CovBSM: An Optimization Approach to RF Beacon Deployment for Indoor Localization

Raphael Falque[1], Mitesh Patel[2], and Jacob Biehl[2]

***Abstract*— In this paper, we propose a novel solution to optimize the deployment of Radio Frequency (RF) beacons for the purpose of indoor localization. We propose a system that optimizes both the number of beacons and their placement in a given environment. We propose a novel cost-function, called CovBSM, that allows to simultaneously optimize the 3-coverage while maximizing the beacon spreading. Using this cost function, we propose a framework that maximize both the number of beacons and their placement in a given environment. The proposed solution accounts for the indoor infrastructure and its influence on the RF signal propagation by embedding a realistic simulator into the optimization process.**

Keywords: Indoor localization, Beacon deployment, Bluetooth Low Energy (BLE), k-Coverage, Wireless sensor networks (WSN).

I. INTRODUCTION

Localization problems can be solved effectively using Global Navigation Satellite Systems (GNSS). However, for indoor localization, GNSS is most often not available. As an alternative, localization solutions using different modalities such as RGB-D cameras [1], Inertial Measurement Unit (IMU) [2], and wireless access point (Wi-Fi) [3] have been developed by various researchers. Additionally, researchers have also developed multi-modal localization approaches by fusing two or more sensors [4], [5].

With wireless communication technologies becoming widespread and ubiquitous, they have been studied extensively as a solution to indoor positioning systems [6]. Various solutions based on the Bluetooth Low Energy (BLE) technology [5], [7] have been proposed. Most often, these methods use the Received Signal Strength Indication (RSSI) from the BLE beacons to estimates the user position. In indoor environments, the radio signals are severely impacted due to shadowing and multipathing effects which lead to noisy data. To receive better RSSI signal from BLEs, they are placed in locations where RSSI signals are available through Line-Of-Sight (LOS) thereby going through minimal multipathing.

Scaling BLE based indoor positioning becomes a challenging problem as it involves placing beacons at the optimal location to get a high level of localization accuracy. In most of the indoor positioning systems [5], [7], beacons are placed using expert knowledge about the environmental condition in order to achieve good localization accuracy. Hence the optimal placement of BLE beacons has been studied by researchers from the Wireless Sensor Network (WSN) and the robotics communities. The optimal BLE map is a function of different parameters such as the number of beacons, the signal frequency and transmission power, the indoor environment, and the required localization accuracy.

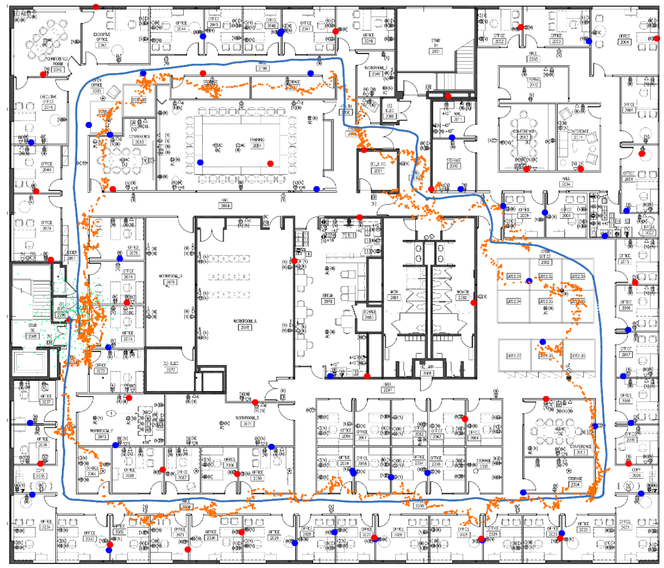


Fig. 1: Sample of an optimized BLE beacon deployment (red circles) chosen from the available beacon positions (blue and red circles). The deployment is evaluated by comparing the localization of a robot within the map (orange points) to the associated ground truth (blue line).

In this paper, we propose an approach to achieve optimal beacons placement that can be used for localization of a user/robot in a given environment. An illustrative example of beacon placement achieved using this approach is shown in Figure 1. The main content of the paper can be defined as follows:

i. We propose a novel technique to optimize the number and placement of beacons needed for localization of user/robot in a given environment. Unlike traditional Radio Frequency (RF) based beacon placement techniques [5], our proposed technique accounts for the interferences between RF signals and the environmental infrastructure. This is achieved by embedding a realistic simulator into the optimization process. Additionally, we propose a novel cost-function, which is inspired by several high-level approaches from the literature.

ii. We provide experimental results on trajectories logged with ground truth for indoor environments. The beacon map generated using our proposed technique is com-

[1]Raphael Falque is with Centre for Autonomous System, Faculty of Engineering and IT, University of Technology Sydney, Ultimo, NSW 2007, Australia Raphael.H.Guenot-Falque@student.uts.edu.au

[2]Mitesh Patel, and Jacob Biehl are with FX Palo Alto Laboratory Inc., Palo Alto, CA - 94304, USA {mitesh,biehl}@fxpal.com

pared to a deployment designed by human operators where beacons are deployed in all the power plugs available. The results generated using the beacon map is evaluated with different localization techniques such as Linear Least Square (LLS) and Particle Filter (PF). Through our experiments, we demonstrate that the number of required beacons were reduced by 53% while reducing the localization accuracy by only 0.6 m compared to the results obtained when using the complete distribution of beacons.

II. Related Work

The optimization of landmarks/beacons placement for indoor localization on a predefined path is well studied in the literature. Using the prior of a predefined path, near-optimal solutions can be obtained with greedy approaches. These methods incrementally add beacons at the locations that minimize the localization uncertainty [8] or some other confidence bounds, such as a guaranty of minimum deviation from the desired path [9]. As an alternative to a predefined path, it is also possible to consider a set of trajectories similarly to the approach proposed by [10] where the visual landmarks are iteratively added while considering a set of independent trajectories. Despite providing bounds related to the localization performances, these methods require a path known in advance to formulate the optimization problem. Hence, they are not practical for human localization or unplanned exploration of a floorplan.

In cases where the path followed by the robot/user is unknown, one can use the methods developed by the WSN research community and to formulate the problem as a coverage problem (in this case the sensors are replaced by the set of beacon emitters). Solutions to the coverage problem have been surveyed by Wang *et al.* [11] and a standard approach is to use regular lattices in order to design the deployment of the WSN. The evaluation of different lattices' pattern (e.g., square grid, hexagonal lattice) has been studied by Chen *et al.* using trilateration and they concluded that a deployment based on a triangular lattice is the best pattern for localization tasks.

Similarly, it is possible to formulate the coverage problem as an *art gallery problem* which is known to be *NP-hard*. In such case, meta-heuristics strategies such as Genetic Algorithm (GA) offer good candidates to find near-optimal solution [12]. In this context, Yoon *et al.* developed an efficient variant of GA which has better performances than standards GA approaches for the specific case of the coverage problem [13]. Evaluation of their technique was done with Monte Carlo simulations. Similarly, Seo *et al.* used GA to optimize the sensor placement in a battlefield environment with a GPU implementation to achieve maximum coverage [14]. The authors divided the terrain into a grid and calculated the probability of vehicle detection using generational GA when it passes through to each cell. Similarly, Carter and Ragade use a probabilistic model that accounts for the detection probabilities of the sensing devices which may decay with distance, environmental conditions and hardware configurations [15].

While the aforementioned methods provide potential solutions, the coverage problem is not designed to optimize the placement of landmarks for localization purpose. Therefore, other metrics such as the Geometric Dilution Of Precision (GDOP), which was originally defined in order to quantify the quality of the satellite positioning for localization purpose, can provide better information for optimizing the beacon deployment. Roa *et al.* have shown that a beacon deployment based on the optimization of the GDOP can outperform regular lattices [16]. The optimization of the GDOP on a floorplan has been done for different modalities, such as angular emitters used for tiangulation [17] and more recently, Rajagopal *et al.* introduced the concept of Uniquely Localizable (UL) which can be combined to GDOP for the deployment of sound beacons [18]. The deployment has been tested on simulated datasets; however, this concept seems incompatible with the through wall propagation of RF beacons.

An alternative approach has been proposed by Zou *et al.* in the Virtual Force Algorithm (VFA). This method iteratively updates the position of beacons by moving them away or towards each other according to the inter-sensor distances [19]. Once optimized, the algorithm would ideally reach a configuration where all beacon are spaced by a predefined distance. They evaluate the performance of the VFA by using the beacon deployment for localization purpose. Their localization approach is based on matching the sensor readings at any given point of the environment with a predefined probabilistic table.

Most of the literature does not account properly for the environment in the optimization of the beacon deployment. The methods that maximize the inter-beacon distances are inherently independent of the environment. The other methods optimizing the coverage are often based on simple sensor models that do not account for the environment. Our proposed approach differs from the prior methods in the following ways:

i. Our proposed system optimizes for both the number of beacons required in a given environment and optimizing the location of those beacons.
ii. We propose a novel cost-function that combines the k-coverage methods and the beacon spreading.
iii. Our proposed system is generalized whereby it accounts for beacon placement constraints due to environmental conditions as well as noisy signal characteristics due to multi-path issues.

III. Problem Definition

In this paper, we consider the deployment of BLE beacons for indoor localization. We consider both the problem of finding the number, n, of beacons required, and the search of their optimal placements in order to maximize localization accuracy. We define $\mathscr{B}$ to be the space where the beacons can be deployed and B the set of the chosen beacons' position,

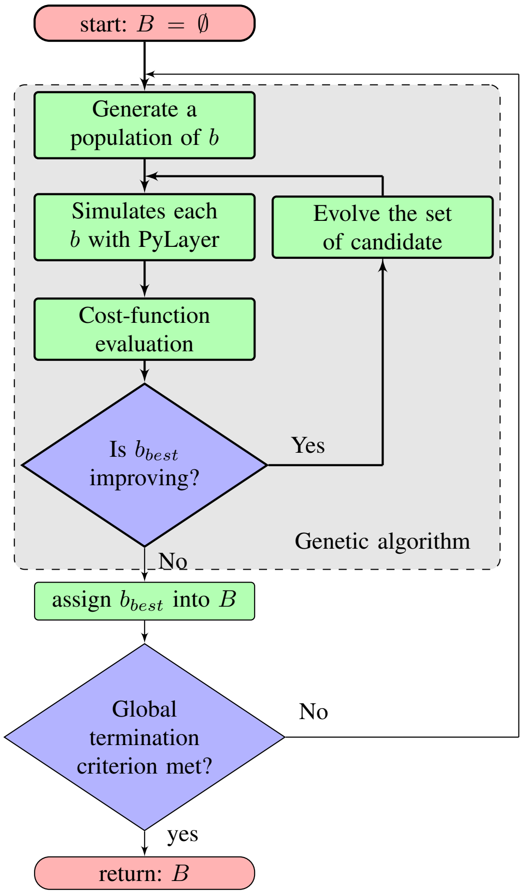


Fig. 2: Initial greedy deployment

which is more formally defined as $B = \{\boldsymbol{b}_1, ..., \boldsymbol{b}_n\}$ with $\boldsymbol{b}_i \in \mathscr{B}$ the i^{th} beacon position.

IV. Methodology

During the deployment stage, the number of BLE beacons required to cover an area is often unknown. Estimating this quantity is often a hard task as the required number of beacon depends on the environment. Therefore, we first use a greedy algorithm to estimate the number of beacons required. Once this quantity is known, we then consider a global optimization formulation to obtain a more optimal solution.

A. Algorithm architecture

We first explain the architecture of the main algorithm before providing more details for each part of the algorithm in the following sub-sections.

The proposed greedy algorithm initializes the set of deployed beacon B as an empty set. A GA is then used to find the beacon position that would maximize our cost-function f_{CovBSM}. In other words, the added beacon b_{best} is defined as $\underset{b}{\text{argmax}}(f_{\text{CovBSM}}(B \cup b))$. The process is repeated iteratively until a termination criterion is met. The flowchart of the greedy algorithm is shown in Figure 2.

As the previous greedy approach can not be optimal [20], we perform a re-deployment step using the number of beacons obtained from the greedy approach. As shown

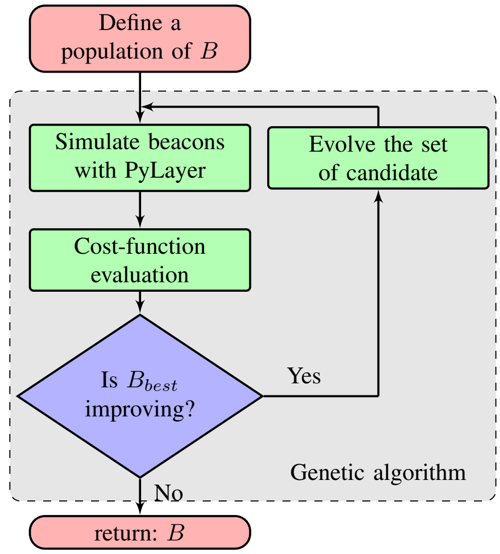


Fig. 3: Global re-deployment

in Figure 3, the approach is quite similar to the greedy approach. However, conversely to the greedy approach where the optimized parameter is a single beacon location b, we now consider the optimization of a full set of beacons locations, i.e., the optimized parameter is now $B = \{b_1, ..., b_n\}$ with n the number of beacons obtained from the greedy approach.

In the following sub-section we explain how the sensor model is defined. We then provide the definition of the cost-function which is built on top of the sensor model. Finally we explain how the GA algorithm optimizes the sensor placement using the cost-function.

B. Sensor modeling

As discussed in [21], different sensor models can be used to model the propagation of BLE beacon over a 2D plan. The models commonly used in the literature are the disk model, the Friis model[1], and the direct LOS model. The behavior of these models for a simple scenario is shown in Figure 4.

While some of these models are a good approximation of the sensor behavior in an open environment — especially the disk model — they are a bad approximation of the reality in

[1]The Friis model is also referred to as the probabilistic model and exponential model.

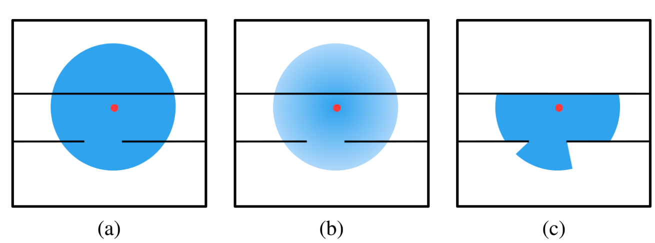
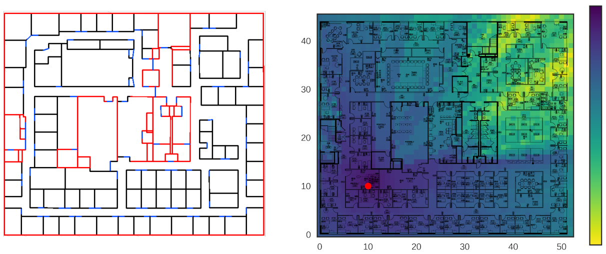


Fig. 4: Sensor model commonly used in the literature: in (a) the disk model, in (b) the Friis model [22], and in (c) the direct line-of-sight model.



(a) (b)

Fig. 5: In (a), the floor plan with the doors in blue, the partition in black and the concrete walls in red. In (b), a sample of the simulation result of a beacon located at the position $(10, 10)$ using the PyLayer simulator.

an office and building environment where walls are present, and the propagation and attenuation through the walls should be accounted for.

To accommodate for the impact of the environmental factors on the RF signal, we use the PyLayer simulator [23] to simulate the RSSI for a given floor plan. PyLayer is an open source Python library which allows simulating the propagation of radio signals in complicated environments. As an input, the simulator requires the information of the environment. In our scenario, we consider three categories of infrastructure: (i) the doors, which are modeled as air, (ii) the soft partitions, and (iii) the concrete walls. An example of the floorplan information feed into the simulator is shown in Figure 5a.

Additionally to the infrastructure, the properties of the Bluetooth emitters are also an input of the simulator (i.e., emission power and beacon location within the map). Using this environment and the beacon properties, the simulator uses ray-casting to compute the attenuation of the signal at every point of the map. As a result, the simulator provides us with a path loss that accounts for the environment such as the one shown in Figure 5b.

C. Cost function

While considering the minimization of the localization error might seems to be the natural approach, including the localization error into the cost-function leads to a lack of generalization as the sensor deployment would be designed for a specific localization algorithm. Furthermore, using a localization approach built around a Bayesian fusion of the prior belief and the sensor update would require to define a specific path which would also decrease the generalization of our approach.

For these reasons, we propose a hybrid approach between the VFA [19] and a k-coverage optimization. As discussed in [24], the coverage problem is a proxy to minimizing the localization error; therefore, an optimal sensor network design can not be obtained by considering the k-coverage problem by itself. However, a 3-covered area would inherently provide with a good localization and also offer a better generalization as it is independent of the localization algorithm used.

1) k-covered and beacon spreading maximization: By using the definition of k-covered given in [25], the cost-function is defined as the percentage of the combined 3-covered part of the map and a measure of the spreading of the beacons on the map. More formally, considering a grid-map of the floor plan of size $X \times Y$ and the coverage function c, the cost function is defined as follows

$$f_{\text{CovBSM}} = \underbrace{\frac{1}{3XY}\sum_{x=1}^{X}\sum_{y=1}^{Y}\min\big(c(x,y),3\big)}_{\text{maximizes 3-coverage}} + \underbrace{\lambda\sum_{i=1}^{n}\min_{\forall b_j\in\{B/b_i\}}||b_i-b_j||}_{\text{maximizes beacons spreading}}, \quad (1)$$

where $b_i \in B$ is the set of beacons distributed on the map, and $c(x, y)$ is the k-coverage at the location (x, y). Given the definition of the cost-function, we formulate here the problem as a maximization problem.

The coverage c is obtained by thresholding the output from the PyLayer simulator for each beacon. The threshold is obtained by analyzing the surrounding electromagnetic noise, in our environment -100 dB is a reasonable value. More formally c is defined as follows

$$c(x,y) = \sum_{i}^{n}\begin{cases}1 \text{ if } \mathrm{RSSI}_i(x,y) > -100 \text{ dB}\\ 0 \text{ otherwise}\end{cases} \quad (2)$$

The parameter λ is determined by finding the upper bound of the spreading parameter. It can be obtained by solving the *circle packing problem* within the area of the given floor plan. Indeed, for a given surface, the circle packing problem is defined as finding the arrangement of circles which maximize the areas of the surface covered by the circles. It can be shown that the optimal arrangement is obtained with a *hexagonal lattice* [26] which provides an optimal packing density η equal to

$$\eta = \frac{\pi}{2\sqrt{3}}. \quad (3)$$

Considering this optimal arrangement, for a rectangular surface such as the one shown in Figure 6, we aim to find the maximal circle radius r which can be used to pack n circles within the surface[2]. Note that we allow having the center of the circles to be placed on the edges of the surface resulting in a surface area S defined as

$$S = (L + 2r)(W + 2r). \quad (4)$$

With the surface area of a single circle S_c defined as

$$S_c = \pi r^2, \quad (5)$$

[2]In the circle packing problem, n is the number of packed circles. In our case, it corresponds to the number of deployed beacons.

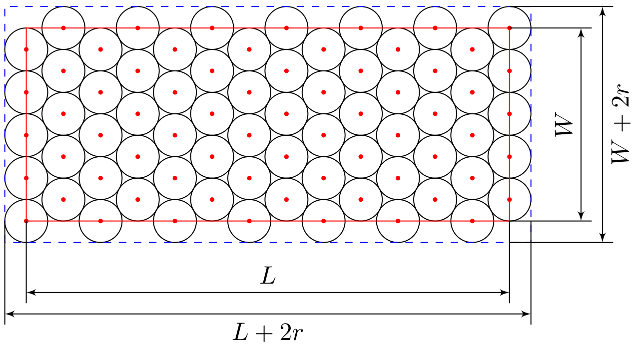


Fig. 6: Example of a hexagonal lattice which maximizes the inter-distance between beacons. This distribution is obtained by solving the circle packing problem.

the optimal packing results in

$$
\begin{aligned}
& S = nS_c\eta & (6)\\
\equiv\ & (L+2r)(W+2r) = n\pi r^2 \frac{\pi}{2\sqrt{3}} & (7)\\
\equiv\ & r^2\left(4 - n\frac{\pi^2}{2\sqrt{3}}\right) + r(2L+2W) + LW = 0 & (8)
\end{aligned}
$$

The upper bound of the optimal radius can be obtained by solving this quadratic equation which results in

$$
r = \frac{-(2W+2L) - \sqrt{(2W+2L)^2 - 4(4 - n\frac{\pi^2}{2\sqrt{3}})LW}}{2(4 - n\frac{\pi^2}{2\sqrt{3}})} \tag{9}
$$

The normalization parameter λ is then defined as the inverse of the inter-beacon distance, i.e. $2r$, for an optimal distribution such as

$$
\lambda = \frac{1}{n2r}. \tag{10}
$$

The evolution of λ for a different number of beacons is shown in Figure 7; considering the evolution of λ, this normalization parameter is important to track the performance of the cost-function.

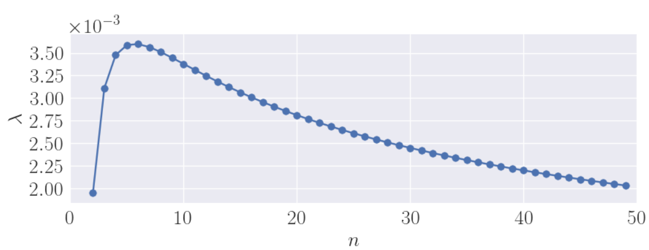


Fig. 7: The relationship between the normalization parameter λ and the number of deployed beacons n.

2) Localization error: In order to compare our approach with a more traditional method we also introduce here another cost-function which directly minimizes the localization error estimated from the simulated RSSI sensor readings. We use here the simple hyperbolic algorithm [27] (also referred to Linear Least Squares localization interchangeably in our paper).

Let us consider the minimization of the localization error e defined as follows

$$
e = \sum_{i=1}^{K} \left(\sqrt{(\hat{x} - x_i)^2 + (\hat{y} - y_i)^2} - d_i \right) \tag{11}
$$

where $(\hat{x}, \hat{y})$ is the estimated position of the robot, (x_i, y_i) is the position of the i^{th} beacon, K the number of received RSSI readings, and d_i is the distance estimated with the sensor model. In order to infer d, we use the Friis free space model [22] which is defined as follows,

$$
P_r = P_t \frac{G_t G_r \lambda_{tr}^2}{(4\pi d)^2 L_{tr}}, \tag{12}
$$

where P_r and P_t are the respective received and transmitted signal power, G_r and G_t are the respective gain of the receiver and transmitter, λ_{tr} is the wavelength of the signal, L_{tr} is the system losses, and d is the distance between the transmitter and receiver. Extracting d from Equation (12) results in the following sensor model:

$$
d = \sqrt{P_t \frac{G_t G_r \lambda_{tr}^2}{(4\pi)^2 L_{tr} P_r}}. \tag{13}
$$

By linearizing the set of K equation defined in Equation (11) we can obtain the following formulation:

$$
d_i^2 = (\hat{x} - x_i)^2 + (\hat{y} - y_i)^2. \tag{14}
$$

Similarly to [27], we use the position of the first beacon as a basis — i.e. $(x_1, y_1) = (0, 0)$ — and subtract the equation of the first reading to the others equations resulting into:

$$
d_i^2 - d_1^2 = x_i^2 - 2\hat{x}x_i + y_i^2 - 2\hat{y}y_i \tag{15}
$$

The set of $K - 1$ equation can now be formulated in a vector form such as

$$
\underbrace{\begin{bmatrix} 2x_2 & 2y_2 \\ \vdots & \vdots \\ 2x_K & 2y_K \end{bmatrix}}_{\boldsymbol{H}} \underbrace{\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}}_{\hat{\boldsymbol{x}}} = \underbrace{\begin{bmatrix} x_2^2 + y_2^2 - d_2^2 + d_1^2 \\ \vdots \\ x_K^2 + y_K^2 - d_K^2 + d_1^2 \end{bmatrix}}_{\boldsymbol{b}}, \tag{16}
$$

and solved with linear least squares:

$$
\hat{\boldsymbol{x}} = (\boldsymbol{H}^T \boldsymbol{H})^{-1} \boldsymbol{H}^T \boldsymbol{b}. \tag{17}
$$

This localization estimation is computed for a set of J evenly distributed locations and the error is summed to obtain the final cost-function:

$$
f_{\text{LLS}} = \frac{1}{J} \sum_{j=1}^{J} \sqrt{(\hat{x_j} - x_j)^2 + (\hat{y_j} - y_j)^2} \tag{18}
$$

Remark 1: Equation (16) shows that at least three equation (two for solving the problem and one used as a basis), are required for solving the localization inference. In other words three beacons are required to solve the localization problem, which shows the importance of maximizing the 3-coverage in Section IV-C.1.

D. Genetic algorithm

Considering the sensor model used herein (i.e. the PyLayer simulator), having a search approach which is agnostic from the cost-function helps avoiding gradient computation. Indeed, as the sensor is modeled using a simulator, the computation of an analytical gradient is not possible. For this reason and the arguments related to metaheuristics provided in the related work section, we use a GA for optimizing our solutions. We used the DEAP library for the implementation of the GA optimization [28].

GA is an optimization process inspired by Darwin's theory of evolution. In the GA's terminology, the parameters are referred as *genomes*, the full set of parameters that form a solution of the optimization problem is called *individual*, and the set of individuals is called *population*. The GA optimization starts by an initialization of the population. An *evaluation* of the population is then performed with the cost-function mentioned in the previous section, and given the results of the evaluation, a *selection* process is executed in order to keep the good candidates. An evolution strategy is then performed by using *crossover* and *mutation*. The process is then iteratively repeated from the evaluation stage.

The crossover consists of switching parameters between different individuals. The crossover is characterized by a hyperparameter called the probability of crossover. The mutation consists of adding noise to the parameters values and is characterized by the probability of mutation. In practice, the crossover is used in order to jump to another place in the space of the parameters and is similar to the jumping in *basin hoping*. The mutation is used to explore the neighborhood of the good candidates.

Once the GA is setup, we embed it into the systems described in Figure 2 and 3 to find the required quantity of beacons and where to deploy the beacons. Specific examples of the GA parametrization is given in the following section.

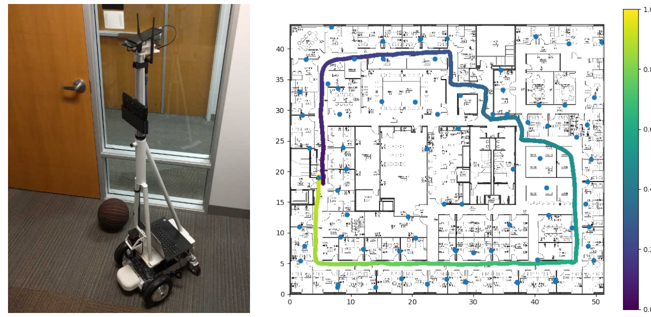
V. EXPERIMENTAL RESULTS

In this section, we introduce the results of both the greedy and global optimization. The datasets collected in our experiment are available upon request.

A. Experimental setup

The evaluation of the proposed approach is done by considering the performance of two different localizations algorithms given the optimized beacons deployment in the real world. The environment used for evaluation is an indoor office-like space of size 40×50 m^2 shown in Figure 8b. We use a robotic platform in order to collect both the ground-truth and the Bluetooth RSSI measurement.

The robotic platform — shown in Figure 8a — is equipped with a 2D LIDAR and a Nexus 6P smartphone (for the collection of the BLE data). The ground truth consists of the robot poses which are obtained through the mapping of the environment. More specifically, we use *hector_mapping* [29] to map the environment and have an estimate of the robot poses. The pose estimates from hector_mapping are then merged with the robot odometry in order to improve the robot poses. The robot was remote controlled, and the data were collected at a speed of 0.65 m$\cdot$s^{-1}.



(a) (b)

Fig. 8: Robot used for the data collection (a), and ground truth generated while moving in the building (b).

As a result, the data collection consists of the bluetooth RSSI measurements collected at 7 Hz and the ground truth defined by the robot poses, for a path of 150 m which is shown in Figure 8b. Both data were saved in a ROS bag along with the timestamps which allow matching the data together.

B. Greedy approach

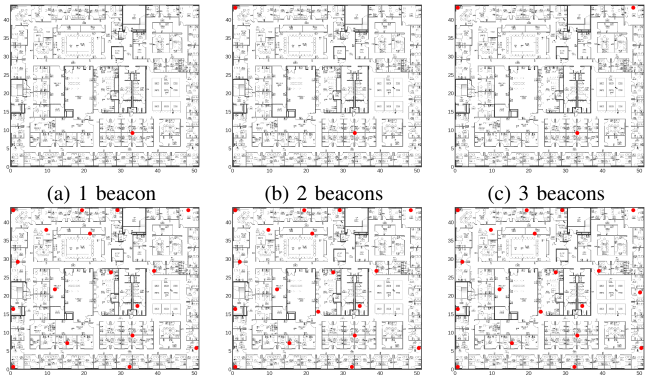
GA is a stochastic process; therefore, there is no guarantee that the search would reach consistently the same solution for different randomizer initialization. Therefore, we have to decide how many time the experiment has to be repeated in order to claim the result as valid. Each experiment has been repeated 50 times with different random seeds for evaluation with quantitative results.

Prior to providing the results, we first describe the setup of the GA's hyperparameters. The optimized parameters are defined as the tuple (x, y); therefore, the space of possible beacon positions $\mathscr{B}$ is defined as a two-dimensional interval which is a subset of $\mathbb{R}^2$ and bounded by the building dimensions. In other words, $\mathscr{B} = [0, 51] \times [0, 41]$ which correspond to the dimensions of the building in meters. As the optimization consists of finding the best position of a single beacon on a 2D map, the crossover does not have a meaning; therefore, the probability of crossover is set to 0.0 and the probability of mutation to 0.3. The population is set to 50 and the number of generations to 10. We terminate the greedy process once 90% of the map being covered by at least 3 beacons, i.e. 3-coverage equal to 0.9.

A sample of the Greedy optimization is shown in Figure 9, with the beacon placement in Figure (a) to (c) largely influenced by the spreading term of the cost function. The optimization of the coverage is more apparent in Figure (d) to (f). The outcome of the greedy optimization results in 37 beacons being required to cover the full environment. This number is then fed into the global optimization process.

C. Global optimization

In the global optimization, the parameters optimized by the GA are the full set of deployed beacons. While the greedy



(a) 1 beacon (b) 2 beacons (c) 3 beacons

(d) 17 beacons (e) 18 beacons (f) 19 beacons

Fig. 9: Greedy addition of beacons. The position of the added beacon is found by maximizing Equation (1). In Figure (a) to (c), the position of the added beacon is clearly chosen in order to maximize the beacon spreading. In Figure (d) to (f), the position of the added beacon is chosen in order to maximize the coverage.

part of the problem is solved in a simulated environment, the global optimization has to be deployed in the real world. In our case, we use beacons that require a power supply, hence we limit the space of the possible beacon positions to 79 different locations in the building where an electric power plug is available, i.e. $\mathscr{B} = \{l_1, ..., l_{79}\}$ with l_i the i^{th} possible location. Given the definition of $\mathscr{B}$, the GA is formulated as selecting for the best 37 beacon positions amongst the set of possible locations. Therefore, using the mutation in the GA is irrelevant and we set the probability of mutation to 0.0 and the probability of crossover to 0.3. As the search is more complex and there is no use of mutation, the population has to be increased in order to achieve a good solution. We set the population to 200 and the number of evolution iterations to 20.

Remark 2: Restricting the possible beacon positions into a finite set of places is not a theoretical limitation of the proposed framework. For example, formulating the GA into searching through a space of $[0, 51] \times [0, 41] \times 37$, i.e. each beacon location defined on a two-dimensional interval. In such situation, both the mutation and the crossover should be used.

In the global optimization, we evaluate the performance of the localization using two different localization methods. The first method consists of the LLS localization, described in Section IV-C.2; the second method is based on a PF [7]. Sample of the results are shown in Figure 10 and summarized for multiple initializations in Figure 11, under the form of a Cumulative Distribution Function (CDF) for both the different localization methods.

In our approach, by using the greedy optimization and the cost-function f_{CovBSM}, we use 47% of the beacons deployed by the human operator but still obtain similar performances. The optimization of f_{CovBSM} outperform the optimization of the hyperbolic localization. More formally, using the same evaluation method as the EvALL competition [30], in 75% of the times, the mean error for the multiple runs using the PF[3] is less than 2.34 ($\pm$ 0.03) m for all the beacons, 3.02 ($\pm$ 0.28) m using the f_{CovBSM} cost-function, and 3.56 ($\pm$ 0.62) m using the f_{LLS} cost-function.

(a) (b)

Fig. 10: In (a), a sample of an optimized deployment obtained with the f_{CovBSM} cost-function and the associated localization using the particle filter. In (b), a similar figure showing the performance of the f_{LLS} cost-function. The set $\mathscr{B}$ is shown with dots — blue and red — and the selected ones are displayed in red.

VI. DISCUSSION & FUTURE WORK

In this paper, we propose an approach that solves for both the problem of automatically finding the required number of BLE beacons and how to deploy them to get a near-optimal beacon deployment specifically purposed for indoor localization. Our approach accounts for the RF signal attenuation due to the environment and is independent of the localization algorithm used. The experiments show that our cost function outperform the minimization of the trilateration error. Furthermore, we our method obtains relatively similar performances than the exhaustive use of all the beacons while

[3]Given the non-deterministic nature of the PF, we run each experiment 50 times and report the difference of error within 2 σ bounds

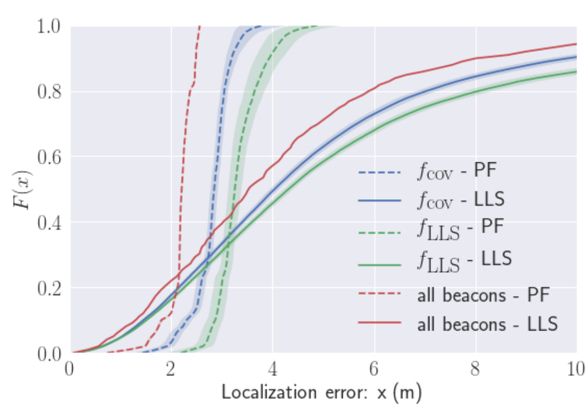


Fig. 11: Cumulative Distribution Function of the localization error, with in green the deployment optimized using the cost-function f_{CovBSM}, in blue the minimization of the cost-function f_{LLS}, and in red the 79 beacons deployed in every available positions. The plots are showing the mean values of the CDF and the 95% confidence bounds.

using significantly less beacons (i.e. the total number of beacons needed were reduced by 53%).

Solving the deployment of RF beacons by either optimizing the coverage problem or maximizing the spreading of the beacons in the map, e.g. triangular lattice and VFA, is common in the literature. Our proposed cost-function combines these two approaches by using both the 3-coverage and the maximization of the inter-beacon distances. Our formulation of the beacon spreading in the cost-function is more flexible than forcing triangular lattice and do not require a hyperparameter in the cost function compared to the VFA. Furthermore, considering the 3-coverage problem over the 1-coverage problem allows solving the localization for the most basic localization algorithms such as trilateration or hyperbolic localization.

For completeness, we also compare the performance of our proposed cost-function that combines the maximization of both the 3-coverage and the inter-beacon distances with direct minimization of the hyperbolic localization error as the cost-function. Our proposed cost-function provided slightly better results which can be explained by the fact that the minimization of the hyperbolic localization error is performed in the simulated environment, which is an inherently different from the real world application. Furthermore, the minimization of the hyperbolic localization error is performed on a uniform grid and not specifically on the path followed by the robot.

As part of the future work, we would like to explore other metaheuristic optimization methods for the greedy approach such as BI-POP cma-es which has proven to outperform other black box optimization techniques [31]. Furthermore, we want to perform a fair comparison of our approach to the optimization of the GDOP.

References

[1] S. Ito, F. Endres, M. Kuderer, G. Diego Tipaldi, C. Stachniss, and W. Burgard, "W-rgb-d: Floor-plan-based indoor global localization using a depth camera and wifi," *IEEE International Conference on Robotics and Automation*, pp. 417–422, 2014.

[2] R. Zhang, A. Bannoura, F. Hflinger, L. M. Reindl, and C. Schindelhauer, "Indoor localization using a smart phone," *IEEE Sensors Applications Symposium*, pp. 38–42, 2013.

[3] S. Chan and G. Sohn, "Indoor localization using wi-fi based fingerprinting and trilateration techiques for lbs applications," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 3826, pp. 1–5, 2012.

[4] P. K. Yoon, S. Zihajehzadeh, B. S. Kang, and E. J. Park, "Adaptive kalman filter for indoor localization using bluetooth low energy and inertial measurement unit," *IEEE Engineering in Medicine and Biology Society*, pp. 825–828, 2015.

[5] R. Faragher and R. Harle, "Location Fingerprinting With Bluetooth Low Energy Beacons," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 11, pp. 2418–2428, 2015.

[6] G. Yanying, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 13–32, 2009.

[7] M. G. Jadidi, M. Patel, and J. Valls Miró, "Gaussian processes online observation classification for rssi-based low-cost indoor positioning systems," *International Conference on Robotics and Automation*, pp. 6269–6275, 2017.

[8] M. P. Vitus and C. J. Tomlin, "Sensor placement for improved robotic navigation," *Robotics: science and Systems VI*, p. 217, 2011.

[9] M. Beinhofer, M. Mller, and W. Burgard, "Effective landmark placement for accurate and reliable mobile robot navigation," *Robotics and Autonomous Systems*, vol. 61, no. 10, pp. 1060–1069, 2013.

[10] V. Magnago, L. Palopoli, R. Passerone, D. Fontanelli, and D. Macii, "A nearly optimal landmark deployment for indoor localisation with limited sensing," in *Indoor Positioning and Indoor Navigation (IPIN), 2017 International Conference on*. IEEE, 2017, pp. 1–8.

[11] Y. Wang, S. Wu, Z. Chen, X. Gao, and G. Chen, "Coverage problem with uncertain properties in wireless sensor networks: A survey," *Computer Networks*, vol. 123, pp. 200–232, 2017.

[12] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: overview and conceptual comparison," *ACM Computing Surveys*, vol. 35, no. 3, pp. 189–213, 2003.

[13] Y. Yoon and Y. H. Kim, "An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks," *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1473–1483, 2013.

[14] J.-h. Seo, Y. Yoon, and Y.-h. Kim, "An Efficient Large-Scale Sensor Deployment Using a Parallel Genetic Algorithm Based on CUDA," *International Journal of Distributed Sensor Networks*, vol. 12, no. 3, p. 8612128, 2016.

[15] B. Carter and R. Ragade, "A probabilistic model for the deployment of sensors," *IEEE Sensors Applications Symposium*, pp. 7–12, 2009.

[16] J. O. Roa, A. R. Jiménez, F. Seco, J. C. Prieto, and J. Ealo, "Optimal placement of sensors for trilateration: Regular lattices vs meta-heuristic solutions," in *International Conference on Computer Aided Systems Theory*. Springer, 2007, pp. 780–787.

[17] P. Nazemzadeh, D. Fontanelli, and D. Macii, "Optimal placement of landmarks for indoor localization using sensors with a limited range," in *Indoor Positioning and Indoor Navigation (IPIN), 2016 International Conference on*. IEEE, 2016, pp. 1–8.

[18] N. Rajagopal, S. Chayapathy, B. Sinopoli, and A. Rowe, "Beacon placement for range-based indoor localization," in *Indoor Positioning and Indoor Navigation (IPIN), 2016 International Conference on*. IEEE, 2016, pp. 1–8.

[19] Y. Zou and Krishnendu Chakrabarty, "Sensor deployment and target localization based on virtual forces," *IEEE Computer and Communications Societies*, vol. 2, no. 1, pp. 1293–1303, 2004.

[20] G. Gutin, A. Yeo, and A. Zverovich, "Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP," *Discrete Applied Mathematics*, vol. 117, no. 1-3, pp. 81–86, 2002.

[21] M. Hefeeda and H. Ahmadi, "A Probabilistic Coverage Protocol for Wireless Sensor Networks," *IEEE International Conference on Network Protocols*, pp. 41–50, 2007.

[22] H. Friis, "A Note on a Simple Transmission Formula," *Proceedings of the I.R.E. and Waves and Electrons*, vol. 34, no. 5, pp. 254–256, 1946.

[23] N. Amiot, M. Laaraiedh, and B. Uguen, "PyLayers: An open source dynamic simulator for indoor propagation and localization," *IEEE International Conference on Communications Workshops*, pp. 84–88, 2013.

[24] Y. Chen, J.-a. Francisco, W. Trappe, and R. Martin, "A Practical Approach to Landmark Deployment for Indoor Localization," *IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, vol. 11, pp. 365–373, 2006.

[25] C.-F. Huang, Y.-C. Tseng, and L.-C. Lo, "The coverage problem in a wireless sensor network," *Mobile Networks and Applications*, vol. 10, no. 04, pp. 519–528, 2005.

[26] H.-C. Chang and L.-C. Wang, "A Simple Proof of Thue's Theorem on Circle Packing," 2010. [Online]. Available: http://arxiv.org/abs/1009.4322

[27] P. Tarrío, A. M. Bernardos, and J. R. Casar, "Weighted Least Squares Techniques for Improved Received Signal Strength Based Localization," *Sensors*, vol. 11, no. 12, pp. 8569–8592, 2011.

[28] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, 2012.

[29] S. Kohlbrecher, J. Meyer, O. von Stryk, and U. Klingauf, "A flexible and scalable slam system with full 3d motion estimation," *Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, 2011.

[30] "Evaluating AAL systems through competitive benchmarking," http://evaal.aaloa.org/2016/competition-results, accessed: 2018-Feb-24.

[31] "Coco / black-box optimization benchmarking (bbob)," http://coco.gforge.inria.fr/, accessed: 09-10-201.