

# Particle Filter-based Trajectory Estimation with Passive UHF RFID Fingerprints in Unknown Environments

[Draft Version]

Philipp Vorst and Andreas Zell

**Abstract**—In this paper we present a novel approach to estimating the trajectory of a robot by means of inexpensive passive RFID tags and odometry in unknown environments. We show how trajectory estimation, a prerequisite of mapping RFID transponder positions without a reference positioning system, can be achieved using a particle filter. The presented technique is based on a non-parametric model of spatial relationships between RFID measurements. It overcomes the noisy nature of RFID measurements and the absence of distance and bearing information. The accuracy of our method is investigated in a series of experiments with a mobile service robot.

## I. INTRODUCTION

Over the last decade, considerable progress has been achieved in enabling robots to autonomously map unknown environments. Usually, a solution involves simultaneous localization and mapping (SLAM), where map building and estimating the trajectory of the robot are performed concurrently. A crucial point for the generation of consistent maps is data association, that is, the association of observations to previously seen landmarks. Radio frequency identification (RFID) solves this issue trivially by the contactless identification of objects via radio waves. Still, there is little literature on RFID-based SLAM, and the mapping of transponder positions usually requires a reference localization system. One reason for this is that the SLAM-inherent issue of loop detection is easy with RFID, but closing the loop *accurately* is not trivial: Long-range RFID detections reveal large position uncertainty while distance and bearing to a transponder are unknown. Passive ultra-high frequency (UHF, 433-950 MHz) RFID, the prevailing technology for product tagging, has a read range of up to 7m and is characterized by a lot of false-negative detections.

In this paper, we show how to tackle these difficulties: We decouple the task of estimating the trajectory of the robot from the task of mapping transponder positions. Our algorithm addresses the first task such that – given the tracked path of the robot – tag positions can be estimated efficiently afterwards by means of well-known related approaches. The technique is motivated by the observation that in warehouses, logistic centers, and supermarkets, more and more products are being labeled with inexpensive RFID tags. Robots with

RFID readers will be able to perform inventory, detect and transport goods, or serve as shopping guides. As a side effect, RFID can be used cost-efficiently also for localization; thousands of transponders which are attached to objects stored in shelves can be utilized as navigation landmarks. Compared to laser range finders, which are commonly used for localization and mapping, RFID readers are expected to be cheaper in the long run. Moreover, in contrast to vision-based SLAM, computationally expensive image processing is not required when RFID is used. Additionally, RFID does not depend on line of sight and is independent of lighting conditions.

Our method works as follows: We exploit the fact that the degree of similarity of two RFID measurements taken at two positions indicates how close the two positions are. Based on training data, we learn non-parametric models of the likelihoods of different degrees of similarities, conditioned on the distance between two measurement positions. The learned likelihood models are used to reweight the samples of a particle filter that tracks the pose of the robot. As the result, our method returns the most likely trajectory of the robot. In conjunction with the RFID measurements, it can be used to estimate the positions of RFID tags, or to localize the robot with a fingerprinting-based positioning technique. In general, the learned RFID observation model could also be used to augment particle filter-based SLAM approaches using laser range finders or cameras. Our choice to use particle filtering is motivated by the seamless integration of the proposed observation model as well as reported good accuracy in the context of SLAM (e.g., [1], [2]).

This paper is structured as follows. In Sect. II, we present a survey of related approaches, and in Sect. III we examine how RFID can be used for trajectory estimation. Thereafter, we clarify the background of particle filter-based trajectory estimation in Sect. IV. Section V treats the derivation and utilization of RFID fingerprint models, before we present experimental results with a mobile robot in Sect. VI. Finally, we draw conclusions in Sect. VII.

## II. RELATED WORK

The approach presented in this paper is closely related to SLAM solutions, although transponder positions are not simultaneously mapped: The consistent reconstruction of the robot trajectory involves its correction via sensor readings and loop closure, as with SLAM. Especially, our work relates to SLAM with (Rao-Blackwellized) particle filters such as

This work was funded by the Landesstiftung Baden-Württemberg in the scope of the research program BW-FIT, research cooperation AmbiSense.

P. Vorst is with the Chair of Computer Architecture, headed by A. Zell, Computer Science Department, University of Tübingen, Sand 1, D-72076 Tübingen, Germany {philipp.vorst, andreas.zell}@uni-tuebingen.de

FastSLAM, which was introduced by Montemerlo *et al.* [1]. They presented how to generate consistent maps computationally efficiently with laser range finders (or cameras). Laser-based FastSLAM was also employed by Hähnel *et al.* [3] to estimate the trajectory of their robot and the positions of passive UHF RFID tags in an office environment. A probabilistic sensor model formed the basis to reweight the samples in particle filter-based mapping. Given the map of transponders, the robot was finally able to localize itself with RFID and odometry alone. Our approach enables a robot to do the same, but eliminates the potentially expensive reference positioning system. Kleiner *et al.* [4] showed how trajectory correction and graph-based SLAM can be performed with sparsely spread, passive transponders. Their approach relies on the direct proximity to a short-range tag (and thus high certainty in the relative position between the robot and the tag), and it can be pursued by multi-robot teams in a distributed fashion. Tanaka’s work [5] is based on a similar setup, but he used a more recent stochastic gradient descent framework to optimize the constraints given by ego-motion and RFID measurements [6], [7]. In [8], we also investigated a graph-based SLAM framework for trajectory estimation. The setup was very similar to the one in the work at hand, i.e., long-range passive RFID and quite high tag densities. Graph-based SLAM has the advantage that potentially large and nested loops can be closed, which is tricky when using particle filters. Moreover, the trajectory estimate can be revised upon arrival of new observations; past poses in a standard particle filter are never revised. However, we found that the graph-based SLAM approach can be sensitive to erroneous loop closures. That is the reason why in this paper we examine the suitability of particle filtering for trajectory estimation as an alternative.

With regard to active RFID tags, Kantor *et al.* [9] and Djugash *et al.* [10] utilized an extended Kalman filter for localization, mapping, and SLAM. Their methods exploit measured signal strength between the transponders, which is not a standardized feature in passive RFID systems. Fingerprinting has already been successfully applied to RFID-based localization [11], [12], [13] and is based on the observation that the position of a robot can be inferred from the comparison of the currently detected tags with a database of measurements obtained during training. The area of appearance-based SLAM (e.g., see [14], [15], [16]) is technically similar to fingerprinting-based trajectory estimation: Similarities to the visual signatures of previously visited places are found in order to close loops. This is done without the reconstruction of feature positions in world coordinates. By contrast, a trajectory reconstruction approach which relies on the fusion of forward and backward odometry only was presented by ten Hagen *et al.* [17].

### III. RFID AND FINGERPRINTS

Radio frequency identification (RFID) is a technique for the identification of objects via electromagnetic waves. An RFID system consists of an RFID reader with antennas and a number of small transponders (also called tags or labels)

attached to objects. The reader is frequently called to perform an *inquiry*, that is, all transponders within read range are requested to send back their unique IDs. Passive tags obtain their energy for the reply only from the electromagnetic field of the reader. Yet, they feature a read range of up to 7 m.

With RFID systems of this kind, only the *detection* of a transponder is possible, but its exact location inside the reader field cannot be determined. So, neither bearing nor distance to a tag are known. This is a great difference as compared to active RFID tags, where commonly signal strength measurements can be used to estimate distances.

Read attempts of passive tags may fail even if the tags are within read range, because detections depend on a number of parameters. Some factors are hard to include in predictions, for example absorption and the reflexion of the transmitted electromagnetic waves. They are accepted as undisclosed noise. But most importantly, the chance of successfully detecting a tag correlates with the tag’s relative position and orientation with respect to the reader antenna (Fig. 1). This property is one of the fundamentals of this work.

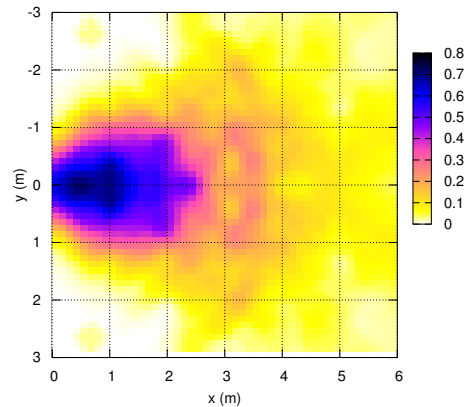


Fig. 1. Dependence of tag detection rates on the relative displacement ( $x,y$  coordinates only) of a tag with respect to the antenna of the RFID reader. The latter is located in the origin and points to the right.

Our method makes use of RFID *fingerprints*, which follow the usual notion of fingerprinting: Sensor measurements are associated with the positions where they are recorded. The idea behind this technique is that raw sensor data are considered to characterize the locations in which they are recorded. Consequently, if the lists of detected RFID tags in two fingerprints are similar, we will conclude that the fingerprints are likely to stem from similar positions. We focus on fingerprinting-based sensor models because they yielded good accuracy in self-localization [12], [13]. The snapshot approach by Schneegans [12] can be used, for instance, for localizing the robot against pose-corrected observations obtained with the method of the work at hand.

Formally, we represent an RFID fingerprint  $F$  by a pair  $F = (\mathbf{f}, \mathbf{x})$ .  $\mathbf{f} = (\mathbf{f}_{ant_1}, \dots, \mathbf{f}_{ant_A})$  is the result of an inquiry that was carried out for all  $A$  antennas connected to the RFID reader, and  $\mathbf{x} = (x, y, \theta)$  is the position where  $\mathbf{f}$  was recorded.  $\mathbf{f}_{ant_j} = (f_{ant_j}^{(1)}, \dots, f_{ant_j}^{(L)})$  is a binary vector in

which  $f_{ant_j}^{(l)}$  states whether antenna  $j$  detected tag  $l$  (1) or not (0).  $L$  is the total number of different tags observed.

A very important property of such fingerprints is that they are implicitly *orientation-dependent*. If the robot is equipped with more than one antenna and these antennas are mounted in different directions, measuring the same tags with the same antennas in two fingerprints means that the orientation of their recording positions must be similar. Consequently, without knowing the true orientation of the robot, there is a correlation between the difference in heading and the similarity of two fingerprints if they are compared in an antenna-wise manner.

In this work, we assume densely tagged environments, that is, the robot is able to detect several tags at a time from most positions. This assumption guarantees that RFID measurements are spatially expressive. In supermarkets filled with tagged products, this assumption is naturally satisfied.

#### IV. TRAJECTORY ESTIMATION WITH PARTICLE FILTERS

Trajectory estimation means to incrementally localize a robot traversing an unknown environment while maintaining consistency of the pose estimates based on the current and previous observations. In other words, the position of the robot is tracked without an a-priori map. This task resembles simultaneous localization and mapping (SLAM), where the goal is to incrementally build a map of an unknown environment while the robot is localizing itself against the created map. The challenge lies in the uncertainty of sensor readings and especially in noisy odometry; building a map when the trajectory of the robot is known is an easier subproblem. The SLAM problem itself is well-studied, has brought up a variety of approaches and can be regarded as solved from a theoretical perspective. A good overview is presented in [2], from which we have adopted some of the notations in the following section.

Trajectory estimation concerns the posterior density

$$p(\mathbf{X}_{0:t} | \mathbf{Z}_{0:t}, \mathbf{U}_{0:t}, \mathbf{x}_0), \quad (1)$$

of the trajectory  $\mathbf{X}_{0:t} = \{\mathbf{x}_0, \dots, \mathbf{x}_t\}$  (the history of robot poses  $\mathbf{x}_i$ ) at each discrete time step  $t$ , given the histories of sensor measurements  $\mathbf{Z}_{0:t}$  and robot movements  $\mathbf{U}_{0:t}$  as well as the initial pose  $\mathbf{x}_0$  (typically zero). In the 2D case (as in our case), the robot pose  $\mathbf{x}_t = (x_t, y_t, \theta_t)$  contains the 2D coordinates of the robot as well as a global heading  $\theta_t$ .  $\mathbf{Z}_{0:t} = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$  comprises all sensor readings and  $\mathbf{U}_{0:t} = \{\mathbf{u}_1, \dots, \mathbf{u}_t\}$  all odometry measurements.

A particle filter approximates (1) at time  $t$  by a set of  $N$  particles, where each particle is associated a trajectory  $\mathbf{X}_{0:t}^{(i)}$  and a weight  $w_t^{(i)}$ . (In a full SLAM approach such as FastSLAM [1], each particle would also feature a density  $p(\mathbf{m} | \mathbf{X}_{0:t}^{(i)}, \mathbf{Z}_{0:t})$  representing the map.) Note that particle filters are generally suited for estimating the state of a dynamic system with non-linear dynamics and non-Gaussian noise, as in the task at hand. The particle filtering algorithm takes the following steps in a recursive fashion:

- 1) *Prediction*: A new sample is drawn from the proposal distribution, which is the motion model in our work:

$$\mathbf{x}_t^{(i)} \sim p(\mathbf{x}_t | \mathbf{x}_{t-1}^{(i)}, \mathbf{u}_t) \quad (2)$$

The model probabilistically describes the transition of the system from the old state for a given odometry measurement  $\mathbf{u}_t$ .

- 2) *Correction*: Samples are reweighted according to the latest sensor readings:

$$w_t^{(i)} = \eta_t w_{t-1}^{(i)} p(\mathbf{z}_t | \mathbf{X}_{0:t}^{(i)}, \mathbf{Z}_{0:t-1}) \quad (3)$$

$p(\mathbf{z}_t | \mathbf{X}_{0:t}, \mathbf{Z}_{0:t-1})$  is the observation model and compensates for the discrepancy between the proposal and the true distribution. In Sect. V we dedicate to the derivation of an observation model  $p(\mathbf{z}_t | \mathbf{X}_{0:t}, \mathbf{Z}_{0:t-1})$  for RFID fingerprints.  $\eta_t$  ensures that  $\sum_{i=1}^N w_t^{(i)} = 1$ .

- 3) *Resampling*: Samples are drawn with replacement, where the probability of choosing a particle corresponds to its weight. By this step, samples representing inconsistent trajectories are removed. All new samples are given equal weights  $1/N$ . An option is to resample not always, but only if the estimate  $\hat{N}_{eff} = 1 / \left( \sum_{i=1}^N (w_t^{(i)})^2 \right)$  of the *effective sample size* falls below some threshold, e.g.,  $N/2$ . Generally, resampling should be performed as rarely as possible in order to maintain particle diversity.

SLAM particle filters would also require a forth step, the map update, in which landmark positions are updated, given the estimated robot pose. Our approach does not require landmark updates, because the likelihood  $p(\mathbf{z}_t | \mathbf{X}_{0:t}, \mathbf{Z}_{0:t-1})$  used in (3) directly operates on  $\mathbf{X}_{0:t}$  and  $\mathbf{Z}_{0:t}$ . On the one hand, this means that computation time with respect to map updates is saved. On the other hand, the time complexity for correcting the particle weights in step 2 becomes time-dependent: FastSLAM, for instance, has reweighting costs of  $\mathcal{O}(N \cdot L)$  for  $N$  particles and  $L$  landmarks; the costs in our approach are  $\mathcal{O}(t \cdot L' + N \cdot s_{max})$ , where  $t$  is the current time index and  $s_{max}$  is a constant which determines how many previous fingerprints are used for reweighting the samples. Consequently, the computation time will theoretically grow over time. However,  $L' \ll L$  is a nearly constant value which reflects that, due to technical reasons, each RFID inquiry only concerns a bounded number of landmarks. Moreover, not all  $t$  measurements have to be considered in likelihood computation. In Sect. VI we actually show that the asymptotic behavior seems irrelevant in practice, although further studies on this issue may be subject to future research.

If desired, the locations of RFID tags can be estimated later, following the particle filter-based mapping method by Hähnel *et al.* [3] or similar approaches using a fuzzy sensor model [18] or a multi-path propagation model including specified antenna characteristics [19]. As can be seen in Fig. 1, the observation model for the detection of a single RFID tag is neither Gaussian nor sharply peaked. This is the reason why one cannot simply use EKFs as in FastSLAM,

estimate the positions of the landmarks (RFID tags), and localize against them simultaneously.

## V. OBSERVATION MODEL FOR FINGERPRINTS

In this section, we design a likelihood function  $p(\mathbf{z}_t | \mathbf{X}_{0:t}, \mathbf{Z}_{0:t-1})$  that relies on statistics of the similarity of RFID measurements and their spatial distribution. For this purpose, we elaborated a non-parametric model which is based on fingerprints recorded during a training phase. We first depict how training data are acquired, and then we explain the derivation of the likelihood function used in the particle filter.

### A. Acquisition of Training Data

The acquisition of training data is performed as follows: RFID tags are placed in the environment – arbitrarily, but similar in density to the arrangement in the application environment in which trajectory estimation is to be performed. That is, for instance, the tags should be attached to walls at different heights if the target environment will later also comprise transponders at varying heights, e.g., on products in shelves. Then, the robot should be located at various known positions, i.e., at different distances and with different relative orientations to the tags. RFID measurements are recorded and annotated with their true recording positions. The outcome of this training phase is a set  $\{F_i\}_{i=1,2,\dots}$  of fingerprints as defined in Sect. III.

### B. Derivation of an Observation Model for Fingerprints

In order to estimate the robot’s trajectory from RFID and odometry data only, any kind of observation model must exploit the fact that the detection rates of RFID tags strongly depend on the distance and relative orientation between the tags and the reader antenna. In prior experiments, we investigated different similarity measures for comparing two RFID fingerprints. In particular, we examined the relationship between the measured similarity on the one hand and the mean distance between the positions at which they were taken on the other hand. We observed that the *number of equal tag IDs* in two measurements, summed up over all antennas, yielded one of the highest values of (absolute) Pearson correlation and also mutual information, as also compared to cosine similarity (cf. [8]) and histogram intersection. Due to this result, the simple parameterizability of the model, and the very efficient computability, we decided to use the number of equal identifiers as a measure for our algorithm.

The number of equal tags in two fingerprints  $\mathbf{f}_i$  and  $\mathbf{f}_j$  is computed as the sum of detections of tags  $l$  that were detected (with the same reader antenna  $a$ ) both at times  $i$  and  $j$ :

$$EQU(\mathbf{f}_i, \mathbf{f}_j) = \sum_{a=1}^A \left| \left\{ l \mid f_{i,ant_a}^{(l)} = 1 \wedge f_{j,ant_a}^{(l)} = 1 \right\} \right| \quad (4)$$

Let  $Z$  be the random variable which represents the number of equal tags in two inquiries. Furthermore, let  $d$  be the random variable which denotes the Euclidean distance  $\|\mathbf{x}_i - \mathbf{x}_j\|$  between the positions where two fingerprints were taken.

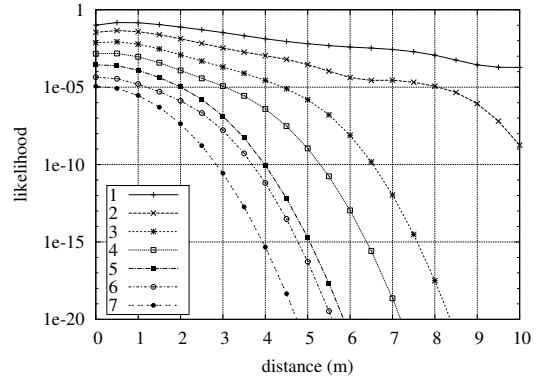


Fig. 2. Learned likelihood functions  $p(Z = k|d)$  depending on the distance  $d$  and for different numbers (1-7) of equal tags in two compared RFID fingerprints. Please note the logarithmic scale of the vertical axis.

Then  $p(Z = k|d)$  stands for the likelihood of having  $k$  equal tags in two fingerprints recorded at a given distance  $d$ . In order to compute this likelihood, we apply Bayes’ formula:

$$p(Z = k|d) = p(d|Z = k) \frac{p(Z = k)}{p(d)} \quad (5)$$

The term  $p(Z = k)$  can directly be estimated from the training data by simple counting how often which number of equal tags in any pair of training measurements is observed. The terms  $p(d)$  and  $p(d|Z = k)$ , however, require density estimation, i.e., the estimation of probability densities, based on the discrete number of sample distances. In our implementation, we employ *kernel density estimation*, which is a non-parametric means of estimating the (arbitrarily shaped) probability density function of a random variable (e.g., [20]). For a Gaussian kernel, as in our implementation, the optimal kernel bandwidth can be determined via the normal reference rule  $h = 1.06\sigma n^{-1/5}$ , where  $n$  is the number of input distances and  $\sigma$  is the standard deviation among them. While  $p(d)$  is estimated from all pairs of training fingerprints, the estimation of  $p(d|Z = k)$  for a fixed  $k$  only involves pairs of training fingerprints which have exactly  $k$  tag identifiers in common. Fig. 2 shows the likelihood functions which we estimated from our experiments for different numbers of equal tags.

In the recall phase,  $p(Z = k|d)$  can then be used to compute  $p(Z = EQU(\mathbf{f}_i, \mathbf{f}_j) | \|\mathbf{x}_i - \mathbf{x}_j\|)$  for a single pair of fingerprints,  $\mathbf{F}_i$  and  $\mathbf{F}_j$ . In the particle reweighting step at time  $t$ , we first compare  $\mathbf{z}_t := \mathbf{f}_t$  to all  $\mathbf{z}_j := \mathbf{f}_j, 0 \leq j < t$ , by computing  $e_{tj} := EQU(\mathbf{f}_t, \mathbf{f}_j)$ . We claim that  $e_{tj} \geq e_{min}$  for some threshold  $e_{min}$ , because only if sufficiently many tags are re-detected, sufficient evidence of revisiting the same place is provided. So, if  $e_{tj} \geq e_{min}$ , measurement  $j$  is a candidate to be used for reweighting. Since it may happen that in some areas of the environment there are many similarity candidates whereas in others there are only few, it is reasonable to bound the number of candidates in order to enforce homogeneity in assigned likelihoods. This is achieved by first computing a set of candidates  $S(\mathbf{z}_t, \mathbf{Z}_{0:t-1})$  as described before and then choosing the best  $s_{max}$  measurements of highest similarity to  $\mathbf{z}_t$ . The algorithm

**Input:** Current and recent RFID measurements  $\mathbf{z}_t, \mathbf{Z}_{0:t}$   
**Output:** The set  $S(\mathbf{z}_t, \mathbf{Z}_{0:t-1})$  of similarity candidates

```

 $S \leftarrow \{\}$ 
for  $j \leftarrow 0$  to  $t-1$  do
   $e_{tj} \leftarrow \text{EQU}(\mathbf{z}_t, \mathbf{z}_j)$  // number of equ. tags in  $\mathbf{z}_t, \mathbf{z}_j$ 
  if  $e_{tj} \geq e_{min}$  then
     $S \leftarrow S \cup \{(e_{tj}, j)\}$  // similarity candidate
  end
end
if  $|S| > s_{max}$  then
  sort  $S$  descending w.r.t. to first tuple component
  trim  $S$  to first (i.e., best)  $s_{max}$  elements
end
return  $S$ 

```

Fig. 3. Computation of  $S(\mathbf{z}_t, \mathbf{Z}_{0:t-1})$ , which contains indices of RFID measurements similar to  $\mathbf{z}_t$  and the number of equal tag IDs.

in Fig. 3 summarizes these steps. Finally, all particles are reweighted according to the chosen similar measurements and based on the trajectories  $\mathbf{X}_{0:t}^{(i)}$  that the particles embody. Here, we assume that the likelihoods of all similarities are independent and compute the product of single likelihoods:

$$p(\mathbf{z}_t | \mathbf{X}_{0:t}^{(i)}, \mathbf{Z}_{0:t-1}) = \prod_{\substack{(k,j) \in \\ S(\mathbf{z}_t, \mathbf{Z}_{0:t-1})}} p(Z = k | \|\mathbf{x}_j^{(i)} - \mathbf{x}_t^{(i)}\|)$$

Let us conclude this section with the remark that our model could also be easily extended to orientations, rather than distances only. Yet, although we do not weight a particle based on its current orientation, its orientation is implicitly assessed: Two particles with identical locations but with different orientations will end up in distinct positions if the robot travels some distance. On the other hand, if a pair of fingerprints along the robot’s path has actually been taken in same place, but with different orientations of the robot, the RFID measurements will be likely to differ entirely. This will prevent the fingerprint pair from being selected as a similarity candidate and hence prevent a particle representing the same position from erroneously being assigned a low weight.

## VI. EXPERIMENTS

We conducted several experiments with a B21 robot. The robot is equipped with a laser range finder (240° field of view), an Alien Technology ALR-8780 UHF RFID reader (EPC Class 1 Gen. 2, 2 W EIRP transmitting power), and two pairs of RFID antennas, which point at angles of approx. 45° to either side of the robot (Fig. 4).

In the training phase, we collected RFID data and laser-based reference positions as described in Sect. V-A. The training environment was a corridor (28 m × 3 m) in which we attached 240 UHF RFID tags to the walls at distances of 0.25 m on average, roughly at the height of the upper antennas. We steered the robot manually on arbitrary paths along the corridor. On a trajectory length of 1210 m corresponding to a duration of 68 minutes, 8123 fingerprints were taken in total. The recorded data were used to compute the fingerprint model shown in Fig. 2, which took a few seconds only.

In order to investigate the quality of trajectory estimation, we attached more than 400 tags to walls and furniture at

different heights in a different environment (Fig. 4). The traversable area was approx. 195 m<sup>2</sup>. We recorded laser-based reference positions (only for ground truth), odometry, and RFID data on 13 trajectories. To be able to test the loop closing capability of our approach, all paths contained loops and ended in the same position where they had started. In this, the deviation between the initial and final positions of the robot is a good measure of accuracy, since odometric errors sum up and typically lead to a large final position deviation. The mean and maximal deviations over all test trajectories dead-reckoned by pure odometry were 3.40 m and 7.63 m, respectively. As a better measure of the consistency of the estimated trajectory, we computed the mean residual Cartesian error between the estimated positions associated with the most likely particle and the actual positions along the entire path after an alignment at each other. The alignment minimizes errors, but is useful to measure the consistency of the generated trajectory: The coordinate frame of the reference positions is different from the initial frame of the particles, and a particle’s trajectory can be quite consistent even if it is initially distorted. The mean residual Cartesian error with odometry alone was 1.23 m. Finally, we always used a very low resampling threshold of 5% of the effective sample size in order to avoid particle impoverishment.

### A. Parameter Choice for Similarity Candidates

In a first series of experiments, we investigated the influence of the choice of the parameters  $e_{min}$  (minimal number of equal tags) and  $s_{max}$  (number of similarity candidates used for reweighting). The particle filter was run ten times with each configuration and a fixed number of  $N = 1000$  particles. The results are visualized in Fig. 5. The first observation is that the larger  $e_{min}$  (the minimal number of equal tags of two fingerprints related to each other) the better the accuracy. This can be explained by the fact that only similarity candidates are selected which strongly indicate that a place is revisited. Furthermore, the number of similarity candidates,  $s_{max}$ , should be chosen carefully. That is, a smaller number seems to be beneficial. An explanation

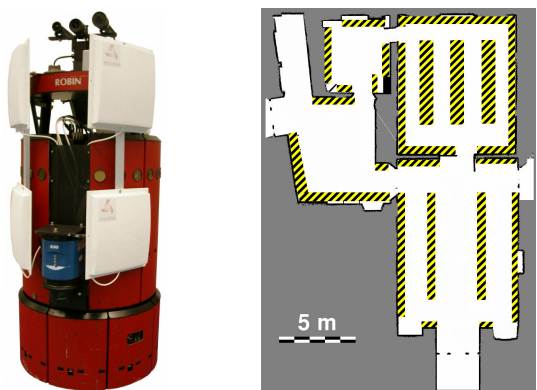


Fig. 4. Left: Our robot with its UHF RFID antennas (white), spanning an angle of approx. 90°. Right: The environment in which we conducted our experiments. RFID tags were situated in the hatched areas.

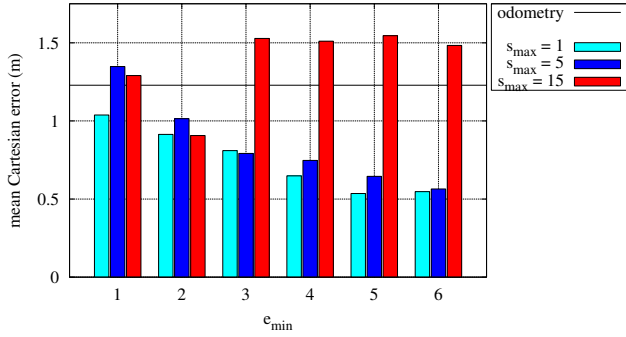


Fig. 5. Mean residual Cartesian errors of the estimated trajectories, depending on the different parameters

TABLE I

IMPACT OF THE NUMBER OF PARTICLES ( $N$ ) ON THE MEAN RESIDUAL CARTESIAN ERROR AND THE FINAL POSITION DEVIATION OF THE ROBOT ALONG THE ESTIMATED TRAJECTORY (IN M)

$N$	Residual Cartesian error		Final deviation
	Mean $\pm$ std. dev. (in m)	Maximum (in m)	Mean $\pm$ std. dev. (in m)
30	$0.7662 \pm 0.2650$	1.4089	$1.7016 \pm 0.9211$
100	$0.5987 \pm 0.1717$	0.9952	$1.0555 \pm 0.6130$
300	$0.5189 \pm 0.1430$	0.9896	$0.7995 \pm 0.4429$
1000	$0.4706 \pm 0.1492$	0.8362	$0.5455 \pm 0.3599$
3000	$0.4242 \pm 0.1430$	0.8089	$0.4192 \pm 0.2635$

for this is that for larger  $s_{max}$ , the likelihood function  $p(\mathbf{z}_t | \mathbf{X}_{0:t}^{(i)}, \mathbf{Z}_{0:t-1})$  separates particle weights too drastically. The smallest mean deviation in this experiment was 0.54 m for  $e_{min} = 5$  and  $s_{max} = 1$ . Thus, compared to the mean residual error of 1.23 m for odometry only, the estimated trajectories are significantly improved.

### B. Number of Particles

The impact of the number of particles,  $N$ , on the accuracy of the estimated trajectory was examined in the second series of experiments. Again, we ran the particle filter ten times on five selective data sets. We set  $s_{max} = 1$  and  $e_{min} = 5$ , based on the findings of the previous experiments. The results are listed in Tab. I. Not surprisingly, the largest number of particles achieves the best accuracy, in this case a residual error of less than half a meter. On the other hand, the final error is quite large for 30 particles, resulting in an inconsistent trajectory. An example of an estimated trajectory obtained with 1000 particles is shown in Fig. 6.

### C. Variation of Motion Noise

In the last series of experiments, we investigated the stability of the proposed approach under the variation of rotational noise in the robot's movements. The motivation behind this is that SLAM and localization are typically highly sensitive to rotational errors. We therefore added zero-mean Gaussian noise with standard deviation  $\lambda \cdot \Delta\theta$  to the rotational component  $\Delta\theta$  of odometry measurements and varied the factor  $\lambda$ . Without the addition of artificial noise, we assumed Gaussian rotational noise with a standard

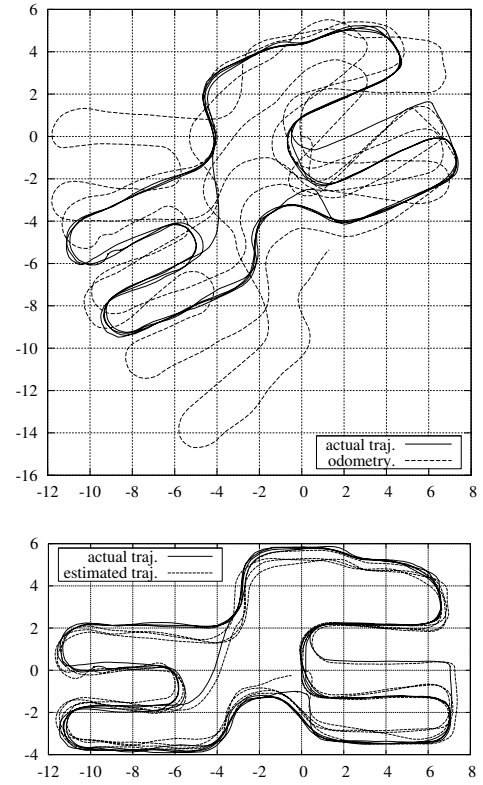


Fig. 6. Top: Actual trajectory vs. raw odometry data. Bottom: Actual trajectory vs. sample estimated trajectory. It was generated with 1000 particles and has a mean residual Cartesian error of 0.3975 m.

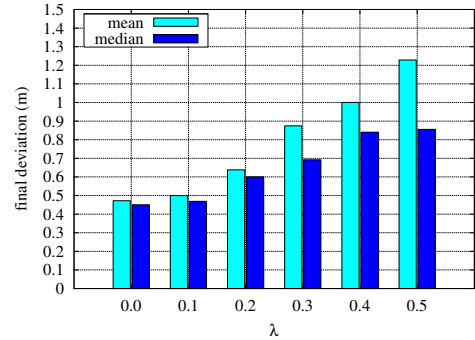


Fig. 7. Mean final deviations for increased motion noise

deviation of  $0.05 \cdot \Delta\theta$ . The results are shown in Fig. 7. For moderate rotational noise, the residual Cartesian errors increase only slightly, while for  $\lambda \geq 0.3$ , the accuracy is clearly worse. We draw the conclusion that our results can be generalized to other types of robots with similar or even slightly larger odometric errors.

### D. Run Times

While running the experiments of Sect. VI-B, we measured the run-times of our algorithm. Table II lists the mean durations of the last 100 correction steps of the particle filter on a 3 GHz processor. Although the asymptotic run-time complexity of the algorithm depends on the trajectory length, the number of particles obviously has a much larger influence in practice. All computations can be performed by far in real



TABLE II

MEAN RUN-TIMES OF THE FINAL 100 CORRECTION STEPS OF THE PARTICLE FILTER, CONDITIONED ON THE NUMBER OF PARTICLES ( $N$ )

Number of particles	30	100	300	1000	3000
Mean duration	0.9 ms	1.0 ms	1.1 ms	1.6 ms	3.4 ms

time, and RFID measurements arrive approx. every 0.7 s. The extrapolation of the measured times indicates that only after more than 24 hours of continuous autonomous exploration the robot would not be able to process sensor data in time. An oblivion mechanism for too old RFID measurements, however, would be an example of a simple, effective solution.

## VII. CONCLUSION

In this paper, we presented a solution to trajectory estimation by incorporating passive UHF RFID measurements and odometry only. First, a sensor model of relating two RFID measurements (fingerprints) from different positions is learned. Transponders are permitted to be spread arbitrarily over the environment; the given tag infrastructure is utilized. The only requirement is that the robot should be able to detect several tags from each position. The learned model is used in the correction step of a particle filter in order to estimate the trajectory of the robot, based on the estimated positions of previous fingerprints.

The presented approach permits a robot to estimate its trajectory without prior map in an RFID-tagged environment, using particle filtering. Hence, RFID-based mapping is possible without an extra reference localization system. This is promising insofar as in the densely tagged environments of the future (e.g., supermarkets and storehouses), one will be able to map the places of labeled products without reference positioning system. In our experiments, we achieved a mean accuracy of approx. 0.5 m. This is almost the same as for our graph-based approach [8]. This result also shows, however, that RFID-based SLAM is still far from the accuracies known from laser-based SLAM which allows for highly consistent maps with few particles only. Due to the low measurement rate (less than 2 Hz in our experiments), a detection field radius of up to 7 m as well as missing distance and bearing information, RFID-based trajectory estimation will probably always suffer from increased uncertainty. Yet, we think that it is worth developing more complex observation models in order to investigate and improve its accuracy. Of course, an option is to incorporate other (preferably low-cost) sensors or to augment particle filter-based SLAM based on other sensors with our method when data association is crucial.

In this work, we did not address moving tags, which is an issue in the target scenarios. The particle filtering framework and the observation model already reveal some robustness to small numbers of tag relocations. As future work, however, we are going to treat moving tags, too.

Finally, a valuable future extension to our approach is to incorporate models of different RF power levels in inquiries,

because different levels allow for switching between large-area inventory scans and short-range measurements with lower position uncertainty.

## REFERENCES

- [1] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the AAAI National Conference on Artificial Intelligence*. Menlo Park, CA, USA: American Association for Artificial Intelligence, 2002, pp. 593–598.
- [2] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–108, June 2006.
- [3] D. Hähnel, W. Burgard, D. Fox, K. P. Fishkin, and M. Philipose, "Mapping and localization with RFID technology," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA)*, 2004, pp. 1015–1020.
- [4] A. Kleiner, C. Dornhege, and S. Dali, "Mapping disaster areas jointly: RFID-coordinated SLAM by humans and robots," in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR)*, Rome, Italy, 2007.
- [5] K. Tanaka, "Multiscan-based map optimizer for RFID map-building with low-accuracy measurements," in *Proceedings of the 19th International Conference on Pattern Recognition (ICPR)*, December 2008, pp. 1–4.
- [6] E. Olson, J. Leonard, and S. Teller, "Fast iterative optimization of pose graphs with poor initial estimates," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA)*, May 2006, pp. 2262–2269.
- [7] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard, "A tree parameterization for efficiently computing maximum likelihood maps using gradient descent," in *Robotics: Science and Systems III*, Atlanta, GA, USA, June 2007.
- [8] P. Vorst, B. Yang, and A. Zell, "Loop closure and trajectory estimation with long-range passive RFID in densely tagged environments," in *Proceedings of the 14th International Conference on Advanced Robotics (ICAR)*, Munich, Germany, June 2009.
- [9] G. Kantor and S. Singh, "Preliminary results in range-only localization and mapping," in *Proceedings of the IEEE Conference on Robotics and Automation (ICRA)*, May 2002.
- [10] J. Djugash, S. Singh, and P. I. Corke, "Further results with localization and mapping using range from radio," in *International Conference on Field and Service Robotics (FSR)*, ser. Springer Tracts in Advanced Robotics, vol. 25. Springer, July 2005, pp. 231–242.
- [11] A. Lim and K. Zhang, *Advances in Applied Artificial Intelligence*, ser. Lecture Notes in Computer Science. Springer, 2006, vol. 4031, ch. A Robust RFID-Based Method for Precise Indoor Positioning, pp. 1189–1199.
- [12] S. Schneegans, P. Vorst, and A. Zell, "Using RFID snapshots for mobile robot self-localization," in *Proceedings of the 3rd European Conference on Mobile Robots (ECMR)*, Freiburg, Germany, September 2007, pp. 241–246.
- [13] P. Vorst, S. Schneegans, B. Yang, and A. Zell, "Self-localization with RFID snapshots in densely tagged environments," in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nice, France, September 2008, pp. 1353–1358.
- [14] R. M. Eustice, H. Singh, and J. J. Leonard, "Exactly sparse delayed-state filters for view-based SLAM," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1100–1114, 2006.
- [15] P. Newman, D. Cole, and K. Ho, "Outdoor SLAM using visual appearance and laser ranging," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2006, pp. 1180–1187.
- [16] P. E. Rybski, S. I. Roumeliotis, M. Gini, and N. Papanikolopoulos, "Appearance-based minimalistic metric SLAM," in *Proceedings of the International Conference on Robotics and Intelligent Systems (IROS)*, Las Vegas, USA, 2003, pp. 194–199.
- [17] S. ten Hagen and B. Kröse, "Trajectory reconstruction for self-localization and map building," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 2002, pp. 1796–1801.

- [18] A. Milella, G. Cicirelli, and A. Distanto, "RFID-assisted mobile robot system for mapping and surveillance of indoor environments," *Industrial Robot: An International Journal*, vol. 35, no. 2, pp. 143–152, 2008.
- [19] T. Deyle, C. C. Kemp, and M. S. Reynolds, "Probabilistic UHF RFID tag pose estimation with multiple antennas and a multipath RF propagation model," in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nice, France, September 2008, pp. 1379–1384.
- [20] D. W. Scott and S. R. Sain, *Handbook of Statistics*. Elsevier, Amsterdam, 2004, vol. 23: Data Mining and Computational Statistics, ch. Multi-Dimensional Density Estimation.