

## Next-Best-Sense: a multi-criteria robotic exploration strategy for RFID tags discovery

Riccardo Polvara\*, Manuel Fernandez-Carmona\*, Gerhard Neumann<sup>\*+x</sup> and Marc Hanheide\*

**Abstract**—Automated exploration is one of the most relevant applications for autonomous robots. In this paper, we propose a novel online coverage algorithm called *Next-Best-Sense (NBS)*, an extension of the Next-Best-View class of exploration algorithms which optimizes the exploration task balancing multiple criteria. NBS is applied to the problem of localizing all Radio Frequency Identification (RFID) tags with a mobile robot. We cast this problem as a coverage planning problem by defining a basic sensing operation – a scan with the RFID reader – as the field of “view” of the sensor. NBS evaluates candidate locations with a global utility function which combines utility values for travel distance, information gain, sensing time, battery status and RFID information gain, generalizing the use of Multi-Criteria Decision Making. We developed an RFID reader and tag model in the Gazebo simulator for validation. Experiments performed both in simulation and with a robot suggest that our NBS approach can successfully localize all the RFID tags while minimizing navigation metrics, such sensing operations, total traveling distance and battery consumption. The code developed is publicly available on the authors’ repository <sup>1</sup>.

**Index Terms**—Reactive and Sensor-based Planning, Surveillance Systems, Service Robots, RFID, Next-Best-View

### I. INTRODUCTION

Mobile robot patrolling is attracting growing attention, since it allows to perform tasks (e.g., exploration, surveillance, search and rescue) efficiently and continuously in potentially hazardous conditions [1]. Our *Next-Best-Sense (NBS)* approach adapts an online Next-Best-View (NBV) method that uses Multi-Criteria Decision Making (MCDM) [2] to choose the next location a robot should reach. This exploration strategy is applied in this work to Radio Frequency IDentification (RFID) tag discovery. In an RFID system there is at least one reader and typically multiple tags. The tags are simple radio devices, capable of transmitting a short signal after a reader request. They do not require a battery, as they can be powered by the reader’s radio signal itself and they are extremely cheap. Existing work was focused on finding tags in a given environment based on *a-priori* knowledge about their position [3], [4]. Our contribution does not require any knowledge about the inventory distribution, casting the inventory management problem into an environment coverage problem with RFID reader measurements. Using *Next-Best-Sense* to schedule the

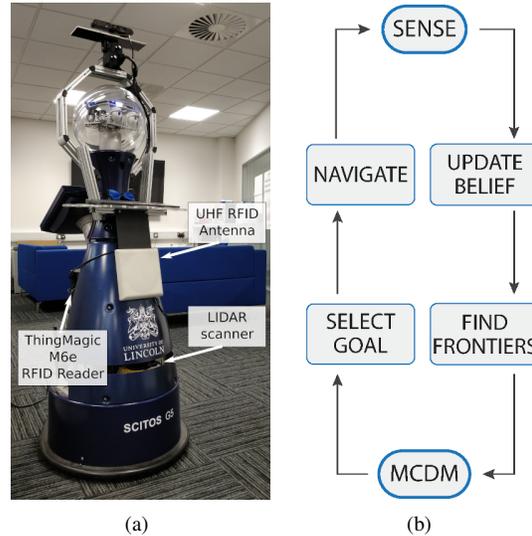


Fig. 1: The Linda robot (a). The Next-Best-Sense algorithm (b).

exploration, we can minimize the time required to cover an environment while finding all RFID tags in a target area. At each step, after performing a local sensing operation, new candidate locations are identified along the boundary between the scanned and the unscanned space. While navigating a robot, either in a tele-operated or autonomous way, the user always try to achieve multiple objectives at the same time. Few examples are given by traversing the shortest path while maximizing the area covered by the sensors. At the same time, it would also be important to reduce the impact over the battery in order to guarantee longer deployment time for the robot. These objectives are usually considered inside a utility function used to select the next destination of the robot. Differently from approaches using *ad-hoc* aggregation function [5], [6], MCDM offers a more principled theoretical foundation to the decision making process. Next-Best-Sense is an online greedy algorithm, it always selects the best candidate location according to the given criteria without considering long-term planning. In this way, it favorably scales to large environments and can also be used in unknown environments. The entire NBS loop is given by Fig. 1(b). The experiments performed both in simulation and real world (using the Linda robot shown in Figure 1(a)) proved that NBS achieves a fast map coverage and high tag localization precision (<1m). Similarly to [2], our NBS represents a theoretical contribution for many search applications. For example, it offers a valid option for planning a safe path for a robot deployed in a nuclear facility

\*Authors are with Lincoln Center for Autonomous Systems (L-CAS), School of Computer Science, University of Lincoln, United Kingdom. {rpolvvara, mfernandezcarmona, gneumann, mhanheide}@lincoln.ac.uk

<sup>+</sup>Bosch Center for Artificial Intelligence, Renningen, Germany.

<sup>x</sup>Karlsruhe Institute of Technology, Karlsruhe, Germany.

<sup>1</sup><https://github.com/LCAS/nbs>

with the goal to identify all the possible radiation sources [7].

The remainder of the article is organized as follows: in Sec. II we survey the related literature while in Sec. III we define the problem statement. In Sec. IV and in Sec. V we introduce the sensor model and our algorithm, respectively. In Sec. VI we report the results of experiments, performed both in simulation and with a physical platform. Finally, in Sec. VII we summarize our approach, highlighting the main concepts, its advantages and limitations, and future works.

## II. RELATED WORK

Our work focuses on a novel exploration strategy where multiple criteria can be optimized at the same time. This strategy is particularly relevant to RFID-tagged object detection, as it usually involves covering large areas and requires an efficient planning. This section briefly surveys related works in both areas.

### A. Exploration Strategies

The majority of exploration strategies for initially unknown environments take decisions in a greedy fashion and are often called Next-Best-View (NBV) algorithms. Usually, in NBV systems, the candidate next locations for the robot are chosen to be on the frontier between the already explored free space and the unknown area, and they need to be reachable from the current position of the robot. The identified candidate positions are then evaluated according to an utility function. In [8], they use the *traveling cost*, according to which the next best observation location is the nearest one. In a similar way, in [5], [6] the traveling cost is combined (linearly and exponentially, respectively) with the *information gain*, defined as the expected amount of new information the robot can acquire from the candidate location. In [9], traveling cost and information gain are instead combined with semantic information. In [10], the authors introduce a technique based on relative entropy, while [11] uses several criteria and a multiplicative function to obtain a global utility value. More recently, in order to overcome the limitation of local planning, frontier-based exploration has also been combined with Receding Horizon NBV (based on sampling methods) to prevent the robot gets stuck [12]. The aforementioned strategies define *ad hoc* aggregation methods that combine the values assumed by the criteria considered. To solve this issue, [2] proposed the adoption of Multi-Criteria Decision Making, which is more theoretically founded and of inspiration for our work. More recently, some data-driven solutions have also been proposed following the growing interest generated by deep learning in the robotic community. In [13], the robot is equipped with a long-term memory which allows to learn the global map using only raw sensory input and to plan immediate reactive actions during the exploration task. Spatial memory is used also in [14], where the authors adopt imitation learning and coverage rewards for bootstrapping learning the exploration policy. [15] proposes to learn an actor-critic model that selects the best frontier among those available. Similarly to the NBV methods described before, also these approaches

take a decision solely based on the current partial observation on the environment. However, their use is still limited due to the large amount of samples required for training the model.

### B. RFID Technology

RFID is essentially a form of radio-frequency (RF) communication. As a result, any RF localization technique can be also applied to RFID: Time Of Arrival, Time Difference Of Arrival, Trilateration, Received Signal Phase or radar [16]. These techniques are based on RF propagation models, including the physical aspects of RFID communication. They determine a set of RF propagation parameters and use them to locate the tags based on received power, phase or delay. Other techniques rely on a purely statistical approach modeling the tag detection event. They are data-driven approaches, use Machine Learning techniques to not take into account any radio signal propagation model. Unfortunately, both RF and statistical techniques usually require non conventional readers or tags [17] and multiple readers [18], or at least multiple antennas to provide spatial diversity. A compromise solution that can rely on commercial RFID readers is followed by recent Bayesian approaches. They include some physical properties of RFID radio propagation into an essentially statistical model [19]. Those approaches have proven to be useful for robot exploration. For example, [20] uses a Bayesian model to track tags and move robot towards the most likely location, but still requires detecting the tag to plan its next movement in a reactive way. Zhang *et al.* [3] propose small changes to Dynamic Window Approach (DWA) to move closer to obstacles where tags are supposed to be, following a fixed global plan covering the environment. A-priori knowledge about the tags locations is also considered by Li *et al.* [4] in their library inventory management system. Another usual approach is to use or even deploy tags as landmarks to help robot localization. For example, MF tags have a very reduced detection range ( $\sim 30$ cm.). This makes them suitable to improve exploration in simultaneous localization and mapping (SLAM) systems [21]. Our work does not assume any knowledge about the tags location, uses a way larger detection range and focuses on the opposite problem of efficiently surveying all the tags in a known map. It aims to overcome the limitations on the applied exploration strategies on RFID inventory systems.

The contribution of this paper is to use MCDM, originally introduced in robotics for human search and rescue, for RFID tags detection with an online greedy approach that considers novel criteria. Differently from [2], where a team of robots looks for victims, we do not use the probability of communication between the members of the team. Instead, we use the *sensing time*, defined as the time required for performing an inventory operation. We also consider the *RFID information gain*, namely the amount of uncertainty regarding the presence of a tag around a given location. As last, the current *battery charge* is another fundamental criterion when deploying a robot with long-term autonomy in mind. Moreover, we also differ from existing exploration

methods for RFID tags such as [3], [4], [20] because we do not have any a priori assumption on the tag location but only the map of the environment to explore.

### III. PROBLEM FORMULATION

In this work, we address the problem of tag detection by a single mobile robot in a known environment, represented by a binary occupancy grid, where cells are labeled either as free or as obstacle. Hence, the robot pose  $c$  is defined by the free grid cell  $g$  where it is located and its orientation  $\theta$ . We assume that the robot is always in the center of a cell and the orientation belongs to a finite set  $\Theta$  of possible orientations, with values equally spaced in  $[0, 2\pi)$ . We use a Moore neighborhood connectivity to model the robot's traversability over free cells, i.e., a robot can move from pose  $c$  to any of the 8 free poses  $c'$  around it. The robot can perform a *sensing operation* to analyze the presence of RFID tags in its surrounding. This operation consists in the emission of an interrogation signal, that will be replied by the tags with their identification number (ID), if they receive enough power to back scatter their reply. This is mostly an instantaneous operation but if the environment is dense enough and given the limited power on the reader, this may be a noticeable time. The *detection field* is highly limited by the specific antenna and tag radiation patterns, as well as by the environment. Based on the probabilistic sensor model we describe in Sec. IV, the detection field can be defined using the incoming received power.

A free grid cell  $g'$  is considered *sensible* from a robot pose  $p = (g, \theta)$  if the line segment spanning from the center of  $g$  to the center of  $g'$  does not intersect any obstacle cell and if the center of  $g'$  is inside the detection field. We assume that obstacles fully occupy grid cells, obstructing the line of sight of a sensing operation. The problem of planning a path for tag discovery is given by finding the optimal sequence of sensing operations  $\langle ((g_1, \theta_1)), ((g_2, \theta_2)), \dots, ((g_n, \theta_n)) \rangle$  to be performed in order to sense all the free cells of the environment (namely, each cell should be sensed from *at least* one pose  $((g_i, \theta_i))$ ). Pose  $(g_1, \theta_1)$  is the starting pose of the robot in the environment and is not required to be equal to pose  $(g_n, \theta_n)$ , i.e., we are looking for a path and not for a tour. Performance metrics for optimality include the final level of map coverage, the number  $n$  of sensing operations, the total distance traversed by the robot, the final battery level and the detection accuracy and precision.

### IV. SENSOR MODEL

A RFID reader emits radio frequency waves with a configurable power, using random frequencies. If a tag receives enough power to activate its chip, it sends back its identification number (ID). If that back-scattered radio wave gets to the reader, the reader will report the ID, the received power and the incoming wave phase. Usually there are multiple waves that make it to the antenna (other tags, diffractions) and the propagation conditions change, so there are significant variations in the received power and

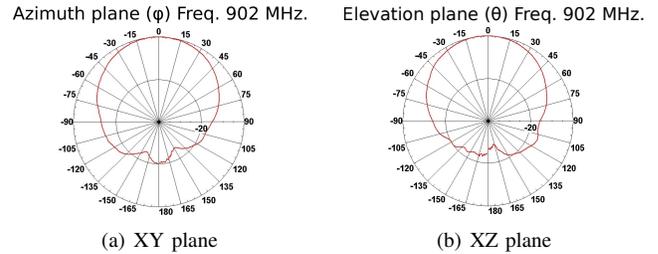


Fig. 2: RFID antenna Gain for the antenna used in this study. Antenna gain has a single maximum along axis X.

phase. Our probabilistic sensor model that accounts for these phenomena based on known UHF propagation models.

#### A. Friis equation

The Friis equation describes received power in terms of transmitted power, antenna-separation distance, frequency and antenna gains in free-space communication [22]. It is only valid in free space and in far field ( $d \gg \lambda$ ), but is often used as reference.

$$P_r = P_t + G_t(\theta, \varphi) + G_r(\theta, \varphi) + 10 \log\left(\frac{\lambda^2}{(d4\pi)^2}\right) \quad (1)$$

The Friis equation contains the following terms:  $P_t$  Transmitted power (decibels),  $P_r$  Received power (decibels),  $G_t$  Transmitter Antenna gain (decibels),  $G_r$  Receiver antenna gain (decibels),  $\lambda$  Carrier wavelength (meters),  $d$  distance between tag and antenna (meters). Antenna gains (reader, tag) are usually defined empirically by manufacturer data and depend on the azimuth angle  $\varphi$  and elevation angle  $\theta$  of the receiver w.r.t the transmitting antenna. The RFID reader antenna used in this work has a cardioid radiation pattern, shown in Fig. 2, whereas most RFID tags are dipole antennas [23]. In our experiments, we used a *Smartrac FROG 3D* tag that has a quasi isotropic propagation model [24].

In the studied scenario, there is a two way communication with just the initially transmitted power. After the RFID reader transmits, the tag receives the signal first and then it transmits back using the received power to the reader. Because of that, the Friis equation must be applied twice to find the final received power:

$$P'_r = P_t + 2G_t(\theta, \varphi) + 2G_r(\theta, \varphi) + 20 \log\left(\frac{\lambda^2}{(d4\pi)^2}\right). \quad (2)$$

This equation can be used to build a map (see Fig. 3(a)) of the expected power received by an antenna at the origin of coordinates from a tag placed elsewhere. Next subsection will describe additional noise factors included to model perturbations due to multiple path propagation and shadows.

#### B. Log-distance path loss model

This empirical model accounts for energy losses inside buildings or dense areas. It was first proposed by Seidel [25], based on field measurements. According to data, they modeled power noise in UHF band frequency as a log-normal distribution:

$$L(d) = -G_t(\theta, \varphi) - G_r(\theta, \varphi) - 10\gamma \log\left(\frac{\lambda}{d4\pi}\right) + X_g, \quad (3)$$

where  $L(d) = P_t(d) - P_r(d)$  Path losses (decibels),  $\gamma$  path loss exponent, an empirical factor,  $X_g$  Gaussian noise

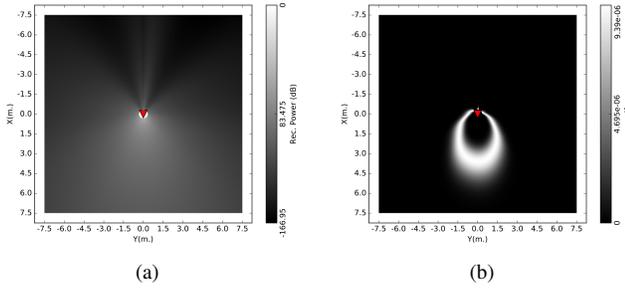


Fig. 3: (a) Received power ( $P_t = 0\text{dB}$ ,  $\text{Freq} = 902\text{MHz}$ ) using Friis equation. (b) Received power probability distribution ( $P_t = 0\text{dB}$ ,  $P_r = -95\text{dB}$ ,  $\text{Freq} = 902\text{MHz}$ ).

with zero mean, reflecting the attenuation (in decibel) caused by fading. The  $\gamma$  factor accounts for the obstacles inside the building. The noise factor, described by  $\sigma$ , describes the multi-propagation effect itself. For example, according to Seidel [25], in an office environment with soft partitions, a common noise factor has a  $\sigma = 3.92$  and  $\gamma = 2.2$ .

### C. Implemented Probabilistic Power Model

Our model is based upon the Friis equation and the noise model from the log-distance path loss model. First, the Friis equation provides the *expected* received power from a tag located at any position around the robot ( $P_r$ ). According to the log-distance model described above, the measurement noise of the received power  $P_r'$  is represented by  $X_g \sim \mathcal{N}(0, \sigma^2)$ . Hence, the likelihood of receiving power  $P_r'$  given the distance  $d$ , azimuth  $\varphi$  and elevation  $\theta$  can therefore be written as:

$$p(P_r'|d, \varphi, \theta) = \mathcal{N}(P_r' - P_r(d, \varphi, \theta), \sigma^2), \quad (4)$$

where  $P_r$  has been defined in Equation 2. In the further discussion, we will disregard the elevation angle  $\theta$  as we only concentrate on 2D maps. The likelihood function given a specific measurement of  $P_r$ , a signal frequency and noise variance  $\sigma^2 = 16$  is given in Fig. 3(b).

## V. THE NEXT-BEST-SENSE APPROACH

The Next-Best-Sense algorithm adopts the iterative approach already presented in Figure 1(b) to explore the environment. The robot executes sensing actions and updates its map with information from the most recent observation (detected tag identifiers in current scan). This information is then used to choose the next sensing cell and the robot finally executes the navigation task.

### A. Information Integration

One of the main advantages of NBS is that it focuses on a reduced set of frontiers. Frontiers are cells on the boundary between the scanned and the unexplored environment. For each of them, we consider multiple candidate robot *poses*, one for each orientation  $\theta \in \Theta$ . In order to choose the best pose among the candidates, we define five *criteria* relevant for the problem at study. (i) *Information gain (IG)*, computed as the number of free unscanned cells that the robot will be

able to sense from the candidate pose. (ii) *Travel distance (TD)*, the distance between the current robot pose and the candidate pose. (iii) *Sensing time (ST)*, the time required to sense all the free unscanned cells senseable from the candidate pose. (iv) *RFID Information gain (RFID)*, corresponding to the amount of uncertainty with reference to a RFID tag position in the proximity of the considered pose. (v) *Battery Status (BS)*, the expected amount of battery charge left after reaching the candidate pose. Information gain values are computed using *ray casting*: we take each free cell inside the sensor detection field. The RFID Information gain is calculated as the total entropy  $H$  of the surrounding polygon  $S$  of the candidate pose, where the entropy is a function of the cell probability of containing a tag. Since the tag presence in a cell  $i$  is a Bernoulli variable with probability  $p_i$ , the entropy can be computed as

$$H = \sum_{i \in S} -p_i \log_2 p_i - (1 - p_i) \log_2 (1 - p_i). \quad (5)$$

### B. Candidate Pose selection

In order to select the best candidate pose, we use an utility function that combines all the aforementioned criteria in a balanced way. This function is defined using the MCDM method, which has been proven useful in robot exploration [2]. MCDM deals with problems in which a decision maker has to choose among a set of alternatives and its preferences depend on different, and sometimes conflicting, criteria. Each criterion  $k_i$  is characterized by an utility function  $u_i$  which returns values between 0 and 1. MCDM offers a principled way to combine criteria and to account for their dependencies. Depending on the features estimated, two criteria can be related by a relationship of *redundancy* or *synergy*. Therefore, their joint contribution to the utility function can be less or greater, respectively, of the sum of the marginal ones. Differently from *ad hoc* exploration strategies [5], [6], MCDM exploits an aggregation method called *Choquet Fuzzy Integral* to model relations of redundancy and synergy. For our purposes (see [2] for a complete description), we first introduce a function  $\eta : P(N) \rightarrow [0, 1]$ , where  $N$  is the set of criteria considered ( $|N| = 5$  in our case) and  $P(N)$  is the power set of  $N$ , with the following properties:  $\eta(\{\emptyset\}) = 0$ ,  $\eta(N) = 1$ , if  $A \subseteq B \subseteq N$ , then  $\eta(A) \leq \eta(B)$ . The function  $\eta$  is used to specify weights for each subset of criteria, which have to be chosen by the user. The weights specified by  $\eta$  capture the above relations among criteria: if two criteria  $k_1, k_2$  are redundant, then we specify that  $\eta(\{k_1, k_2\}) < \eta(\{k_1\}) + \eta(\{k_2\})$ , while if they are synergic  $\eta(\{k_1, k_2\}) > \eta(\{k_1\}) + \eta(\{k_2\})$ ; in case  $\eta(\{k_1, k_2\}) = \eta(\{k_1\}) + \eta(\{k_2\})$  we say that the criteria are *independent*. For a given pose  $c$ , we first sort the criteria according to their utility such that

$$u_{(1)}(c) \leq \dots \leq u_{(|N|)}(c) \leq 1, \quad (6)$$

where we also assume that  $u_{(0)}(c) = 0$ . Furthermore, we define the set  $A_j$  as

$$A_{(j)} = \{i \in N | u_{(j)}(c) \leq u_{(i)}(c) \leq u_{(|N|)}(c)\}, \quad (7)$$

i.e., the set of all criteria with utility larger than the utility from  $k_j$ . The global utility function  $f(u_c)$  of a candidate pose  $c$  is computed by a *discrete* Choquet integral which uses the ranking of the utilities as well as the sets  $A_j$ , i.e.,

$$f(c) = \sum_{j=1}^{|N|} (u_{(j)}(c) - u_{(j-1)}(c)) \eta(A_{(j)}). \quad (8)$$

We define the weights of the five criteria as follows:  $x_1$  is the weight of *information gain* criterion,  $x_2$  is that of the *travel distance*,  $x_3$  is that of the *sensing time*,  $x_4$  is the weight of the *RFID information gain* criterion and finally  $x_5$  is the weight for the *battery status* criterion. Individual weights must be greater or equal than zero and their sum must add up to one. Finding the best weights for sets of criteria is a difficult but fundamental task. Given that we have only five criteria, we adopted a simple *ad-hoc* selection of weights, keeping in mind that  $\eta(N) = 1$ , identifying a subset of interesting configurations to test. Firstly, the one associated with the vertices of the parameters spaces, in which one criterion is associated with maximum weight ( $x_i = 1.0$ ) and the others with null. Then, we sampled a configuration in which all the criteria have the same importance ( $x_i = 1/n$ , with  $n = 5$ ). In the end, we selected five configurations in which one criterion at each time has a big associated weight  $x_i$  and the others have identical importance ( $x_j = (1 - x_i)/(n - 1) \forall j \neq i$ ). The intuition behind the choice of the weights is that, for example, maximizing *information gain*, *travel distance*, and *sensing time* criteria is expected to correspond to a robot behavior that quickly covers most of the environment, that travels short distances, and that achieve full coverage with less sensing operations, respectively. These configurations are then evaluated experimentally on different maps and the best is identified as the one optimizing an objective function, in our case minimizing the number of sensing operation and the traveled distance, with less impact on the battery, while having high precision in finding the tags.

### C. RFID belief map

In addition to the map used for navigation, our NBS approach iteratively updates a belief map each time a sensing operation is performed. We describe now how to update the cells for the RFID grid.

At the beginning of the task, we assume each grid cell maintains a probability  $p_{ij}$ , which represents the belief that  $p_{ij} = p(O_{ij} = 1)$  contains a tag. The prior for  $p_{ij}$  is initialized uniformly, for example if we have  $N$  grid-cells, then  $p_{ij} = 1/N$ . We also have a likelihood function for each power measurement  $P'_h$  defined by Equation 4. Note that, for each measurement  $h$ , we can compute distance  $d$  and azimuth  $\varphi$  of the robot to each grid-cell  $ij$ . We will therefore write:

$$p(P'_h | ij, c_h) = p(P'_h | d(ij, c_h), \varphi(ij, c_h)) \quad (9)$$

as likelihood of our measurements, where  $c_h$  is the configuration of the robot (position and orientation) at the

$h$ th measurement. Given the likelihood, the new posterior belief can be obtained by a Bayesian update, i.e.,

$$p(O_{ij} = 1 | P'_h, c_h) = \frac{p(P'_h | ij, c_h) p_{ij}}{\sum_{ij} p(P'_h | ij, c_h) p_{ij}}. \quad (10)$$

This Bayesian update needs to be applied after every measurement  $P'_h$ , where the old posterior is again used as prior  $p_{ij} = p(O_{ij} = 1)$  for the next update. A graphical representation of the Bayesian update over a map is given in Fig. 4 where the red arrow represents the robot pose and the green circle is the ground truth position of the marker.

## VI. EXPERIMENTS

We tested the proposed NBS approach both in simulation and in the real world, using different maps and weight configurations.

### A. Parameters and Evaluation Metrics

The RFID tags discovery problem is affected by multiple parameters. (i) RFID probabilistic sensor model. It depends on the antenna, tags, transmitted power used and noise factor considered in the experiment. We always use a noise factor  $\sigma = 3.92$ , as expected for an office setup. (ii) The number of possible orientations  $\Theta$  the robot can be in at a given position. (iii) The configurations of the five criteria described in Sect. V-A, namely Information Gain, Travel Distance, Sensing Time, RFID Information Gain and Battery Status.

The metrics considered in our experiments are given as: (i) Coverage: the final percentage of sensed cells over the total of free ones. (ii) #Config: the number of sensing operations, namely how many times the robot stops sensing for a tag. (iii) Distance: the total traversed distance covered by the robot from the beginning of the exploration task. (iv) Battery: the battery level at the end of the exploration. Differently from the distance, it is also affected by the number of rotations performed by the robot. (v) Accuracy: the percentage of tag correctly detected. (vi) Precision: the tag localization precision, expressed as euclidean distance from the ground truth.

### B. Simulation Experiments

Multiple tests are run in simulation to evaluate our approach and to identify the best combinations of criteria weights for addressing the RFID detection problem. We selected five different environments: a corridor of the Orebro Universitet's Teknikhuset in Sweden (OREBRO), a warehouse of the National Centre for Food Manufacturing in UK (NCFM), and three environments at the Isaac Newton Building at Lincoln University in UK – an office (INB\_3123), a corridor at the School of Engineering (INB\_ENG) and the Atrium (INB\_Atrium) –. All of the maps have been created using the robot, cleaned from noise and discretized as binary images (Fig. 5). For each map, we tested 11 weights criteria combinations sampled from the parameter space in the way described before. For space limitation, in Table I we report only the weights associated to the individual criteria. Given the nature of the criteria selected,

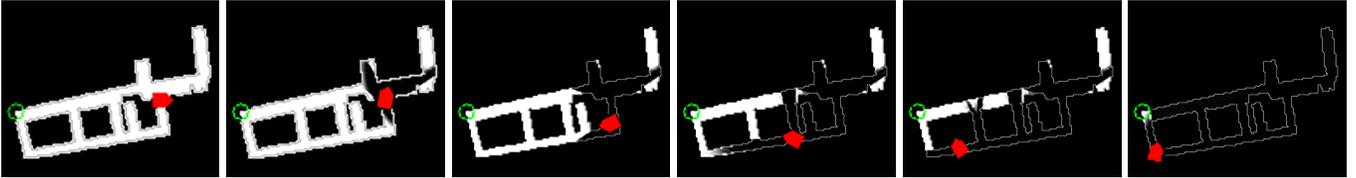


Fig. 4: Bayesian update for the RFID belief map. The red arrow is the robot pose, the green circle is the ground truth position of one RFID tag. White cells are associated with a high probability of containing the tag. At the beginning (left), all the cells have uniform probability. During the exploration, the probability is updated using eq. 10. After emitting a signal, if the robot does not receive a reply, the surrounding cells within the detection field and in the line-of-sight are updated with a lower value, while the probability the tag is located elsewhere in the map is increased. Differently, if a reply is received, the probability is increased in the surrounding of the robot and decreased elsewhere. Please note how the probability move towards the correct map position at different exploration stages (left to right).

we model a synergic relationship among all of them, i.e., for a set of two criteria, we set  $\eta(\{k_i, k_j\}) = \eta(\{k_i\}) + \eta(\{k_j\}) + b$ , with  $b = 0.1$ . We compared NBS with a *RandomWalk* solution and the one we call *RandomFrontier*, which are now described. In *RandomFrontier*, after scanning a local portion of the map, a random destination is chosen among the frontiers. *RandomWalk*, instead, at each iteration randomly samples the next destination from the list of *all* the free cells in the map. Differently from the other solutions, *RandomWalk* can select as next destination a pose that has already been adopted by the robot, in an area of the map already scanned. For this reason, *RandomWalk* represents a very basic baseline. In all of the tests, we placed 10 tags in random positions uniformly sampled from the free space. For each configuration, every single map has been solved three times using a different robot starting position and tags location.

The results of the simulations are reported in Table I and in Fig. 6. Here at the top, the coverage rate of the three approaches is plotted for each one of the environments. *RandomWalk* never accomplishes full map coverage due to battery discharge. NBS proves instead to be faster than *RandomFrontier* because of a better policy in choosing the next destination for the robot. Figure 6-(bottom), instead, shows the traveled distance covered by the robot while following the different policies. Based on this metric, it is possible to notice how NBS performs better than both *RandomFrontier* and *RandomWalk*.

From these simulations we can draw some conclusions regarding which weights configuration to use. Looking at the Table I, it is possible to notice how simply focusing on maximizing one criterion does not lead to the best



Fig. 5: The maps used: from left, OREBRO ( $242 \text{ m}^2$ ,  $0.5 \text{ m/px}$ ), NCFM ( $386 \text{ m}^2$ ,  $0.25 \text{ m/px}$ ), INB\_3123 ( $262 \text{ m}^2$ ,  $0.5 \text{ m/px}$ ), INB\_ENG ( $364 \text{ m}^2$ ,  $0.25 \text{ m/px}$ ), INB\_Atrium ( $355 \text{ m}^2$ ,  $0.33 \text{ m/px}$ ).

overall performance, namely obtaining the best value across all the metrics considered. For example, assigning all the importance to InformationGain (IG), we minimize the number of times the robot stops for a sensing operation but the traversed distance is on average  $> 2x$  the best result. Interestingly, maximizing the TravelDistance (TD) weight do not lead to a minimum traversed path, but this make sense as we use a greedy optimization. The most satisfying results are obtained when all criteria are associated with a positive weight. More importantly, almost all the NBS configurations always guarantee better ( $> 2x$ ) performances than *RandomFrontier* if just looking at the overall traversed distance. *RandomWalk*, instead, always performs much worst than the other two approaches in all the environments, running out of battery even before covering the 60(%) of the entire map. To conclude, this study showed that adopting MCDM to balance the importance of multiple optimization objectives always leads to a better performance either than a random heuristic or than maximizing a single criterion. In particular, the configuration of the weights favoring TravelDistance (TD) tends to be the best setting for most of the metrics considered regardless of the structure of the environment. Another interesting observation is that aiming only at detecting the tags (RFID = 1) still allows to find all tags with high precision regardless of partial map coverage. Yet, this configuration would leave the robot with very little battery left in most of the experiments due to the higher travel distances.

### C. Gazebo and Real World Experiments

In order to better quantify the efficiency of the proposed approach for RFID tag detection, we tested NBS both in Gazebo and in the real world using as test-bed the environments INB\_3123 and INB\_ENG. For our experiments, we used the Linda robot already presented in Sec. I. Our idea is to show the robustness of NBS to increasing complexity environments and setting. In the previous section, we showed that NBS is able to successfully solve a given two dimensional grid and identify multiple RFID tags with an accuracy lower than one meter. We now move forward, presenting results obtained in Gazebo where we used the full robotic navigation stack offered by ROS. At the same time, we compare the results obtained within the simulator with those obtained with a prototype robot implementation, where

TABLE I: Simulation results on five grid worlds. The performance metrics are expressed as mean and standard deviation across three runs per configuration. Best results are reported in bold.

Criteria	NBS Configurations												Random_frontier	Random_walk
	BS	1	0	0	0	0.6	0.1	0.1	0.1	0.1	0.2	0.2		
RFID	0	1	0	0	0	0.1	0.6	0.1	0.1	0.1	0.2			
ST	0	0	1	0	0	0.1	0.6	0.1	0.1	0.1	0.2			
TD	0	0	0	1	0	0.1	0.1	0.1	0.1	0.1	0.2			
IG	0	0	0	0	1	0.1	0.1	0.1	0.1	0.1	0.2			
INB_3123	Coverage(%)	100	100	100	100	100	100	100	100	100	100	100	100	60(20)
	Battery(%)	83.44(1.4)	50.18(2.65)	35.29(7.13)	81.92(0.58)	69.51(1.37)	<b>87.55(0.88)</b>	81.98(1.53)	82.03(0.51)	86.87(0.43)	80.87(1.72)	85.45(1.25)	62.88(2.9)	0
	#Config	784.67(47.82)	477.33(16.86)	869.33(63.51)	785.67(57.01)	<b>240.33(4.16)</b>	312.67(10.97)	321.03(34.04)	406.0(21.66)	327.0(15.87)	272.0(19.08)	333.33(11.59)	367.33(14.38)	125.66(7.41)
	Distance(m)	370.63(31.1)	825.45(25.07)	1009.47(102.57)	372.84(13.9)	501.53(20.96)	253.72(14.41)	334.73(29.46)	355.87(11.37)	<b>257.37(2.0)</b>	332.38(19.02)	281.15(22.88)	560.13(35.02)	2121.94(91.53)
	Accuracy(%)	100	100	100	100	100	97(6)	100	100	100	100	100	100	83(6)
INB_ENG	Precision(m)	<b>0.61(0.19)</b>	0.7(0.12)	0.66(0.14)	0.68(0.11)	0.66(0.1)	1.01(0.55)	0.68(0.12)	0.73(0.13)	0.77(0.18)	0.68(0.14)	0.76(0.15)	0.68(0.07)	1.51(1.17)
	Coverage(%)	100	98(1)	89(11)	100	100	100	100	100	100	100	100	100	51(3)
	Battery(%)	79.54(1.67)	8.57(15.12)	0	76.61(4.23)	0	57.0(4.34)	79.65(2.5)	58.17(3.99)	69.17(1.99)	<b>82.54(2.1)</b>	69.4(3.74)	72.7(2.93)	45.45(2.84)
	#Config	673.33(71.4)	574.33(88.52)	1079.0(15.13)	713.67(126.62)	<b>220.67(13.58)</b>	399.67(18.5)	307.33(28.04)	499.67(14.57)	323.0(14.11)	282.33(27.32)	410.0(9.0)	381.0(13.95)	89.0(6.68)
	Distance(m)	230.76(23.73)	823.31(129.89)	781.79(7.37)	241.91(39.75)	389.59(46.5)	212.31(29.87)	367.03(34.87)	274.98(15.9)	<b>166.97(17.98)</b>	273.69(35.87)	270.83(35.68)	498.65(13.92)	1066.90(61.75)
INB_ATRIUM	Accuracy(%)	97(6)	100	87(12)	100	93(6)	100	100	100	90(0)	83(6)	100	100	60(10)
	Precision(m)	0.44(0.35)	0.35(0.08)	0.83(0.62)	0.38(0.07)	0.61(0.5)	0.33(0.06)	0.36(0.09)	<b>0.30(1.2)</b>	0.73(0.69)	1.0(0.76)	0.38(0.06)	0.49(0.36)	1.72(0.86)
	Coverage(%)	100	98(0)	97(2)	100	100	100	100	100	100	100	100	100	32(2)
	Battery(%)	65.28(5.23)	0	4.55(7.95)	62.68(3.13)	36.14(3.37)	73.36(3.91)	51.34(4.33)	48.3(1.03)	<b>73.47(3.69)</b>	59.7(3.36)	68.56(5.4)	7.23(2.06)	0
	#Config	1277.0(169.78)	635.33(22.85)	1049.33(45.79)	1298.33(127.63)	378.67(13.65)	606.33(6.51)	578.0(62.51)	886.0(13.0)	527.33(31.53)	<b>488.33(4.93)</b>	612.33(8.39)	654.33(15.43)	88.66(4.49)
NCFEM	Distance(m)	521.76(73.15)	1195.09(8.99)	1029.88(74.16)	516.18(42.21)	743.42(44.9)	370.73(50.76)	596.06(45.37)	644.06(18.46)	<b>347.54(51.82)</b>	503.23(55.09)	413.69(61.87)	1042.29(42.54)	1433.06(70.15)
	Accuracy(%)	97(6)	97(6)	87(6)	97(6)	87(6)	97(6)	93(12)	97(6)	90(0)	83(6)	97(6)	100	67(6)
	Precision(m)	0.68(0.43)	<b>0.62(0.5)</b>	1.05(0.76)	0.64(0.41)	1.12(1.07)	0.64(0.46)	0.76(0.65)	0.64(0.45)	0.93(0.7)	1.24(1.03)	0.62(0.48)	0.79(0.57)	2.08(1.37)
	Coverage(%)	100	95(3)	75(23)	100	100	100	100	100	100	100	100	100	41(0)
	Battery(%)	70.02(3.46)	0	0	71.13(5.19)	43.73(4.31)	74.24(5.7)	50.38(7.92)	49.45(9.54)	<b>81.93(1.54)</b>	62.98(3.35)	66.6(5.06)	14.28(5.5)	0
OREBRO	#Config	992.33(78.78)	572.67(40.77)	761.0(56.51)	921.0(143.5)	<b>317.33(18.58)</b>	560.67(48.64)	500.67(65.76)	759.0(75.66)	420.0(47.09)	439.67(67.0)	606.67(43.66)	481.33(14.38)	70.66(3.78)
	Distance(m)	351.36(36.18)	936.42(4.16)	879.27(6.22)	309.2(50.3)	520.05(33.57)	273.77(40.44)	471.1(70.6)	467.77(79.76)	<b>197.16(14.27)</b>	349.88(19.33)	349.53(48.38)	782.35(16.29)	1126.18(54.04)
	Accuracy(%)	93(6)	73(15)	100	100	83(6)	90(10)	87(6)	93(12)	87(6)	93(6)	93(12)	93(12)	40(17)
	Precision(m)	0.61(0.46)	0.58(0.47)	1.26(0.69)	<b>0.39(0.08)</b>	0.97(0.65)	0.68(0.75)	0.84(0.59)	0.69(0.59)	0.9(0.56)	0.63(0.51)	0.66(0.51)	0.74(0.51)	2.43(1.02)
	Coverage(%)	100	100	100	100	100	100	100	100	100	100	100	100	54(3)
Battery(%)	84.04(0.7)	47.14(12.06)	35.47(3.38)	83.04(1.33)	76.24(1.04)	88.05(0.53)	82.07(0.62)	80.99(2.66)	<b>88.52(1.05)</b>	82.99(1.68)	85.52(2.4)	65.15(1.82)	0	
#Config	740.67(33.38)	491.0(85.85)	684.33(44.28)	<b>200.0(12.0)</b>	684.33(44.28)	316.0(2.0)	290.67(12.34)	405.67(44.09)	293.33(14.01)	223.33(9.29)	297.33(30.06)	328.86(4.49)	109.66(2.05)	
Distance(m)	566.42(12.28)	908.17(188.39)	1005.74(88.62)	352.39(21.18)	399.9(20.89)	251.16(8.19)	360.02(11.06)	362.76(47.48)	<b>232.84(17.18)</b>	316.88(29.82)	279.44(50.86)	531.83(3.26)	2170.44(36.85)	
Accuracy(%)	100	100	100	100	100	100	100	100	100	100	93(6)	100	90(10)	
Precision(m)	<b>0.65(0.14)</b>	0.63(0.09)	0.7(0.05)	0.65(0.2)	0.72(0.18)	0.75(0.15)	0.69(0.18)	0.7(0.09)	0.78(0.11)	1.22(0.89)	0.71(0.1)	0.71(0.0)	1.49(1.06)	

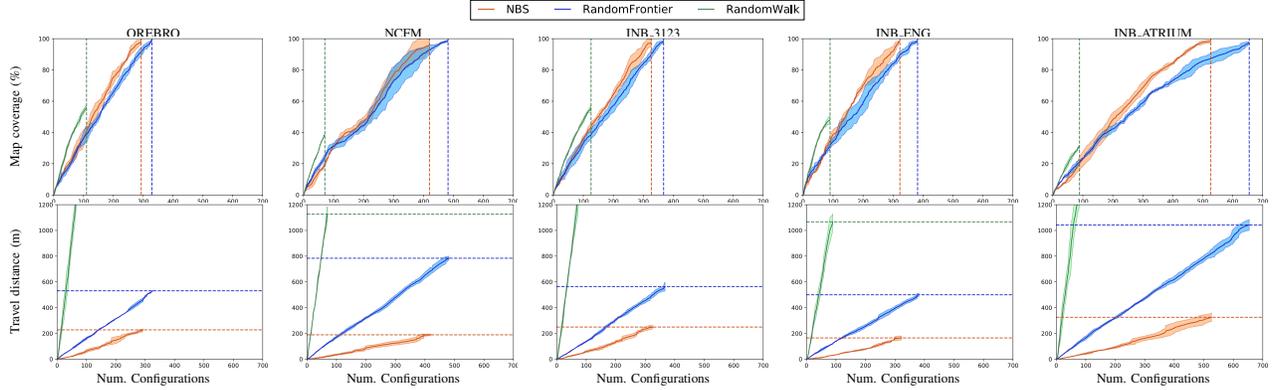


Fig. 6: Simulations results while solving five grid-world scenarios. The first row shows the map coverage rate of the three strategies, while in second row, the total traveled distance (for NBS, we used IG=0.1, TD=0.6, ST=0.1, RFID=0.1, BS=0.1).

miss-localization and path occlusions can lead to a drop in performance. The results are reported in Fig. 7. RandomWalk proves again to be the slowest exploration strategy to converge to full map coverage (under the assumption of unlimited battery charge). Surprisingly, it is the fastest policy to find all the four RFID tags located in INB\_ENG. This can be explained because RandomWalk does not plan locally as for the other two strategies. It uniformly samples a random destination from the list of all free cells in the map. However, it's not possible to draw any significant conclusion on the behavior showed by the robot given the randomness of the approach. RandomFrontier, instead, performs almost twice as better than RandomWalk regarding the map coverage, while it proves to be the slowest approach to find all the tags. As already proved in the previous section, NBS outperforms the other exploration policies. Interestingly, our real implementation performs really close to the one in simulation. This result has a really significant importance because validates the probabilistic model used in this study. Moreover, it proves the idea of a full sim-to-real pipeline without a significant drop in performance.

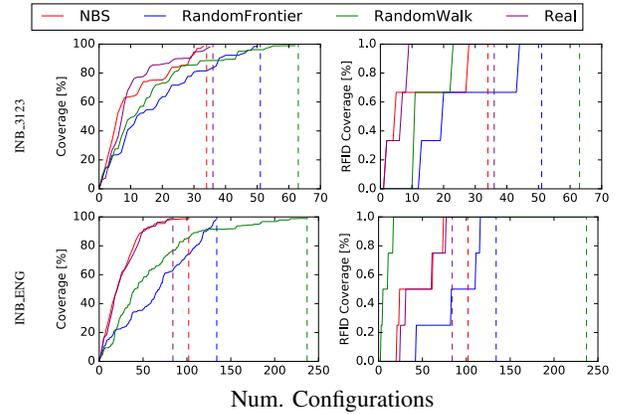


Fig. 7: Map and RFID coverage results obtained in the Gazebo simulator and with a prototype robot.

## VII. CONCLUSIONS

In this paper, we addressed the problem of planning coverage paths for RFID tags discovery with a mobile robot equipped with an RFID reader. The solution we presented follows an online approach and uses MCDM to define a

Next-Best-Sense strategy that chooses the next sensing pose by optimizing multiple criteria. Experimental evidence, both in real world and simulation, shows the potential of NBS in finding effective exploration strategies which lead to high tag localization precision (less than a meter). The main limitation of this work is linked with the resolution of the map which affects the computational time and the tags detection precision. Future work will address this trade-off, for example using a topological map [9] for the navigation and a finer resolution for the tag detection or by adapting the resolution in the areas of interest. Another extension is to scale the exploration to the spatiotemporal dimension [26], to find tags which change position over time (inventory management applications) in an environment with moving obstacles [27], [28]. Moreover, we will work on improving the tag detection precision. For example, allowing variable transmitted power can reduce the number of required robot configurations, reducing overlapping in exploration. Finally, the probabilistic sensor model of the RFID tag detection could be extended to take into account also the phase of the received signal. This, together with a finer resolution map, would allow an even more precise tag localization.

#### Acknowledgements

The research leading to these results has received funding from EPSRC under grant agreement EP/R02572X/1 (National Center for Nuclear Robotics) and the European Union's Horizon 2020 research and innovation programme under grant agreement No 732737 (ILIAD).

#### REFERENCES

- [1] V. Scilimati, A. Petitti, P. Boccadoro, R. Colella, D. Di Paola, A. Milella, and L. Alfredo Grieco, "Industrial internet of things at work: a case study analysis in robotic-aided environmental monitoring," *IET Wireless Sensor Systems*, vol. 7, no. 5, pp. 155–162, 2017.
- [2] F. Amigoni and N. Basilico, "Exploration strategies based on multi-criteria decision making for searching environments in rescue operations," *Autonomous Robots*, vol. 31, no. 4, pp. 401–417, November 2011.
- [3] J. Zhang, Y. Lyu, T. Roppel, J. Patton, and C. P. Senthilkumar, "Mobile robot for retail inventory using rfid," in *2016 IEEE International Conference on Industrial Technology (ICIT)*, pp. 101–106, March 2016.
- [4] Renjun Li, Zhiyong Huang, E. Kurniawan, and Chin Keong Ho, "Auross: An autonomous robotic shelf scanning system," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6100–6105, Sep. 2015.
- [5] H. Gonzáles-Baños and J.-C. Latombe, "Navigation strategies for exploring indoor environments," *The International Journal of Robotics Research*, vol. 21, no. 10-11, pp. 829–848, 2002.
- [6] C. Stachniss and W. Burgard, "Exploring unknown environments with mobile robots using coverage maps," in *Proc. IJCAI*, 2003.
- [7] F. Mascari, T. Wilson, C. Papachristos, and K. Alexis, "Radiation source localization in gps-denied environments using aerial robots," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6537–6544, 2018.
- [8] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, pp. 146–151, 1997.
- [9] C. Wang, D. Zhu, T. Li, M. Q. Meng, and C. W. de Silva, "Efficient autonomous robotic exploration with semantic road map in indoor environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2989–2996, 2019.
- [10] F. Amigoni, V. Caglioti, and U. Galtarossa, "A mobile robot mapping system with an information-based exploration strategy," in *Proc. ICINCO*, pp. 71–78, 2004.
- [11] B. Tovar, L. Muñoz-Gomez, R. Murrieta-Cid, M. Alencastre-Miranda, R. Monroy, and S. Hytchinson, "Planning exploration strategies for simultaneous localization and mapping," *Robotics and Autonomous Systems*, vol. 54, no. 4, pp. 314–331, 2006.
- [12] M. Selin, M. Tiger, D. Duberg, F. Heintz, and P. Jensfelt, "Efficient autonomous exploration planning of large-scale 3-d environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1699–1706, 2019.
- [13] J. Zhang, L. Tai, J. Boedecker, W. Burgard, and M. Liu, "Neural slam: Learning to explore with external memory," *arXiv preprint arXiv:1706.09520*, 2017.
- [14] T. Chen, S. Gupta, and A. Gupta, "Learning exploration policies for navigation," in *7th International Conference on Learning Representations, ICLR 2019*, 2019.
- [15] F. Niroui, K. Zhang, Z. Kashino, and G. Nejat, "Deep reinforcement learning robot for search and rescue applications: Exploration in unknown cluttered environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 610–617, 2019.
- [16] X. Fu, A. Pedross-Engel, D. Armitz, and M. S. Reynolds, "Simultaneous sensor localization via synthetic aperture radar (sar) imaging," in *2016 IEEE SENSORS*, pp. 1–3, Oct 2016.
- [17] E. DiGiampaolo and F. Martinelli, "Mobile Robot Localization Using the Phase of Passive UHF RFID Signals," *IEEE Transactions on Industrial Electronics*, vol. 61, pp. 365–376, Jan 2014.
- [18] L. Geng, M. F. Bugallo, A. Athalye, and P. M. Djurić, "Indoor Tracking With RFID Systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 96–105, Feb 2014.
- [19] J. Zhang, Y. Lyu, J. Patton, S. C. G. Periaswamy, and T. Roppel, "Bfvp: A probabilistic uhf rfid tag localization algorithm using bayesian filter and a variable power rfid model," *IEEE Transactions on Industrial Electronics*, vol. 65, pp. 8250–8259, Oct 2018.
- [20] R. Liu, G. Huskić, and A. Zell, "Dynamic objects tracking with a mobile robot using passive uhf rfid tags," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4247–4252, Sep. 2014.
- [21] V. A. Ziparo, A. Kleiner, B. Nebel, and D. Nardi, "RFID-Based Exploration for Large Robot Teams," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 4606–4613, 2007.
- [22] J. Shaw, "Radiometry and the friis transmission equation," *American Journal of Physics*, vol. 81, pp. 33–37, 01 2013.
- [23] C. Qin, L. Mo, H. Zhou, and H. Zhang, "Dual-Dipole UHF RFID Tag Antenna with Quasi-Isotropic Patterns Based on Four-Axis Reflection Symmetry," *International Journal of Antennas and Propagation*, vol. 2013, p. 194145, 2013.
- [24] SMARTRAC, "Smartrac frog 3d tag datasheet," May 2018. [Online].
- [25] S. Y. Seidel and T. S. Rappaport, "914 MHz. path loss prediction models for indoor wireless communications in multifloored buildings," *IEEE Transactions on Antennas and Propagation*, vol. 40, pp. 207–217, Feb 1992.
- [26] J. M. Santos, T. Krajník, J. P. Fentanes, and T. Duckett, "Lifelong information-driven exploration to complete and refine 4-d spatio-temporal maps," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 684–691, 2016.
- [27] G. Qi, H. Wang, M. Haner, C. Weng, S. Chen, and Z. Zhu, "Convolutional neural network based detection and judgement of environmental obstacle in vehicle operation," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 80–91, 2019.
- [28] Y. Wu, Y. He, P. Shivakumara, Z. Li, H. Guo, and T. Lu, "Channel-wise attention model-based fire and rating level detection in video," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 117–121, 2019.