November 2013

Time-Shifted Multilateration for Mobile Object Localization using RFID Crowdsourcing

Technical Report 2013-615

Telecommunications Research Lab (TRL)
School of Computing
Queen's University
Kingston, Ontario, Canada, K7L 3N6

Lobna M. Eslim
Hossam S. Hassanein
Walid M. Ibrahim
Abdallah Alma'aitah
Email: {lobna.eslim, hossam, walid, abdallah}@cs.queensu.ca

Queen's School of Computing

Queen's University

Abstract

RFID technology as an enabler of Internet of Things (IoT) is extensively utilized for object localization. Existing RFID-based object localization techniques follow a central and coordinated approach. Indeed, none is designed to leverage RFID crowdsourcing for the purpose of object localization. In this paper, we propose a system for localizing mobile RFID tags using heterogeneous, distributed and dynamic mobile RFID readers. We introduce the concept of Time-Shifted Multilateration (TSM) to enhance location estimation accuracy of mobile tags when sufficient synchronous detection information is not available. We validate the proposed system and technique through extensive simulations using ns-3 and empirical experiments using actual RFID readers and tags. Results show that our approach can achieve accurate location estimation in typical IoT settings.

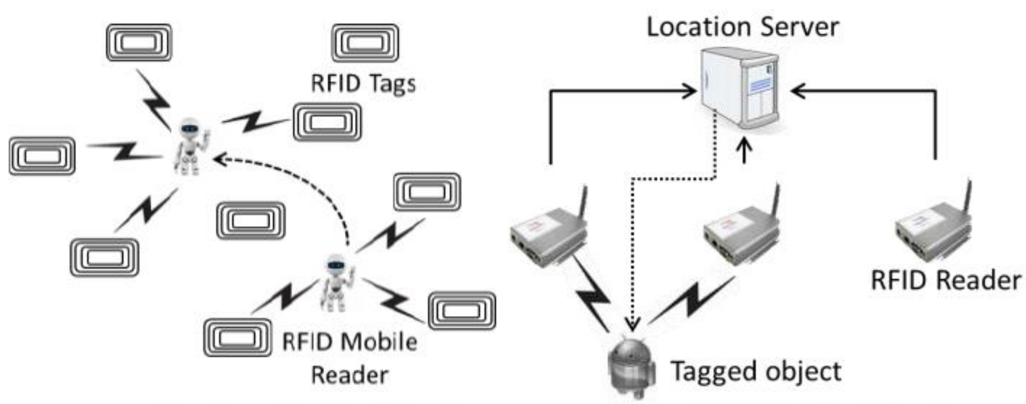
I. Introduction

The term "Internet of Things (IoT)" is broadly used to refer to a new generation of the current Internet with millions or even billions of spatially disseminated smart objects or simply "things". These things are equipped with different sensors and actuators that allow them to; be identifiable, communicate and exchange information among themselves and/or with humans [1]. Applications under the umbrella of IoT span a wide and diverse range of domains such as: smart environments (i.e. smart homes, smart buildings and smart cities), healthcare, environmental monitoring, smart transportation, etc. [2]. Typically, these applications are rooted in our physical world to offer users more convenient context-and location-aware services. Thus, smart objects should be aware of their locations and/or localized to take advantage of such context. Providing localization while considering the IoT characteristics in terms of scalability and objects heterogeneity, and mobility, is a challenging problem.

Radio Frequency Identification (RFID) is one of pivotal enabling technologies of IoT for the purpose of identification. RFID development achieved unceasing technical progress in addition to cost reductions and standardization in the past few years [3]. The demand for embedding the localization capability into the IoT infrastructure; sparked the use of RFID systems for object localization [7]- [16]. RFID-based localization systems can be broadly categorized into; reader localization and tag localization [4] as shown in Fig. 1. In the reader localization (see Fig. 1(a)), objects are equipped with RFID readers and localized, based on connectivity information with a set of active and/or passive tags deployed at known locations. Whereas in the tag localization (see Fig. 1(b)), objects are attached with RFID tags and localized through a set of coordinated RFID readers that report to a central server for location estimation.

In IoT settings, it is typical to have an environment that is comprised of a large number of RFID-tagged objects and a considerable group of ad hoc mobile RFID readers which are possibly heterogeneous and un-coordinated. Eminent examples of such environments are shopping malls, airports, attractions, etc. Participants in such environments are typically interested in localizing some objects depending on their changing needs and context. For instance, in city attractions, visitors need to keep track of their children and/or their belongings. Providing an object localization service in these environments is challenging.

In this paper, we propose a distributed object localization system based on crowdsourcing. The system takes advantage of the following: (1) smart objects are identified by either passive or active RFID tags, (2) embedded RFID readers in mobile devices are being rapidly adopted (e.g., smart phones or small embedded RFID scanners) [5] and (3) RFID tags are capable of storing a large amount of data in addition to their unique identifiers [6]. Our proposed system can operate



(a) Reader localization (b) Tag localization

Fig. 1: Classification of RFID-based localization systems

in a central or distributed manner. In the centralized approach, the Detectors (heterogeneous, independent and dynamic RFID readers) periodically detect tags in their interrogation zones and send the detection information to a central server, which in turn localizes tags and offers localization service. In the distributed approach, the system utilizes the tag as the focal point for storing reader proximity and location information obtained from passing readers. Users interested in the location of an object can send a query to pull the information. We remark that this approach is fundamentally different from exiting tag localization techniques.

Existing localization techniques operate with the assumption that available detection information is sufficient and synchronous. Such assumption, however, may not be plausible in mobile and/or dynamic environments. The dynamicity in terms of number of readers, readers' detection ranges and mobility of both tags and readers prevent having sufficient and synchronous detection information for each tag. Therefore, we propose the novel localization paradigm "Time-Shifted Multilateration (TSM)". TSM utilizes asynchronous time-shifted detection information to localize tags when sufficient synchronous detection information is not available. To the best of our knowledge our approach is the first to:

- develop an RFID-based localization system to localize mobile objects based on reader crowdsourcing,
- utilize tags' memory to store reader detection information and location information that can be read by other passing readers, and
- use time-shifted detection information to enhance localization in the following cases: (1) the concurrent spatial information is not sufficient to localize a tag and (2) the ad hoc mobile RFID readers have relatively short reading ranges which are not sufficient to follow mobiles tags.

We validate the proposed system through extensive simulations using ns-3 and empirical experiments using actual RFID readers and tags. Results show that our approach can achieve accurate location estimation in typical IoT settings.

The remainder of this paper is organized as follows: Section II reviews some of the related work and motivates our proposed approach. In Section III we propose a system for localizing mobile RFID tags using heterogeneous, distributed and dynamic mobile RFID readers. The TSM technique is explained in section IV and a use case is presented in Section V. Section VI presents performance evaluation of the proposed system. Finally, our conclusion is given in Section VII.

II. RELATED WORK AND MOTIVATION

Several RFID-based localization systems have been proposed in the literature, which can be broadly categorized into; reader localization and tag localization.

In the reader localization systems [7] - [11], typically a large number of active and/or passive tags are deployed at known locations in the area of interest to represent landmarks for mobile objects. Each mobile object, which is equipped with an RFID reader, estimates its location based on the connectivity information with those landmarks. For instance, in reference [7], the authors attach reference tags to the floor and ceiling in a square pattern to localize a mobile reader using the weighted average method and a weighting function. While in reference [8], the same approach is followed but the accuracy of localization is enhanced by rearranging the reference tags in a triangular pattern. However, in such systems, the required number of reference tags is relatively high. To alleviate such a requirement, SLAC-RF [9] propose a specialized tag named supertag, which is an array of RFID tags arranged to emulate a virtual antenna array. Based on the phase difference of received signals with respect to those supertags and the inertial navigation system (INS) measurements, a mobile reader can estimate its position. The work in [9] and [11] propose localization methods based on the geometric knowledge of the identification region in 3D space to provide a finer degree of localization. However, reference [11] considers the fault frequency in localization and proposed a quality index to measure the quality of localization results. Reader localization systems are inherently distributed and provide good accuracy through cost effective infrastructure. However, they suffer from the high cost of associating an RFID reader with every object, rendering such an approach ineffective for IoT settings.

In tag localization systems [12]- [16], an infrastructure of RFID readers, which detect tags and report detection information to a central server for location estimation, is used. LANDMARC [12] uses an RFID reader infrastructure along with reference tags. By comparing the Received Signal Strength (RSS) from the targeted tag with those of reference tags, the server estimates the tag location based on the locations of the k-nearest reference tags. VIRE [14] and L-VIRT [15] use virtual reference tags instead of the dense deployment of reference tags. For instance, VIRE calculates the RSS of each virtual reference tag using the RSS of the surrounding reference tags and a linear interpolation algorithm. Then, it compares a tag's RSS to that of reference tags either real or virtual, identifies all plausible locations and filters them using an elimination algorithm. An attempt to localize tagged objects using mobile readers is proposed in reference [13] with support of landmarks. The reader-tag distance and tag-landmark distance are used to estimate the tag location. The centralized and fixed infrastructure-based systems provide limited scalability and may not be a practical solution for IoT settings.

The objective of our work is to design an accurate tag localization system for dynamic and/or mobile IoT settings, which is scalable and requires minimal central infrastructure.

III. READER CROWDSOURCING SYSTEM

Our approach aims to provide a localization service in large-scale and dynamic environments; where deploying and maintaining a fixed central infrastructure for localization is expensive and/or infeasible. Our proposed system relies on crowdsourcing detection information from ad hoc readers that are mobile and uncoordinated.

A. System Model and Components

The system has two components:

- ***Tags*** – representing the objects to be localized. These objects can be either stationary and/or mobile and are identified by RFID tags. The number of tags is much larger than the *Detectors* in the scenarios under study.

- ***Detectors*** – Representing the RFID readers in the area, which are predominantly dynamic, heterogeneous, and uncoordinated. They have a common need, which is localizing objects of interest in the environment. Such *Detectors* may be the smart phones of visitors to an attraction or handheld RFID scanners carried by caregivers in a hospital. The *Detectors* are assumed to be capable of acquiring their locations at any given time using any of the localization systems for mobile readers (e.g., GPS, WiFi, etc.).

A key player in tag localization is the detection events. When a *Detector* detects a tag successfully, it generates a detection record, which contains temporal and spatial information about the tag with respect to the *Detector*. Detection events of a tag within a specific time interval are then used to localize the tag. Given a set of RFID tags (*Tags*) and a set of ad hoc mobile RFID readers (*Detectors*), each tag is allowed to know its estimated position at any given time and locally store such information in its memory.

For consistency, we use the following notations:

- $T = \{t_1, t_2, t_3,....,t_n\}$ is the set of n *Tags*.
- $D = \{d_1, d_2,....,d_m\}$ is the set of m *Detectors*.
- *tolerance interval* is the time interval within which detection events are eligible to contribute to localization.

In this work, we assume that *Detectors* can interrogate all *Tags* in the given environment and update their memory. Access rights and security issues are considered to be beyond the scope of this paper.

B. Exchanged Information

Two types of information are created during the system operation. These are:

Detection information, which contains temporal and spatial information about a Tag ti with respect to *Detector* d_j as shown in TABLE I. It is built during the tags identification process and used later in location estimation.

Location information, which contains the estimated locations of t_i. Each location is identified by its estimation time and a Location Accuracy Indicator (*LAI*) as shown in TABLE II. *LAI* represents the number of detections positively contributing to the tag location estimation, and affects the location accuracy.

TABLE I: SCHEMA OF DETECTION INFORMATION

Field	*Description*
time	*The time at which a detector d_j detects tag t_i and creates the detection record.*
position	*The 2D position of the detector d_j at time of detection, it is represented by x, y coordinates.*
distance	*The tag to detector distance, measured by means of RSS, time difference of arrival , angle of arrival, etc.*

TABLE II: SCHEMA OF LOCATION INFORMATION

Field	*Description*
time	*The time at which a detector d_j estimates the location of tag t_i based on its detection information.*
location	*The estimate location of t_i, it is represented by x, y coordinates.*
LAI	*Number of detections used by d_j to estimate the location of t_i.*

C. System Operation

The reader crowdsourcing system can operate using one of two approaches: centralized or distributed. Fig. 2 (a) and (b) represent the general framework of the system for the centralized and distributed, respectively. In the following subsections, we explain the two approaches.

1) Centralized approach

In this approach, the infrastructure provider deploys a central location server. This server collects detection information from all *Detectors* in the area, estimate the tags locations and responds to location queries. As depicted in Fig. 2 (a), each *Detector* periodically interrogates the tags within its proximity and sends the detection information to the central location server. The server uses this information along with an estimation of tag speed to update the location information every *tolerance interval*. Then, the server can be contacted by any wireless device via an app1 for tags locations queries. The localization algorithm is common for both centralized and distributed approaches and is detailed later.

2) Distributed approach

As shown in Fig. 2 (b), instead of employing a central location server, Tags are used as the focal point. The *Detectors* periodically: (1) detect tags in their interrogation zones and write detection information on the interrogated tags memory and (2) retrieve detection information, estimate tags locations accordingly and update the tags location information. Location queries can be carried out between *Detectors* using a pull strategy similar to that in reference [17]. In the distributed approach it is the *Detectors* that are required to estimate the locations of tags in their proximity, update their location information and reply to location queries. Thereafter, we focus on the distributed approach.

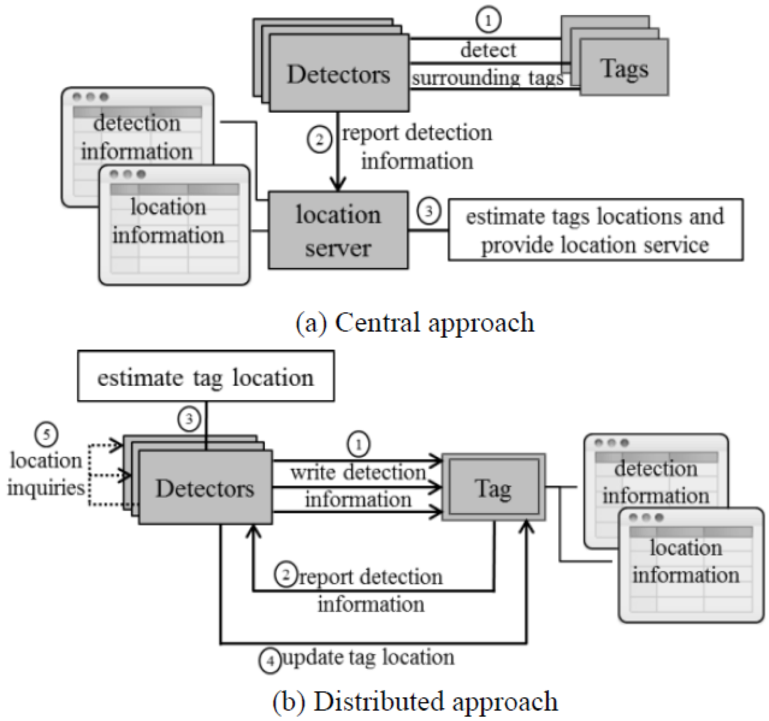


Fig. 2: General framework of reader crowdsourcing system.

a) Tags notification

Tags' memories are updated by passing *Detectors* for two objectives: (1) maintain detection records on the tag memory to be used by other passing *Detectors* for tag localization and (2) allow the tag to know its estimated position at every *tolerance interval*. For the former, each *Detector* d_j in D interrogates tags in its proximity. For each successfully identified tag t_i, dj creates a detection record and writes such record into t_i memory (see Fig. 3). As shown in the figure, updating the tag memory by a subset of D allows the tag to hold multiple time-stamped detection records. If the tag is static, most of these detection records positively contribute to localization accuracy. However, in case of a mobile tag, a time constraint should be considered when localizing the tag, to effectively ignore the outdated records with respect to *tolerance interval*.

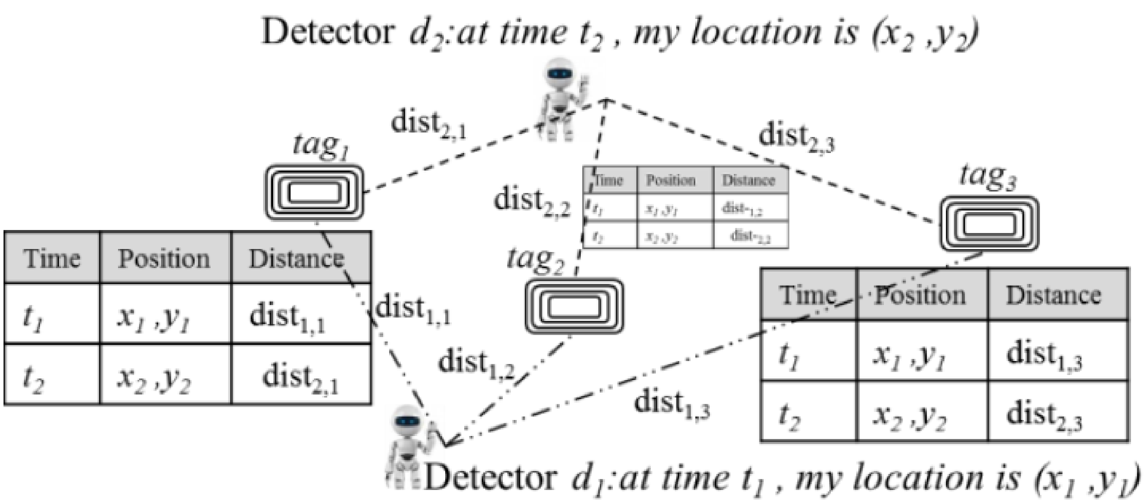


Fig. 3: Snapshot of tags notification.

b) Tags localization

Every *tolerance interval*, *Detectors* interrogate tags in their proximity, fetch the tags detection information, estimate the tags locations and update the tag location information accordingly. Algorithm I lists the tag localization algorithm, where it is assumed tags and readers are stationary. This may also correspond to a snapshot of the dynamic readers and tags case.

In Algorithm I, the detection records are processed first to filter out the outdated records with respect to *tolerance interval* (lines 5-9). Then the remaining detection records are filtered to exclude detections that do not positively contribute to the intersection area of the more recent detections (lines 10-20). The two sequenced filtrations may result in only one detection record, resulting in less localization accuracy. Otherwise, the Multilateration technique is applied and the number of used detections is added to the location record as the *LAI*. The *LAI* can be used to decide which location is more accurate in case multiple locations are estimated within a same tolerance interval. The tag location information is then updated.

Algorithm I: Tags localization Algorithm

Input: detection information **Output:** location information record

```
for each tolerance interval do
    for each t_i in my proximity do
        set Detect_info(t_i) = get t_i .detection information
        set filtered_info(t_i)
        for each record r_j in Detect_info (t_i) do
            if r_j.time < current time – tolerance interval then
               Detect_info(t_i).delete( r_j)
            end if
        end for
        for each record r_j in Detect_info (t_i) do
            if j = 1 then   filtered_info(t_i).add(r_j)
            else
                for each r_k in filtered_info(t_i) do
                    if r_j do not intersect with r_k then
                        Detect_info(t_i).delete( r_j) and break
                    end if
                end for
                filtered_info(t_i).add (r_j)
            end if
        end for
        set LAI = filtered _info(t_i).size
        t_i .position = Estimate_Loc (filtered_info(t_i))
       Update t_i .location information ( Get (current time),
                              t_i.position, LAI)
    end for
end for
```

Tags notification process can result in large accumulated detection information, which may be outdated after the tolerance interval. To release *Tags* resources, *Detectors* periodically delete this outdated detection information along with location information and maintain only *S* most recent locations. The parameter *S* determines the location history maintained in each tag for the purpose of tag speed estimation, which will be explained later.

c) Location query

Tags notification and tags localization allow each tag to hold its own location, limiting the localization service for only RFID readers. To adapt the localization service for all application users regardless of their sensing capabilities; a pull strategy similar to reference [17] can be adopted among wireless devices via apps designed for localization service. In this strategy, a wireless device broadcasts a query asking for the location of tag(s) of interest. Each *Detector* receiving this query interrogates such tag(s) in its vicinity, retrieves its location if it exists and replies back to the requestor. If the tag does not exist, the *Detector* ignores the query. If the interested wireless device does not receive a response within a certain timeout, it initiates another location query.

Wireless devices acquire the most recent and most accurate locations for tags of interest. Algorithm I, uses Multilateration which assumes that the detection records within a *tolerance interval* are synchronous. This assumption provides reasonable localization accuracy for stationary objects (e.g. products in shopping malls or items in drug stores). However, in dynamic environments such as sports activities or public gatherings, objects to be localized are typically mobile and the assumption of synchronized proximity readings is not always valid. To overcome this negative effect, we next introduce an asynchronous multilateration technique.

IV. TIME-SHIFTED MULTILATERATION (TSM) TECHNIQUE

Typically most distance-based localization techniques assume that the measured spatial information, even those from mobile anchors, is synchronous and sufficient to localize objects [18]. Thus, they estimate the object position based on the intersection of the given spatial information (i.e., lateration, bounding box, etc.). This assumption is not reliable in a typical dynamic environment where the mobile anchors are mobile ad hoc RFID readers typically with short reading ranges. Three challenges arise in this case: (a) insufficient spatial information, (b) non-intersecting spatial information, or (c) the intersection may not reflect the object's real location. As a result, the difference between the actual and the estimated location may be significant. We propose *Time-Shifted Multilateration (TSM)* where the spatial information is shifted based on the tag speed and time differences to provide better accuracy. Thereafter, each detection record is considered as circle in 2D, centered at the *Detector* position at detection time with the radius equal to *Tag* to *Detector* distance.

The TSM technique takes two inputs: asynchronous detection information during a specific period (*tolerance interval*) and a tag location history and works as follows. First, if the tag has no previous estimated locations, TSM considers an initial tag speed based on the attributes of the mobile object it is attached to (e.g., walking speed for pedestrians). Otherwise, TSM estimates the tag speed using an exponentially weighted moving average. Second, TSM performs a time-shifting process, TSM enlarges each detection based on the both the tag speed and the time difference between such detection and the most recent one; resulting in a synchronized detection list. Last, TSM applies Multilateration to the synchronized detection list to estimate tag location. Fig. 4 illustrates an example of the TSM technique and shows how the time-shifting process takes place for 4 detections. We next validate the positive effects of time-shifting on location accuracy. We then detail the tag speed estimation and the time-shifting processes.

Definition 1 (detection set): Given the set of *Detectors D*, the detection set of a tag *ti* is the spatial information measured by a subset of *D* within a specific time interval (*tolerance interval*), denoted as $P(t_i)$ and ordered chronologically.

Each element p_k in $P(t_i)$, is represented by $p_k.t$, $(p_k.x, p_k.y)$ and $p_k.r$, which are defined in TABLE I. As in typical localization schemes, each p_k is prone to two sources of errors: *Detector* position and tag to *Detector* distance errors.

Knowing the speed of a mobile tag, the tag can be localized at time *t* using detection information from time t-Δt. Accordingly, we can establish the following theorem.

Therom1: a mobile tag, which is localized by a detection $p_k = \{ p_k.t, (p_k.x,p_k.y), p_k.r \}$, can be localized after time Δt by a detection $p'_k = \{ p_k.t+\Delta t, (p_k.x,p_k.y), p_k.r+(s * \Delta t)\}$, given its average speed is *s*.

Proof: Given the mobile tag speed *s*, the maximum distance a tag can travel during a period Δt is $\Delta r = (s * \Delta t)$. So if the tag is localized by the detection p_k as shown in Fig. 5(a); the worst case is when the tag is located at a point on the circumference of the circle at time $p_k.t$ and moves perpendicularly outside the circle. Considering the maximum distance *Δr*, if the tag is detected in a circle centered at $(p_k.x,p_k.y)$ and has a radius $p_k.r$; after the period *Δt*, the tag cannot reach a point outside the circle centered at $(p_k.x,p_k.y)$ and has a radius $p_k.r+\Delta r$. ■

Therom2: a mobile tag, which is localized by detections: $p_k = \{ p_k.t, (p_k.x,p_k.y), p_k.r \}$ and $p_j = \{ p_j.t, (p_j.x,p_j.y), p_j.r \}$ such that $p_k.t$ is more recent than $p_j.t$, is expected to be located in the area of intersection between the circle centered at $(p_k.x,p_k.y)$ with a radius $p_k.r$ and the circle centered at $(p_j.x,p_j.y)$ with a radius $(p_j.r+s*(p_k.t-p_j.t))$, given its average speed is *s*.

Proof: If the tag is localized by the detection p_k as shown in Fig. 5(b) then at time $p_k.t$, the tag is located at an arbitrary point in the circle centered at $(p_k.x,p_k.y)$ with a radius $p_k.r$. According to *theorem 1*, the tag is also located at an arbitrary point in the circle centered at $(p_j.x,p_j.y)$ with a radius $(p_j.r+s*(p_k.t-p_j.t))$. Thus, such an arbitrary point would be in the area of intersection between the above mentioned two circles. ■

A. Tag speed estimation:

TSM starts updating tag speed after estimating two or more locations for the tag. With two previous locations in hand, we measure the tag speed based on the traveled distance between them. For three or more previous locations, we use the exponentially weighted moving average [19].

Definition 2 (location set): Given the set of *Tags T*, the location set of a tag $t_i \in T$ is the tag's estimated locations computed by any $d_j \in D$, denoted as $Loc(t_i)$ and ordered chronologically.

Each element loc_j in $Loc(t_i)$, is represented by $loc_j.t$, $(loc_j.x, loc_j.y)$ and $loc_j.LAI$, which are defined in TABLE II.

Definition 3 (distance between two locations): Given two consequent locations for tag t_i: $loc_j(t_i)$ and $loc_{j+1}(t_i)$, the distance between the two locations is the Euclidean distance between $(loc_j.x, loc_j.y)$ and $(loc_{j+1}.x, loc_{j+1}.y)$, denoted as $d(loc_j,loc_{j+1})$.

The traveling speed of tag t_i from $loc_j(t_i)$ to $loc_{j+1}(t_i)$ is:

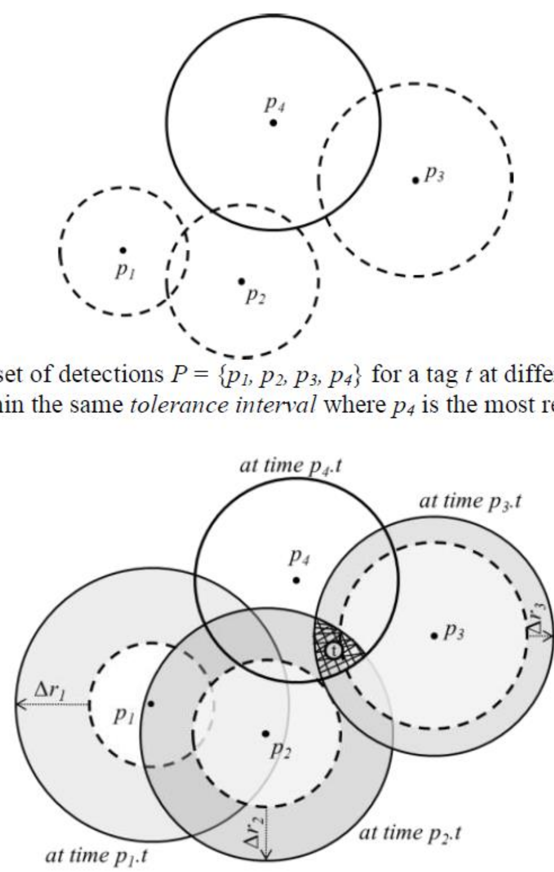
$$speed_{j+1}(t_i) = d(loc_j,\ loc_{j+1})\ /\ (loc_{j+1}.t - \ loc_j.t) \quad (1)$$

where *j = 1...k-1 | k* is the number of previous estimated locations for tag t_i. Given *k-1* successive speeds for tag t_i such that $speed_{k-1}(t_i)$ is the most recent one, the exponentially weighted moving average speed for t_i can be computed using the following equation:

$$EMAspeed_{j+1}(t_i) = \alpha * speed_{j+1}(t_i) + (1-\alpha) * EMAspeed_j(t_i) \quad (2)$$

where α is a constant factor between 0 and 1 which controls the rate of coefficients decreasing; attenuating the contribution of older speeds to the estimated speed. Thereafter in the paper, we consider α as *(1/k-1)*.

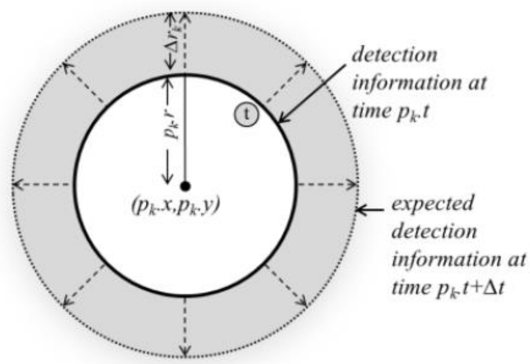
If there are no previous estimated locations for the tag to be localized, the initial speed is used though. Otherwise, Equation 1 is used in case of two previous locations existing and Equation 2 is used in case of *3* or more previous locations are available.



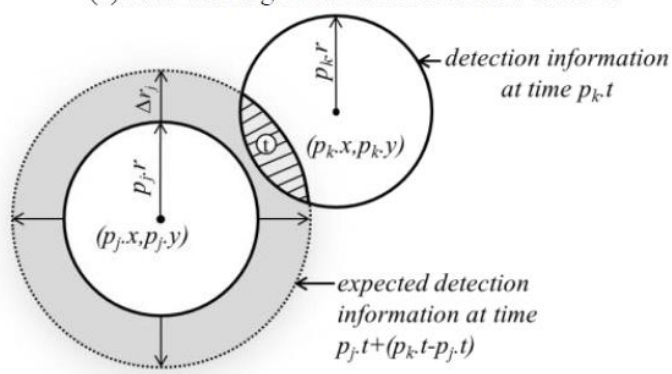
(a) The set of detections $P = \{p_1, p_2, p_3, p_4\}$ for a tag t at different times within the same *tolerance interval* where p_4 is the most recent.

(b) The set of shifted detections after expanding p_1, p_2, p_3 by Δr_1, Δr_2 and Δr_3 respectlively, the tag t is expected to be in the shaded area.

Fig. 4: Example on TSM technique.



(a) time-shifting of one detection after time Δt.



(b) How time-shifting affects localization accuracy

Fig. 5: The concept of time-shifting process

B. The time-shifting process and Multilateration:

Algorithm II is designed to perform the time-shifting step in our proposed technique. It takes as input a number of asynchronous detections and based on tag speed and time difference, it expands the radius of detections accordingly and outputs a new synchronized set to which the common Multilateration can be applied.

Algorithm II Time-shifting Algorithm

Input: asynchronous detections **Output:** synchronous detections

set $P(t_i)$ **= set of** k **detections of tag** t_i **chronologically ordered**
set $speed(t_i) =$ *estimate tag speed using equation 1 or 2*
for $j=1$ **to** k-1 **do**
  $\Delta r_j = speed(t_i) * (p_k.t - p_j.t)$
  $p_j.r = p_j.r + \Delta r_j$
end for
return $P(t_i)$

Using Multilateration, the coordinates of the tag *(x, y)* should satisfy the following equation:

$$(x - x_i)^2 + (y - y_i)^2 = d_i^2 \tag{3}$$

where (x_i , y_i) are the *x* and *y* coordinates of the *ith* anchor node and the d_i is the measured distance between such anchor node and the tag to be localized. This equation can be modified to include the time-shifting step as follows:

$$(x - p_j.x)^2 + (y - p_j.y)^2 = (p_j.r + (s * (p_k.t - p_j.t)))^2 \tag{4}$$

The same solution applied to equation (3) [20] can be applied to solve equation (4) without affecting the overall complexity which is *O(k3)* where *k* is the size of the set $P(t_i)$.

Theorem 3: Given a set of *k* asynchronous spatial information about a mobile tag t_i, $P(t_i)$, TSM technique can accurately localize the mobile tag t_i, given its average speed is *s* regardless of the direction of its trajectory.

Proof: For each spatial information $p_j(t_i)$ and according to Theorem 2, the mobile tag is located at a point in the area of intersection between $p_k(t_i)$ and $p'_j(t_i)$ where $p'_j.r = p_j.r+(s*(p_k.t-p_j.t))$. Considering the *k-1* pair $\{(p_k(t_i),p'_j(t_i)) \mid j=1..k-1\}$, we have at most (*k-1)* areas of intersection, where the mobile tag is expected to be located at time $p_k.t$. Subsequently, the mobile tag is located at a point in the area of intersection of the above mentioned (*k-1)* areas. ■

To enable the TSM technique in the reader crowdsourcing system, Algorithm II should be executed just before step 10 in Algorithm I.

V. USE CASE SCENARIO

Suppose that Tom plans to attend a Fair that came to town with his active young son Max. Upon his arrival, he receives a notification on his mobile device indicating that he has the option to contribute to a participatory localization service at the Fair area. Tom likes the idea as he is interested in keeping track of MAX. So he accepts the notification, hence, an app is installed on his mobile device along with supportive quick help. Also, he is instructed to pick up a wristband

RFID tag from the site administration for Max. A considerable number of participants have the same interest as Tom, thus they participate in the localization service as well. For the sake of illustration, we define the following potential types of actions that take place in the system:

- **Action A**: A mobile RFID reader interrogates a tag and writes such detection into the tag's memory.
- **Action B**: A mobile RFID reader interrogates a tag, fetches detection information from the tag memory, localizes the tag accordingly and writes the estimated location into the tag's memory.
- **Action C**: A mobile device broadcasts a query asking about location of certain tag(s).
- **Action D**: A mobile RFID reader receives a location query, triggers Action B with respect to the tag of interest and replies to the requestor.

Fig. 6 depicts several locations and events over a time window of Tom's tour. Within this time window, there are 7 mobile RFID readers contributing to localization service including Tom's mobile device. At location 1, R1 and R3 executed a type A action in relation to Max's tag. At location 2, another Action A was taken by R4; consequently Max's tag holds three asynchronous detection records. When Tom and Max were at location 2, a science show attracted Max so he moved to location 3 to enjoy it without Tom. While Max was enjoying the science show, R2 conducted Action A while R5 conducted Actions A & B. When performing Action B, R5 uses the TSM technique to localize Max (at location 3) based on detection records created by itself, R2, R3 and R4 (by now the detection from R1 is outdated.) After a while, Max discovered that he was lost so he started running toward his father but unfortunately, it was in a wrong direction. Tom did not realize that, thus he followed his path as shown in Fig. 6. When Max reached location 4, Action A was taken by R6. At the same time, Tom realized that his son was not around; he used his mobile device and carried out Action C with respect to Max's tag. During this time Max moved from location 4 to location 5, getting out of R6's coverage area. R7 carried out Action D, which includes Action B as well. In Action B, R7 uses the TSM technique to localize Max (at location 5). R7 estimated Max's moving speed based on his location history and expanded the detection record created by R6 accordingly, resulting in better location accuracy. Tom received a message from R7 indicating that Max was now at location 5. Very relieved, Tom then rushed to this location for Max.

At the end of the day, Tom decided to go home. At the exit gate, he received a message indicating that his mobile device was unregistered from the localization service and the app may then be uninstalled, releasing any resources on Tom's mobile device.

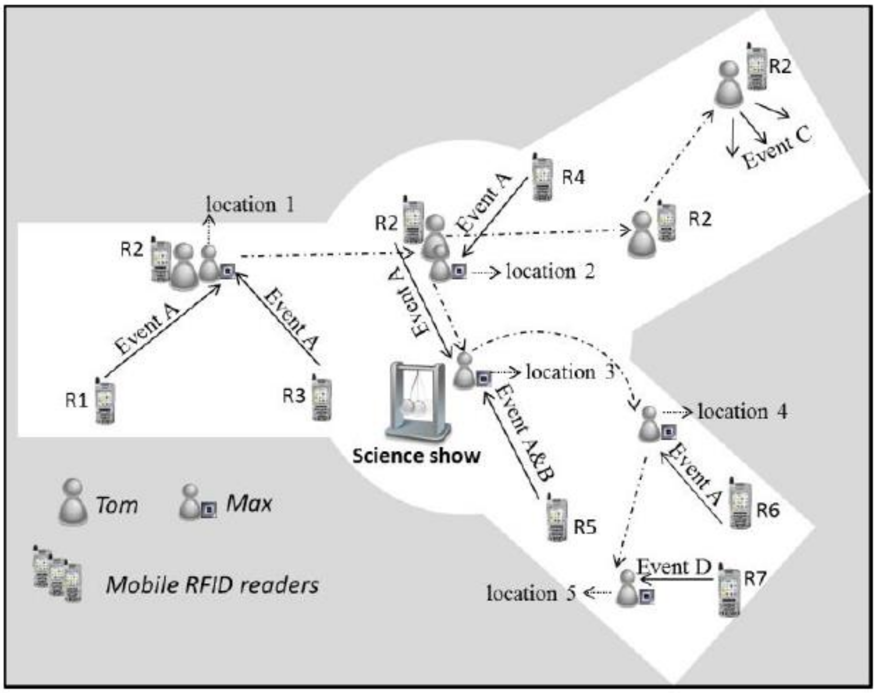


Fig. 6: Use case scenario.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the reader crowdsourcing system along with the TSM technique through extensive simulation experiments using ns-3 and as well as a testbed using actual readers and tags. We aim to: (1) validate the system under realistic scenarios, (2) investigate the effect of applying the TSM technique on the system performance under different dynamicity settings and (3) assess the effects of various parameters on the performance of TSM.

A. Simulation Setup

The reader crowdsourcing system suits a multiplicity of applications. Among which we chose to simulate a mini attraction area as illustrated in Fig. 7. Using the ns-3 network simulator [22] and based on Graph-Based Mobility Model for Ad Hoc Network [23], we simulate an area of *200m x 200m* containing *14* point of interest, which are linked using pathways of *4m* width. During the simulation, the mobile nodes, represented by mobile RFID readers and tags, are only allowed to move on those pathways to a randomly selected point of interest. We also allow them to pause for a period of time (say *10* sec) at each point of interest during their tour. After the pause period, each mobile node changes its speed and moves to another randomly selected point of interest. The speed of mobile nodes is pedestrian speed ranging from *0.75m/sec* to *1.25m/sec* [24]. Each mobile RFID reader has a random reading range from *3m* to *5m* and interrogates surrounding tags every *1sec* while the *tolerance interval* is set to *10sec*. In measuring the distance between a tag *ti* and a reader *dj*, we consider the range measurement noise $\epsilon_{i,j}$ as a zero-mean white Gaussian process $\mathcal{N}(0, \sigma_{i,j}^2))$ where σ is a variance correlated to the noise free distance and signal to noise ratio (SNR) as $\sigma^2 = (noise\_free\ distance)^2/SNR$ [25].

Without loss of generality, we start our simulation by deploying the mobile nodes randomly at points of interest and allow them to move based as aforementioned. We perform the simulation experiments under different settings in terms of the number of mobile readers, pause time and mobility speed of both readers and tags. We are interested in the following performance metrics: (1) average location error, (2) localization delay and (3) tracking quality. The location error is the Euclidean distance between the actual location of a tag and its estimated location. We calculate such an error for all localized tags at each time a tag is localized during the simulation and take the average to represent the average location error. The localization delay is the time the system takes to localize all tags. The tracking quality represents the percentage of time during which a tag is localizable. For each performance metric we study the behavior of TSM technique and the Multilateration while running the system using the distributed approach to localize *1000* tags. The total simulation time is *2500sec;* values are averaged over ten different independent runs with distinct random seeds after dumping the first *500sec.*

B. Simulation Results

We examine the simulation results for two cases; when a tag location is estimated using any number of detections ($LAI \geq 1$) and when three or more detections are used in location estimation ($LAI \geq 3$). The latter will naturally result in higher localization accuracy, but may not be always feasible.

1) Average location error:

Fig. 8 depicts the impact of the number of mobile readers on the average location error while considering $LAI \geq 1$. Increasing the number of mobile readers helps the system to localize more tags and/or increase the number of detections used in localization, thus both Multilateration and TSM show better average location error. However; TSM shows an average enhancement of up to *10%* over Multilateration. This enhancement is a result of the time-shifting process, which adapts detections based on the estimated tag speed, allowing more detections to contribute to the localization estimation. The Impact of the tags' speed on the overage location error is depicted in Fig. 9. Both schemes have better accuracy at low mobility and/or when $LAI \geq 3$. Note though that TSM is less affected by tags' speed than Multilateration. For both $LAI \geq 1$ and $LAI \geq 3$, the average location error of TSM and Multilateration converge at low tags' speed values (*0.7m/s*) with TSM outperforming Multilateration by *3%* and *9%*, respectively. At high tags' speed (*1.5m/s*), the average location error of Multilateration for both $LAI \geq 1$ and $LAI \geq 3$ converge due to lack of useful detections at high mobility. On the other hand, TSM maintains its performance in terms of the average location error as the tags' speed estimation and time-shifting processes alleviate the negative effect of tag' speed. In fact, the accuracy of TSM with $LAI \geq 3$ at high mobility shows *100%* improvement over Multilateration.

2) Localization delay:

In Fig. 10, we study the impact of the number of mobile readers on the localization delay while considering both $LAI \geq 1$ and $LAI \geq 3$. As shown in Fig. 10, Multilateration and TSM show almost the same localization delay for $LAI \geq 1$ with lower average location error for TSM (see Fig. 8). However, for $LAI \geq 3$, TSM shows a *77%* reduction in localization delay versus *47%* reduction in case of Multilateration, when the number of readers is doubled from *100* to *200*. In addition to this reduction, TSM keeps track of the tags better than Multilateration (see Fig. 12). While Multilateration awaits for three or more useful detections to localize a tag; TSM, through the time-shifting process, turns otherwise unusable detections into useful ones; reducing the localization delay.

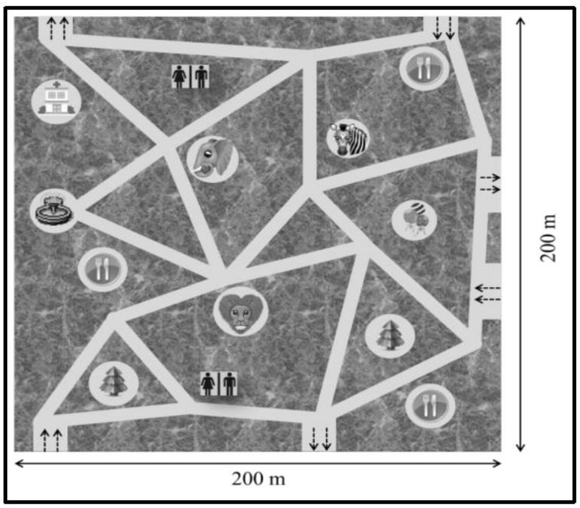


Fig. 7: Simulation Setup.

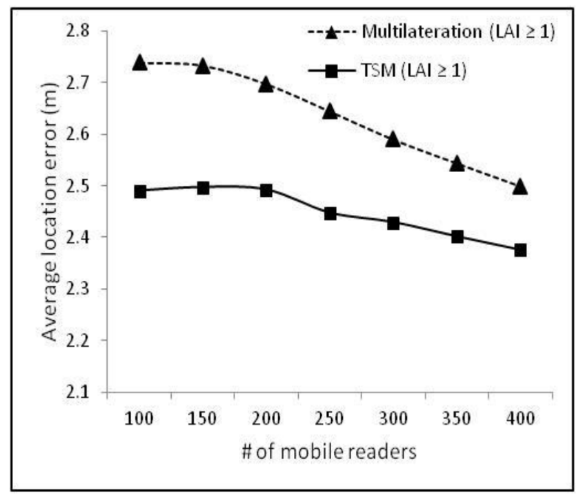


Fig. 8:Effect of # of mobile readers on average location error

3) Tracking quality:

Fig. 11 and Fig. 12 respectively show the impact of tags' speed and number of mobile readers on the tracking quality. As depicted from Fig. 11, for *LAI* ≥ *3*, TSM maintains almost the same tracking quality even at higher tags' speed (conforming to the accuracy results in Fig. 9). Multilateration tracking quality is comparable to TSM at lower speeds, but is up to 30% lower than TSM at tags' speed (*1.5m/s*). In Fig. 12, although the tracking quality is higher for *LAI* ≥ *1*, it comes at the expense of average location error (see Fig. 8). For *LAI* ≥ *3*, TSM and Multilateration have similar performance for low number of readers, whereas, when the number of mobile readers is high (*400*), TSM outperforms Multilateration by an average of *24%*.

C. Field Experiments

We carried out an outdoor RSSI based localization experiment using actual RFID readers and tags to validate our proposed system. The RFID system components included ZR-USB active tag readers and ZT-50 programmable active RFID tags [21]. The reader has a whip omni-directional antenna and is attached to a Windows® based mobile device via universal serial bus

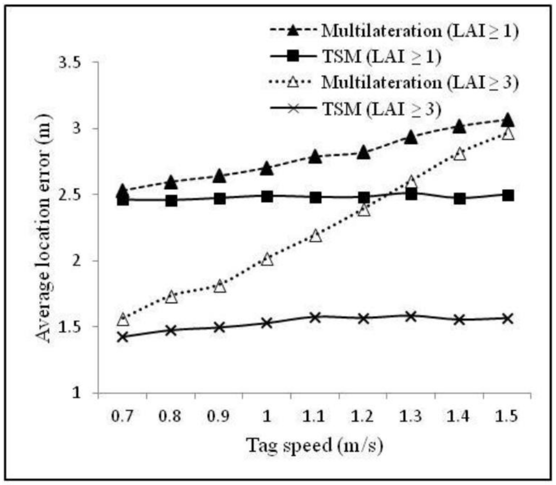


Fig. 9: Effect of tags' speed on average location error (*300 mobile reader,10s pause time*)

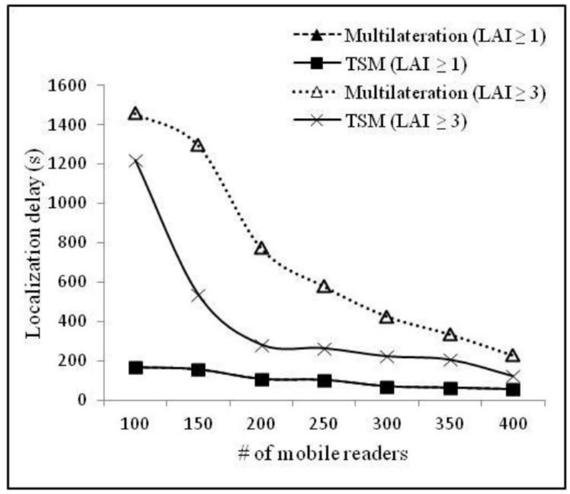


Fig. 10: Effect of # of mobile readers on localization delay.

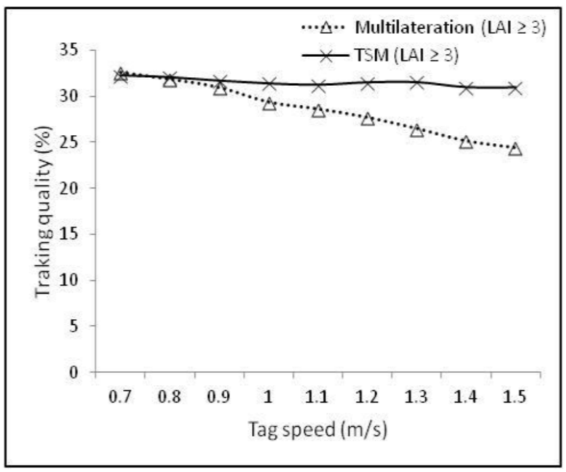


Fig. 11: Effect of tags' speed on tracking quality (*300 mobile reader,10s pause time*).

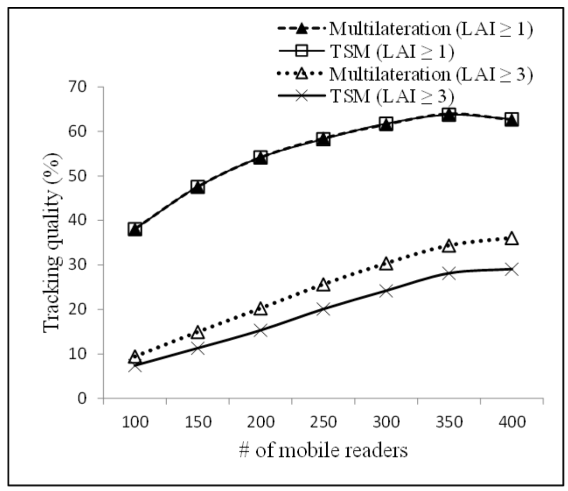


Fig. 12: Effect of # of mobile readers on tracking quality.

(USB) interface. To consider the energy limitation of the mobile device, the reader power is set to *10dbm* and showed a maximum reading range of *32m* in line of sight (LOS). We performed the experiment in the Cricket field at Queen's University where fairs are usually held.

First, we generated a reference curve to map the RSSI from the tag to its distance. To do so we took *17* readings of RSSI at the reader with respect to *17* different tag positions. At each position, we set a time period of *5* seconds and we obtained the average of the RSSI values within that period. The reference curve is shown in Fig. 13. Second, *4* active readers (that are attached to mobile devices) and *21* active tags were placed into a grid in a square area of *25m x 25m* with a spacing of *5m* as shown in Fig. 14. Each reader interrogated the surrounding tags (for *5* seconds in our settings) and updated its detection information on both the tag and the mobile device. Finally, the mobile device shared tag location information in its proximity with the other devices. In our experiment, we estimated the tags' locations by mapping RSSI values to their associated distance in the reference curve executing multilateration equations using a simple C++ based program. Localization results are shown in Fig. 15 (average location error is *2.88m*).

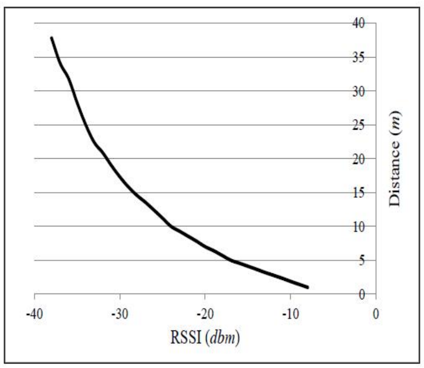


Fig. 13: RSSI to distance reference curve.

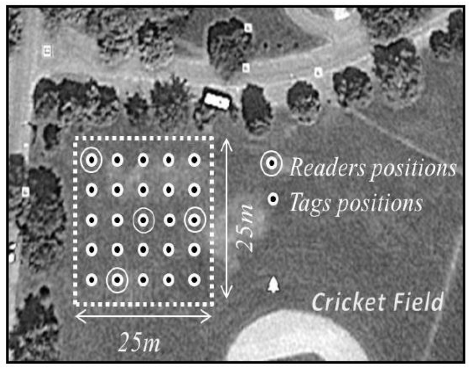


Fig. 14: Experiemnt setup.

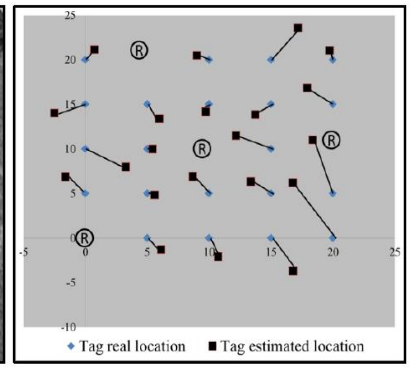


Fig. 15: Experiemnt results.

VII. CONCLUSION

In this paper, we propose a distributed RFID-based mobile object localization system for large dynamic environments as in typical IoT applications. The system leverages the prevalence of RFID readers along with the RFID tags' capability to store a large amount of data, to provide a localization service. In addition, we propose a new localization paradigm "*time-shifted multilateration*" (TSM), which accommodates the environments' dynamics while localizing the tags. TSM estimates the tags' mobility speed and adapts the asynchronous detection information accordingly for better localization. The novelty of our approach is that it: (1) employs RFID crowdsourcing to localize mobile tags, (2) utilizes the tag as the focal point and seizes its memory to store detection and location information and (3) uses time-shifted detection information to enhance localization. We validate our proposed system and study the performance of TSM technique through extensive experiments and a testbed. The results show that TSM can maintain the system performance under different dynamicity settings.

Our approach is based on crowdsourcing from distributed ad hoc RFID readers. When the number of readers is sparse the location accuracy indicator is more likely to be low (fewer than *3* readings). In such a case, it is beneficial to investigate the deployment of a hybrid system with stationary reference readers in addition to the ad hoc ones. On the other hand, in large-scale RFID systems, location query broadcasting may result in large overhead. In our work we utilized the pull strategy in reference [17] for location queries. We plan to investigate the use of alternative pull strategies that can better scale in large RFID systems.

REFERENCES

[1] L. Atzori , A. Iera and G. Morabito The Internet of Things: A Survey, Computer Networks, vol. 54, no. 15, pp. 2787–805, 2010

[2] D. Miorandi, S. Sicari, F. D. Pellegrini, and I. Chlamtac, “Internet of things: Vision, applications and research challenges,” Ad Hoc Networks, vol. 10, no. 7, pp. 1497 – 1516, 2012.

[3] E. Welbourne "Building the Internet of Things Using RFID: The RFID Ecosystem Experience", IEEE Internet Computing, vol. 13, no. 3, pp.48 -55 2009

[4] T. Sanpechuda and L. Kovavisaruch "A review of RFID localization: Applications and techniques", Proc. 5th Int. Conf. Elect. Eng./Electron., Comput., Telecommun. Inf. Technol., pp.769 -772 2008

[5] J. Yu and S. Lim, "Design of an Electrically Small, Circularly Polarized, Parasitic Array Antenna for an Active 433.92-MHz RFID Handheld Reader," Antennas and Propagation, IEEE Transactions on , vol.60, no.5, pp.2549,2554, May 2012

[6] EPC Global, Class 1 Generation 2 UHF air interface protocol standard v1.2, http://www.epcglobalinc.org, 2008

[7] H.J. Lee and M.C. Lee, "Localization of Mobile Robot Based on Radio Frequency Identification Devices," SICE-ICASE, International Joint Conference, pp.5934-5939, October 2006

[8] S. Han, H.-S. Lim, and J.-M. Lee, “An efficient localization scheme for a differential-driving mobile robot based on RFID system,” IEEE Trans. Ind. Electron., vol. 54, no. 6, pp. 3362–3369, Nov. 2007.

[9] V. Ekambaram and K.Ramchandran, "SLAC-RF: Simultaneous 3D Localization of mobile readers and Calibration of RFID supertags," EECS Department, University of California, Berkeley, Technical Report UCB/EECS-2012-188, 2012.

[10] C. Wang, H. Wu, and N.-F. Tzeng. RFID-based 3-D positioning schemes. In Proc. of INFOCOM, pages 1235-1243, 2007.

[11] W. Zhu; J. Cao; Y. Xu; L. Yang; and J. Kong, "Fault-tolerant RFID reader localization based on passive RFID tags," INFOCOM, 2012 Proceedings IEEE , vol., no., pp.2183,2191, 25-30 March 2012

[12] L. M. Ni , Y. Liu , Y. C. Lau and A. P. Patil "LANDMARC: Indoor location sensing using active RFID", Wireless Netw., vol. 10, no. 6, pp.701 -710 2004

[13] A. Bekkali, H. Sanson, and M. Matsumoto. RFID indoor positioning based on probabilistic RFID map and kalman filtering. In Proc. of WiMOB, 2007.

[14] Y. Zhao, Y. Liu, and L.M. Ni. VIRE: Active RFID-based localization using virtual reference elimination. In Proc. of ICPP, 2007.

[15] M. Bouet and G. Pujolle. L-virt: Range-free 3-d localization of rfid tags based on topological constraints. Computer Communications, 32:1485- 1494, 2009.

[16] C. Hekimian-Williams, B. Grant, and P. Kumar, "Accurate localization of RFID tags using phase difference," in Proceedings of the IEEE International Conference on RFID, 2010.

[17] L. Eslim, W. Ibrahim and H.S. Hassanein, "GOSSIPY: A Distributed Localization System for Internet of Things using RFID Technology" to appear, IEEE GLOBECOM, 2013 [Available: http://queenstrl.ca/GOSSIPY.pdf]

[18] S. Zhang; J. Wang; X. Liu; J. Cao, "Range-free selective multilateration for anisotropic wireless sensor networks," Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2012 9th Annual IEEE Communications Society Conference on , vol., no., pp.299,307, 18-21 June 2012

[19] C.C. Holt, Forecasting seasonals and trends by exponentially weighted moving averages, International Journal of Forecasting, Volume 20, Issue 1, January–March 2004, Pages 5-10

[20] G.S. Kuruoglu, M. Erol, and S. Oktug, "Localization in wireless sensor networks with range measurement errors, " Advanced International Conference on Telecommunications, vol. 0, pp. 261-266, 2009.

[21] TagSense Inc. [Online] Available: http://www.tagsense.com/

[22] The network simulatorns-3 [Online]. Available: http://www.nsnam.org/

[23] J. Tian, J. Hahner, C. Becker, I. Stepanov and K. Rothermel. Graph-based Mobility Model for Mobile Ad Hoc Network Simulation, in the Proceedings of 35th Annual Simulation Symposium, in cooperation with the IEEE Computer Society and ACM. San Diego, California. April 2002.

[24] RL. Knoblauch, MT. Pietrucha, M. Nitzburg Field studies of pedestrian walking speed and start-up time. Transportation Research Record 1538. Washington (DC): National Research Council, Transportation Research Board; 1996 Dec. p. 27–38

[25] H. Wei, Q. Wan, Z. Chen, and S. Ye, “A novel weighted multidimensional scaling analysis for time-of-arrival-based mobile location,” IEEE Transactions on Signal Processing, vol. 56, no. 7, pp. 3018–3022,2008