Problem called *robust guarding*, i.e., placing minimum number of transmitters such that all points of the domain are covered by two sensors from sufficiently different directions. We prove that this problem is NP-hard and provide combinatorial upper and lower bounds for the number of sensors needed. Furthermore, we show that $n/2$ guards are always sufficient and sometimes necessary for rectilinear polygons. In the system evaluation, we developed a testbed using low cost off-the-shelf IR sensors for indoor device-free localization. Experiments with both simulation and real system show that our solution outperforms existing algorithms on sensing accuracy and coverage significantly with almost negligible overhead.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design

General Terms

Design, Experimentation, System, Performance

Keywords

Sensor networks, Line-of-sight sensor, Art gallery problem, Localization

1 Introduction

Various non-isotropic sensors, such as camera, ultrasound, and infrared sensors, have been deployed for safety surveillance [22], occupancy monitoring [8], and target tracking [6]. These non-isotropic sensors rely on signals that propagate in a directional manner, casting shadows on targets, which can be exploited for indoor monitoring and sensing purposes. These signals are preferred over radio-frequency (RF) for the relatively simple propagation model (as compared to the complex propagation properties of RF in an indoor environment) as well as the relatively low interference with other signal sources in the same domain.

However, while the signal propagation by infrared or camera sensors can be easily described and modeled, the line of sight constraint creates non-trivial challenges for sensor deployment and scheduling. The environment in which the sensors are deployed is typically an indoor setting with complicated geometric features. Therefore to ensure full coverage we often immediately encounter the classical Art Gallery Problem – placing a minimum number of sensors such that any point in the domain has a direct line of sight path to at least one sensor. In the setting for indoor monitoring, we may have additional constraints to guarantee quality of sensing. First, for robustness it often happens that we wish the targets to be visible to multiple sensors. In addition, by using the interaction of the signals/sensors one may infer additional knowledge other than just the existence of targets. For example, infrared sensors cast shadows when targets are around. The shadow could be used to infer the location of the target if the target is visible to at least two sensors – trace the line between the infrared emitters and their respective shadows; the target is located at the intersection. In addition, the two sensors better shine on the targets from different angles and the quality of the sensing outcome depends on the angular separation. In the extreme setting if the two sensors are from exactly the same direction they behave as the same sensor. The closer they are from the same direction the worse the quality of resolution we obtain. In the setting of camera sensors, it is often desirable to have pictures of the target taken from different views. Again it would be good if the target is visible to multiple sensors from different angles.

The main contribution in this paper is to examine the sensor deployment and coverage problem with line of sight con-

straints and the angular requirement. We model the input domain as a polygon with holes (e.g., obstacles). We formulate the problem as the α-robust guarding, in which each emitter casts signal in all directions which is only blocked by obstacles/walls. Further, we wish each point in the domain is visible to at least two emitters, such that the angle between them is above a given threshold α, $0 < \alpha \leq \pi/2$. This is a special variant of the classical Art Gallery Problem and has not been investigated much. Almost nothing is known theoretically about this problem even for the most basic questions such as: how many emitters do we need in a polygon with n vertices? how many emitters are always enough? How many emitters are at least needed? What if the polygon is rectilinear (with only horizontal and vertical edges)? In this paper we initiate the study of this problem and provided various upper and lower bounds. We show that finding a minimum number of guards to α-robustly guard a simple polygon is NP-hard. For a general polygon of n vertices, $\lfloor 3n/4 \rfloor$ guards are sometimes necessary (while for a standard art gallery problem $\lfloor n/3 \rfloor$ guards are both sufficient and sometimes necessary). For rectilinear polygons and placements of guards on the vertices, we show that $\lfloor n/2 \rfloor$ guards are sometimes necessary, and surprisingly, always sufficient – we prove that placing guards in alternating order along the polygon boundary is an α-robust guarding solution. This provides not only tight bounds for robust guarding in rectilinear cases but also a practical algorithm for sensor deployment in indoors when the region can be approximated by a rectilinear polygon.

Further, we apply our results for sensor placement in an environment sensing and tracking application. We developed Infrared Localization nodes, called iLoc nodes, with off-the-shelf infrared (IR) emitters and sensors with component costs of only a few dollars. In this testbed, the infrared emitters are deployed using our deployment algorithm and the infrared sensors are placed along the wall/furniture to capture the shadows caused by the targets and/or the reflections from the targets.

We develop algorithms for both single target and multiple target detection using our testbed. Experimental results show that our design detects human location in an office setting in real-time with an average error of 11.7 cm which outperforms many other device-free techniques. The contributing factors for this high precision localization are the accurate detection of shadows and reflections by the new hardware design, the collaborative sensing of the localization scheme, and our guard deployment algorithm.

2 Related Work

Robust guarding. The problem of α-robust guarding has not been studied much. The problem was first proposed in [7] as one of the two models used to ensure that a polygon would be robustly guarded by a set of guards. Then a $O(\log|\mathrm{OPT}|)$ approximation was proposed for the robust guarding problem inside a simple polygon using the standard technique of a geometric set cover where OPT is the minimum number of guards needed. Some more observations were made in [2] for the case when α is in between $\pi/2$ and π. Specifically, it was shown that any polygon of n vertices can be $2\pi/3$ robustly guarded by the n vertices and this is sometimes necessary. The upper and lower bounds are generalized for other values of $\alpha \in [\pi/2, \pi]$. From an algorithmic perspective, it proposed algorithms for detecting if a simple polygon can be α-guarded by a given set of guards and, if not, by computing the region of α-robustly guarded points.

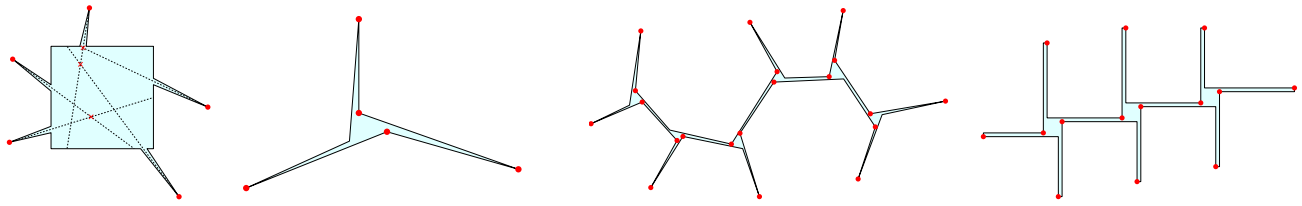
The results of this paper add to the collection of results above. Our hardness results and tight bounds for rectilinear polygons are new. The problem of robust guarding is also related to the coverage problems encountered when using cameras. A popular model of robust coverage when using cameras as sensors models each camera as a fixed angle cone with a radius r. In certain applications such as surveillance, it is important to capture the front face image of the target person which requires that every point in the domain is covered by all directions. Under this model, Wang and Cao [22] developed an algorithm to test whether a given set of cameras could cover all points of the domain and they also provided bounds on the density of cameras needed in a random deployment.

Indoor Localization. Device-based localization assumes that the target carries a special device. Such a device either measures the distances to nearby anchor nodes or generates signals for nearby anchor nodes to use to localize the device. Various device-based localization systems have been developed based on wireless [5, 4, 28, 21, 26, 24], optical [9, 16], and acoustic signals [18, 10]. Some other works introduces inertial sensing [11, 25], magnetic sensing [1] and even indoor GPS [14] to assist in localization. These works can achieve high localization accuracy but their applicability may be limited due to privacy and cost concerns.

In contrast, device-free localization techniques remove the burden of users carrying around devices and have received a lot of attention recently [12, 27]. Existing systems include indoor GPS [14], radar [12], and camera networks [22, 3]. These localization systems achieve high localization accuracy with specially designed devices but their cost is relatively higher. Other works use ultrasound sensors [8] and pressure sensors [15]. These systems detect targets reliably in specific deployment locations [19] but they do not cover the whole building due to limited sensing range.

A number of device-free localization works use WiFi to capture signal variations caused by target movement. In [17], the authors designed a probabilistic algorithm to detect a single target based on radio signal strength readings. In [23] the authors used CSI to perform fine-grained activity recognition. These systems have very low cost but the localization accuracy varies with different environmental conditions since the wireless signals are sensitive to multi-path, shadowing, and interference effects.

In contrast, we are exploring IR signals and hope to build a system that is complementary in terms of performance metrics to the systems above. Infrared is very robust to ambient environmental changes. The cost of systems using off-the-shelf hardware is relatively low. There have been IR based localization systems in the past. For example, authors of [6] develop a real-time tracking system with binary IR motion sensor networks which focused on tracking in a hallway scenario and used IR sensors as 'tripwires'. Our goal is to apply IR sensors for full coverage in an indoor setting.



(i) (ii) (iii) (iv)

Figure 1. (i) A "spike box." (ii) 5 vertex guards are required to α-robustly guard this polygon. (iii) It requires $\lfloor 3n/4 \rfloor + 1$ vertex guards. (iv) A rectilinear polygon that requires $n/2$ vertex guards.

3 Robust Guarding: Combinatorics and Algorithms

To achieve full coverage in an indoor environment, we need to deploy the iLoc devices effectively and efficiently. In this section, we model the sensor deployment and its quality requirement as the robust guarding problem: the indoor environment is regarded as a polygon and iLoc guard nodes as point guards. We deploy iLoc strips of sensors along the boundary of this polygon to detect the arrival angle for localization.

An α-robust guarding solution for a polygon P requires that each point p of P is visible to at least two guards g_1, g_2 such that the angle between the two rays pg_1 and pg_2 is at least α. We provide an upper bounds on the number of guards required as well as algorithms for guard placement in any rectilinear polygon and a lower bound for the worst case. We focus on the case of placing guards only at vertices of P and we consider the case when $\alpha \in (0, \pi/2]$.

3.1 Hardness and Lower Bounds

THEOREM 1. *Let P be a simple polygon with n vertices. Finding an optimal set of guards to α-robustly guard every point $p \in P$ is NP-hard.*

PROOF. The proof uses a reduction from the NP-hard problem Minimum Line Covering Problem [13]. In this problem, a set of n lines $L = \ell_1, ..., \ell_n$ is given in which no two lines are parallel and not all lines intersect at the same point. One asks for a minimum-cardinality set of intersection points S of L such that each line ℓ_i has at least one point of S on it. We now turn this problem to an instance of α-robust guarding. We draw a large box that contains all intersections of the arrangement of L. For each line ℓ_i, we chop away (arbitrarily) one of the rays extending outside the box, and, for other ray, we turn it into an extremely narrow "spike". See Figure 1 (i) for an example. We form a polygon, commonly called a spike box, with complexity $O(n)$. If we make the angle at each of the spikes to be smaller than α, it is necessary to place one guard at the tip of each spike to cover this tip, and the other guard somewhere near or on the line. Now we argue that an optimal solution to minimum line covering problem, together with the guards at the tips of all spikes, gives a α-robust coverage for the spikebox.

First we argue that the interior of the box is α-guarded. When not all lines intersect at the same point, the minimum line covering problem uses at least two points to cover L. We make α smaller than any angle pg_1, pg_2, where g_1, g_2 are any two intersections of L and p is any point inside the box. Thus, placing guards at the solution of the minimum line covering problem will be sufficient to guard the interior of the box. For the points inside each spike i, they are guarded by the guard at the tip of the spike and the guard that covers the line ℓ_i. Thus if the minimum line covering problem uses k points, then the α-robust guarding of the spikebox requires at least $k+n$. □

THEOREM 2. *Let P be a simple polygon with n vertices. $\lfloor 3n/4 \rfloor + 1$ vertex guards are sometimes necessary to α-robustly guard P.*

PROOF. For the polygon with 6 vertices in Figure 1 (ii), we need 5 vertex guards (in red) to α-robustly guard any point with the angle constraint. We can form a chain of copies of the polygon in Figure 1 (ii) resulting in a larger polygon as in Figure 1 (iii). With the addition of each new copy of the base polygon we increase the number of vertices by 4 and so we need 3 additional guards. Therefore, we need $\lfloor 3n/4 \rfloor + 1$ vertex guards to guard such a polygon. □

3.2 Rectilinear Polygons

For indoor localization, a realistic assumption is to consider only rectilinear polygons whose edges are horizontal or vertical. Many indoor environments can be modeled or approximated by a rectilinear polygon. In this subsection we prove tight bounds of the α-robust guarding problem. Note that it is necessary for the number of vertice n of the rectilinear polygon to be even ($n/2$ is an integer).

THEOREM 3. *Let P be a simple rectilinear polygon with n vertices. Then, $n/2$ vertex guards are always sufficient and sometimes necessary to α-robustly guard P.*

We first show an example that requires $n/2$ guards in Figure 1(iv). This is an example like that of Figure 1(iii) where only horizontal and vertical edges are used. In each iteration, we increase the number of vertices by 8 resulting in needing 4 additional guards. Therefore, we need $n/2$ vertex guards to guard such a rectilinear polygon.

To establish the upper bound, we show that the simple solution of placing guards at alternating vertices of the boundary of P will always α-robustly guard P, for any $\alpha \in (0, \pi/2]$. We will begin by building some structural lemmas of rectilinear polygons to prove the theorem.

Let p be a point in P. We define P_{max} to be the rectangle inside P with maximum area containing p. All edges of P_{max} overlap with some portion of the boundary of P. For an edge e of P_{max}, the overlapping line segments are defined as $l(e)$ where $l(e) = e \cap P$. Thus, there are only three possible cases for $l(e)$: (a) $l(e)$ contains both endpoints of e, (b) $l(e)$ contains no endpoint of e, (c) $l(e)$ contains only one endpoint of e. Figure 2 show the three cases.

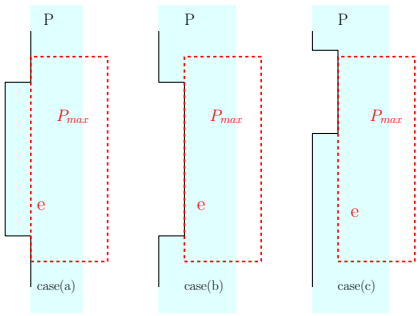


Figure 2. Different cases for intersection between the boundaries of P_{max} and P.

LEMMA 1. *Any two neighboring edges e and e' of P_{max} must have at least one vertex of P. Further, the boundary of P_{max} contains at least four vertices of P.*

PROOF. Consider two neighboring edges, e and e', sharing a corner vertex $v = e \cap e'$ of P_{max}. First, an edge e of P_{max} does not contain a vertex of P only if e is completely in the interior of an boundary edge of P. Therefore, for two neighboring edges e and e' of P_{max} sharing a corner vertex v, they must at least contain one vertex of P because if $l(e) = e$ and $l(e') = e'$, then the vertex v' where $v' = e \cap e'$ must be a vertex of P .

For the second claim, if $l(e) \cap l(e') \neq \emptyset$, the vertex v must also be a vertex of P. If $l(e) \cap l(e') = \emptyset$, there is a vertex of P on e if $v \notin l(e)$ and a vertex on e' if $v \notin l(e')$. There are four pairs of neighboring edges in a rectangle. Therefore, the edges of P_{max} must contain at least four vertices of P. □

LEMMA 2. *There must exist two points g and g' on the boundary of P_{max} such that g, g' are selected as guards in $g_v(P)$.*

PROOF. By Lemma 1 there are only three cases for the location of the vertices of P on the boundary of P_{max}: Case(a) all edges of P_{max} contain at least a vertex of P, Case(b) only one edge of P_{max} does not contain any vertex of P, Case(c) a pair of opposite edge of P_{max} does not contain any vertex of P. If P_{max} contains more than four vertices of P, pick any four vertices such that any neighboring edges of P_{max} must contain at least one vertex.

If any two vertices v_i and v_j out of the four selected is on the same edge e of P_{max}, one of them is chosen as a guard if they are connected by odd number of edges in P. If they are connected by even number of edges, then one of the incident edge of v_i or v_j is completely on the edge e of P_{max}. So there must be a vertex of P on e such that it is in $g_v(P)$.

This leaves only the case that two vertices of the four vertices are on neighboring edges of P_{max}. If the incident edge of v_j is completely on the boundary of P_{max}. Then either an endpoint of this edge is the vertex guard. Otherwise, the incident edge is only partially contained and let v_k be the the first vertex among the four that is traversed from v_j in anticlockwise direction. See Figure 3. If v_j and v_k is connected by odd edges, either one is in $g_v(P)$. Otherwise, the incident edge of v_k is completed on the boundary of P_{max}. So there must be a vertex of P on the edges of P_{max} that contains v_j and v_k.

Case(a): Since each edge of P_{max} contains a vertex, we get two pairs of two vertices that are on different neighboring edges. We can apply the argument above for each pair. Therefore, there are at least two vertices in $g_v(P)$.

Case(b): If the edge opposite to the edge without any vertex of P contains two out of the four vertices, we group each vertex on this edge with the vertex on the neighboring edge. We have two pairs of two vertices on different neighboring edges. Otherwise, we group the only vertex with the vertex on the neighboring edge that contains one vertex. Then, the remaining edge contains two vertices. In both cases, we get at least two vertices in $g_v(P)$.

Case(c): Since there is a pair of opposite edges of P_{max} that does not contain any vertex of P, the remaining two edges each contains two vertices. Thus, there are two vertices in $g_v(P)$. □

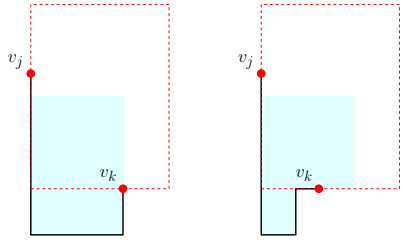


Figure 3. Vertices on two neighboring edges

We are now ready to show that $n/2$ vertex guards are always sufficient to α-robustly guard a rectilinear polygon.

PROOF. By the above lemma, for any point $p \in P$, there are two points g and g' from $g_v(P)$ on the boundary of the maximum rectangle P_{max} that contains p. Since p is visible by any boundary point of P_{max}, p is visible to at least two guards g and g' in $g_v(P)$.

If $\angle gpg' \geq \alpha$, the point must be α-robustly guarded. If $\angle gpq < \alpha$, we must be able to draw a line l through p such that g and g' lie on the same side of line and l must be parallel to some edges of the polygon P. Then we draw a line l' through p particular to l. We divide the polygon into four regions. If we discover another vertex guard g'' such that it lies on the opposite side of l, we can check whether p is now α-robustly guarded. If it fails, then the three vertex guards g, g', g'' must lie into two neighboring regions. We can draw a new separating line and repeat the process. Now the four guards must lie in three different regions. Then the angle constraint must be satisfied. An example is shown in Figure 4.

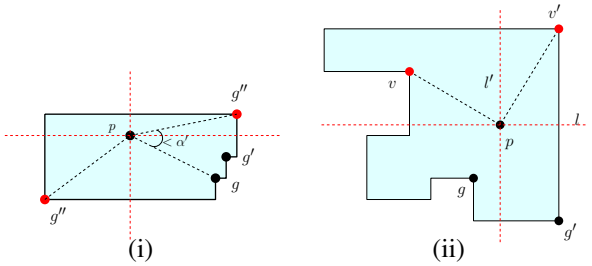


Figure 4. (i) Searching for g'' (ii) Sweeping for v and v'.

Now we need to show that if there is a separating line, we are always able to find such a new vertex guard. We

use a sweeping strategy, taking the line l as our sweep line and moving it away from the points g and g'. The sweeping stops when we first find two vertices, say v and v', of P that are visible to the point p and they are on the opposite of l'. See Figure 4(ii). When the sweep line discovers v (or v'), it encounters the other endpoint of the edge containing v (or v') and parallel to the sweep line. If this endpoint is also visible to p, then both endpoints of this edge is visible to p and either one of them must be in $g_v(P)$. Then we find the new vertex guard g''. So the only case we need to prove is that only v and v' are visible after the sweeping stops. We can connect v and v' by extending them towards the line l, shown with the dotted blue line in Figure 5. This dotted blue line and the edges of P that connects v and v' in the other way form a new rectilinear polygon. Since v and v' are separated by three edges across the blue line, they must also be separated by odd number of edges on the other way. Therefore, in the original polygon p, one of v and v' must be in the set $g_v(P)$. □

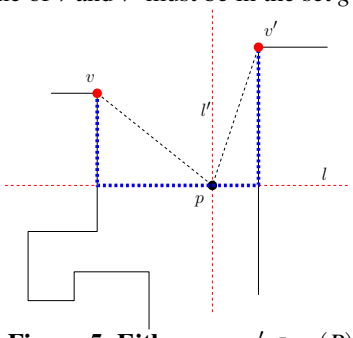


Figure 5. Either v or $v' \in g_v(P)$

THEOREM 4. *Let P be a rectilinear polygon with rectilinear holes. The total number of vertices including vertices on the holes and outer boundary is n. Then, $n/2$ vertex guards are always sufficient to α-robustly guard any point $p \in P$.*

PROOF. For rectilinear polygons with holes, the proof is similar. The algorithm is different only in the sense that the alternative selection of vertices on the holes is opposite to the selection on the boundary of the polygon. Now the sweep line also searches for vertex guards on the holes. If v and v' are on different holes, we draw a narrow bridge along the line l to connect the two holes such that v and v' are separated by three edges. If they belong to the same hole, we draw a line along l to extend the hole. Again, v and v' are separated by three edges. Therefore, the sweep line always finds a vertex in $g_v(P)$. □

We remark in our problem formulation that we only require a lower bound of α for the difference of direction of the angles of arrival from two different emitters at any point inside the domain. In the formulation in [7], the direction from the two emitters must lie in the interval $[\alpha, \pi - \alpha]$, i.e., avoiding two emitters and the target location being nearly collinear. This model also makes sense in applications. Still, there are polygons that cannot be robustly guarded even if we place guards at all vertices – for any given angle α consider an extremely skinny rectangle and the center of mass p, the directions from all vertex emitters are within α or greater than $\pi - \alpha$. Thus studying the combinatorial bounds for this version is not very meaningful.

4 Collaborative Sensing Framework

After the placement of sensors to achieve α-robust coverage, we ensure that we have full coverage with reasonably good sensing quality. The next step is that we need to interpret the signals to reason about the environment. In this section, we present a general framework of perception strategies that are specific to line-of-sight sensors while general to different applications. These strategies are motivated by the nature of how we utilize such directional sensors in a device-free manner. Shadowing and reflection both gives us opportunities to detect and understand the events happening in the domain. But pinpointing the precise details of an event involving multiple targets require non-trivial strategies presented below. Then we present a case study to demonstrate how these strategies are applied in a localization application.

Sensor fusion. As a collaborative sensing system, multiple sensors are deployed in the environment and more than one reasoning technique (shadowing and reflection) are used. We need to incorporate the information from these multiple sources. Since the signals may align or contradict to each other, a probabilistic framework is flexible enough to integrate the varied information. The exact procedure on how to transform the signals into probabilistic beliefs of the real world depends on the application and implementation. The possible related techniques are filtering, signal processing and machine learning with a rich set of literature. After developing the beliefs, an optimization approach should be used to find out the best explanation.

Iterative reasoning. In a multiple target environment, the signals are generated by different targets. Since the targets are not carrying any device, we don't have any identification on the signals. In other words, the impact of each individual target is tangled in the signals. Therefore, we suggest to use an iterative manner to isolate the effects of each target. In each iteration, we allocate the contribution of a subset of the signals to a single target. The procedure continues until all targets are identified.

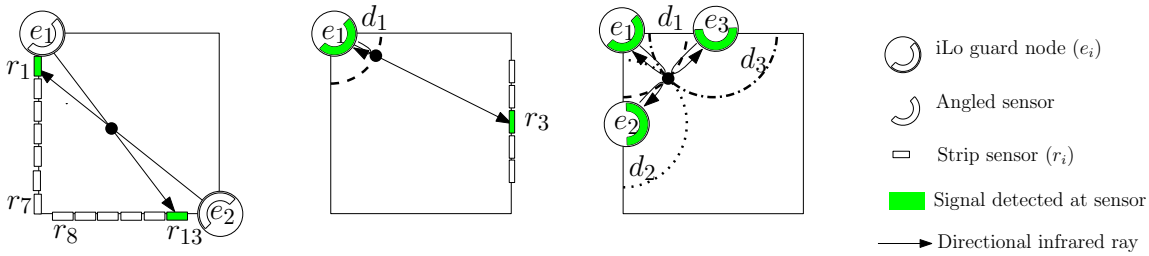
Signal structure. The line-of-sight constraints impose combinatorial constraints on the signal's structure. Both the existence and absence of some signals provide hints about the events in the domain. For example, a target completely blocking the emitter creates shadows everywhere. If some line of sight is not blocked, we can disfavor this situation.

4.1 A Case Study: Infrared Localization

Here we use a localization system as an example to demonstrate our collaborative sensing framework.

4.1.1 Single Target Localization

The localization scheme is used to determine the location (x, y) of an object. It can utilize the signals detected at sensors to determine the angle of the object from known emitters. For example, if the line of sight between an emitter at (x_i, y_i) and a sensor at (x_j, y_j) is blocked, the strength of light received decreases and a shadow signal is detected at the sensor. Then we know that the object should be located on the line of sight between them. Its distance from the line



(a) A shadow is detected by two emitter-sensor pairs $(e_1, r_{13}), (e_2, r_1)$ since target is blocking their lines of sight

(b) Beside a shadow, a reflection is reported by angle sensor on emitter e_1 with a distance of d_1 since the target is close

(c) Three reflections are captured since the target is within the detection range of the angle sensors on emitters e_1, e_2, e_3

Figure 6. To determine the location of a target (represented by a black dot), we need either two shadows in subfigure (a), one shadow and one reflection in subfigure (b) or three reflections subfigure (c) where there is no sensor on boundary

should be zero:

$$\frac{|(y_j - y_i)x - (x_j - x_i)y + x_j y_i - y_j x_i|}{\sqrt{(y_j - y_i)^2 + (x_j - x_i)^2}} = 0$$

With two such lines that are not parallel, we can uniquely determine the location of an object causing these two signals.

The angled sensors on the emitter can also report reflection sensor readings if an object is within the reflection detection range t. If this is the case, a reflection signal is reported with an estimated distance d_i between the emitter and the object:

$$d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2} < t$$

With either two shadow signals or one shadow and one reflection signal, we can uniquely determine the location as illustrated in Figure 6(a) and 6(b). With three reflection signals in 6(c), trilateration is also possible without any shadow signals (which means no strip sensor is needed on the boundary). But for our iLoc guard's reflection module, the range is usually short relative to the room's size so simultaneous detection of multiple reflection signals is unlikely.

In the complete system, we have multiple iLoc guard nodes (emitters and angled sensors) at corners and iLoc strip nodes (sensors) on the entire boundary so that more signals are detected than needed. Therefore, the probabilistic framework for the *sensor fusion* can integrate multiple and possibly noisy signals to estimate the final location. To achieve it, we developed a metric for the belief and localization scheme to optimize against the metric considering efficiency to compute on the controller which has limited resources.

For the localization scheme, suppose there are r strips of sensors (iLoc strip node) and e points where emitters (iLoc guard node) are placed with their locations known to the localization procedure. To uniquely determine the identification of an emitter, we need modulation so that the sensor can decide where the source of light is from.

In our system, e emitters take turn emitting light. As an example, when the i-th emitter is emitting a pulse, the j-th strip sensor processes the signal and reports a probability p^j_{shadow}. For reflection, the angled sensors on the emitter report a probability $p^i_{reflection}$ with a distance d_i. Then we can define a metric for the estimated location (x, y) for the object based on the IR pulse emitted by the i-th emitter as

$$L^i(x,y) = \frac{1}{\sum_{k=1}^{r} \lceil p^k_{shadow} \rceil} \times \sum_{j=1}^{r} \Big(p^j_{shadow} \times \frac{|(y_j - y_i)x - (x_j - x_i)y + x_j y_i - y_j x_i|}{\sqrt{(y_j - y_i)^2 + (x_j - x_i)^2}} \Big) + p^i_{reflection} \times \Big| \sqrt{(x - x_i)^2 + (y - y_i)^2} - d_i \Big| \quad (1)$$

Intuitively, if the shadow probability is strong, the object should be on the line between the emitter and strip sensor. Similarly for reflection signal, it should be at distance d from the emitter. Each emitter is equal therefore the normalizing factor $\frac{1}{\sum_{k=1}^{r} \lceil p^k_{shadow} \rceil}$ for the shadow signal can help in the case where the object is close to an emitter and a lot of shadow signals are detected. To integrate the signals from different emitters, we minimize the summation of metrics for all e emitters together so that the estimated location aligns with all the signals as much as possible:

$$(x^*, y^*) = \underset{x,y}{\operatorname{argmin}} \sum_{i=1}^{e} L^i(x,y)$$

After minimization, the object is best estimated at (x^*, y^*).

4.1.2 *Multiple Target Localization*

When there are multiple human objects in the sensing environment ambiguity may arises. For a rectangular room, we need two guards to guarantee two shadows for a single target. But when there is more than one target, it creates the possibility of phantom points such that we cannot determine what the real locations of the objects are. See Figure 7 for an example. To resolve this issue, we can add one additional guard (not co-linear with the previous two). Unless any two of the targets and one guard form a line, 3-coverage is enough to uniquely determine the location and number of targets.

In addition, we will modify the signal processing algorithm accordingly to accommodate possibly multiple targets.

Recall that in the cost function, $\frac{|(y_j - y_i)x - (x_j - x_i)y + x_j y_i - y_j x_i|}{\sqrt{(y_j - y_i)^2 + (x_j - x_i)^2}}$ represents the distance between a target estimated at (x, y)

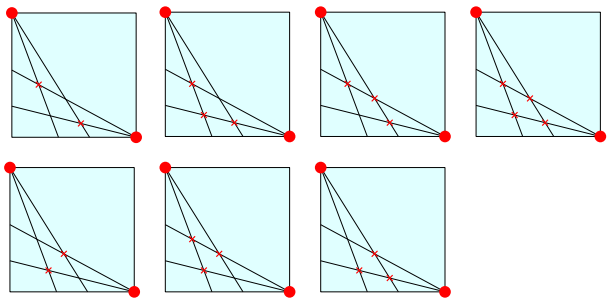


Figure 7. Phantom targets in multiple target environment. With 4 blocked lines of sight, 7 possible configurations of targets' location (red crosses) could explain detected shadows. Ambiguity of true locations and number of targets arises.

and the blocked line of sight connecting emitter i and receiver j. If this distance is larger than the typical width of a human body, this shadow is very unlikely to be casted by this human target. With this intuition, we should modify the scheme in the sense that we apply *iterative reasoning* to attribute a subset of all remaining signals to a newly estimated location. In each iteration, we perform the optimization to search for a target's location that its metric $\sum L^i(x,y)$ summing over the subset of signals attributed to this location is smallest. Then we remove this subset of signals from the following iterations. We stop when all signals are removed or the remaining signals are not strong enough to reasonably represent a human target.

To explain the detected signals, multiple targets can always be assumed to be in front of each emitter. However, this assumption cannot reason about the absence of some shadow/reflection signals. Therefore, for a estimated location, if signals that should be detected do not exist, we should increase the metric of this location in the localization scheme such that the optimization (minimization of the metric) step will not favor this location. The *signal structure* helps to disambiguate events.

Figure 8 illustrates an example of localizing two targets in the environment with the modified scheme.

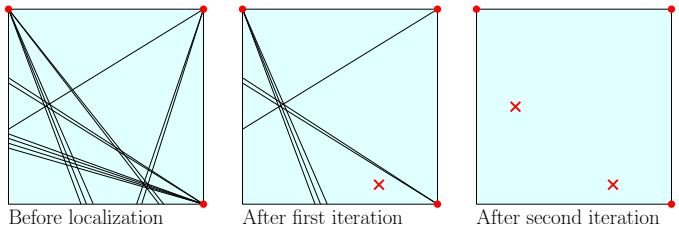


Figure 8. Iterative localization of two targets: Each iteration localize one target and remove the signals attributed to the target. Algorithm stops after the two targets are located.

5 Evaluation

We built a device free localization testbed using infrared sensors, which helps us to evaluate and demonstrate our deployment algorithms.

5.1 Design and Implementation

Our localization system is comprised of three types of nodes: iLoc guards, iLoc strips, and a iLoc coordinator, as shown in Figure 9 (a). IR emitters on the guard node emit light into the localization domain where it is monitored by IR intensity sensors on the strip nodes to detect shadows. Three desirable traits for the iLoc system are to have: 1) reliable long range IR emitters; 2) IR sensors that can be easily deployed along the localization domain; and 3) use a minimum number of nodes to robustly monitor the domain.

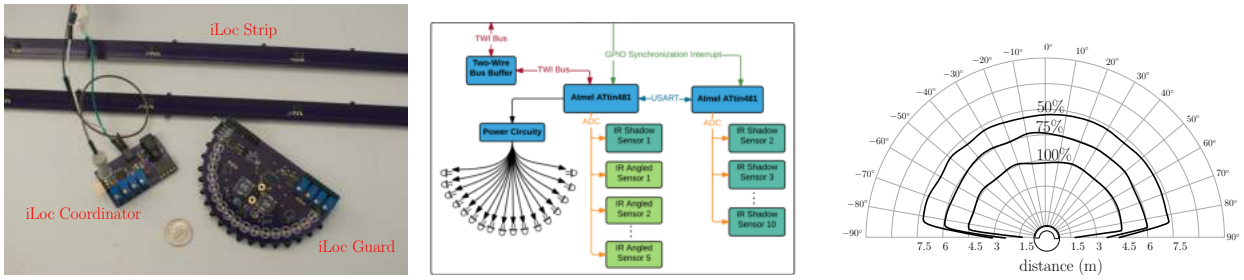
iLoc guard. Typical off-the-shelf infrared emitters produce irregular cone-shaped dispersion patterns that make it difficult to effectively illuminate, and therefore interpret, a 2d cross-section of a room. In contrast, we designed the iLoc guard to emit a long range high intensity IR pulse of short duration over a narrow 170-degree horizontal band. Since the IR pulse only needs to occur long enough for the system to collect a snapshot of the localization environment, the iLoc guard produces a long range high intensity pulse that is possible only with a low duty cycle. In addition, we equipped the iLoc guard with angled IR light intensity sensors to infer the direction of sensed light. The aim is to use these sensors to detect reflections off of the target when the target is close to a pulsing emitter.

iLoc strip. To complement the iLoc guard, we designed iLoc sensor strips to have a high density horizontal row of IR light intensity sensors. The iLoc sensor strips are connected to a local controller that allows them to be coordinated by the system, aggregate sensed data, and report the aggregated data in real time. Due to the distributed nature of the system and the iLoc guards' pulsed emission, we designed a coordinator node called the iLoc coordinator that is responsible for system wide data collection synchronization and emitter coordination.

iLoc coordinator. The iLoc coordinator is used to implement time division based modulation to allow sensors to differentiate between emitters. In addition, the iLoc coordinator is also used to: 1) optimize data communications by selectively targeting nodes where a shadow is expected to appear first; 2) dynamically control the system duty cycle to balance energy saving with real-time target tracking: and 3) adapt to interferences from ambient light sources.

The distributed nodes of the system are connected by a 400 kHz Two-Wire Interface (TWI) bus and a single GPIO interrupt line. For ease of implementation and testing, We connected an iLoc guard and two iLoc strips as a single unit. Each contains two base components: 1) a TWI bus repeater that increases the system's scalability by increasing the possible range between nodes, and 2) two Atmel Attiny841 microcontrollers that work in parallel to complete the tasks of data aggregation, analog to digital conversion of each light intensity sensor output, as well as control the LED power circuitry. Figure 9 (b) provides an overview of components on a single iLoc unit.

For an iLoc guard, 17 directional LEDs were placed every 10 degrees in a 170 degree fan-out configuration. Each LED will produce 550 mW/Sr at its maximum forward current rating of 100 mA. These LEDs are driven with 500 mA for a period of 2 milliseconds. At this current, the LEDs emit



(a) System Components (b) iLoc Node Component Diagram (c) Sensing Range

Figure 9. iLoc Localization System

approximately 4 times the radiant intensity than when driven at 100 mA. Although it only takes 234 microseconds to complete the analog to digital conversions of the sensor outputs, a 2 ms pulse is necessary to allow the LED intensity to settle before taking the reading. This high intensity 15 Watt IR pulse allows for shadow detection to up to 23 m. The LEDs can sustain a pulse frequency of up to 600 Hz when driven at this current and pulse length. The node uses two high current buck regulators to convert a 5 V power rail to 1.8 V which is then used to drive the LEDs. Using buck regulators in this configuration allows the system to better tolerate the large current draw of the 15 W IR pulse. The node is also equipped with 5 angled low sensitivity light intensity sensors that are intended primarily for reflection sensing. Sensors face 30 degrees apart, allowing the system to identify the reflection direction. Low sensitivity sensors were used to prevent saturation by the emitter.

A single LED used in the emitter has a relatively narrow viewing angle which is specified in the data sheet. Our iLoc guard is designed to overcome this shortcoming. We conducted empirical tests to evaluate this property in real deployment. Figure 9 (c) demonstrates the results of infield tests to determine the effects of angle and distance on sensor readings. The contours verify the omni-directional emitting property of our iLoc guard node except at extreme angles. This wide angle property continues to be preserved when the distance is 7.5 m and the intensity drops by 50%.

Table 1. Reflection Intensity vs. Distance

Distance (cm)	28	35.5	50	75	101.5
Reflection Strength (%)	100	75	50	15	10

Table 1 shows the change of intensity of the reflected IR by a human body. The intensity drops by 50% when the target is located 50 cm away from the emitter. This limits the effective range for reflection detection.

5.2 Performance Evaluation with Robust Constraints

The effect of α on localization accuracy is investigated. We placed two iLoc guard nodes separated by the angle α to localize an human target which was 1 meter away. The average accuracy is 15.7 cm, 4.9 cm and 2.8 cm when α is $\pi/6$, $\pi/3$ and $\pi/2$ respectively. Based on this observation, we use $\alpha = \pi/2$ for the following simulations and experiments.

5.3 Performance Evaluation with Simulations

We simulated the coverage quality for different sensor deployment algorithms. We compare our solution against two algorithms for the general art gallery problem which are 1) greedy set cover and 2) 3-coloring for standard sensor coverage. In the greedy set cover, we find the greedy set to ensure all points are visible to at least a single guard. In the 3-coloring algorithm, the input polygon is first decomposed into a triangulation. Then we find a 3-coloring of the vertices of the triangulation and place guards at the vertices of the least used color. This algorithm is guaranteed to find a valid art gallery solution. It will use at most $\lfloor n/3 \rfloor$ guards.

For alpha robust guarding, we propose to run a greedy algorithm to iteratively pick guards to increase coverage while satisfying the robust constraint. In this algorithm, we always select the next pair of guards that can cover the most number of targets robustly until all targets are covered. The final guard set is the union of all selected pairs which is also the objective that we want to minimize. Then we have a problem of submodular cover with submodular objective function. By [20], an $O(\log n)$ is obtained immediately.

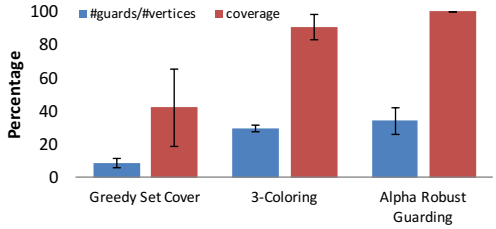


Figure 10. Guard set size & coverage for deployment algorithms

We run the three algorithms on 100 random rectilinear polygons with $\alpha = \pi/2$. As a result, Figure 10 shows the average number of guards deployed as a fraction of the polygon's size and the corresponding coverage as a fraction of targets that is alpha-robustly guarded. Obviously, the greedy set cover and 3-coloring cannot guarantee a full robust guarding. Despite only some small extra guards over 3-coloring, our greedy α robust guarding deployment provides full coverage. In theory, the lower bound is $n/2$ for the worst case as shown in Section 3 where n is the size of polygon. Our simulation shows that empirically we need $0.338n$ guards on average to achieve α-robust guarding.

Figure 11. Deployment Setup

5.4 Performance Evaluation with Testbed

Experimental Setup. A series of localization experiments have been conducted to test the ability of the system to locate a human in an office room. The test room was a 409 cm by 255 cm space shown in Figure 11. The geometry of the room was visualized by the boundary of polygon in Figure 12. Since it is a polygon with 6 vertices, we placed iLoc guard nodes at the vertices (represented by the red dots) according to Theorem 3. iLoc sensor strips (represented by the white dots) were deployed on all walls at about the same height. For each of the 16 locations (represented by the black dots), a sensor snapshot of the readings was taken and fed in real time to the localization program described in Section 4.1.1 with a human subject standing at the location and facing the right wall. We repeated the tests 10 times at each testing point to obtain statistical results.

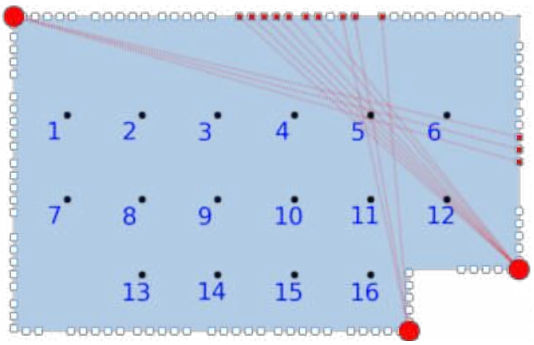


Figure 12. Testbed Layout. The three red dots at vertices represent iLoc guard nodes; the white dots along the edges represent iLoc sensors; the black dots represent the testing locations. It demonstrates the case when a person is at location 5.

Experimental Results. We implemented a simple threshold based signal processing algorithm to determine $p^i_{reflection}$ and p^j_{shadow}. The two probabilities equal to the percentage increase and decrease of the sensor reading respectively. For reflection, we used Table 1 to determine d_i which is calculated by mapping $p^i_{reflection}$ to the distance in the table. Based on the current implementation, the effective range of reflection detection is relatively small (around 50 cm) compared to the effective sensing range for shadows (23 m). For shadow, we use a threshold λ. If the infrared light intensity reading of a sensor drops more than λ from the reading taken when there is no target, a shadow is detected at the sensor:

$$p^j_{shadow} = \frac{\Delta reading^j}{reading^j} \text{ if } \frac{\Delta reading^j}{reading^j} > \lambda$$

This approach utilizes background subtraction which does not require any learning or modeling of the testing environment. It is also adaptive to more permanent changes in the room such as furniture movement.

When λ is set to 30%, the algorithm achieves the lowest average localization error with smallest standard deviation: the average accuracy is 11.7 cm while the worst-case localization error is 20.0 cm. Choosing a lower threshold may take into account signals caused by noise and a higher threshold may ignore potential shadows.

Figure 13 shows the accuracy result for all evaluation points. For the evaluation points represented by red triangles, they are visible to all three emitters. With the extra shadow, the average localization error is 9.1 cm, which is much lower compared to 16.1 cm for the other points (black square) that report shadows from only two emitters. This shows the integration of multiple and excessive signals through probabilistic beliefs enhances the localization accuracy.

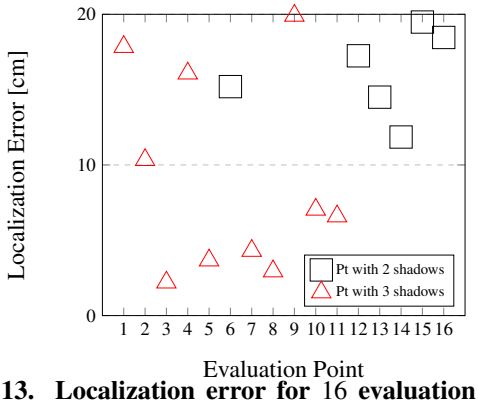


Figure 13. Localization error for 16 evaluation points (λ=30%)

To compare different deployment algorithms, Greedy Set Cover deploys only 1 guard at the top left and 3-Coloring deploys one located at the top left and another at the bottom right. In Figure 14, Greedy Set Cover resulted in a average error of 249.9 cm because only one shadow can be generated by the single emitter and it is not enough to uniquely determine location. For 3-Coloring, the average error is 89.1 cm. Much error is contributed by the poor estimation for point 6, 12 since they are not visible to the bottom right emitter.

Experiments were performed to evaluate the localization system in a multiple target setting. For each setting of 2 targets, 3 targets and 4 targets, we randomly selected 30 combinations of the 16 testing points with multiple human subjects standing on them. Then the signals were analyzed by the scheme in Section 4.1.2. The localization error is reported in Figure 15.

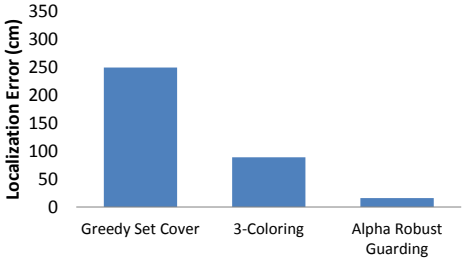


Figure 14. Localization error for deployment algorithms

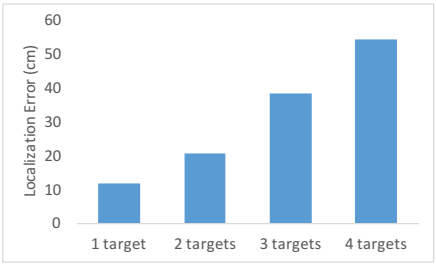


Figure 15. Localization error for multiple targets

6 Conclusions

In this paper, we presented both theoretical results and experimental results for device-free indoor localization using infrared tripwire sensors. For the theoretical part, we have a tight bound on α-robust guarding for rectilinear polygons. For the experiments, we studied target localization and showed promising deployment results for realistic settings. For future work, we will extend it to α-robust guarding for general shapes. More sophisticated signal processing algorithms can also enhance the localization.

7 Acknowledgments

K. S. Liu, and J. Gao would like to acknowledge the support through NSF CNS-1217823, CCF-1535900, CNS-1618391, and DMS-1737812. K. S. Liu and S. Lin are grateful for support through NSF CNS-1553273 and CNS-1463722.

8 Appendix

Power consumption. Each of the 15 light intensity sensors on the iLoc node draw 1.1 mA totaling 16.5 mA. The high current switching regulators that convert the system input voltage of 5 V to the LED driving voltage of 1.8 volts give tolerance for significant voltage drops on the 5 V rail without causing system brown-outs. As the switching regulators operate at about 92% efficiency, the 15 Watt LED pulse draws approximately 16.5 Watts from the power rail. When the switching regulators are not driving the LEDs, they are in sleep mode, each drawing 1 mA. The power consumption of both micro-controllers combined is less than 1 mA. Hence altogether the system draws about 17.5 mA per node when idle and about 3.3 A during the 2 ms period when the node is emitting an LED pulse.

Sampling rate. The frequency that the system is able to take a snapshot of the localization environment depends on two factors: the ability of the system to perform all necessary communications in a single sampling period, and the length of the LED pulse. It takes approximately 12.5 microseconds for the coordinator to receive data from a single node. As the number of nodes increases, the time it takes to collect all of the data increases proportionately. Therefore, a system such as the 10 node system used in our evaluation will have a minimum period size of 6.375 ms which results in a maximum sampling frequency of about 157 Hz.

9 References

[1] T. E. Abrudan, A. Markham, and N. Trigoni. A case for magneto-inductive indoor localization (best poster award). In *The 11th European Conference on Wireless Sensor Networks (EWSN 2014)*, 2014.

[2] P. Brass, H. Na, and C. Shin. Guarding a polygon from two nearly-opposite directions. *Int. J. Comput. Geometry Appl.*, 20(3):327–339, 2010.

[3] T. Brown, Z. Wang, T. Shan, F. Wang, and J. Xue. Obstacle and connectivity aware wireless video sensor deployment for 3d indoor monitoring. In *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation*, pages 305–306. ACM, 2017.

[4] G. Chandrasekaran, M. A. Ergin, J. Y. 0003, S. Liu, Y. Chen, M. Gruteser, and R. P. Martin. Empirical evaluation of the limits on localization using signal strength. In *SECON*. IEEE, 2009.

[5] K. K. Chintalapudi, A. P. Iyer, and V. Padmanabhan. Indoor localization without the pain. In *Mobicom*. Association for Computing Machinery, Inc., September 2010.

[6] D. De, W.-Z. Song, M. Xu, C.-L. Wang, D. Cook, and X. Huo. FindingHuMo: Real-time tracking of motion trajectories from anonymous binary sensing in smart environments. In *Proceedings of the 2012 IEEE 32Nd International Conference on Distributed Computing Systems*, ICDCS '12, pages 163–172, Washington, DC, USA, 2012. IEEE Computer Society.

[7] A. Efrat, S. Har-Peled, and J. Mitchell. Approximation algorithms for two optimal location problems in sensor networks. In *Proceedings of the 2nd International Conference on Broadband Networks (BroadNets 2005)*, pages 714–723, 2005.

[8] T. W. Hnat, E. Griffiths, R. Dawson, and K. Whitehouse. Doorjamb: Unobtrusive room-level tracking of people in homes using doorway sensors. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, SenSys '12, pages 309–322, New York, NY, USA, 2012. ACM.

[9] P. Hu, L. Li, C. Peng, G. Shen, and F. Zhao. Pharos: Enable physical analytics through visible light based indoor localization. In *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, HotNets-XII, pages 5:1–5:7, New York, NY, USA, 2013. ACM.

[10] P. Lazik and A. Rowe. Indoor pseudo-ranging of mobile devices using ultrasonic chirps. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, SenSys '12, 2012.

[11] S. Lee, B. Kim, H. Kim, R. Ha, and H. Cha. Inertial sensor-based indoor pedestrian localization with minimum 802.15.4a configuration. *IEEE Trans. Industrial Informatics*, 2011.

[12] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. *Trans. Sys. Man Cyber Part C*, 37(6):1067–1080, Nov. 2007.

[13] N. Megiddo and A. Tamair. On the complexity of locating linear facilities in the plane. *Oper. Res. Lett.*, 1:1940197, 1982.

[14] S. Nirjon, J. Liu, G. DeJean, B. Priyantha, Y. Jin, and T. Hart. Coin-gps: Indoor localization from direct gps receiving. In *Proceedings of the 12th International Conference on Mobile Systems, Applications and Services (MobiSys'2014)*, 2014.

[15] J. Ranjan, Y. Yao, and K. Whitehouse. An rf doormat for tracking people's room locations. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 797–800, New York, NY, USA, 2013. ACM.

[16] K. Römer. The lighthouse location system for smart dust. In *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services*, MobiSys '03, 2003.

[17] M. Seifeldin, A. Saeed, A. E. Kosba, A. El-Keyi, and M. Youssef. Nuzzer: A large-scale device-free passive localization system for

wireless environments. *IEEE Trans. Mob. Comput.*, 12(7):1321–1334, 2013.

[18] S. P. Tarzia, P. A. Dinda, R. P. Dick, and G. Memik. Indoor localization without infrastructure using the acoustic background spectrum. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, MobiSys '11, pages 155–168, New York, NY, USA, 2011. ACM.

[19] I. Vlasenko, I. Nikolaidis, and E. Stroulia. The smart-condo: Optimizing sensor placement for indoor localization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(3):436–453, 2015.

[20] P.-J. Wan, D.-Z. Du, P. Pardalos, and W. Wu. Greedy approximations for minimum submodular cover with submodular cost. *Computational Optimization and Applications*, 45(2):463–474, 2010.

[21] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury. No need to war-drive: Unsupervised indoor localization. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, MobiSys '12, pages 197–210, New York, NY, USA, 2012. ACM.

[22] Y. Wang and G. Cao. On full-view coverage in camera sensor networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 1781–1789, April 2011.

[23] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu. E-eyes: Device-free location-oriented activity identification using fine-grained wifi signatures. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, MobiCom '14, pages 617–628, New York, NY, USA, 2014. ACM.

[24] K. Whitehouse, C. Karlof, A. Woo, F. Jiang, and D. Culler. The effects of ranging noise on multihop localization: An empirical study. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, IPSN'05, 2005.

[25] Z. Xiao, H. Wen, A. Markham, and N. Trigoni. Lightweight map matching for indoor localisation using conditional random fields. In *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, IPSN '14, 2014.

[26] Z. Yang, C. Wu, and Y. Liu. Locating in fingerprint space: Wireless indoor localization with little human intervention. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 269–280, New York, NY, USA, 2012. ACM.

[27] M. Youssef, M. Mah, and A. Agrawala. Challenges: Device-free passive localization for wireless environments. In *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, MobiCom '07, pages 222–229, New York, NY, USA, 2007. ACM.

[28] Z. Zhong and T. He. Achieving range-free localization beyond connectivity. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, SenSys '09, 2009.