

Exploiting Space-time Correlations in an RFID Tag Field for Localization and Tracking

Victor K.Y. Wu* and Nitin H. Vaidya
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Email: {vwu3, nhv}@illinois.edu

Abstract—We consider exploiting space-time correlations of passive RFID tags for localization and tracking. Consider tags with storage memory distributed densely in space. Users move through the field, scanning the tags. Subsets of these tags are space-time correlated with respect to the user’s trajectory. For example, if two adjacent tags are scanned close together in time, they are highly correlated. We can represent this correlation value as a weighted virtual arrow pointing from the earlier scanned tag to the later scanned tag. A user continuously calculates these arrows, and stores them locally in the tags, forming a digital path, for localization and tracking of herself by other users. We specify several of these correlation functions and evaluate them experimentally. We provide statistical measures of performance, as well as visualizations of the resulting arrow fields created by users.

I. INTRODUCTION

In this paper, we propose an RFID-based localization and tracking scheme. Consider stationary passive RFID tags with storage memory distributed densely in space. For example, they could be embedded beneath the floor tiles of a large warehouse or inside the barks of trees in a national park. Users equipped with RFID readers move through the tag field, continuously scanning the tags. As a user moves, stationary tags enter and exit her scan range (relative to her position). A subset of tags are said to be *space-time correlated* with respect to the user’s trajectory. The correlation value is high if the tags are close together in space, and are scanned by the user close together in time. The correlation also has a direction, indicating the movement of the user in that space-time locality. This correlation value and direction form a “virtual arrow”. The value (arrow weight) is stored in the tags themselves, and the direction is stored by writing a tag ID in another tag. These weighted arrows form a “digital path” of the user for localization and tracking. In this paper, we focus on different ways to calculate the correlation. We evaluate them experimentally using performance metrics and visualizations.

The rest of the paper is organized as follows. In Section II, we discuss related work and how our scheme compares. In Section III, we present our scheme in the context of a system description. We also specify several correlation functions. In Section IV we evaluate our scheme experimentally. Finally, we conclude in Section V and provide future work.

II. RELATED WORK

A. Localization and tracking

Localization is the estimation of the positions of mobile entities (e.g. people or objects) to a required level of accuracy. Tracking is determining these positions over time, either in real-time, or after the fact (historical data). In [1], the authors use RFID readers and active tags (serving as reference points) to locate passively tagged objects. In [2], [3] the authors use mobile robots to build maps and track mobile objects. The authors in [4] propose deploying passive tags in space over large areas in a highly dense and redundant fashion. Vehicles can leave traces by writing IDs to tags, allowing them to be retraced in the future. The authors in [5] track conference attendees using a system called IntelliBadge. Attendees are each assigned a uniquely identifying RFID tag upon registration, and can access a variety of applications at web kiosks. The authors in [6] use RFID to locate objects in a 3D space, such as a shipping container

Since [1] uses stationary readers, it is constrained to relatively small spaces, such as inside an office building. Conversely, our scheme is similar to [2], [3], [4] in that we deploy passive tags over a large area. In [2], [3], robots create centralized maps for localization and tracking. In our scheme, as users move through the tag field, they only store space-time correlation information locally. The scheme is entirely distributed and tag locations do not need to be known in general. Our scheme also benefits from easy deployment and maintenance. For example, if we require high tracking accuracy, the spatial tag density should be higher. If we require high reliability (against random tag failures or external damage (e.g. a fire)), we can again deploy more tags per area or use more reliable tags.

B. Search and rescue

Search and rescue (SAR) can be viewed as a specific case of localization and tracking, since we want to locate people as quickly as possible, bringing them to safety. Tracking is a secondary goal that is only beneficial (which is often the case) if it aids in the primary goal. For example, safety authorities in Texas provide wristbands with passive tags to evacuees in times of emergency [7]. The evacuees are tracked (officials scan them) as they move into and out of emergency

*This work is supported in part by NSF grant CNS-0519817.

reception centers. If an evacuee is lost, SAR can begin in the last known scan point of that evacuee. In Virginia, patients suffering Alzheimer's (or similar diseases) wear wristbands, each with an active RFID chip emitting a unique identifier [8]. If a patient wanders away, searchers use RFID readers tuned to the patient's identifier to quickly locate her.

Our scheme is suitable in a large area where passive tags can be distributed in space. For example, in [9], we describe a system where tags are embedded in the barks of trees. Users store digital paths as they move through the space. If a user becomes lost, searchers use her digital path as a guide to locate her.

III. SYSTEM DESCRIPTION

A. System model

Consider a large set of RFID tags \mathcal{T} distributed in space. Each tag has position \mathbf{p}_i , where $i \in \mathcal{T}$. We assume each tag has a large storage memory (relative to the application requirements). (The TegoChip has 32 kilobytes [10].) That is, our system is not storage-limited.¹ Each tag is identified by a unique ID u_i , which is stored in the tag itself. \mathbf{p}_i may also be stored in the tag. Users, each equipped with one RFID reader, move through the tag field and scan the tags, reading and writing to them. The set of users is \mathcal{R} .

B. Space-time correlation functions

Users exploit the space-time correlations of scanned tags as they move through the tag field according to various trajectories. For example, if a user with a small scan range scans two tags within a short time period, they are highly correlated to each other in both space and time (with respect to that user at that point in her trajectory). We can then store a "virtual arrow" with a large weight in the two tags, pointing from the earlier scanned tag to the later scanned tag. At a later time, if another user encounters this arrow, she can confidently guess that the previous user moved in the arrow's direction at this position in the past.

The positions of tags may be stored in the tags themselves. Or, users may have a map $M : u_i \rightarrow \mathbf{p}_i$. In either case, users can determine a tag's position by scanning its ID. If the positions are unavailable, a user may estimate relative tag positions $\hat{\mathbf{p}}_i$ (distance between tags), for example, using her scan range. Alternatively, tag positions may not be used at all. Assuming users are continuously scanning tags, let $t_{i,r}^{(in)}$ and $t_{i,r}^{(out)}$ be the times when tag i enters and exits the scan range of user r during her trajectory, respectively, where $r \in \mathcal{R}$. Note that tags are assumed stationary, and users are mobile. Envisioning tags moving into and out of users' scan ranges provides greater insight to the situation. (For simplicity, we assume a user encounters a tag only once in her trajectory. Otherwise, we would require enter and exit times for each encounter.) In general, user r progressively acquires these space and time values (i.e. \mathbf{p}_i , $t_{i,r}^{(in)}$ and $t_{i,r}^{(out)}$) as she moves, which can be mapped through a correlation function \mathbf{f} , into weighted directions ("arrows"). These arrows are stored in the

tags, to be queried later possibly by other users. This general setup allows for a large class of correlation functions. That is, a user can use her entire history of tag scans to calculate many different correlations. However, for practical implementation's sake, we consider six correlation functions, where at time t , user r only uses the set of tags being currently scanned, $\mathcal{T}(t)$. (That is, these are the tags within her current scan range.) There are three different possible weightings and two different possible directions for the correlation functions. The weightings are

$$w^{(equ)} = 1, w^{(num)} = |\mathcal{T}(t)|, \text{ and } w^{(inv)} = \frac{1}{\max_{i,j} (t_{i,r}^{(in)} - t_{j,r}^{(in)})}. \quad (1)$$

$w^{(num)}$ gives more weight to arrows when more tags are being scanned. $w^{(inv)}$ gives more weight to arrows when the tags being scanned are "closer" in time. (That is, the weight is inversely proportional to the maximum entrance time difference of tags currently being scanned.) Intuitively, these latter two weightings put more emphasis on arrows when tags are relatively denser in space and time. The directions are

$$\mathbf{d}^{(new)} = \mathbf{p}_k - \mathbf{p}_l, \mathbf{d}^{(cen)} = \mathbf{p}_m - \mathbf{p}_l, \quad (2)$$

where $k = \arg \max_i t_{i,r}^{(in)}$, $l = \arg \min_i t_{i,r}^{(in)}$,
and $\mathbf{p}_m = \frac{1}{|\mathcal{T}(t)|} \sum_{i \in \mathcal{T}(t)} \mathbf{p}_i$.

$\mathbf{d}^{(new)}$ points from the oldest tag to the newest tag in $\mathcal{T}(t)$ that entered the user's scan range. Intuitively, this estimates the instantaneous direction of the user's trajectory. However, since new tags encountered by the user may not be directly on her trajectory, $\mathbf{d}^{(cen)}$ provides another estimate of the instantaneous direction. $\mathbf{d}^{(cen)}$ points from the oldest tag to the centroid of tags (spatial average). The six correlation functions are

$$\mathbf{f}_1(t) = w^{(equ)} \frac{\mathbf{d}^{(new)}}{\|\mathbf{d}^{(new)}\|}, \mathbf{f}_2(t) = w^{(equ)} \frac{\mathbf{d}^{(cen)}}{\|\mathbf{d}^{(cen)}\|} \quad (3)$$

$$\mathbf{f}_3(t) = w^{(num)} \frac{\mathbf{d}^{(new)}}{\|\mathbf{d}^{(new)}\|}, \mathbf{f}_4(t) = w^{(num)} \frac{\mathbf{d}^{(cen)}}{\|\mathbf{d}^{(cen)}\|} \quad (4)$$

$$\mathbf{f}_5(t) = w^{(inv)} \frac{\mathbf{d}^{(new)}}{\|\mathbf{d}^{(new)}\|}, \mathbf{f}_6(t) = w^{(inv)} \frac{\mathbf{d}^{(cen)}}{\|\mathbf{d}^{(cen)}\|}. \quad (5)$$

(The directions are normalized to have unit length so that the weight of each correlation function comes only from (1)).

In general, a user can continuously calculate correlations in time as she moves along her trajectory, and write them to tags. This is difficult to implement and requires heavy computing resources (such as memory in the reader and large storage in tags). Our correlation functions allows us to discretize the problem to a smaller dimensionality and provides a practical implementation method. In particular, every time a tag enters or exits a user's scan range, she calculates a correlation (represented as an arrow) and writes it to the tags. Specifically, the arrow weight is written to at least one of the tags. For $\mathbf{d}^{(new)}$, the arrow direction can be stored by writing tag IDs in the appropriate tags. In the simplest case, if the arrow is pointing from tag i to tag j , where $i, j \in \mathcal{T}(t)$, then the user can store u_j in tag i . Note that this eliminates the need for an explicit coordinate system. Implementation-wise, this means

¹In [9], we consider a system where tags are storage-limited. Users thus may choose to replace existing information on a tag. In that work, we investigate such replacement algorithms.

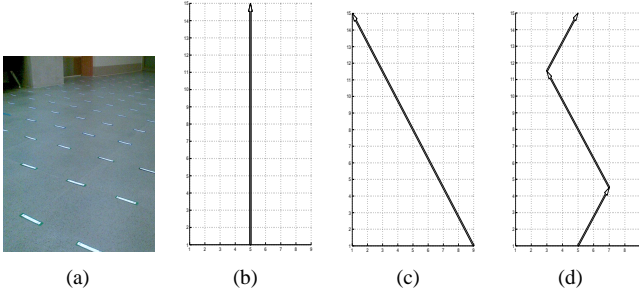


Fig. 1. (a): RFID tag field. (b), (c), (d): Tag grid and trajectories. Each vertex of the 8 feet by 14 feet grid has a tag. The outlined arrows indicate the trajectory.

$\mathbf{p}_i, \forall i \in \mathcal{T}$ does not need to be known by users at all, and can be fully eliminated from the system. However, $\mathbf{d}^{(cen)}$ does require some notion of tag positions in order for the user to calculate the spatial centroid of tags.

IV. EXPERIMENTAL EVALUATION

A. Setup

We perform experiments using the Alien ALR-9650 RFID reader with an ALR-9611-CR external antenna, and ALL-9650-02 passive tags. These are UHF EPC Gen 2 compliant hardware. Our RFID field consists of 135 tags positioned on the vertices of an 8 feet by 14 feet grid, as shown in Fig. 1(a). We move the reader through the grid according to one of three trajectories (Figs. 1(b), 1(c), 1(d)), while the reader continuously scans tags and records scan times. We take the first and last scan time of a tag as the entrance and exit time, respectively, as explained in Section III-B. For each trajectory, we use three RF power levels readily available on the Alien reader. P_3 is the maximum power, $P_2 = P_3 - 2$ dB, and $P_1 = P_3 - 4$ dB. (The reader outputs a maximum of 4 Watts EIRP. The external antenna is circular polarized with maximum gain of 6 dBi. It's 3 dB beamwidth is 40 degrees.) For each of these nine configurations (trajectory and power level), we repeat the experiment five times.

B. Results

1) *Localization and tracking*: For localization and tracking, we estimate a user's entire trajectory. Therefore, a performance measure is the average error of an arrow with respect to the trajectory. That is, we first calculate the arrows in each experiment run. For each point on each arrow, we find the closest distance to the trajectory (perpendicular distance). We then integrate this distance over the entire arrow. This is the arrow error. We average this over all arrows using the arrow weights (normalized), and then average it again over the five experiment runs. Results are shown in Fig. 2 for the six correlation functions in Section III-B. We see that in all three trajectories, $\mathbf{f}_2, \mathbf{f}_4$, and \mathbf{f}_6 (corresponding to $\mathbf{d}^{(cen)}$) all perform better than $\mathbf{f}_1, \mathbf{f}_3$, and \mathbf{f}_5 (corresponding to $\mathbf{d}^{(new)}$). $\mathbf{d}^{(new)}$ points from the oldest tag to the newest tag in the user's scan range, while $\mathbf{d}^{(cen)}$ points from the oldest tag to the centroid of tags. Thus, we see that the latter strategy gives a more

accurate estimate of the instantaneous direction of the user. At low power, $\mathbf{f}_2, \mathbf{f}_4$, and \mathbf{f}_6 perform similarly. But at high power (P_3), we see that \mathbf{f}_6 gives the best performance. Thus, using the inverse of the maximum entrance time difference is a good strategy for weighting the arrows.

Another performance measure is how effective arrows are at pointing towards the trajectory. For each arrow, we calculate the change in error (perpendicular distance to trajectory) from the arrow tail to the arrow head, normalized by the arrow tail error. This is averaged over all arrows using weights (normalized) and the five experiment runs. Note that a positive value means that arrows are moving away from the trajectory on average and a negative value means that arrows are moving towards it. Thus, the lower (towards negative infinity) the change in error, the better the arrows are at guiding later users to track the original user's trajectory. Results are shown in Fig. 2 for the six correlation functions in Section III-B. We see again that in all three trajectories, $\mathbf{f}_2, \mathbf{f}_4$, and \mathbf{f}_6 all perform better than $\mathbf{f}_1, \mathbf{f}_3$, and \mathbf{f}_5 . This is due to the omnidirectional nature of the RFID reader antenna. In other words, as a user moves, she encounters more tags off of her trajectory (line of motion) than on her trajectory. Thus, since $\mathbf{d}^{(new)}$ points from the oldest tag to the newest tag in the user's scan range, many of the arrows point away from the trajectory. Conversely, since $\mathbf{d}^{(cen)}$ uses a spatial average of scanned tags, the arrows point toward a better estimate of the user's trajectory. (In fact, if the antennas were not only omnidirectional, but highly isotropic, using the spatial average would give a very good estimate of the user's trajectory.) On the average, arrows are usually pointing away from the trajectory. (The plot points are mostly positive.) Only in the second and third trajectory, at high power (P_3), the arrows are pointing toward the trajectory on average.

We also evaluate our correlation functions using visualizations. We create an *arrow field* by drawing the arrows on the tag grid. For simplicity, we ignore the correlation weights and focus only on the correlation directions $\mathbf{d}^{(new)}$ and $\mathbf{d}^{(cen)}$. (Thus, all arrows are weighted equally). The arrow fields for $\mathbf{d}^{(new)}$ with the medium power level are shown in Fig. 3. The arrows start and end at tag locations, corresponding to vertices on the grid. The arrow fields for $\mathbf{d}^{(cen)}$ are shown in Figs. 4, 5, and 6. Since our goal is to estimate the user's trajectory accurately, we want a dense arrow field that surrounds the trajectory tightly. $\mathbf{d}^{(new)}$ is poor since the arrow fields are "fat", agreeing with the average arrow error and average normalized arrow error change discussions above. $\mathbf{d}^{(cen)}$ is thus the better choice. At low power (Fig. 4), the reader is not able to scan enough tags. Thus, the arrow field is too sparse. At high power (Fig. 6), the reader scans many tags, resulting in a dense arrow field. However, the reader range is too large, resulting in the arrow field being too fat. Medium power (Fig. 5) gives a reader range that results in an arrow field that is both dense and narrow. (Some of the arrow fields are slightly shifted linearly away from the trajectory. This is due to the error of not following the trajectory exactly when performing the experiments.)

2) *Search and rescue*: For search and rescue, the goal is to locate the lost person as quickly as possible. The goal is then different from that of accurately estimating a user's entire trajectory. Rather, we want the digital path left by a lost user (the arrow field) to be dense and fat. This allows searchers to quickly locate the digital path, follow it, and locate the user. That is, we do not care about where the user was exactly at specific points in space. We just want to locate and stay on the path as efficiently as possible. Therefore, $\mathbf{d}^{(new)}$, as shown in Fig. 3 is a better choice than $\mathbf{d}^{(cen)}$. However, to create an even denser arrow field, we can modify $\mathbf{d}^{(new)}$ and $\mathbf{d}^{(cen)}$. In $\mathbf{d}^{(sar,new)}$, every time a tag enters or exits a user's scan range, create $|\mathcal{T}(t)| - 1$ arrows, where the i^{th} arrow points from the i^{th} non-newest tag $\in \mathcal{T}(t)$ to the newest tag $\in \mathcal{T}(t)$, where $i = 1, \dots, |\mathcal{T}(t)| - 1$. In $\mathbf{d}^{(sar,cen)}$, every time a tag enters or exits a user's scan range, create $|\mathcal{T}(t)|$ arrows, where the i^{th} arrow points from the i^{th} tag $\in \mathcal{T}(t)$ to the centroid of $\mathcal{T}(t)$, where $i = 1, \dots, |\mathcal{T}(t)|$. Results are shown in Figs. 7 and 8. We see that $\mathbf{d}^{(sar,new)}$ is better for search and rescue.

One goal of search and rescue is to minimize the search time required to locate the lost person, which is directly proportional to the area searched. Consider a generic search algorithm where a lost person has left behind a set of ordered discrete markers in space. Starting at a given position, the searcher progressively expands her circular search range (starting from zero) until a marker is found. She then repositions herself at the new marker, and repeats the expanding search range until the next higher order marker with is found. This is repeated until she finds the lost person. This search algorithm can model an RFID reader progressively increasing it's scan range, or a searcher physically moving in a spiraling motion between markers. The cumulative search area is $A = \sum_{i=1}^N \pi r_i^2$, where r_i is the i^{th} search range, and if there results in N such searches. As N (number of markers) becomes large and they are distributed approximately uniformly so that $r_i \approx \frac{L}{N}$ for some fixed L , $A \rightarrow 0$. Intuitively, we want more markers to minimize the search area. In our case, we take the starting position as the start point of the trajectory and the lost person's position as the end point of the trajectory. We take arrow heads as the markers. They are ordered according to how the arrows themselves are created, according to Section III-B. We calculate the cumulative search areas when using $\mathbf{d}^{(new)}$ and $\mathbf{d}^{(cen)}$. (Note that $\mathbf{d}^{(sar,new)}$ and $\mathbf{d}^{(sar,cen)}$ give the same results, as $\mathbf{d}^{(new)}$ and $\mathbf{d}^{(cen)}$, respectively, since they introduce additional arrows for the same arrow head position.) These are averaged over the five experiment runs. Results are shown Fig. 9. $\mathbf{d}^{(cen)}$ provides better performance, since there are more arrows, agreeing with our intuition of reducing the cumulative search area by increasing the number of markers (arrow heads in our case). This is also the reason for the poor performance at low powers where the number of scanned tags is small, resulting in large distances between consecutive found tags in the search algorithm. $\mathbf{d}^{(cen)}$ improves in performance as the power level increases. However, for $\mathbf{d}^{(new)}$, performance actually may degrade at high powers. This is because the large scan range may create arrows pointing far away from

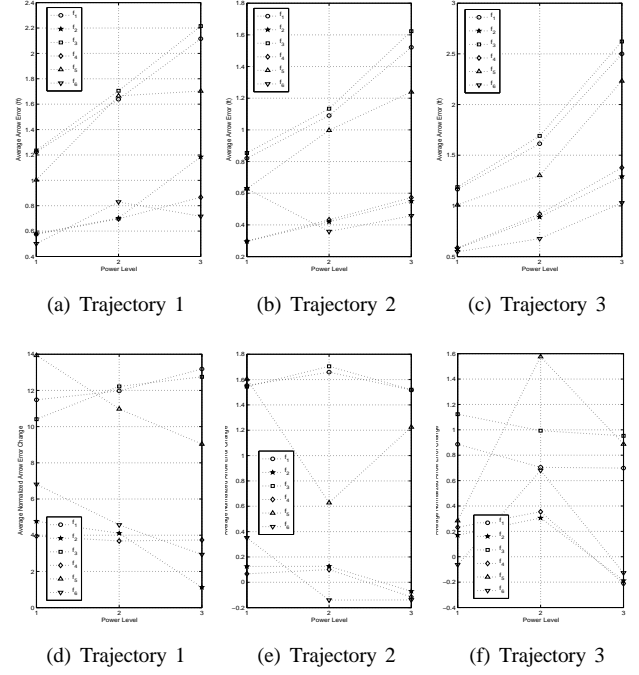


Fig. 2. (a), (b), (c): Average arrow error. (d), (e), (f): Average normalized arrow error change.

the trajectory, resulting in a large search areas in the vicinities of these particular arrows. In contrast, this does not happen for $\mathbf{d}^{(cen)}$.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a scheme to exploit the space-time correlations in a RFID tag field. We consider several correlation functions and interpret them as arrows. We evaluate our scheme using experiments. We consider the efficacy of the arrows for localization and tracking, as well as search and rescue using both statistical measures and visualizations.

Future work includes generalizing our tag field beyond a grid structure. We can consider different spatial distributions of tags, modeling the different types of tag deployment constraints or requirements. For example, we may want a denser tag deployment in a high security area in a warehouse to closely track mobile robots. The tag field can also suffer failures, due to tags failing or external damage. Such an environment may demand a more robust system design.

REFERENCES

- [1] L.M. Ni, Y. Liu, Y.C. Lau, and A.P. Patil, "LANDMARC: Indoor Location Sensing Using Active RFID," in *Proc. IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Forth Worth, TX, Mar. 2003, pp. 407-415.
- [2] D. Hähnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and Localization with RFID Technology," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, LA, Apr. 2004. vol. 1, pp. 1015-1020.
- [3] A. Kleiner, J. Prediger, and B. Nebel, "RFID Technology-based Exploration and SLAM for Search And Rescue," in *Proc. IEEE International Conference on Intelligent Robots and Systems (IROS)*, Beijing, China, Oct. 2006, pp. 4054-4059.

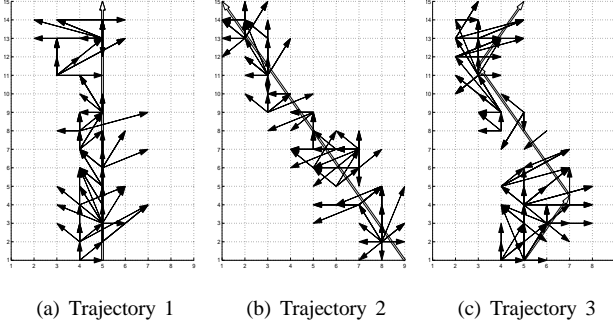


Fig. 3. Arrow field with $d^{(new)}$ from one run. Antenna power = P_2 .

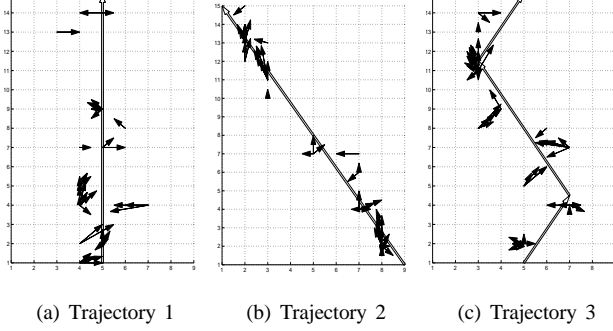


Fig. 4. Arrow field with $d^{(cen)}$ from one run. Antenna power = P_1 .

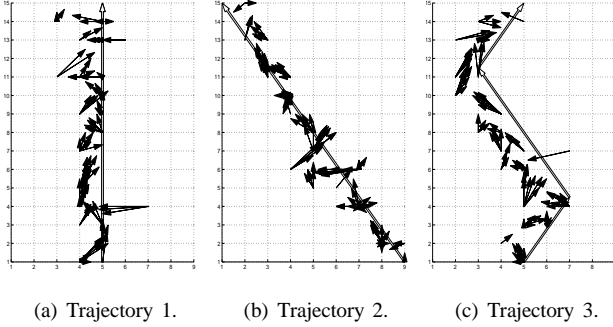


Fig. 5. Arrow field with $d^{(cen)}$ from one run. Antenna power = P_2 .

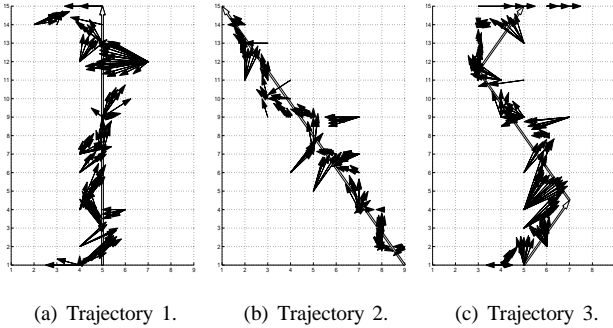


Fig. 6. Arrow field with $d^{(cen)}$ from one run. Antenna power = P_3 .

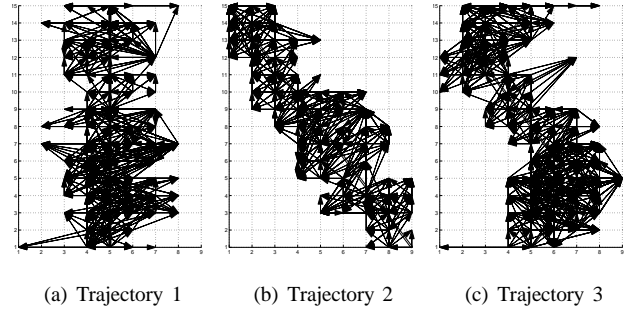


Fig. 7. Arrow field with $d^{(sar,new)}$ from one run. Antenna power = P_3 .

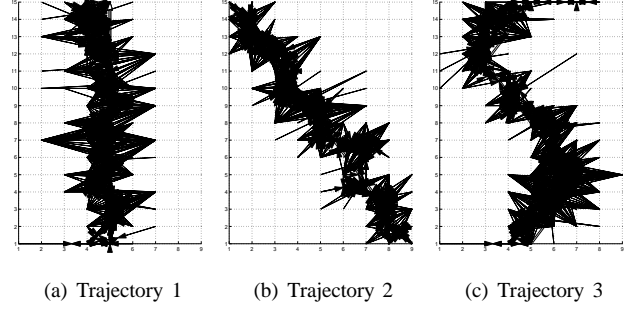


Fig. 8. Arrow field with $d^{(sar,cen)}$ from one run. Antenna power = P_3 .

- [4] J. Bohn and F. Mattern, "Super-Distributed RFID Tag Infrastructures," in *Proc. European Symposium on Ambient Intelligence (EUSAI)*, Eindhoven, Netherlands, Nov. 2004. *Lecture Notes in Computer Science (LNCS)*, 2004, vol. 3295, pp. 1-12.
- [5] D. Cox, V. Kindratenko, and D. Pointe, "IntelliBadge: Towards Providing Location-Aware Value-Added Services at Academic Conferences," in *Proc. ACM International Conference on Ubiquitous Computing (UbiComp)*, Seattle, WA, Oct. 2003. *Lecture Notes in Computer Science (LNCS)*, 2003, vol. 2864, pp. 264-280.
- [6] C. Wang, H. Wu, and N.-F. Tzeng, "RFID-based 3-D Positioning Schemes," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Anchorage, AK, May 2007, pp. 1235-1243.
- [7] J. Burnell, "Texas Turns to RFID for Emergency Evacuation System," in *RFID Update*. [Online]. Available: <http://www.rfidupdate.com/articles/index.php?id=1513>.
- [8] C. Swedberg, "Search Teams Put RFID to the Rescue to Help Find the Missing," in *RFID Journal*. [Online]. Available: <http://www.rfdjournal.com/article/articleview/3362/1/1/>.
- [9] V.K.Y. Wu and N.H. Vaidya, "RFID Trees: A Distributed RFID Tag Storage Infrastructure for Forest Search and Rescue," in *Proc. IEEE Conference on Sensor, Mesh, and Ad Hoc Communications and Networks (SECON)*, Boston, MA, Jun. 2010.
- [10] "TegoChip." [Online]. Available: <http://www.tegoinc.com>.

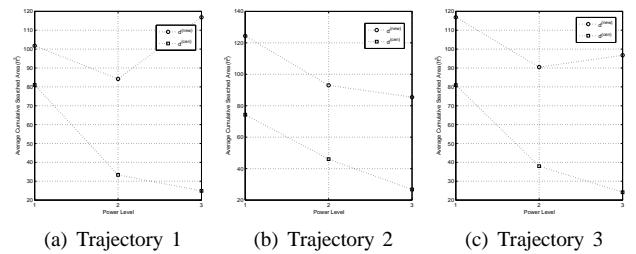


Fig. 9. Average cumulative searched area.