GOSSIPY: A Distributed Localization System for Internet of Things using RFID Technology

Lobna M. Eslim, Walid M. Ibrahim, and Hossam S. Hassanein

Telecommunications Research Lab
School of Computing, Queen's University, Kingston, ON, Canada K7L 3N6
{lobna.eslim, walid, hossam}@cs.queensu.ca

***Abstract*—The popularity of smart objects in our daily life fosters a new generation of applications under the umbrella of the Internet of Things (IoT). Such applications are built on a distributed network of heterogeneous context-aware devices, where localization is a key issue. The localization problem is further magnified by IoT challenges such as scalability, mobility and the heterogeneity of objects. In existing localization systems using RFID technology, there is a lack of systems that localize mobile tags using heterogeneous mobile readers in a distributed manner. In this paper, we propose the GOSSIPY system for localizing mobile RFID tags using a group of ad hoc heterogeneous mobile RFID readers. The system depends on cooperation of mobile readers through time-constrained interleaving processes. Readers in a neighborhood share interrogation information, estimate tag locations accordingly and employ both proactive and reactive protocols to ensure timely dissemination of location information. We evaluate the proposed system and present its performance through extensive simulation experiments using ns-3.**

I. INTRODUCTION

The ability to embed intelligence into everyday objects creates what is known as "smart objects". Such smart objects sense, process and communicate under certain circumstances with each other and/or with people [1]. The possibility of having millions of such smart objects sprouted the Internet of Things (IoT), which was originally used by the RFID development community, and refers to a new evolution of the current Internet. IoT unlocks the door for a new generation of applications and solutions, where each smart object plays a role in a distributed network of heterogeneous context-aware devices [2] [3]. Position information in such applications is intrinsic for the context knowledge to be useful [4] [5]. This adds more attention to object localization in IoT scenarios as: (1) objects may scale from millions to billions, (2) typically large numbers of objects are mobile and (3) objects are heterogeneous and vary in terms of local capabilities and communication characteristics.

Although RFID is a prominent technology for object identification, it does not have its own localization principles. However, the same localization principles for wireless networks can be applied with adaptation to the characteristics of such technology [6]. Several RFID localization systems have been proposed to localize either mobile or stationary objects and are categorized into; reader localization and tag localization [7]. In the former, objects are attached with RFID readers and localized by deploying a large number of active and/or passive tags at known locations. While in the latter, objects are identified by RFID tags and localized using fixed infrastructure of RFID readers that report to a central localization server. In a typical IoT scenario, it is common to have an environment that contains a large number of tagged objects along with group of ad hoc heterogeneous and independent mobile RFID readers. Prominent examples are malls, airports and hospitals. In such scenarios, the impossibility of attaching an RFID reader to each object of interest, along with the non feasibility of deploying a fixed infrastructure, necessitate a distributed localization system that is robust, scalable and provides acceptable accuracy according to the application.

This paper proposes GOSSIPY, a distributed localization system, for localizing mobile objects using heterogeneous, independent and dynamic RFID readers. The GOSSIPY system leverages mobile RFID readers in place to localize surrounding RFID tags based on information sharing under time constraints. GOSSIPY operates in three interleaving processes, each mobile reader periodically: (1) collects tags proximity information and exchanges this information with readers in the neighborhood, (2) estimates tags locations based on collected and exchanged proximity information and (3) disseminates the tags estimated locations using the push/pull strategy in a peer-to-peer mechanism. Exchanging proximity information between neighbors is proposed to improve location accuracy and to alleviate the collinearity problem [8]. In addition, the dissemination of estimated locations allows readers to know the locations of tags beyond their interrogation zones. Fig. 1 represents the generic architecture for the GOSSIPY system.

The remainder of this paper is organized as follows. Section II reviews some of the related work and highlights the motivations behind introducing GOSSIPY system. Section III describes the system's components, assumptions and its interleaving processes. Section IV explains the performance evaluation methodology and analyzes the results. Finally, our conclusion is given in Section V.

II. RELATED WORK AND MOTIVATION

Several localization systems based on RFID technology have been proposed in the literature. These systems can be broadly classified as reader localization and tag localization.

In reader localization systems, readers are usually mobile (e.g., robots) and localized by deploying a large number of active and/or passive tags at known locations. These tags, named reference tags, work as landmarks in the area of interest and are used by the mobile reader to estimate its absolute location, orientation and even velocity [9] - [12]. In reference [9], tags are attached to the floor and arranged in a

978-1-4799-1353-4/13/$31.00

(a) proximity information collection and sharing (b) location information sharing

Fig. 1: Generic architecture of the GOSSIPY system

square pattern. When the reader detects some of these tags, while knowing their locations, it can estimate its location using weighted average method. However, to provide good accuracy, the distance between tags should be reduced. So, a large number of tags are required which adds to the system cost and feasibility. Another similar system is proposed in [10], which enhances location accuracy but avoids using large numbers of tags by arranging them in a triangular pattern. To avoid the dense deployment of reference tags, SLAC-RF [12] proposes specialized tags named supertags. Each supertag is an array of RFID tags which are arranged to simulate a virtual antenna array. A mobile reader which navigates the area estimates its position using the phase difference of received signals with respect to supertags along with inertial navigation system (INS) measurements. Although the aforementioned systems provide good accuracy using cost effective and easy to setup infrastructure, their usability is limited to support autonomous mobile robot applications.

Tag localization systems [13] - [19] are devised to locate tags based on detection events at multiple readers. Such detection events are sent back to a central location server to estimate tag location using the trilateration, fingerprinting, or probabilistic technique. Based on the inter-tag distances at multiple fixed readers, systems in [13] and [14] have been proposed where SpotON [13] is the pioneer. SpotON laterates tag location using an aggregation algorithm; ignoring measurement uncertainty caused by the environment dynamics. Based on the fingerprinting technique and in order to decrease the effect of RSS measurement uncertainty hence improve the accuracy, LANDMARC [15], a well-known RFID-based localization system, introduces the concept of reference tags to calibrate the environment dynamics. To localize a tag, LANDMARC compares its RSS measurements to those of reference tags, and estimates the coordinates of the tag based on the locations of the k-nearest reference tags. Systems such as [17] and [18] follow the same approach as LANDMARC but both avoid the dense deployment of reference tags by using virtual reference tags instead. In such systems, the RSS readings of each virtual reference tag for each reader are calculated using those of surrounding real reference tags. For example, VIRE [17] calculates the RSS of virtual reference tags by the linear interpolation algorithm. RSS readings of the tracked object are then compared to those of reference tags either real or virtual. Positions with similar readings are considered as plausible locations which are then filtered using an elimination algorithm.

Tag localization systems cater to a wide range of applications. However, they rely on expensive and static infrastructures and provide limited robustness and scalability, rendering them unfit for IoT settings. The objective of our work is to design a localization system which localizes tagged objects using relatively small numbers of mobile readers, while providing good location accuracy through cooperation among readers.

III. GOSSIPY SYSTEM

The goals of GOSSIPY are to:

1) *localize mobile objects and be able to control the location accuracy based on the application needs,*
2) *leverage the available mobile RFID infrastructure in place for localization, and*
3) *Sustain robustness and scalability.*

To accomplish these, GOSSIPY is composed of three interleaving processes: (1) proximity information collection and sharing, (2) location estimation and (3) location information dissemination. The following subsections elaborate the system's components, define our assumptions, formalize the problem and explain in detail each of its interleaving processes.

A. System Components and Used Notations

Given an RFID system of n tags and m mobile readers, we consider a two-dimensional localization problem of the tags using the mobile readers in an outdoor area. The system has the following components:

1) *Mobile RFID readers:* A set of m mobile readers which move in random trajectories.
2) *RFID active or passive tags:* A set of n mobile or stationary tags. The tags move in random trajectories.
3) *Proximity table:* A table for each mobile reader. It contains time-stamped proximity information about any tag in its interrogation area along with proximity information received from neighbors.
4) *Location table:* A table for each mobile reader. It contains the estimated positions of detected tags. The table size is of order (mn).

Thereafter in the paper, we use the following notations:

- $R = \{r_1, r_2, ..., r_m\}$ is the set of *m* mobile readers.
- $T = \{t_1, t_2, t_3, ..., t_n\}$ is the set of *n* mobile or stationary tags.
- $CR_i = \{cr_{i1}, cr_{i2}, ..., cr_{ik}\} \subseteq R$ is a subset of the mobile readers that cover a tag t_i at time τ.
- $NR_i = \{nr_{i1}, nr_{i2}, ..., nr_{il}\} \subseteq R$ is a subset of the mobile readers that are in neighborhood of reader r_i at time τ.
- *detection interval* is a time interval at which each $r_i \in R$ detects tags in its vicinity, updates its proximity table and shares such information with NR_i.
- *localization interval* is a time interval at which each $r_i \in R$ estimates the locations of detected tags based on proximity information in hand.
- *update timeout* is the validity time of location update messages in the location dissemination process.

Typically, *detection interval* **<** *localization interval***<** *update timeout.*

B. Assumptions and Problem Definition

We assume that: (1) each mobile reader can acquire its position at any given time using one of the positioning systems for mobile readers (e.g., GPS, WiFi, anchors, etc.) and (2) the readers can reach neighbor readers to share location information. Formally, the tag localization problem is defined as follows. Given a set of RFID tags and a set of ad hoc mobile RFID readers, find the coordinates of each tag in two dimensions at any given time and share this location information with those interested. This problem can be formulated using the aforementioned notations as follows. Given *T* and *R*, for each $r_i \in R$ and for each *localization interval*, estimate the coordinates (x_j, y_j) of all detected $t_j \in T$ using proximity information of readers in CR_i and share such location information with readers in NR_i.

C. Proximity Information Collection and Sharing

In this process, mobile readers maintain their proximity table while they are moving. At every *detection interval*, each mobile reader r_i in *R* interrogates tags in its proximity, creates a detection record in its proximity table for each detected tag, and shares such records with all readers in NR_i. Each detection record contains: *time, tag ID, reader ID, reader position, tag to reader distance, and a flag*. The flag of value 0 distinguishes the first hand proximity information to limit its sharing among one hop neighboring readers only as illustrated in Fig. 2. The number of detection records is limited to a time window which is equal to the *localization interval* after which replacement takes place. Given that the detection interval is smaller than the localization interval; each localization interval may hold multiple time-stamped detection records for the same tag either from a reader itself or from one-hop neighbors. This in turn provides more accuracy and decreases the probability of collinear detections. To accommodate mobile tags, we can acquire more detection records within each localization interval by decreasing the detection interval with respect to the *localization interval.*

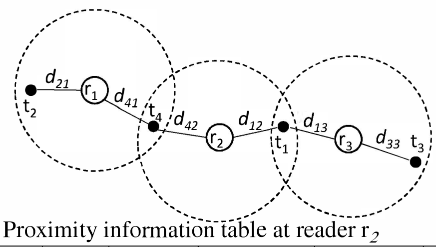
D. Location Estimation

In case of stationary tags, typically most detection records positively contribute to localization accuracy. However, due to the probability of tag mobility, not all detection records are useful in localization. As a preprocessing step for location estimation, detection records should be filtered to exclude those that may negatively affect the localization accuracy, taking detection time into consideration. To do so, we consider each detection record as a circle, which is centered over the reader position and has radius equal to the distance from the reader to the tag of interest. Starting with the most recent circle and follow with others one by one, we exclude circles that do not contribute to the intersect area of all previous circles. The worst case is when one detection is included resulting in a less accurate location. Otherwise, the common lateration or multilateration technique is used according to the number of useful detections. The entire location estimation process is illustrated in Algorithm I, and the filtering steps (3-15) are depicted in Fig. 3 for only one tag for simplicity. In Fig. 3, the dotted circles are excluded since they negatively contribute to the location accuracy

Algorithm I: location estimation

Input: proximity table **Output:** location table

```
1  for each localization interval do
2    for each tag_i in proximity table do
3      for each detection d_j in proximity table for tag_i do
4        if j = 1 then
5          filtered_proximity_info_list.add (d_j)
6        else
7          to_add_flag = True
8          for each d_k in filtered_proximity_info_list do
9            if (distance between (d_j.reader position, d_k.reader
               position) > (d_j.distance + d_k.distance) then
10             to_add_flag = False
11             break
12           end if
13         end for
14         if (to_add_flag = True)
15           filtered_proximity_info_list.add (d_j)
16         end if
17       end if
18     end for
19   tag_i.position = Estimate_Loc (filtered_proximity_info_list)
20   Update location table (Reader ID, Get (current time),
       tag_i.ID, tag_i.position, filtered_proximity_info_list.size)
21   end for
22 end for
```



Proximity information table at reader r_2

Time	Tag ID	Reader ID	Reader position	T-R distance	Flag
..	t_4	r_2	..	d_{42}	0
..	t_1	r_2	..	d_{12}	0
..	t_4	r_1	..	d_{41}	1
..	t_1	r_3	..	d_{13}	1
..	t_2	r_1	..	d_{21}	1
..	t_3	r_3	..	d_{33}	1

Fig. 2: snapshot explains proximity information sharing

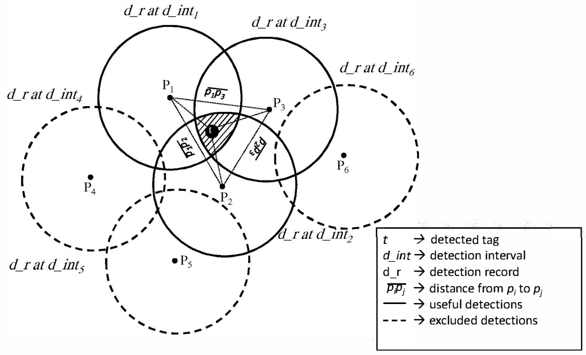


Fig. 3: detections filtering technique

At the *update location table* step in Algorithm I, the number of useful detections (*ud*) used in localizing a tag is added to the location information record. Hereinafter, we refer to this number as the *accuracy level*, which is used to decide which location is more accurate (within the same *localization interval*) in the event of conflict during the location dissemination process. Readers may keep old location information and use a tracking technique to reduce the amount of information, while keeping track of a tag's location on the go. We remark though that tracking granularity is beyond the scope of this paper.

E. Location Information Dissemination

The proximity information collection and sharing, and location estimation processes we have discussed so far help each reader to know the locations of a subset of tags. However, the issues of exchanging such locations and/or knowing locations of undetected tags have not yet been considered. In GOSSIPY, each reader upon updating its location information table, may share information in hand with all other readers. A reader may also send a location query when a certain tag's location is required. This solution can be implemented using one of two approaches: centralized or peer-to-peer. In the centralized approach, readers periodically push their location information table to a central server, and query this server for tags of interest. This approach requires mobile readers to share the same communication protocol and have access to a common database (not the case in GOSSIPY). In the peer-to-peer approach (which we adopt), location dissemination is carried out between neighbors and is achieved either by using push or pull strategy.

Peer-to-peer push strategy:

In this strategy, each mobile reader behaves proactively with location information updates. When a reader updates its location information table during either location estimation or location information dissemination process, it sends an update message to all readers in NR_i. Consequently, each reader in NR_i considers the *update timeout* interval and handles the update message differently. The reader ignores the message if it is a loopback or outdated with respect to *update timeout*. However, if the outdated update message is about an unknown tag; the reader will accept it but will not forward it. Otherwise, the reader updates its location information table if the received location is more recent or has a higher accuracy level, hence, pushes such updates to its neighbors. For the sake of illustration, Algorithm II details how this process takes place. The advantage of this strategy is that tag locations will be available to all readers as soon as they are estimated. The disadvantage however is the reader's payload in terms of the amount of information to be received from or forwarded to neighbors.

Algorithm II: location information dissemination – push

Input: location updates **Output:** updated location tables

```
for each location update do
 if location update.Time > current time – update timeout then
    set update message = Get location record (T_ID)
    set NR_i = current neighbors
    send update message to NR_i
  end if
end for
if update message UM is received then
  if UM.Reader ID is my ID then
     Ignore message
  else
     Accept Update Message (UM)
  end if
end if
----------------
procedure Accept Update Message (update message)
    if update message.Time <current time – update timeout then
        if update message.Tag ID is unknown then
              Update location table (update message)
        else Ignore message
        end if
    else if update message.Time >current time – localization
            interval then
        if update message.ud > current location record.ud then
              Update location table (update message)
        end if
end procedure
```

Peer-to-peer pull strategy:

In this strategy, a mobile reader does not share its location information until it receives a query message from an interested reader. Such a query can ask for multiple tags. For simplicity, we explain the process (see Algorithm III) assuming that readers break down the query into multiple single tag query messages. The process starts with a reader sending a time-stamped query message asking for a tag of interest to its neighbors. Each reader receiving this query plays the role of either responder or forwarder. If it has the tag's location, it generates a reply message containing the location information and sends it to its neighbors. Else, it forwards the query to its neighbors. Doing so allows the query to disseminate between readers until a responder is found or the query has expired. The reply message travels the same way allowing the forwarder readers to update their location table accordingly. As in the push strategy, readers ignore the query or reply message if it is a loopback or outdated with respect to *update timeout*. Obviously, the peer-to-peer pull strategy has less overhead in terms of number of messages and amount of exchanged information compared with peer-to-peer push strategy. In addition, readers acquire the most recent and accurate locations for tags of interest. However, the disadvantage is that location information dissemination only takes place for tags of interest.

Algorithm III: location information dissemination – pull

Input: Tag of interest **Output:** updated location tables

```
if Tag_i is to be localized then
        Send Query Message (Tag_i)
end if
if query message QM is received then
    if (QM.R_ID is my ID or QM.Time <current time –
            localization interval) then
        Ignore message
    else if QM.T_ID exists in location table then
        set NR_i = current neighbors
        set reply message (R_ID, Time, QM.T_ID, Location, ud)
        send reply message to NR_i
    else
        set QM.R_ID = My ID
        set NR_i = current neighbors
        send QM to NR_i
end if
if reply message RM is received then
      if RM.R_ID is my ID then
        Ignore message
      else if RM.Time <current time – update timeout then
          if RM.T_ ID is unknown then
            Update location table (RM)
          end if
      else if RM.Time > location record (RM.T_ID).Time then
          Update location table (RM)
          set RM .R_ID = My ID
          set NR_i = current neighbors
          send RM to NR_i
      end if
end if
-----------------
procedure Send Query Message (T_ID)
   set query message = (R_ID, Time, T_ID)
   while T_ID is not localized do
      for each localization interval do
         set NR_i = current neighbors
         send query message to NR_i
      end for
end procedure
```

A hybrid strategy might also be a solution to reduce the overhead of disseminating all the tags locations while keeping the opportunity to acquire the location of tags of interest. This can be achieved by categorizing objects into different classes, pushing location information for only important classes, and use the pull strategy for others. Such classification is domain specific. For instance, in a healthcare domain the most important objects may be patients and hazardous materials, while other objects may be of less importance.

IV. PERFORMANCE EVALUATION

In GOSSIPY, a number of parameters control the system performance. These parameters include but are not limited to the number of mobile readers, how frequently they interrogate surrounding tags, their mobility speed, and which location dissemination strategy they deploy. In this section we study the effect of the aforementioned parameters on the system performance in terms of location accuracy and localization delay as shown in Figs 4, 5 and 6.

A. Simulation Setup

We extended the ns-3 network simulator to support an RFID system. In our simulation, we randomly deploy *100* tags in a simulation area with dimension of *250m* x *250m*. The tags move in the area using random way mobility model with speeds ranging from *0m/s* to *2.5m/s* and a pause time of *50s*. In addition, the *localization interval* is set to the value of *60s* while the detection frequency is changed from *5s* to *30s* with *5s* steps and the *update timeout* is set to 10 folds the *localization interval*. The mobile readers have a reading range of *30m* and they move using random way mobility model with speeds ranging from *2.5m/s* to *5m/s* with *0s* pause time. In measuring the distance between a tag and a reader, we consider the range measurement noise as a zero-mean white Gaussian process with a variance correlated to the distance and signal to noise ratio (SNR) which is known a prior [20]. Values of the performance metrics are averaged over ten different independent runs with distinct random seeds for a total simulation time of *1000s* after dumping the first *1000s*.

B. Simulation Results

In our evaluation, we investigate two performance metrics: ***average location error,*** which represents the average mean square error of tag locations at each mobile reader for every localization interval and ***localization delay,*** which represents the maximum time it takes for a tag to be localized by all readers in the push strategy, and at interested readers in the pull strategy. In the first simulation scenario, we study the effect of decreasing the detection frequency within the same *localization interval* on the average location error with respect to different numbers of mobile readers. For a *60s localization interval*, we allow each reader to detect surrounding tags using different detection intervals. Fig. 4 illustrates that the average location error is decreased by increasing the detection frequency and is further decreased by having more mobile readers in the simulated area.

Another factor that decreases the average location error is the speed of the mobile readers. In this scenario we deploy *30* mobile readers and study the effect of using different detection intervals, within a *60s localization interval,* on the average location error using *3* different speeds (see Fig. 5).

From Fig. 5, we conclude that the fast speed of mobile readers, along with high detection frequency, allows them to gain more useful detections and exchange more proximity information with other readers, which decreases the average location error. In the last scenario, we study the effect of the number of mobile readers on localization delay for both push and pull strategies. We allow mobile readers to detect

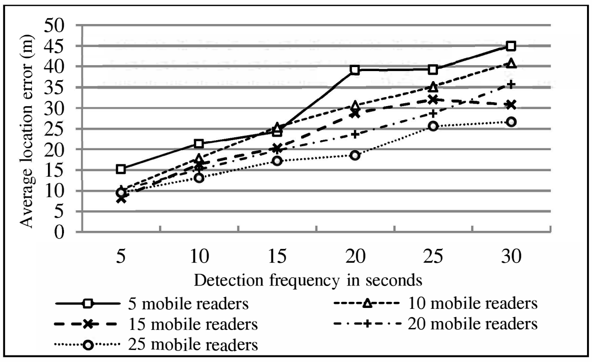


Fig. 4: Relation between average location error and number of mobile readers.

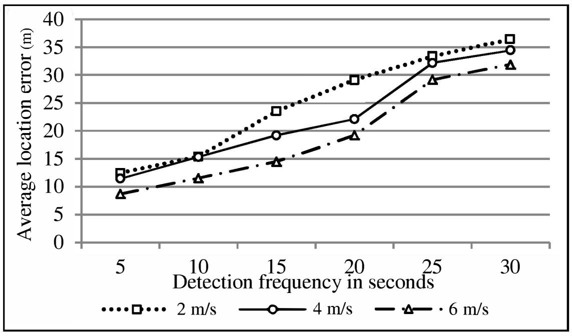


Fig. 5: The effect of mobile readers speed on average location error.

surrounding tags every *10s* within *60s localization interval.*

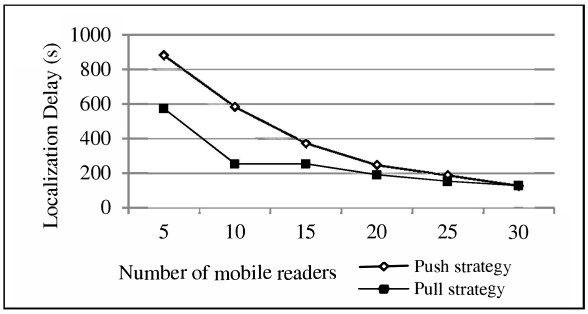


Fig. 6: Localization delay in push and pull strategies

Results are depicted in Fig. 6. We observe at small numbers of mobile readers, pull strategy outperforms push strategy as the location query either being replied or forwarded with no delay, while in push strategy, mobile readers have to wait for the localization interval to push their location information to others. However at large numbers of mobile readers, the localization delays converge.

V. CONCLUSION

RFID technology influences many domains such as transportation, airline and healthcare and plays a significant role in many IoT applications. Its popularity encourages researchers to study the employment of such technology to solve object localization problems. Although many localization systems based on RFID have been proposed in the literature, they do not cope with IoT challenges in terms of scalability, mobility and the heterogeneity of objects. Our proposed GOSSIPY system leverages a group of ad hoc heterogeneous mobile RFID readers to cooperatively localize mobile tags and ensures timely dissemination of their locations. The system is adaptable and its performance is controlled by a number of parameters: *detection interval*, *localization interval* and the numbers of mobile readers along with their speed. These parameters are application-based and their values should be determined according to the application requirements. We plan to study how the aforementioned parameters are correlated and study the system behavior when the mobile readers move in preplanned paths.

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 Sciences and Engineering Research Council (NSERC). The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] G. Kortuem, F. Kawsar, D. Fitton, and V. Sundramoorthy, "Smart objects as building blocks for the Internet of Things, " Internet Computing, IEEE, vol. 14, no. 1, pp. 44-51, 2009

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

[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] Ceipidor, Ugo Biader, et al. "Localization Issues in a ZigBee Based Internet of Things Scenario." The Internet of Things (2010): 157-165.

[5] Haller, Stephan, Stamatis Karnouskos, and Christoph Schroth. "The Internet of Things in an Enterprise Context." Vienna: Springer (Berlin-Heidelberg), 2008. pp. 14-28.

[6] M. Bouet, A. L. dos Santos, "RFID tags: Positioning principles and localization techniques", 1st IFIP Wireless Days, Nov. 2008, pp.1-5

[7] 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

[8] G. Han, H. Xu, J. Jiang, L. Shu, T. Hara, and S. Nishio, "Path planning using a mobile anchor node based on trilateration in wireless sensor networks," Wireless Communications and Mobile Computing, 2011.

[9] 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

[10] 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.

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

[12] 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.

[13] J. Hightower, R. Want and G. Borriello, SpotON: An Indoor 3d Location Sensing Technology Based on RF Signal Strength, Univ. Washington, Seattle, Feb. 2000.

[14] T.F. Bechteler and H. Yenigun. 2-D localization and identification based on SAW ID-tags at 2.5 GHz. IEEE Trans. on Microwave Theory and Techniques, 51(5):1584-1590, 2003.

[15] 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

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

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

[18] M. Bouet and G. Pujolle. A range-free 3-D localization method for RFID tags based on virtual landmarks. In Proc. of PIMRC, 2008.

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

[20] 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