Efficient Beacon Placement Algorithms for Time-of-Flight Indoor Localization

Haotian Wang
Stony Brook University
haotwang@cs.stonybrook.edu

Niranjini Rajagopal
Carnegie Mellon University
niranjir@andrew.cmu.edu

Anthony Rowe
Carnegie Mellon University
agr@ece.cmu.edu

Bruno Sinopoli
Washington University in St.Louis
bsinopoli@wustl.edu

Jie Gao
Stony Brook University
jgao@cs.stonybrook.edu

ABSTRACT

Beacon-based time-of-flight indoor localization systems have shown great promise for applications ranging from indoor navigation to asset tracking. In large-scale deployments, a major practical challenge is determining the placement of a minimal number of beacons that ensures full coverage – each point in the domain has line-of-sight paths to enough beacons to uniquely localize itself. Three beacons with line-of-sight paths are always enough, but two beacons within line of sight may also work, given a favorable geometry. In this paper, we propose two beacon placement algorithms that leverage the floor plan geometry with provable theoretical guarantees. First, we present a greedy algorithm using properties of sub-modular functions to place $O(\mathrm{OPT} \cdot \ln m)$ beacons, where m is the number of discrete location points in the region that need to be localized, and OPT is the size of the optimal solution. Second, we present a random sampling algorithm that places $O(\mathrm{OPT} \cdot \log(\mathrm{OPT}))$ beacons while localizing all targets. We evaluate our algorithms on both real-world and randomly generated floor plans. Our algorithms place on an average $6 \sim 23\%$ and 12% fewer beacons in real-world topologies and randomly generated floor plans respectively, as compared to prior work. We also present a study where we ask users to attempt to place nodes manually and discover that even humans that are well versed on the coverage problem find it hard to balance the trade-off between the number of beacons and area localized.

CCS CONCEPTS

• **Information systems** → **Sensor networks**; • **Networks** → **Mobile networks**.

KEYWORDS

Indoor Localization, Beacon Placement, Approximation Algorithm, Computational Geometry

ACM Reference Format:
Haotian Wang, Niranjini Rajagopal, Anthony Rowe, Bruno Sinopoli, and Jie Gao. 2019. Efficient Beacon Placement Algorithms for Time-of-Flight Indoor Localization. In *27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL '19), November 5–8, 2019, Chicago, IL, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3347146.3359344

SIGSPATIAL '19, November 5–8, 2019, Chicago, IL, USA

ACM ISBN 978-1-4503-6909-1/19/11…$15.00
https://doi.org/10.1145/3347146.3359344

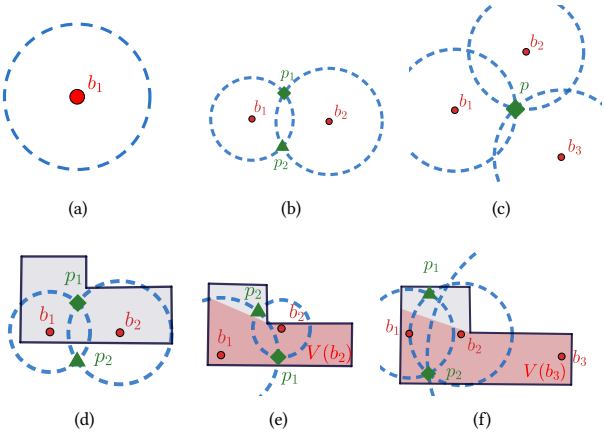


Figure 1: Localization with ToF in a floor plan: (a) An infinite number of solutions with a single beacon; (b) Two solutions with two beacons; (c) A unique solution with three beacons; (d-f) A unique solution with two beacons: (d) p_2 is outside the floor plan and thus is not feasible; (e) p_2 is not visible to b_2 and thus is not feasible; (f) p_2 is visible to a third beacon b_3 and thus is not the correct solution.

1 INTRODUCTION

Several technologies have emerged for indoor localization in the past decade that are able to provide sub-meter ranging using Time-Of-Flight (TOF) or Time-Difference-Of-Arrival (TDOA) ranging. The underlying ranging signals could be based on acoustic, ultrasonic or RF technologies, such as Ultra-Wideband, Bluetooth Low Energy 5 and WiFi 802.11mc. ToF beacon-based systems provide a distance measurement, which is used to estimate a device's location via trilateration. While ToF-based localization is well understood, the problem of where to place the beacons is not. Current methods used to deploy beacons either require domain experts who leverage intuition and heuristics, or let the system installers over-provision indoor spaces with more beacons than required, in order to ensure full localization coverage. As a consequence deployments can waste

resources. Minimizing the number of beacons is particularly important as these systems transition from small deployments, mainly used for demonstration purposes, to commercial ones across large real-world spaces such as airports, museums, malls, and industrial buildings. In these environments an efficient and systematic beacon placement methodology will have a significant impact in terms of cost savings, thus facilitating adoption. Thus in this paper we aim to examine the beacon placement problem systematically, with a focus on both *practicality* in real world setting and *provable guarantees* about coverage and beacon count with respect to optimal.

In our formulation of the localization problem, the beacons emit signals that could be picked up by a mobile device within line-of-sight of the beacons. The device at a location p within line of sight of a beacon b can derive a (fairly accurate) distance estimate from p to b. If a location p has line of sight paths to three or more beacons, by trilateration one can uniquely localize p. If a point is visible to only two beacons b_1, b_2, the two range measurements provide two candidate locations p_1 and p_2, which are mirrors of each other with respect to the line joining the two beacons. With the information from the floor plan we can potentially discover the true location among p_1 and p_2 if the set of beacons visible to one of them differs from $\{b_1, b_2\}$. Figure 1(d)- 1(f) illustrates this subtlety. This property that takes into account the geometry of the floor plan makes the beacon placement problem especially interesting.

We present two algorithms with provable guarantees for the problem of beacon selection for unique localization. We also evaluate their performance on multiple floor plans, compared with prior work and user studies. First, we formally define the conditions for a location to be uniquely localizable without ambiguity and propose efficient algorithms to test whether a given set of beacons provide unique localization for all points in the domain. For this, we employ detailed analysis of the geometric constraints and visibility. On this foundation, we design the beacon placement algorithms.

The first algorithm we propose is a greedy algorithm in which we select the next beacon by optimizing a certain objective function. A greedy algorithm is also favored in practice for its incremental nature. The intuitive approach is to maximize the area that is uniquely localizable (as done in prior work [24]). However, there are a number of problems with this objective function, for example, it is possible that no extra beacon can increase the area that is uniquely localized. There are scenarios where this algorithm can lead to a solution far from the optimal. We overcome this challenge by designing a new objective function that has the submodular property (intuitively, a function with monotonicity and diminishing return) and by optimizing this submodular function, we get provable performance guarantee that the number of beacons selected has an approximation ratio $O(\ln m)$ of the optimal, where m is number of target points to be localized. The key is to consider not only the area to be uniquely localizable but also areas that are covered by at least one beacon and at least two beacons.

We propose a second algorithm that uses random sampling for beacon selection. It uses the concept of ε-net and geometric set cover. For the set cover problem, if the sets have constant VC-dimension (which is a measure of the complexity of the sets), then one can approximate the optimal solution up to $O(\log \mathrm{OPT})$, where OPT is the size of the optimal solution [9]. Again, we cannot directly apply this technique to our problem of unique localization, as checking whether a location is uniquely localizable by two beacons depends on the set of beacons covered by the mirror image of this location. We need to carefully work around the issue and argue that even for our localization problem, the random sampling based algorithm works. In the algorithm, we introduce weights to the beacons. Initially all beacons have the same weights. Now define the weight of a point p as the sum of the weights of the beacons that are visible to p. We develop a new proof using the VC-dimension property to show that a randomly chosen set of beacons can uniquely localize all points of high weight. Now considering the points that are not yet covered (which have low weight), we double the weight of the beacons they see. Therefore, in the next round of random sampling these beacons have a higher chance of being selected. It has been shown in [9] that after a number of iterations, this algorithm stops with all the points uniquely localized, as long as the number of beacons selected is at least $O(\mathrm{OPT} \log \mathrm{OPT})$.

We implement both algorithms in a MATLAB-based toolchain for real-world and random floor plans and compare with prior work. We also show how our placement compares to placements performed by users with varying familiarity with indoor localization and beacon placement geometrical problems. In both cases our algorithms perform favorably in terms of both coverage and localization quality

In summary, the main contributions of this paper are:

(1) A mathematical formulation of the minimal beacon placement problem for indoor localization with line-of-sight beacons;
(2) An algorithm for checking if a point or a region (sub-domain) is uniquely localizable given a placement;
(3) Two approximation algorithms for beacon placement: a greedy algorithm of approximation factor of $O(\ln m)$ and a random sampling algorithm with approximation factor of $O(\log \mathrm{OPT})$ guarantee where OPT is the optimal number of beacons.
(4) Implementation and evaluation of the algorithms on a variety of floor plans and comparison with user selected beacon solutions.

2 RELATED WORK

The Art Gallery problem and visibility. Mathematically, the problem of beacon selection for unique localization is closely related to the classical Art Gallery problem in computational geometry [21], where a minimum number of guards/beacons are selected to ensure all points of the domain (region within an indoor geometrical floor plan) have line of sight paths to (equivalently, are covered by) at least one guard/beacon. The Art Gallery problem is NP-hard even for simple polygons [17]. For any simple polygon P with n vertices, it has been proven that $\lfloor n/3 \rfloor$ guards are always sufficient [6] and sometimes necessary. For a polygon P with h holes, it was shown that P can be guarded with $\lceil \frac{n+h}{3} \rceil$ guards [3, 14]. Eidenbenz et al. proved the problem to be APX-hard [8], implying that it is unlikely that any approximation ratio better than some fixed constant. Ghosh [10] showed that a logarithmic approximation may be achieved by discretizing the input polygon into convex subregions. Valtr [31] showed, the set system derived from an art gallery problem has bounded VC dimension, allowing the algorithm

based on ε-net to obtain an approximation ratio as the logarithm of the optimal number of guards [5].

Also related to our work is the *k-coverage* set problem. Here, the goal is to find the minimum set that covers all the points at least k times. Obviously the k-coverage problem is NP-hard as well. Several approximation algorithms have been proposed for this class of problems. Cormen [7] applied a greedy approach for *k-coverage* problem with a $O(k \log n)$-approximation solution, where n is the number of points. The ε-net technique was introduced for this problem when the sets have constant VC-dimensions [5, 12] and was shown to achieve approximation factor $O(\log \mathrm{OPT})$ [9, 13], where OPT is the optimal solution.

Our problem is not the Art Gallery problem as each point needs to see at least two or three beacons. It is not the *k-coverage* problem either. The points in the domain have to be uniquely localized, which can be achieved by two or three beacons, depending on geometry.

Beacon placement for localization. The optimal beacon placement for a single target location is well understood [2, 27]. Onur and Volkan defined the uncertainty of points and proved that the beacon placement problem, to make the uncertainty of target points below the threshold, is NP-complete [30]. Hence, several heuristic-based optimization algorithms have been proposed for general beacon placement [16, 18, 26]. The state-of-art in beacon placement from commercial beacon vendors [1, 4, 20] suggest guidelines to account for the height of beacons and areas where better accuracy is desired and full coverage. However, this is far from a systematic approach for beacon placement.

A class of prior work has studied optimal beacon placement based on optimizing the placement for certain localization accuracy criteria [22, 29]. Though the accuracy is hard to quantify in the general case, under the assumption that the ranging noise is additive Gaussian noise, independent of the range, one can compute the Cramér-Rao bound (CRB) [15] for a beacon-target geometry and aim to optimize it across the region to be localized.

These prior works rely on at least three beacons for each location. It was first proposed in [24] that two beacons for ToF-based indoor localization system may also work, by considering the floor plan. They present a greedy algorithm for beacon placement for optimizing coverage and accuracy which has no theoretical guarantee. We compare our work with this algorithm in our experiments.

3 PROBLEM DEFINITION

In this section, we first provide the background on localizing with line-of-sight (LOS) beacons, state our assumptions, formulate the problem and introduce notation and definitions.

Localizing with LOS beacons: We illustrate the localization problem in Figure 1. Typical localization approaches use three beacons. When the floor plan information is available, which constrains the beacon's coverage, we can sometimes localize with just two beacons. This concept is key to our minimal beacon placement problem.

Assumptions:

2D deployment: We assume that the 2D representation of the floor plan is available, we also perform the beacon placement in 2D. In reality, we deploy the beacons in 3D by varying the heights at which the beacons are deployed. The 2D beacon placement is effective for 3D deployment if we deploy the beacons close to ceiling level and the user holds the device at regular height (around 1m from ground). Most temporary obstructions in the environment such as chairs, tables, etc would not change the beacon coverage. However, our 2D model assumption would not hold for 3D if the beacons are deployed at floor level and blocked by objects or if the beacons are deployed at ceiling and obstructions such as cubicle partitions are much taller than the user and block the device held by the user from the beacon. For practical purposes, the beacons can be deployed at ceiling level in most public spaces such as airports, museums, malls, and the floor plan 2D coverage and assumption will be applicable.

Ray-tracing coverage: While deploying beacons, we assume pure line-of-sight (LOS) coverage. However, while the system is in use, we receive non-line-of-sight (NLOS) signals and cope with it while solving for location. We compute the LOS coverage based on the beacon range and the ray-tracing coverage area. The ray tracing coverage is as follows: If a point is visible from a beacon (in line-of-sight), we receive an exact measurement from the beacon, otherwise, we do not receive a measurement. When NLOS measurements are received under this deployment (due to reflecting off walls or signals penetrating through walls), we adopt two approaches to cope with NLOS while estimating location. First, we apply localization techniques that localize in the presence of LOS and NLOS measurements with [25] or without the floor plan [19, 33]. The second approach is to detect NLOS signals based on the signal strength or statistical properties of the signal [11, 23, 28]. In this way, as we cannot predict the NLOS signals, we design the beacon placement for LOS coverage and cope with NLOS signals while estimating location.

Definitions: Mathematically, we formulate the optimal beacon placement problem as follows. The floor plan is represented by P. Any permanent walls and obstructions inside are modeled as holes in the polygon. Thus, P refers to the region inside the exterior polygon except the holes. Further, B is a set of *candidate beacon locations*, $B = \{b_i | 1 \leq i \leq n, b_i \in P\}$, that guarantees unique localization. The problem is to find a minimum set of beacons $D \subseteq B$ of size k such that the entire polygon P or all the target points of size m can be uniquely localized by beacons located at D.

Definition 3.1. **(Visibility):** Two points $p, q \in P$ are visible to each other if and only if the line segment $\overline{pq}$ is strictly inside P, i.e., does not intersect any point on the boundary of P. We can also say p sees q or vice versa.

Definition 3.2. **(Visible Region):** For a beacon b_i, the set of points seen by b_i, is the visible region of b_i, denoted as $V(b_i)$.

Definition 3.3. **(Visible Beacon Set):** For a point $p \in P$, the visible beacon set of p, $V(p)$, is the set of candidate beacon locations that can see p. To represent the subset of beacons in the set D that are visible at p, we use $V_D(p) = \{b_i \in D | b_i \text{ can see } p\}$.

Definition 3.4. **(Unique Localization of a Point p):** Given a point p and a set of visible beacons $V_D(p)$, we say p can be *uniquely localized*, i.e., $UL_D(p) = 1$, if there is only one location consistent with the range measurements and the visibility information.

If all the points $p \in P$ can be uniquely localized, then we say P can be uniquely localized. We denote by $UL_D(P) = 1$.

The rest of the paper is organized as follows. In section 4, we describe in detail the algorithm to verify unique localization of a point and a region. The two algorithms are presented in section 5 and section 6 respectively. Finally in section 7, we implement the algorithms and present the placement results.

4 ALGORITHM TO VERIFY UNIQUE LOCALIZATION

In this section, we describe algorithms for verifying whether a point $p \in P$ or the domain P is uniquely localized, given a set of beacons D. We use these subroutines in the beacon placement algorithms.

4.1 Unique Localization of a Point

With beacons at D, a point $p \in P$ can receive range measurements from beacons in $V_D(p)$. We define $|V_D(p)|$ as the number of beacons visible to p. If p can see at most one beacon, i.e. $|V_D(p)| \leq 1$, p cannot be uniquely localized. If p can see three or more beacons, $|V_D(p)| \geq 3$, then p is uniquely localized. The case of p seeing exactly two beacons, say b_i and b_j, needs more discussion, as described next. In this case, the range measurements from b_i and b_j generate two circles that intersect (in general) at two points p_1 and p_2. We call these two points *mirrors* of each other; symmetric about the line joining b_i, b_j. Figure 1(b) shows the ambiguity produced with two beacons. Both locations p_1 and p_2 are consistent with the range measurements and the visibility information. So we do not know which one is the true location. In Figure 1(d), p_2 is outside the domain. In Figure 1(e), p_2 cannot see b_2. In Figure 1(f), p_2 can see b_3. In these cases, we can conclude that p_1 is the unique location. Also, if the range information from two beacons leads to only one intersection (when the two circles are tangential), there is a unique solution.

Fact 1. *If a location p is only visible to two beacons b_i and b_j among D, the location of p is uniquely determined by the range information from b_i, b_j if and only if 1) the range information produces exactly one intersection p_1; or, 2) the range information produces a pair of mirror points p_1, p_2 but $V_D(p_1) \neq V_D(p_2)$.*

The test for unique localization of points is summarized below. in Equation 1. For each point $p \in P$, we can enumerate k beacons and check whether p can see these beacons. If $|V_D(p)| = 2$, we check its mirror point, otherwise, we can get the unique localization status of p through the cardinality of $V_D(p)$.

$$UL_D(p_1) = \begin{cases} 1, & |V_D(p_1)| \geq 3 \\ 1, & |V_D(p_1)| = 2, p_1 = p_2 \\ 1, & |V_D(p_1)| = 2, V_D(p_1) \neq V_D(p_2) \\ 0, & |V_D(p_1)| = 2, V_D(p_1) = V_D(p_2), p_1 \neq p_2 \\ 0, & |V_D(p_1)| \leq 1 \end{cases} \quad (1)$$

4.2 Unique Localization of a Region

To check if a domain P is uniquely localizable by beacons in D, we compute the visible regions of the beacons in D.

Definition 4.1. **(Canonical Region):** Given a domain P and a set of beacons D, P can be partitioned into disjoint regions such that all the points in the same region see exactly the same set of beacons. These regions are called the canonical regions, denoted by $Q_1, Q_2, \ldots, Q_c$.

$$Q = Q_1 \cup Q_2 \cup \cdots \cup Q_c \quad (2)$$

$$\forall Q_i, Q_j, i \neq j, Q_i \cap Q_j = \emptyset \quad (3)$$

$$Q_i = \{q \in Q | V_D(q) = V_D(p), p \in Q\} \quad (4)$$

To test the unique localization of a canonical region Q_i, we check the cardinality of $V_D(Q_i)$. When $|V_D(Q_i)| = 2$, we need to further consider the ambiguity due to localization with two beacons. Hence, we define the mirror region of Q_i.

Definition 4.2. **(Mirror Region):** For a canonical region Q_i with $V_D(Q_i) = \{b_i, b_j\}$, the mirror region of Q_i is the region that is symmetric to Q_i with respect to the line through b_i, b_j.

Similarly, if the canonical region Q_i has different visible beacon set from its mirror region's set, it can be uniquely localized.

Thus, a domain P is uniquely localized when all the canonical regions are uniquely localized.

5 A GREEDY ALGORITHM

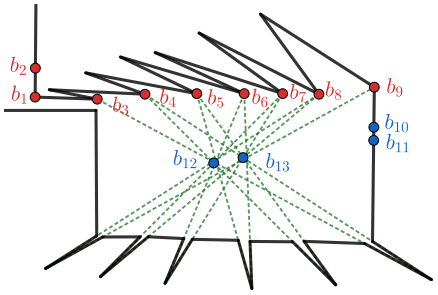


Figure 2: An example for which prior work [24] can be sub-optimal. There are many tips above and below the rectangle in the right and we call them as top tips and bottom tips. b_{10} and b_{11} can see all the points in the top tips. b_{12} and b_{13} see all the points in the bottom tips and part of the top tips. The greedy algorithm in [24] will select the red beacons by the order of their index instead of these four blue beacons, because once the rectangle is uniquely localized, the algorithm will next try to cover each tip that is seen once. When the number of tips is infinite, the greedy algorithm would place infinite beacons while the optimal solution is to place the beacon b_1, b_2 and four blue beacons.

To find a small number of beacons that uniquely localize the entire domain P or all the target points, an intuitive approach is to use a greedy approach based on some optimization criteria. For example, in [24], the beacon that maximizes the extra area that is uniquely localized is chosen. This algorithm, unfortunately, may place a lot more beacons than the minimum number needed, as in Figure 2.

We design a new greedy algorithm by showing a variant that optimizes a different objective function is monotone and submodular. A function on a subset of Ω is monotone and submodular if

- for every $X \subseteq Y \subseteq \Omega$, $f(X) \leq f(Y)$.
- for every $X \subseteq Y \subseteq \Omega$, and $x \in \Omega \setminus Y$, we have

$$f(X \cup \{x\}) - f(X) \geq f(Y \cup \{x\}) - f(Y).$$

We refer to this algorithm as the *Submodular Algorithm*. The submodular property is important because a greedy algorithm that greedily optimizes a submodular function can be shown to have a logarithmic approximation ratio [32]. The area of uniquely localizable region does not have the submodular property. For example, when you add the first beacon, the area of the unique localization region does not increase. When the second beacon is placed, this area is likely to be increased, contrary to the property of the submodular function.

5.1 Algorithm Design

For simplicity of explanation, we describe this algorithm in a discrete problem setting, uniquely localizing all the m target points.

Definition 5.1. **(Visibility Level Region:)** Given a domain P and a beacon location set D, $V_1(D)$ ($V_2(D)$) is the set of target points that see at least one (two) beacon in D. $U(D)$ is the set of target points that are uniquely localized by D.

$$V_1(D) = \{p \in P | \exists b \in D, p \in V(b)\}$$
$$V_2(D) = \{p \in P | \exists b_1, b_2 \in D, p \in V(b_1) \text{ and } p \in V(b_2)\}$$
$$U(D) = \{p \in P | UL_D(p) = 1\}$$

From the definition above, it is clear that $U(D) \subseteq V_2(D) \subseteq V_1(D) \subseteq P$. We define a utility function $F(D)$ below.

$$F(D) = 3|V_1(D)| + 2|V_2(D)| + |U(D)| \tag{5}$$

When all the target points are uniquely localized by D, we have $V_1(D) = V_2(D) = |U(D)| = m$. Thus, the beacon locations need to be selected until $F(D) = 6m$. Our Submodular algorithm selects the next beacon location that maximizes $F(D)$ instead of $U(D)$.

5.2 Approximation Bounds

We prove that the utility function $F(D)$ is submodular and our algorithm can achieve an approximation factor of $O(\ln m)$.

Theorem 5.2. *The utility function $F(D) = 3|V_1(D)| + 2|V_2(D)| + |U(D)|$ is monotone and submodular.*

Proof. Any extra beacon added to D can only help to increase $|V_1(D)|$, $|V_2(D)|$ and $|U(D)|$. Thus For any set $D \subseteq A \subseteq B$, we have $F(D) \leq F(A)$. So $F(D)$ is monotonically increasing.

To prove $F(D)$ is a submodular function, it suffices to show that $F(A \cup \{b\}) - F(A) \leq F(D \cup \{b\}) - F(D)$, where $D \subseteq A \subseteq B$ and $b \in B \setminus A$. We consider the components of the function F separately:

$$F(A \cup \{b\}) - F(A) - (F(D \cup \{b\}) - F(D)) = 3F_1 + 2F_2 + F_3$$

where

$$\begin{aligned} F_1 &= (|V_1(A \cup \{b\})| - |V_1(A)|) - (|V_1(D \cup \{b\}| - |V_1(D)|) \\ F_2 &= (|V_2(A \cup \{b\})| - |V_2(A)|) - (|V_2(D \cup \{b\})| - |V_2(D)|) \\ F_3 &= (|U(A \cup \{b\})| - |U(A)|) - (|U(D \cup \{b\}| - |U(D)|) \end{aligned}$$

In the following proof, we use set subtraction. When one set contains the other completely, the result of subtraction is the same as the subtraction of their value. Thus, we have the following equalities.

$$\begin{aligned} F_1 &= |V_1(b)| - |V_1(b) \cap V_1(A)| - |V_1(b)| + |V_1(b) \cap V_1(D)| \\ &= -|V_1(b) \cap (V_1(A) \setminus V_1(D))|; \\ F_2 &= |V_1(b) \cap (V_1(A) \setminus V_2(A))| - |V_1(b) \cap (V_1(D) \setminus V_2(D))| \\ &= |V_1(b) \cap (V_1(A) \setminus V_1(D))| - |V_1(b) \cap (V_2(A) \setminus V_2(D))| \end{aligned}$$

F_3 considers only target points that become uniquely localizable with b added. F_3 has two parts $F_3 = F_4 + F_5$ where F_4 counts the change of the target points that are seen by three or more beacons and newly become uniquely localizable with the addition of b; and F_5 counts the change of the target points that can be seen by *exactly two* beacons and newly become uniquely localizable.

F_4 can be represented, in the same manner as F_1 and F_2:

$$\begin{aligned} F_4 &= |V_1(b) \cap (V_2(A) \setminus U(A))| - |V_1(b) \cap (V_2(D) \setminus U(D))| \\ &= |V_1(b) \cap (V_2(A) \setminus V_2(D)| - |V_1(b) \cap (U(A) \setminus U(D))| \end{aligned}$$

The trickiest part is to analyze F_5. We consider target points that see two beacons and newly become uniquely localizable.

First, we consider the beacon set $A \setminus D$, the set of beacons in A but not in D. If a point p is seen by two beacons, one is b, and the other is from $A \setminus D$, p can only possibly become uniquely localizable for beacon set A, upon the addition of b. p cannot possibly be uniquely localizable for $D \cup \{b\}$.

Second, when a point p is seen by b and exactly one beacon from D, and its mirror point p' is already uniquely localizable with D, p now becomes uniquely localizable.

Third, when b is added, a point p can become uniquely localizable because its mirror point p' just becomes uniquely localizable (seeing three beacons including b).

Thus, we add the three components together.

$$\begin{aligned} F_5 \leq\ & |V_1(b) \cap (V_1(A) \setminus V_1(D))| + |V_1(b) \cap (U(A) \setminus U(D))| \\ & + |V_1(b) \cap (V_2(A) \setminus V_2(D))| \end{aligned}$$

Now, we can bound these components and obtain the results:

$$\begin{aligned} & F(A \cap \{b\}) - f(A) - (F(D \cup \{b\}) - F(D)) \\ = \ & 3F_1 + 2F_2 + F_4 + F_5 \\ \leq \ & -|V_1(b) \cap (V_1(A) \setminus V_1(D))| - |V_1(b) \cap (V_2(A) \setminus V_2(D))| \\ & -|V_1(b) \cap (U(A) \setminus U(D)| + |V_1(b) \cap (V_1(A) \setminus V_1(D))| \\ & + |V_1(b) \cap (V_2(A) \setminus V_2(D))| + |V_1(b) \cap (U(A) \setminus U(D))| \\ = \ & 0 \end{aligned} \tag{6}$$

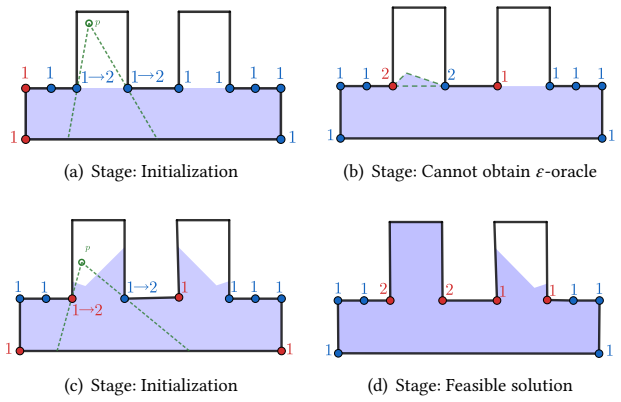
Therefore, $F(D)$ is a submodular function. □

Owing to its submodular properties, the greedy algorithm has an approximation factor following the standard argument [32].

Theorem 5.3. *In discrete problem setting, the approximation ratio of the submodular algorithm is $O(\ln m)$. In the continuous problem setting, the target points are replaced by the domain P, the approximation ratio is $O(\min\{\ln \frac{|P|}{\triangle}, n\})$, where $|P|$ is the area of the domain, $\triangle$ is the minimum increased area for the objective function F with a new guard and n is the number of candidate beacons.*

6 RANDOM SAMPLING ALGORITHM

In this section, we present a different approximation algorithm using random sampling technique with slightly improved approximation factor. This algorithm is motivated by the ε-net based algorithm for geometric set cover which gives an approximation factor



(a) Stage: Initialization (b) Stage: Cannot obtain ε-oracle

(c) Stage: Initialization (d) Stage: Feasible solution

Figure 3: Illustration of Algorithm 1: Each beacon is set the weight and the red beacons are selected beacon locations. The purple region contains all the points with weight greater than $\varepsilon w(B)$.

of log OPT where OPT is the size of the optimal solution [9]. But due to the specific requirement of the unique localization problem, our problem is different and we need to carefully get around the technical difficulties.

6.1 Algorithm Design

In the algorithm we give weights to each candidate beacon location b_i and each point $p \in P$.

Definition 6.1. **(Weight):** Define the weight of a beacon b_i by $w(b_i)$. For a set of beacons D, its weight is $w(D) = \sum_{b \in D} w(b)$. The weight of a point $p \in P$ is defined by the weight of the beacons that can see p, $w(p) = w(V(p))$.

Definition 6.2. **(ε-oracle):** Given a domain P, a candidate beacon location set B and the weight function w, a subset $D \subseteq B$ is an ε-oracle for (P, B, w) if for any $p \in P$ with $w(p) \geq \varepsilon w(B)$, p is uniquely localized by D.

In our algorithm (Alg 1), initially all beacon locations carry the same weight of 1. We randomly select k beacons, in which a beacon is selected with probability proportional to its weight. This random sample has a good probability to be a ε-oracle, by the proof in the next subsection. If the beacons provide unique localization for all points of P, the algorithm terminates. Otherwise, we double the weights of the beacons seen by the points that are not uniquely localized. We iterate this process and can show that this terminates when k is $O(\text{OPT} \log \text{OPT})$, where OPT is the size of the minimum number of beacons supporting unique localization. Since we do not know what is OPT, we start with $k = 2$. When we execute several iterations and still cannot find a feasible solution, k is doubled until a feasible solution is obtained.

Figure 3 illustrates each step of the ε-oracle algorithm. There are 10 candidate beacon locations with weights. We start with $k = 2$ and $\varepsilon = \frac{1}{k}$. In Figure 3(a), all the weights are 1. The total weight is 10. According to Definition 6.2, the region in purple contains all the points with weight greater than $\varepsilon w(B) = 5$. We select two beacons shown in red. They uniquely localize the purple region but

Algorithm 1: Random Sampling Algorithm

Input: The floor plan P, Candidate beacon locations set B
Output: A feasible beacon locations set D that can localize P

```
for k = 2; k ≤ m; k* = 2 do
    ϵ = 1/k;
    Reset all the weights of beacons in B to 1;
    for i = 0; i ≤ (2k/δ) log2(m/k); i++ do
        TotalWeight ← Σ_{b∈B} w(b);
        Prob(b) ← w(b)/w(B);
        Select a beacon set D of size k according to Prob(b);
        if D is ε-oracle then
            if Domain P is uniquely localized by D then
                return D
            else
                Select a point p is not uniquely localized;
                Double the weight of all beacons in V(p);
```

do not uniquely localize the whole region P. Thus, we find a point p (shown as a green node) that is not uniquely localized. There are two candidate beacons that can see p, and their weights are doubled, from 1 to 2. In Figure 3(b), as the weights are changed, the total weight is 12 now and we restart the random sampling. Two red beacons are selected, but the region bounded by the green lines is not uniquely localized. So these beacons are not ε-oracle. After a few trials of random sampling, we pause as we believe that two beacons are not enough.

Then, we double the size k to 4 in Figure 3(c). All the weights are reset to 1. Again the region in purple shows the points of high weight. Four beacons are selected, uniquely localizing the purple region. They are ε-Oracle, but a point $p \in P$ can be found that is not uniquely localized. The weights of the beacons covering p are doubled. Finally, we find the beacon locations shown in Figure 3(d). These four beacons can uniquely localize the whole region P. The solution is obtained and the algorithm terminates.

Notice that in the above algorithm k is always a power of 2. While this is only giving a factor of 2 in the approximation factor theoretically, we would like to optimize k in practice. Thus, we run binary search to find the smallest k such that randomly selected beacons provide unique localization. We set lower bound LB = 1 and higher bound HB = $|B|$. In each iteration, we set $k = (\text{LB} + \text{HB})/2$. If a solution is obtained, HB = k, otherwise LB = $k + 1$. The iterations are executed until LB = HB, when we find the best k.

6.2 Algorithm Analysis

To analyze the Random Sampling algorithm, first we introduce the concept of VC-Dimension [12], used in our following analysis.

Definition 6.3. **(VC-Dimension):** Given a set system $(X, \mathcal{R})$, let A be a subset of X. We say A is shattered by $\mathcal{R}$ if $\forall Y \subseteq A$, $\exists R \in \mathcal{R}$ such that $R \cap A = Y$. The VC-dimension of $(X, \mathcal{R})$ is the cardinality of the largest set that can be shattered by $\mathcal{R}$.

THEOREM 6.4. *[12] For a set system (X, C) with $|X| = n$ and VC-dimension d, $|C| \leq n^d$.*

Now, we consider the VC-dimension of our case. We regard each canonical region as an element. Beacon b with visible region $V(b)$ is a set that contains all canonical regions in $V(b)$. Therefore the beacons correspond to sets in a set system, which is exactly the set system of the Art Gallery problem. The VC dimension of the Art Galley problem is at most 23 by [31].

Notice that our problem of beacon selection for unique localization is not a set cover problem. So we have to develop the bounds from scratch. First, we prove that an ε-oracle can be found by random sampling with high probability. Compared with the previous proof of ε-net [9], our analysis differs in the following aspects:

(1) The definitions and description of some events in the proof are in geometrical ways instead of combinatorial ways.
(2) When calculating the probability of some events, owing to the ambiguity of unique localization, some inequalities are used to simplify the proof.
(3) Through rigorous analysis, a higher probability than the prior work [9] is obtained, which leads to better running time for the algorithm.

Theorem 6.5. *Given (P, B, w) and*

$$k \geq \max\left(\frac{2}{\varepsilon}\log_2\frac{1}{2\delta}, \frac{4d+16}{\varepsilon}\log_2\frac{4d+16}{\varepsilon}\right), \tag{7}$$

let D be k beacons picked randomly from B with probability proportional to their weights. D is an ε-oracle with probability at least $1-\delta$, where d is the VC-dimension of the set system, $d \leq 23$.

Proof. After randomly picking D, we pick another set T in the same way as D. We denote $Z = D \cup T$. Now, we define two events

$$\begin{aligned} E_1 = &\ \{\exists p \in P\ s.t.\ p \text{ is not uniquely localized}, w(p) \geq \varepsilon w(B)\} \\ E_2 = &\ \{\exists p \in P\ s.t.\ p \text{ is not uniquely localized}, w(p) \geq \varepsilon w(B), \\ &\ |V(p) \cap Z| \geq \varepsilon k\} \end{aligned}$$

First, we prove that $\Pr[E_1] \leq 2\Pr[E_2]$, i.e. $\Pr[E_2|E_1] \geq 1/2$ according to the conditional probability. When event E_1 happens, there are some points p that are not uniquely localizable but $V(p) \geq \varepsilon w(B)$. We denote $Y = |V(p) \cap Z|$ and the expected value of Y is at least $2\varepsilon k$, i.e. $E[Y] \geq 2\varepsilon k$, because the selection probability is proportional to their weights. We can also get that the variance of Y is at most $2\varepsilon k$, i.e. $\mathrm{Var}[Y] \leq 2\varepsilon k$. Hence, according to Chebyshev's inequality,

$$\Pr[Y \leq \varepsilon k] \leq \Pr[|Y - E[Y]| \geq \epsilon k] \leq \frac{\mathrm{Var}[Y]}{(\varepsilon k)^2} \leq \frac{2\epsilon k}{(\varepsilon k)^2} \leq 1/2$$

It could be verified that $k \geq \frac{4d+16}{\varepsilon}\log_2\frac{4d+16}{\varepsilon} > \frac{4}{\varepsilon}$ owing that k is given by Equation 7. Thus, we have

$$\Pr[E_2|E_1] \geq \Pr[Y \geq \epsilon k] \geq 1/2$$

Now consider a point p and fix the set Z, when p is not uniquely localized by D, $|V(p) \cap D| < 3$. Thus, we define the event E_p as

$$E_p = \{|V(p) \cap D| < 3, |V(p) \cap Z| \geq \varepsilon k\}$$

We consider the value of $\Pr[E_p]$ with a given Z. Suppose $|V(p) \cap Z| = l$, we have $l \geq \varepsilon k$. There are at most 2 elements of l in D. The selected set D is comprised of $k-2$ elements from $Z - V(p) \cap Z$ and the other 2 elements elsewhere. So

$$\Pr[E_p] = \frac{\binom{2k-l}{k-2}\binom{k+2}{2}}{\binom{2k}{k}} \leq 4(k)^4\frac{\binom{2k-\varepsilon k}{k}}{\binom{2k}{k}} \leq (k)^4 2^{-\varepsilon k+2}$$

The first inequality can be proved by mathematical method. Based on $\Pr[E_p]$, the value of $\Pr[E_2]$ is bounded. For any two points $p_1, p_2 \in P$ with $w(p_1), w(p_2) \geq \varepsilon w(B)$ and $V(p_1) \cap Z = V(p_2) \cap Z$, the events E_{p_1} and E_{p_2} are the same in E_2. Thus, the occurrence of E_p depends only one the intersection $V(p) \cap Z$. Recall that our set has a constant VC-dimension, so we can get the following by Theorem 6.4:

$$\begin{aligned} \Pr[E_2] &\leq \cup_{p|V(p)\cap Z \text{ is unique}} \Pr[E_p] \\ &\leq (2k)^d (k)^4 2^{-\varepsilon k+2} = (2k)^{d+4} 2^{-\varepsilon k-2} \end{aligned}$$

Thus, we can get the probability of E_1.

$$\Pr[E_1] \leq 2\Pr[E_2] \leq (2k)^{d+4} 2^{-\varepsilon k-1} \tag{8}$$

We need to show $\Pr[E_1] \leq \delta$. We can rewrite Equation 8 as:

$$\varepsilon k \geq \log\frac{1}{2\delta} + (d+4)\log(2k)$$

Thus, for the value of k, when we have $\frac{1}{2}\varepsilon k \geq \log\frac{1}{2\delta}$ and $\frac{1}{2}\varepsilon k \geq (d+4)\log(2k)$, the theorem holds. We can verify when k satisfies Equation 7, both inequalities hold.

Hence, when k was given by Equation 7, the selected set D is an ε-oracle with probability at least $1-\delta$. □

Based on the above theorem, we can use random sampling to get an ε-oracle with probability $1-\delta$. However, this ε-oracle may not be the solution to our problem because for some points with low weights, they might not be uniquely localized. This will be fixed by the weight doubling process during the iterative procedure. The number of iterations can be bounded using exactly the same analysis as in [12]:

Theorem 6.6. *Suppose k is the size of an feasible solution, we set $\varepsilon = \frac{1}{k}$. Then $\frac{2k}{\delta}\log_2\frac{m}{k}$ iterations of Algorithm 1 are sufficient to find a feasible solution.*

Based on the Theorem 6.5 and 6.6, we get that Algorithm 1 can get a satisfied solution. When we estimate the optimal solution is OPT and set $\varepsilon = \frac{1}{\mathrm{OPT}}$, we will get a solution of size $O(\mathrm{OPT}\log\mathrm{OPT})$.

6.3 Considering Localization accuracy

In reality, range measurements are noisy and the localization error depends on the range error and the geometry of the beacons. The contribution due to the beacon geometry on the localization error is captured by the Geometric Dilution of Precision (GDOP). A lower GDOP is associated with lower bounds on the variance of the location estimate and hence lower localization errors. The GDOP metric at a location covered by a set of beacons is defined by the angle subtended between the location and pairs of beacons. When two beacon locations are very close, for most of their commonly visible points, the angles between two beacons is very small, which causes the GDOP to be large and hence increases the localization error. Thus, we introduce a heuristic to include the incentive to select beacon locations to be far away from each other.

We add this correction to the random sampling algorithm. In Line 7 of Algorithm 1, the beacon set D are selected according to their weight. We can select these beacons one by one and adjust the other beacons' weights, to avoid selecting beacons that are close to each other. When a beacon location b is selected, from the beacon locations that can see b, we select some whose distances with b

Floor Plan	Range	Beacon Selection Method					
		Opt	RS	Sub	UL	GDOP	3-Place
Map1(1)	inf	8	8	8	8	9	11
Map1(2)	inf	7	8	8	8	9	10
Map1(3)	5m	≤ 13	14	14	14	16	19
Star-Shape	inf	3	3	3	3	3	3
Multi-Room	inf	6	6	8	8	8	9
Map2	10m	≤ 24	24	24	24	28	29
Map3	inf	6	6	6	7	8	9

Table 1: number of beacons placed by various algorithms

are less than a predefined threshold. These beacons' weights are reduced, e.g. taking off 30% of their weight. This way, the probability of selecting a nearby beacon location is decreased. After Line 7, the beacon set D is obtained and their weights are restored to the original weights.

7 EVALUATION BY SIMULATION

For evaluation purposes we implemented the proposed algorithms in a MATLAB-based toolchain. We considered different approaches to generate floor plans, select initial location candidates and beacon placement algorithms.

(1) Floor plan generation: We implemented these types of floor plans:

- User drawn floor plan through our GUI on MATLAB
- Randomly generated simple polygon with user-defined number of vertices
- A pre-defined floor plan. This option is for proving real-world floor plans as inputs.

(2) Candidate beacon locations: Next, we tested with beacon placements with different ranges. The range can be infinite (*Inf*), in which case it is only limited by the floor plan boundaries, or can finite, specified in meters. The beacon locations can be vertices or interior points. The interior points can further be randomly generated or user-specifies through the MATLAB GUI.

(3) Beacon placement algorithms: We implemented these placement schemes in the toolchain:

- **GDOP:** Prior work in minimizing beacons while maximizing expected accuracy based on GDOP [24].
- **UL:** Prior work in minimizing beacons while maximizing coverage [24].
- **RS:** Our proposed random sampling algorithm described in subsection 6.1.
- **Sub**: Our proposed submodular-function algorithm described in section 5.
- **mod-RS**: Our proposed modified random sampling algorithm for accuracy, described in subsection 6.3.
- **Opt**: This is the optimal that our algorithm aims to achieve. We obtain this by searching through all possible solutions.
- **3-Place**: Optimal solution where any point in the domain are covered at least three beacons. This represents the best-case scenario with typical placement methodology.

# of Vertices in the floor plan	# of beacons (% reduction wrt UL)		
	UL	RS	Sub
20	6.2	5.2(**16%**)	6(**3%**)
40	11.6	9.6(**18%**)	11.2(**3%**)
60	17	15.2(**7%**)	16(**6%**)
80	22.2	19.4(**13%**)	20.2(**9%**)
100	30.6	28.8(**6%**)	29.6(**3%**)

Table 2: Performance as random floor plan scales up

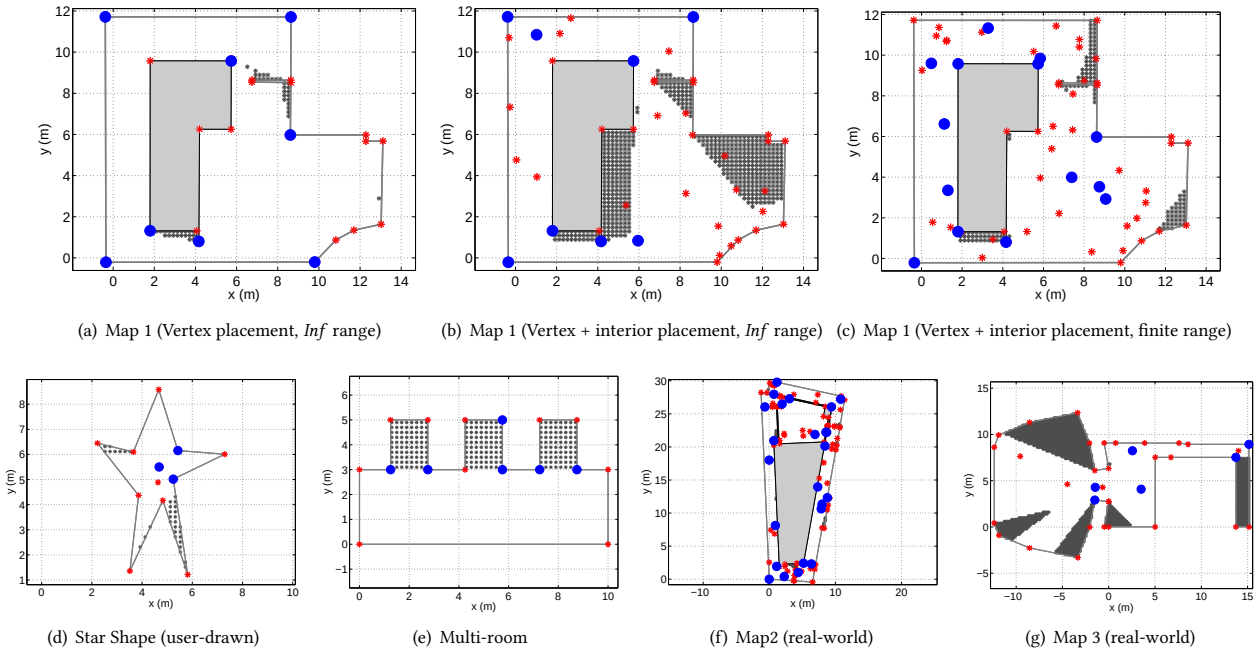
7.1 Number of beacons

In this section, we compare the performance of the different algorithms for various floor plans and potential beacon location settings. These floor plans were chosen to be small enough such that we could compute optimal beacon placement. Figure 4 shows the floor plans. Figure 4(a)-Figure 4(c) show a real-world floor plan with different placement settings (vertex and interior placement, infinite and finite beacon range). Figure 4(d) shows a user-drawn star-shaped polygon. Figure 4(e) shows a floor plan representative of a large area and small rooms. Figure 4(f) and Figure 4(g) are real-world floor plans. The set of blue and red dots together indicate all possible beacon locations. The set of blue dots indicate the beacon locations selected by the proposed *Random Sampling (RS)* algorithm for placement. After placement, all regions are uniquely localized. The regions in white are covered by three or more beacons. The regions in dark grey are covered by 2-beacons and uniquely localized. Table 1 summarizes the results of the number of beacons placed by the different algorithms. We wish to make a few observations: (1) The number of beacons placed by our RS algorithm is similar to prior work, and close to the optimal number of beacons. Hence, we are able to have good performance in beacon placement, with an algorithm that has guarantees. (2) For Map1, there is not much difference in number of beacons with vertex only or vertex and interior placement when the beacon range is unlimited. We also see that even when allowed interior points for placement, half of the beacon locations selected are at vertices due to their high coverage. (3) When the range is limited, interior locations are chosen more often due to them having a higher coverage (4) Our greedy algorithm does better than the previous greedy algorithm (UL) in some scenarios such as Multi-room map. (5) For real-world floor plans where the beacon range is limited by the floor plans, such as Map 1(b) and Map 3, our placement localizes a large part of the regions with just 2 beacons rather than 3, as seen by the large amount of area shaded in grey. The running time of these algorithms are roughly similar.

7.2 Performance at scale

To test the performance of the beacon placement algorithms at scale, we simulated random simple polygons with number of vertices varying from 20 to 100 in steps of 20. In every floor plan, we randomly generated an additional 30 interior points for candidate beacon locations. We generated 5 floor plans for every fixed number of vertices. Table 2 shows the average number of beacons placed with each algorithm as the number of vertices increases. The algorithm in the prior work (UL) is set as the baseline.

We also show the percentage reduction in number of beacons placed by our algorithms. We observe that the number of beacons placed by *RS* is lower in general as the complexity of the floor plan



(a) Map 1 (Vertex placement, *Inf* range) (b) Map 1 (Vertex + interior placement, *Inf* range) (c) Map 1 (Vertex + interior placement, finite range)

(d) Star Shape (user-drawn) (e) Multi-room (f) Map2 (real-world) (g) Map 3 (real-world)

Figure 4: Beacon placement using *Random Sampling* for various floor plans

grows. The reason is - the algorithms that are greedy optimize for coverage and place beacons sequentially, whereas the random sampling algorithm updates the weights of points and places all the beacons newly in every iteration based on the weight of points. This benefit is more evident in floor plans that have small geometrical spaces rather than large open spaces (which is the case with random floor plans).

7.3 Accuracy improvement with modified random sampling algorithm

In this section, we consider the accuracy that can be obtained from the beacon placement, based on the modified random sampling algorithm described in subsection 6.3. In order to quantify the quality of the beacon placement, we evaluate the CDF of the GDOP for the given placement as in [24]. We have presented the results for Map 2 shown in Figure 4. For a fair comparison, we use the first 24 beacons that each algorithm places, and compare the expected accuracy that we can get from each placement. The results are shown in Figure 5. We see that the improvement in performance from *RS* to *RS-mod* with 60% GDOP improving from 1.7 to 1.5, and comparable with the GDOP from prior work that was optimizing for accuracy (*GDOP*). Thus we can accommodate practical metrics that relate to accuracy in or algorithm.

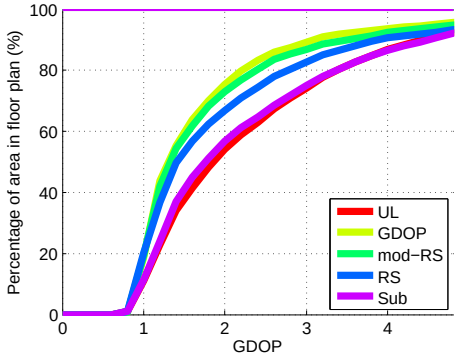
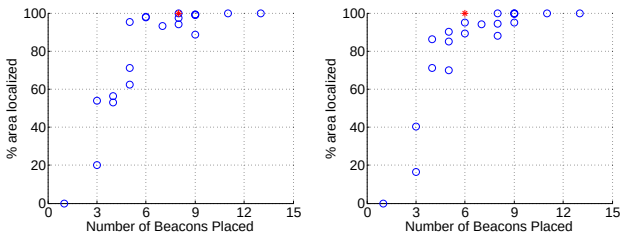


Figure 5: Comparison of expected accuracy with different placement algorithms for Figure 4(f)

7.4 Comparison with user-placed beacons

Since our main contribution is a systematic approach for beacon placement that we believe to be difficult for most humans, we evaluated how our approach performs compares to manual placements from a number of users. We created an online form with four floor plans and asked users to place beacons with the challenge of placing as few beacons as required in order to cover as much of the floor plan as possible. Twenty anonymous users completed the challenge and we did not record any identifying information. We asked the users to indicate their experience with "beacon placement, localization systems, geometry algorithms". 40% indicated no experience, 30% indicated some experience and the remaining indicated reasonable or a lot of experience. The results are shown in Figure 6 for two floor plans. We see that for Map 3 shown in 6(b), the algorithm outperforms the users in terms of the trade-off between number of beacons and area localized. For Map 1 shown in 6(a), the algorithm places 8 beacons to localize the entire floor plan but 90% of the floor plan is localized with 6 beacons. This example indicates that when the constraint of unique localization is relaxed, further savings can be achieved in terms of beacon placement. Across the four tested



(a) Map1 (real-world) (b) Map 3 (real-world)

Figure 6: Comparison between RS-placed beacons (red dot) and user placed beacons (blue circles)

floor plans, perhaps unsurprisingly, we noticed the trend that as the environment becomes larger and more complex, our algorithm begins to more significantly out perform humans.

8 CONCLUSION

This paper presents a rigorous formulation of the unique localization problem in indoor environments. We observe, via simulation, that our beacon placement algorithm performs better than manual placements conducted by a number of volunteers we experimented with, even well versed with the problem. The proposed algorithm also places 5% fewer in real-world floor plans and 12% fewer on random floor plans compared to prior work based on heuristics. Beyond the improved performance, which, even for small percentage gain, can translate into large savings over sizable deployments, we believe that the value of the proposed methodology lies into its ability to provide a baseline to evaluate past and future deployment and to systematize the process of beacon placement. Future work will include relaxation of some key simplifying assumptions, and extensions to 3D environments.

ACKNOWLEDGEMENTS

Wang and Gao acknowledge support through NSF DMS-1737812, NSF CNS-1618391 and NSF CCF-1535900. Rajagopal, Rowe and Sinopoli would like to thank support by the CONIX Research Center (one of six centers in JUMP), a Semiconductor Research Corporation (SRC) program sponsored by DARPA, by the NIST Public Safety Communications Research division, and by National Science Foundation grant #1534114.

REFERENCES

[1] 2016. http://www.oym.co/beacon-placement/.
[2] Adrian N Bishop, Baris Fidan, Brian Anderson, Kutluyıl Doğançay, and Pubudu N Pathirana. 2010. Optimality analysis of sensor-target localization geometries. *Automatica* 46, 3 (2010), 479–492.
[3] Iliana Bjorling-Sachs and Diane L. Souvaine. 1995. An efficient algorithm for guard placement in polygons with holes. *Discrete & Computational Geometry* 13, 1 (1995), 77–109.
[4] Sarah Blaszak. [n. d.]. https://support.trxsystems.com/hc/en-us/articles/218603177-BLE-Beacon-Placement-Guidelines.
[5] Hervé Brönnimann and Michael T Goodrich. 1995. Almost optimal set covers in finite VC-dimension. *Discrete & Computational Geometry* 14, 4 (1995), 463–479.
[6] Vasek Chvatal. 1975. A combinatorial theorem in plane geometry. *Journal of Combinatorial Theory, Series B* 18, 1 (1975), 39–41.
[7] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. 2009. *Introduction to algorithms*. MIT press.
[8] Stephan Eidenbenz, Christoph Stamm, and Peter Widmayer. 2001. Inapproximability results for guarding polygons and terrains. *Algorithmica* 31, 1 (2001), 79–113.
[9] Giordano Fusco and Himanshu Gupta. 2009. ϵ-net approach to sensor k-coverage. In *International Conference on Wireless Algorithms, Systems, and Applications*. Springer, 104–114.
[10] Subir Kumar Ghosh. 2010. Approximation algorithms for art gallery problems in polygons. *Discrete Applied Mathematics* 158, 6 (2010), 718–722.
[11] Ismail Guvenc and Chia-Chin Chong. 2009. A survey on TOA based wireless localization and NLOS mitigation techniques. *IEEE Communications Surveys & Tutorials* 11, 3 (2009), 107–124.
[12] David Haussler and Emo Welzl. 1987. ϵ-nets and simplex range queries. *Discrete & Computational Geometry* 2, 2 (1987), 127–151.
[13] Mohamed Hefeeda and Majid Bagheri. 2007. Randomized k-coverage algorithms for dense sensor networks. In *IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications*. IEEE, 2376–2380.
[14] Frank Hoffmann, Michael Kaufmann, and Klaus Kriegel. 1991. The art gallery theorem for polygons with holes. In *Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on*. IEEE, 39–48.
[15] Steven M Kay. 1993. Fundamentals of statistical signal processing, volume I: estimation theory. (1993).
[16] Nicolaj Kirchhof. 2013. Optimal placement of multiple sensors for localization applications. In *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*. IEEE, 1–10.
[17] Der-Tsai Lee and Arthur K Lin. 1986. Computational complexity of art gallery problems. *Information Theory, IEEE Transactions on* 32, 2 (1986), 276–282.
[18] Tilman Leune, Thorsten Wehs, Marijn Janssen, Christian Koch, and Gerd von Colln. 2013. Optimization of wireless locating in complex environments by placement of anchor nodes with evolutionary algorithms. In *Emerging Technologies & Factory Automation (ETFA), 2013 IEEE 18th Conference on*. IEEE, 1–6.
[19] Xinrong Li. 2006. An iterative NLOS mitigation algorithm for location estimation in sensor networks. *Proc. 15th IST Mobile Wireless Commun. Summit* (2006), 1–5.
[20] Tiffany Morin. 2016. Beacon Placement Best Practices. https://gimbal.com/beacon-placement-best-practices/.
[21] Joseph O'rourke. 1987. *Art gallery theorems and algorithms*. Vol. 57. Oxford University Press Oxford.
[22] Neal Patwari, Joshua N Ash, Spyros Kyperountas, Alfred O Hero III, Randolph L Moses, and Neiyer S Correal. 2005. Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine, IEEE* 22, 4 (2005), 54–69.
[23] Chunyi Peng, Guobin Shen, Yongguang Zhang, Yanlin Li, and Kun Tan. 2007. Beepbeep: a high accuracy acoustic ranging system using cots mobile devices. In *Proceedings of the 5th international conference on Embedded networked sensor systems*. ACM, 1–14.
[24] Niranjini Rajagopal, Sindhura Chayapathy, Bruno Sinopoli, and Anthony Rowe. 2016. Beacon placement for range-based indoor localization. In *Indoor Positioning and Indoor Navigation (IPIN), 2016 International Conference on*. IEEE, 1–8.
[25] Niranjini Rajagopal, Patrick Lazik, Nuno Pereira, Sindhura Chayapathy, Bruno Sinopoli, and Anthony Rowe. 2018. Enhancing indoor smartphone location acquisition using floor plans. In *Proceedings of the 17th ACM/IEEE International Conference on Information Processing in Sensor Networks*. IEEE Press, 278–289.
[26] Javier O Roa, Antonio Ramón Jiménez, F Seco, José Carlos Prieto, and J Ealo. 2007. Optimal placement of sensors for trilateration: regular lattices vs meta-heuristic solutions. In *Computer Aided Systems Theory–EUROCAST 2007*. Springer, 780–787.
[27] Naveed Salman, Hemat K Maheshwari, Andrew H Kemp, and Mounir Ghogho. 2011. Effects of anchor placement on mean-CRB for localization. In *Ad Hoc Networking Workshop (Med-Hoc-Net), 2011 The 10th IFIP Annual Mediterranean*. IEEE, 115–118.
[28] Bruno Silva and Gerhard P Hancke. 2016. IR-UWB-based non-line-of-sight identification in harsh environments: Principles and challenges. *IEEE Transactions on Industrial Informatics* 12, 3 (2016), 1188–1195.
[29] Maurizio Spirito et al. 2001. On the accuracy of cellular mobile station location estimation. *Vehicular Technology, IEEE Transactions on* 50, 3 (2001), 674–685.
[30] Onur Tekdas and Volkan Isler. 2010. Sensor placement for triangulation-based localization. *Automation Science and Engineering, IEEE Transactions on* 7, 3 (2010), 681–685.
[31] Pavel Valtr. 1998. Guarding galleries where no point sees a small area. *Israel Journal of Mathematics* 104, 1 (1998), 1–16.
[32] Peng-Jun Wan, Ding-Zhu Du, Panos Pardalos, and Weili Wu. 2010. Greedy approximations for minimum submodular cover with submodular cost. *Computational Optimization and Applications* 45, 2 (2010), 463–474.
[33] Xin Wang, Zongxin Wang, and Bob O'Dea. 2003. A TOA-based location algorithm reducing the errors due to non-line-of-sight (NLOS) propagation. *IEEE Transactions on Vehicular Technology* 52, 1 (2003), 112–116.