A Cooperative Localization Scheme Using RFID Crowdsourcing and Time-Shifted Multilateration

Lobna M. Eslim, Hossam S. Hassanein, Walid M. Ibrahim and Abdallah Alma'aitah
Telecommunications Research Lab
School of Computing, Queen's University, Kingston, ON, Canada K7L 3N6
{lobna.eslim, hossam, walid, abdallah}@cs.queensu.ca

***Abstract*— RFID technology as an enabler of the Internet of Things (IoT) is extensively utilized for object localization. Existing RFID-based object localization techniques follow a centralized and coordinated approach. Indeed, none is designed to leverage RFID crowdsourcing for the purpose of object localization. In this paper, we propose a cooperative scheme to localize mobile RFID tags using heterogeneous, distributed and dynamic mobile RFID readers in indoor/outdoor environments. In addition, 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 scheme and the TSM technique through extensive simulations using ns-3. Results show that our approach can achieve accurate location estimation in typical IoT settings**.

***Index Terms*—IoT, RFID, localization, crowdsourcing, multilateration.**

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 and smart transportation. [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 be localized to take advantage of such context. Providing localization while considering the IoT characteristics in terms of scalability, heterogeneity and mobility, is a challenging problem.

Radio Frequency Identification (RFID) is one of pivotal enabling technologies of IoT for the purpose of identification. In the past few years, RFID development has achieved unceasing technical progress in addition to cost reductions and standardization [3]; resulting in unconventional utilizations beyond mere identification. The demand for embedding the localization capability into the IoT infrastructure sparked the use of RFID systems for object localization [8]-[17]. RFID-based localization systems can be broadly categorized into reader localization and tag localization [4]. In reader localization, objects are equipped with RFID readers and localized, based on connectivity information with a set of active or passive tags deployed at known locations. Whereas in tag localization, objects are attached with RFID tags and localized through a set of coordinated RFID readers which report to a central server for location estimation. The reader localization approach is not cost effective for IoT settings, whereas the implementation requirements of the tag localization approach make it more suited for indoor environments.

In this paper, we propose a cooperative scheme to localize mobile objects based on crowdsourcing in indoor/outdoor environments. The scheme takes advantage of the following: (1) objects can be easily identified by passive RFID tags, which are inexpensive and widely available, (2) embedded RFID readers in mobile devices are being rapidly adopted due to the great interest of RFID manufacturers, along with the rapid advancements in antenna design for handheld RFID readers [5] and (3) RFID tags are capable of storing data in addition to their unique identifiers [6] and their memories are expected to play a significant role in data exchange [7] . In our proposed scheme, the *Detectors* (heterogeneous, independent and dynamic RFID readers) periodically detect tags in their interrogation zones and utilize 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 existing tag localization techniques.

Existing localization techniques operate with the assumption that available detection information is sufficient and obtained simultaneously. This 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 prevents having sufficient and synchronous detection information for each tag. Therefore, we propose the localization technique *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 object localization system utilizing reader crowdsourcing,
- utilize tags' memory to store reader detection information and location information that can be read by other passing readers, and
- use unsynchronized (*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 mobile RFID readers have relatively short reading ranges which are not sufficient to follow mobile tags.

We validate the proposed system through extensive simulations using ns-3. Results show that our approach can achieve accurate location estimation in typical IoT settings.

978-1-4799-3780-6/14/$31.00 ©2014 IEEE

The remainder of this paper is organized as follows: Section II reviews some of the related work and shows the motivation of our proposed approach. In Section III we propose a cooperative scheme for localizing mobile RFID tags using heterogeneous, distributed and dynamic mobile RFID readers. TSM technique is explained in Section IV and a use case is presented in Section V. Section VI presents the 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 reader localization systems [8]-[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 [8], the authors attach reference tags to the floor and ceiling into a square pattern to localize a mobile reader using the weighted average method and a weighting function. While in [9], the same approach is followed but the accuracy of localization is enhanced by rearranging the reference tags into a triangular pattern. However, in such systems, the required number of reference tags is relatively high. The authors of [10] 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, the work in [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 a cost effective infrastructure. However, they suffer from the high cost of associating an RFID reader with every object, rendering such an approach infeasible for IoT settings.

In tag localization systems [12]-[17], 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. Improvements to LANDMARC were proposed in [13] for reference tags placement and their contribution to tag location estimation. VIRE [15] and L-VIRT [16] use virtual reference tags instead of a 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 [13] and [14] with support of landmarks. The centralized and fixed infrastructure-based systems provide limited scalability and may not be a practical solution for IoT settings especially in outdoor environments.

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

III. Reader crowdsourcing scheme

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

A. Scheme Model and Components

The scheme has two components:

- ***Tags*** – representing the objects to be localized. These objects can be either stationary or mobile and are identified by passive RFID tags. The number of tags is much larger than the *Detectors* in the scenarios under study.
- ***Detectors*** – representing the mobile 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 smartphones or handheld RFID readers. The *Detectors* are assumed to be capable of acquiring their locations at any given time and they are authorized to interrogate all *Tags* in the given environment and update their memory.

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 mobile RFID readers (*Detectors*), each tag stores its estimated position at any given time in its memory. Fig. 1 shows the general framework of our scheme.

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 window within which detection events are eligible to contribute to localization.

B. Exchanged Information

During the operation, two types of information are created:

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

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

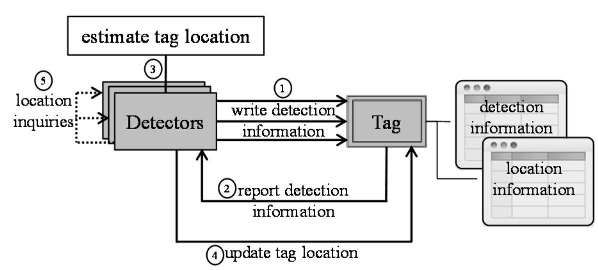

Fig. 1: General framework of reader crowdsourcing scheme.

C. System Operation

As shown in Fig. 1, *Tags* are used as the focal point for storing and exchanging location information. 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 [18]. In our scheme 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. We next explain tags notification, tags localization, and location query process.

1) Tags notification

Tags' memories are updated by passing *Detectors*. The objectives are: (1) maintain detection records on the tag memory to be used by other passing *Detectors* for tag localization and (2) allow a tag to know its estimated position at every *tolerance interval*. For the former, each *Detector* d_j in D interrogates the tags in its proximity. For each successfully identified tag t_i, d_j creates a detection record and writes such record into the memory of t_i (see Fig. 2). As shown in the figure, updating the tag's memory by a subset of D allows the tag to hold multiple time-stamped detection records which are limited to either the tag's memory or the time window of the *tolerance interval*. 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 outdated records with respect to the *tolerance interval.*

2) Tags localization

In 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 tag location information is then updated.

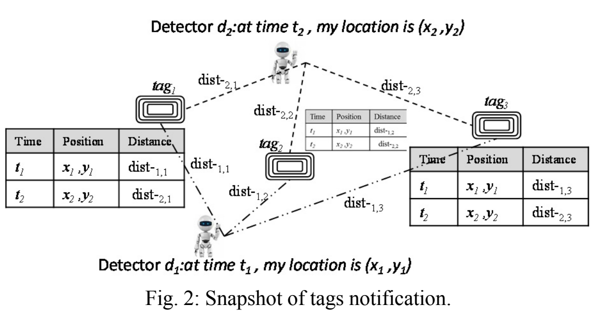

Fig. 2: Snapshot of tags notification.

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.*

Algorithm I: Tags localization Algorithm

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

```
1  for each tolerance interval do
2    for each t_i in my proximity do
3      set Detect_info(t_i) = get t_i .detection information
4      set filtered_info(t_i)
5      for each record r_j in Detect_info (t_i) do
6        if r_j.time < current time – tolerance interval then
7          Detect_info(t_i).delete( r_j)
8        end if
9      end for
10     for each record r_j in Detect_info (t_i) do
11       if j = 1 then    filtered_info(t_i).add(r_j)
12       else
13         for each r_k in filtered_info(t_i) do
14           if r_j do not intersect with r_k then
15             Detect_info(t_i).delete( r_j) and break
16           end if
17         end for
18         filtered_info(t_i).add (r_j)
19       end if
20     end for
21     set LAI = filtered _info(t_i).size
22     t_i .position = Estimate_Loc (filtered_info(t_i))
23     Update t_i .location information ( Get (current time),
                t_i.position, LAI)
24   end for
25 end for
```

Tags notification process can result in 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.

3) 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 the scheme in [18] 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 (the *LAI* can be used to decide which location is more accurate in case multiple locations are estimated within a same *tolerance interval*).

D. User Authentication and Privacy

Our approach suggests that in dynamic environments, the available RFID crowdsourcing in terms of mobile RFID readers along with tags' memories can be leveraged to provide localization service. This is achieved by allowing mobile RFID readers, to locate surrounding RFID tagged objects and update tags' memories accordingly. For user authentication and privacy, scalable light-weight tree-based category of privacy preserving authentication (PPA) protocols such as [19] can be adopted. In such protocols, authenticated keys and their hashed values can be stored in the memories of both tags and readers, under a secure channel, during user's registration in localization service. During the system operation, both readers and tags authenticate each other by matching the received hashed value of the key to the one stored in their memories. Details of user authentication and privacy are the subject of further research.

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 [19][20]. 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 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 (*regardless of its direction of movement*). 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 as discussed later. 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. 3 illustrates an example of the TSM technique and shows how the time-shifting process takes place for four detections. We next detail the time-shifting processes.

Definition 1 (detection set): Given the set of *Detectors D*, the detection set of a tag t_i 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. 4 (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. 4 (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. ■
(Theorem 2 can be generalized for any number of detections.)

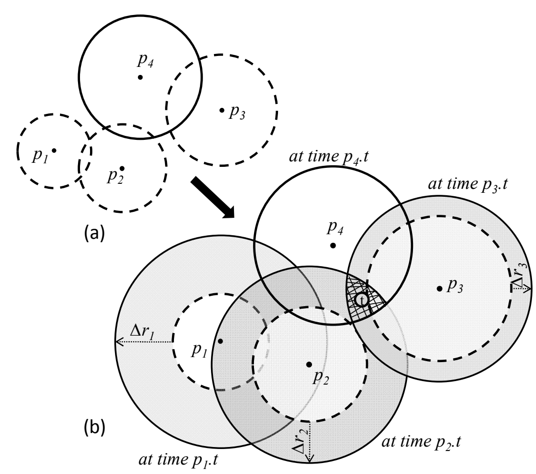

Fig. 3: Example on TSM technique. (a) The set of detections P = {p1, p2, p3, p4} for a tag t at different times within the same tolerance interval where p4 is the most recent. (b) The set of shifted detections after expanding p1, p2, p3 by Δr1, Δr2 and Δr3 resprctlively, the tag t is expected to be in the shaded area.

A. The time-shifting process and Multilateration:

TSM starts updating a tag's 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, methods such as Kalman filter [21] and exponentially weighted moving average [22] can be used to estimate the tag speed. In this work, we adopted the exponentially weighted moving average method. If there are no previous estimated locations for the tag to be localized, the initial speed is used though. 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
for j=1 to k-1 do
    Δr_j = speed(t_i) * (p_k.t – p_j.t)
    p_j.r = p_j.r + Δr_j
end for
return P(t_i)
```

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

$$(x - x_i)^2 + (y - y_i)^2 = d_i^2 \quad (1)$$

where (x_i, y_i) are the x and y coordinates of the i^{th} 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:

$$(x - p_j.x)^2 + (y - p_j.y)^2 = (p_j.r + (s * (p_k.t - p_j.t)))^2 \quad (2)$$

The same solution applied to equation (1) [23] can be applied to solve equation (2) without affecting the overall complexity which is $O(k^3)$ where k is the size of the set $P(t_i)$.

By referring to *Theorem 2*, TSM technique can accurately localize the mobile tag t_i using a set of k asynchronous spatial information about t_i, and its average speed s regardless of its moving direction. 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:

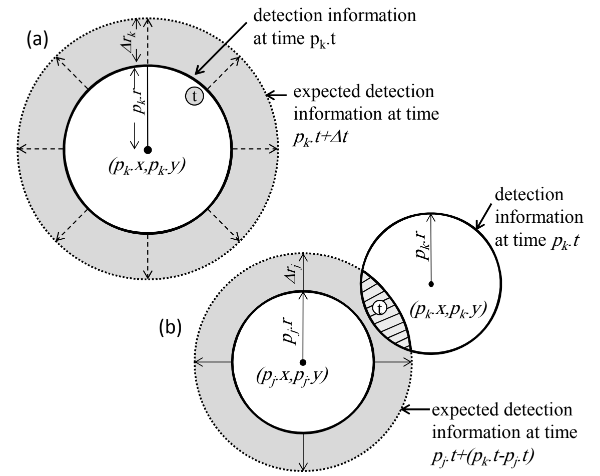

Fig. 4: The concept of time-shifting process. (a) time-shifting of one detection after time Δt. (b) How time-shifting affects location accuracy

- **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. 5 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.

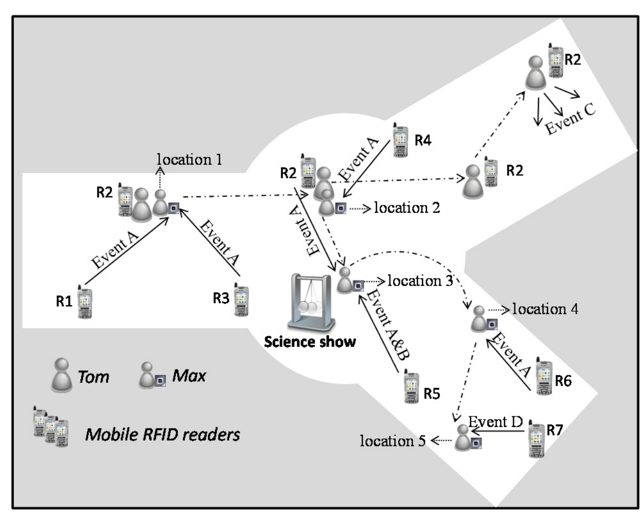

Fig. 5: Use case scenario.

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. 5. 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.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the reader crowdsourcing system along with the TSM technique through extensive simulation experiments using ns-3. 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. Using the ns-3 network simulator [24] and based on Graph-Based Mobility Model for Ad Hoc Networks [25], we simulate an area of *200m x 200m* containing *14* point of interest, which are linked using pathways of *8m* 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* [26]. 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*.

We introduce an error in measuring the distance between a tag and a reader as follows. In measuring the distance between a tag t_i and a reader d_j, 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 $\sigma^2 = (noise\_free\ distance)^2/SNR$ [27]. For location determination of mobile RFID readers in outdoor environments, GPS-based positioning, coupled with street maps, is used with typical accuracies of 1-3 meters. While indoors, where GPS signals are no longer available, wireless technologies such as WiFi, Ultra Wide Band (UWB), Ultrasonic, or RFID can be used for positioning, providing meter-level accuracy [28] [29]. We consider an error in both x and y coordinates of a reader position as a zero-mean white Gaussian process ($\mathcal{N}(0, \sigma_i^2)$), where σ is a value ranged from *0.2m* to *1.4m*. When this error is not mentioned, we assume that mobile RFID readers are accurately localized.

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 one or more detections (*LAI* ≥ *1*) and when at least three detections are used in location estimation (*LAI* ≥ *3*). The latter will naturally result in higher localization accuracy, but may not always be feasible.

1) Average location error:

Fig. 6 depicts the impact of the number of mobile readers on the average location error while considering *LAI* ≥ *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 *6%* 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. 7. Both schemes have better accuracy at low mobility and/or when *LAI* ≥ *3.* Note though that TSM is less affected by tags' speed than Multilateration even when *LAI* ≥ *1* due to the time-shifting process. For *LAI* ≥ *3*, the average location error of TSM and Multilateration converge at low tags' speed values (*0.7m/s*) with TSM outperforming Multilateration by *9%.* At high tags' speed (*1.5m/s*), the average location error of Multilateration for both *LAI* ≥ *1* and *LAI* ≥ *3* increased due to lack of useful detections at high mobility, whereas better result is for *LAI* ≥ *3* but on account of tracking quality (see Fig. 11).On

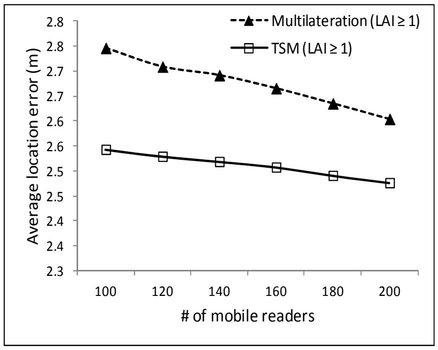

Fig. 6: Effect of # of mobile readers on average location error.

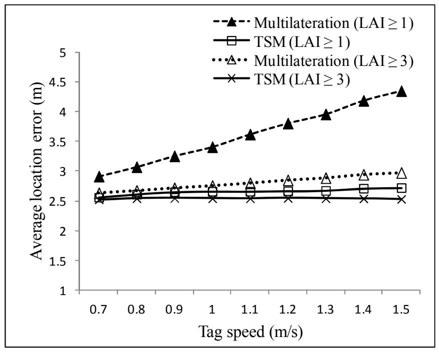

Fig. 7: Effect of tags' speed on average location error (*200 mobile reader,10s*

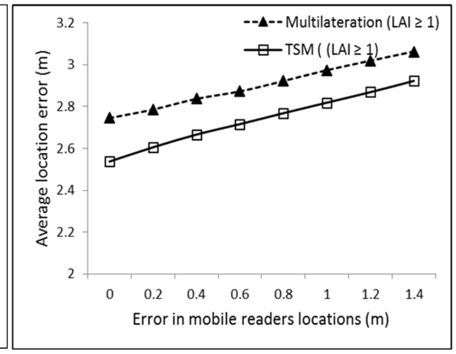

Fig. 8: Effect of error in readers positions on average location error (*200 mobile reader*).

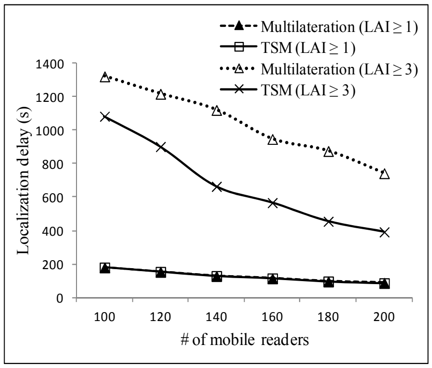

Fig. 9: Effect of # of mobile readers on localization delay.

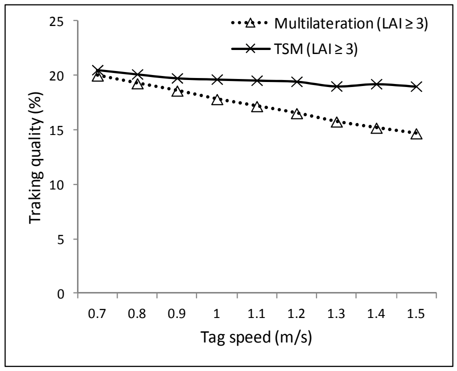

Fig. 10: Effect of tags' speed on tracking quality (*200 mobile reader,10s pause*

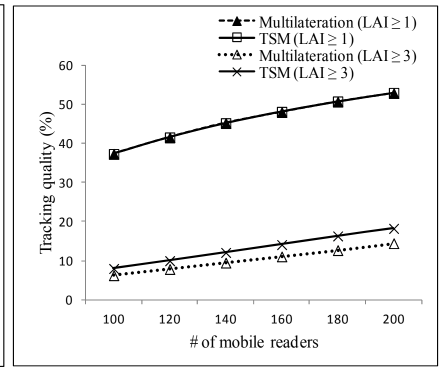

Fig. 11: Effect of # of mobile readers on tracking quality.

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 tags' speed. In fact, the accuracy of TSM with *LAI* ≥ 1 at high mobility shows *100%* improvement over Multilateration. Fig. 8 shows that the error in mobile readers' positions does not aggressively affect the average location error. As shown in the figure, the average location error of Multilateration and TSM are affected by *8%* and *10%* respectively when the error in mobile readers' positions moves from *0.2m* to *1.4m*.

2) *Localization delay:*

In Fig. 9, we study the impact of the number of mobile readers on the localization delay while considering both *LAI ≥ 1* and *LAI ≥ 3*. As shown in the figure, Multilateration and TSM show almost the same localization delay for *LAI ≥ 1* with lower average location error for TSM (see Fig. 6). However, for *LAI ≥ 3*, TSM shows a *64%* reduction in localization delay versus *44%* 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. 11). 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.

3) *Tracking quality:*

Fig. 10 and Fig. 11 respectively show the impact of tags' speed and number of mobile readers on the tracking quality. As depicted from Fig. 10, for *LAI ≥ 3*, TSM maintains almost the same tracking quality even at higher tags' speed (conforming to the accuracy results in Fig. 7). 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. 11, although the tracking quality is higher for *LAI ≥ 1*, it comes at the expense of average location error (see Fig. 6). For *LAI ≥ 3*, TSM and Multilateration have similar performance for low number of readers, whereas, when the number of mobile readers is high (*200*), TSM outperforms Multilateration by an average of *26%*.

VII. Conclusion

In this paper, we propose a distributed RFID-based mobile object localization scheme for large dynamic environments as in typical IoT applications. The scheme leverages the prevalence of RFID readers along with the RFID tags' capability to store data, to provide a localization service. In addition, we propose a localization technique "*time-shifted multilateration*" (TSM), which accommodates the environments' dynamics while localizing 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. 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 per location). 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 utilize the pull strategy in [18] for location queries We plan to investigate the use of alternative pull strategies that can better scale in large RFID systems.

ACKNOWLEDGMENT

This work was made possible by NPRP grant # NPRP4-553-2-210 from the Qatar National Research Fund (a member of The Qatar Foundation). This research is also supported in part by the Canadian Natural Science and Engineering Research Council (NSERC). The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] L. Atzori , A. Iera and G. Morabito The Internet of Things: A Survey, Computer Networks, vol. 54, no. 15, pp. 787–805, 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] Z. Junyi, and S. Jing, RFID localization algorithms and applications—a review, Journal of Intelligent Manufacturing, Vol. 20, pp. 695−707, 2009.

[5] P. V. Nikitin and K. V. S. Rao "Helical antenna for handheld UHF RFID reader", Proc. IEEE Int. Conf. on RFID, pp.166 - 173, 2010.

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

[7] I. Farris, A. Iera, and S. C. Spinella, "A novel paradigm to exchange data in RFID piconets," IEEE International Conference on RFID, pp.215 – 222, 2013.

[8] 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, 2006.

[9] 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, 2007.

[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 , pp.2183 – 2191, 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] G. Jin, X. Lu, and M. S. Park, "An Indoor Localization Mechanism Using Active RFID Tag," IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing , vol. 1, pp. 40 – 43, 2006.

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

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

[16] M. Bouet and G. Pujolle. L-virt: Range-free 3-d localization of rfid tags based on topological constraints. Computer Communications, vol. 32, no. 13-14,pp. 1485-1494, 2009.

[17] 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, pp. 89 – 96, 2010.

[18] L. Eslim, W. Ibrahim and H.S. Hassanein, "GOSSIPY: A Distributed Localization System for Internet of Things using RFID Technology" *Global Communications Conference (GLOBECOM), 2013 IEEE* , pp.162 – 167, 2013

[19] T. Li; W. Luo; Z. Mo; Shigang Chen, "Privacy-preserving RFID authentication based on cryptographical encoding," Proceedings IEEE of INFOCOM, , pp.2174 – 2182, Mar. 2012.

[20] S. Zhang; J. Wang; X. Liu; J. Cao, "Range-free selective multilateration for anisotropic wireless sensor networks," 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), , pp.299 – 307, 2012.

[21] R. Kalman. A new approach to linear filtering and prediction problems, In Transactions of the ASME Journal of Basic Engineering, pp. 34 – 45, 1960.

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

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

[24] ns-3 [Online]. Available: http://www.nsnam.org/

[25] 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, 2002.

[26] R. 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;. pp. 27 – 38, 1996.

[27] 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.

[28] http://spectrum.ieee.org/telecom/wireless/new-indoor-navigation-technologies, [Online], last accessed Mar. 4th 2014

[29] P. Lazik , A. Rowe, Indoor pseudo-ranging of mobile devices using ultrasonic chirps, Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems, 2012