Research and Implementation of an RFID Simulation System Supporting Trajectory Analysis

Tiancheng Zhang, Yifang Yin, Dejun Yue, Xirui Wang, Ge Yu
School of Information Science and Engineering, Northeastern University, China
Email: tczhang@mail.neu.edu.cn

Abstract—**Radio Frequency Identification (RFID) has been playing more and more important roles in domains such as supply chain management, commodity retail and health custody. RFID data can support functions such as object location, object tracking and trajectory analysis. However, at some point, to deploy many RFID devices into a real application scenario would be very difficult. In this paper, we present a RFID simulation system which could support effective data analysis and help users judge the effectiveness of deployment. This system implements part of ISO 18000-6C communication protocol and supports the path loss, backscatter, capture and tag mobility models. It provides a user-friendly visual platform for users to build their own virtual scenarios and deploy RFID devices into it. Further more, in order to improve the efficiency of operations in the application scenarios, we combine R-Tree with TSB-Tree to support the RFID object location and the trajectory analysis.**

Index Terms—**Radio Frequency Identification (RFID), Simulation, ISO 18000-6C, Trajectory analysis**

I. Introduction

As a developing automatic identification and data acquisition technology, Radio Frequency Identification (RFID) has the following features such as noncontact, high-speed, low-cost, long-life, pollution-against, hash environment-adaptive, et al. RFID is considered to be one of the most promising information technologies in the 21st century. It has been widely used in a variety of applications where it is necessary to automatically identify objects which are not proximate. Compared with traditional data, RFID data has the characters of real-time identification, rich semantics, uncertainty and magnanimity. By analyzing the data collected from readers, users can locate the position of a certain object or even track its movement by means of a proper algorithm. It will save a large quantity of labour force since this technology generates practical use.

As fully discussed in [1,2], an RFID system is superior to a conventional barcode system in many aspects. However, it also has some shortcomings such as signal interference and potential privacy invasion. In addition, contrasting with traditional RFID applications such as keyless entry badges where readers could be sparsely deployed, today's RFID applications, most of which are designed for the purpose of target location, are always required to deploy readers densely. but at some point, to deploy many RFID devices into a real application scenario would be very difficult. On one side, it is almost impossible to use real devices to perform the experiment in some certain application scenarios. On the other side, once some significant design weaknesses appear, the devices must be deployed all over again. Therefore, lots of situations must be taken into consideration before the facility deployment. In this paper, we present a novel RFID simulation platform to provide the entity models such as RFID readers and tags. This system is also designed to implement part of the ISO 18000-6C communication protocol. It provides a user-friendly visual platform for users to build their own virtual scenarios and deploy RFID devices into it. Users could print out the simulation results on the console or store them in the database as well. Further more, in order to support the RFID object location and the trajectory analysis, we combine R-Tree (the spatial object index mechanism) with TSB-Tree (Time-split B Tree, the temporal index method). Both the spatial state and the temporal interval of the label objects will be stored in the index entry. Then we can get a hierarchical RT-tree by grouping the close regions into the same cluster in different geographical scales.

II. Related Work

Several RFID simulators have been developed in the past. RFIDSim [3] relies on a discrete event simulator and can be used to simulate large populations featuring thousands of RFID tags. It has also implemented the ISO 18000-6C RFID protocol and supports the path loss, fading, backscatter, capture and tag mobility models, so it can be used to facilitate the relative comparison of different medium access protocols, transmission control strategies, settings in ISO 18000-6C, and privacy and security enhancements. There is also a configurable simulation platform for RFID application deployment [4].

This paper is based on “A Simulation Platform For RFID Application Deployment Supporting Multiple Scenarios”, by Tiancheng Zhang, Yifang Yin, et al, which appeared in Proceedings of the Eighth International Conference on Computataional Intelligence and Security, 17-18 Nov. 2012, Guangzhou, China.

This work was supported in part by the National Natural Science Foundation of China under Grant No.61272180, 61202086, 61272177, 61272179.

doi:10.4304/jsw.9.1.162-168

It provides users a visual developing platform but has not implemented any protocols. All the data related in modeling is sourced from an RFID test database. An RFID simulator called SERFID based on SystemC modeling is presented in [5]. This simulator is able to simulate a complete HF RFID system, from tags to the middleware. It can help to evaluate and optimize the robustness HF RFID systems. A configurable simulator is introduced in [6]. It is designed for RFID-aided supply chains that is capable to create consistent and realistic event data.

All the details of ISO 18000-6C protocol can be found in [8]. It defines the physical and logical requirements for a passive-backscatter, interrogator-talks-first, radio-frequency identification system operating in the 860 MHz–960 MHz frequency range. In this system, an interrogator transmits information to a tag by modulating an RF (Radio Frequency) signal in the 860 MHz–960 MHz frequency range. An interrogator receives information from a tag by transmitting a continuous-wave RF signal to the tag, the tag responds by modulating the reflection coefficient of its antenna, thereby backscattering an information signal to the interrogator. The RFIDSim presented in this paper is designed to feature some main reader commands, tag replies and states to support the simulation.

The path loss model is very important in a wireless simulation system. Hashemi [9] presents a path loss model with variable environmental factor which is extremely suitable for the case of RFID. And Leong et al. [10] improve the path loss model according to the measurement results from their lab. It suggests that walls have been shown to have a great impact on the environmental factor. Within a room, one fixed environmental factor can be used, but it must be increased when a wall is encountered as the distance increases.

During the past years, the trajectory analysis has been performed based on individual location history represented by GPS, RFID or Wireless Sensor trajectories. Using the GPS trajectories generated by multiple users, [11] mined interesting locations and classical travel sequences within a given region. A HITS (Hypertext Induced Topic Search)-based inference model has been proposed to infer a user's travel experience and the interest of a location.

While the RFIDSim discussed in [3] is focused on the physical performance of readers and tags, it can only provide users a simple simulation process of RFID facility deployment. On the other hand, though the RFID simulation platform discussed in [4] can help users to deploy RFID application systems and supply with a user-friendly visualized development platform, it has not implemented any protocols. Thus, we try to develop an improved deployment simulation platform for the RFID application which could implement part of the ISO 18000-6C communication protocol and supports the path loss, backscatter, capture and tag mobility models as well. Moreover, we abstract some common reader APIs (Application Program Interface) and also present a special language to control the behavior of the simulation process. All the data collected during the simulation process would be stored in the database for further processing such as target tracking.

III. OVERVIEW

The objective of RFIDSim is to provide users a visual developing platform for RFID applications. Common objects such as walls and tables are abstracted as elements to build a virtual RFID application scenario. After setting location and motion of RFID devices, a simulation process could be started up. The simulation engine is driven by a discrete event simulator. Since RFIDSim has implemented part of ISO 18000-6C protocol, it could be used to simulate the identification process, memory access etc. The simulation behavior differs due to different programmes that users write by using the special language we have mentioned above. At the end of the simulation process, all the data collected would be stored in the database for further processing and presented to users on the console. Considering a situation that an RFID system is designed for indoor target tracking, this RFIDSim could facilitate users to figure out the balance between the reader deployment and the location algorithm.

IV. SIMULATION MODELS

To support the RFID application deployment, we need to model entities such as readers, tags and walls to build an application scenario, and to model tag movements and reader APIs to initialize a simulation process. It is also necessary to model the reader commands, signal propagation, capture and backscatter to simulate the signal transmission and the reception process. All the models are discussed respectively in the following sections.

A. RFID Reader

At the logical layer, an RFID reader features the main commands and their parameters specified in ISO 18000-6C as shown in Table I. It can also generate proper command sequences according to users' different intentions. For instance, the typical inventory sequences look like a Query command is followed by Ack, QueryRep and QueryAdj commands until a tag population is successfully identified.

TABLE I
MAIN READER COMMANDS SPECIFIED IN ISO 18000-6C

Category	**Command**
Selection	Select
Inventory	Query, QueryRep, QueryAdj, ACK, NAK
Access	Read, Write, Lock, BlockWrite, BlockErase

At the physical layer, an RFID reader radio transmitter is characterized by the carrier frequency and the transmitting power which are required to compute the received signal strength at the RFID tags. Directive reader antennas are adopted here. The radiation pattern of the antenna can be specified as part of the configuration which is shown in Fig. 1 [3]. So the orientation of the RFID reader antenna also needs to be specified.

The capture model for an RFID reader radio receiver chosen to be implemented is the most commonly used power model [5] as equation (1) shows.

$$P_{R_0} \geq c\sum_{i=1}^{k} P_{R_i} \tag{1}$$

Where P_{R_0} denotes the received signal strength of the strongest signal, P_{R_i} denotes the received signal strength of one of the k other tag signals and c is a factor denotes the capture ratio. The P_{R_0} s.t. equation (1) is ass umed to be captured successfully.

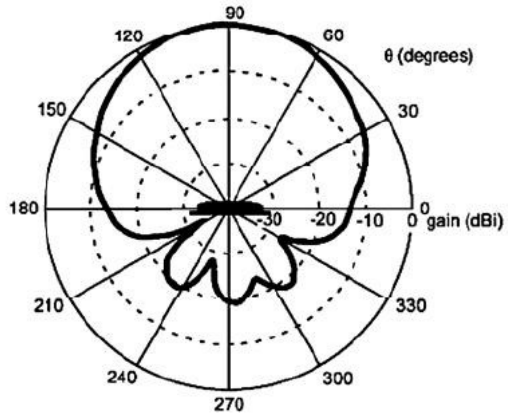

Figure 1. Sample radiation pattern of an RFID reader antenna

B. RFID Tag

An RFID tag model needs to generate the replies and change states according to the commands received from an RFID reader. According to ISO 18000-6C protocol, we implement the main states including ready, arbitrate, reply, acknowledged and secured. The details about ISO 18000-6C protocol can be found in [6].

At the physical layer, an RFID passive tag uses a backscatter model to calculate the backscattered power. The model we choose is a simple linear model [3] that relates the backscattered power P_B to the received power P_R.

$$P_B = \alpha P_R \tag{2}$$

Where α denotes a constant that specifies which proportion of the incident signal is reflected.

C. Signal Propagation

In free space, the path loss model for simulation is considered as a simple function of distance. However, a typical RFID deployment zone is like a warehouse filled with commercial products, so logically a more complex model is required. It is found that a path loss model with variable environmental factor, *n*, is most suitable for the case of RFID as shown in equation (3) [7]:

$$PL(dB) = PL(d_0) + 10n\log\left(\frac{d}{d_0}\right) \tag{3}$$

where d_0 is an arbitrary reference distance, n is the environment factor, d is the separation distance between two antennas and $PL(d_0)$ is the free space path loss for a distance d_0. Considering the fact that the n increases as the distance increases, equation (3) is modified as shown as equation (4) [8]:

$$PL(dB) = \begin{cases} PL(d_0) + 10n_1\log\left(\frac{d}{d_0}\right) & 0 \leq d < 8m \\ PL(d_0) + 10n_2\log\left(\frac{d}{d_0}\right) & d \geq 8m \end{cases} \tag{4}$$

where $n_2 > n_1$. As the experiment shows, a fixed n value can be used within a room. However, n must be increased when a wall is encountered as the distance increases. Thus, for every obstruction abstracted from an RFID application scenario, we set an attribute, *n*, to denote the environment factor discussed here. When a signal transmitted from a reader to a tag meets an obstruction, the attribute *n* of this obstruction will be used in the path loss model. Moreover, users can carry out on-site measurement to determine the best *n* for a certain area and configure it in simulations.

D. Simulation Data Management and Analysis

In this module, the simulation data is used to support a series of functions such as the target location, trajectory analysis, etc. We simulate the procedure where a reader identifies a label id. When a label enters into the sensing field of a reader, the original simulation data including the label id and the sensing time would be generated. After the simulation procedure, this module will process these raw simulation data by some frequently-used RFID processing algorithms such as the label location. Therefore, this system could support the upper lever functions such as the trajectory analysis by utilizing the common RFID data from the low level. Users could output the simulation data on the console or store them in the database alternatively.

RFIDSim is designed to support a range of functions for constructing the virtual RFID application scenario, recording the moving trajectory of the label, etc. In order to improve the efficiency of the insert, delete, update and search operations of the models, we need to adopt an appropriate spatial index to represent the location information of some modes such as the RFID equipments, the entities in the simulation environment, etc.

Since the environment entity will be accessed and queried frequently when the simulation signal is propagated as well as the label is moving, we adopt R-Tree as the index of the simulation environment entity in the virtual scenario. R-Tree is a completely dynamic spatial index structure supporting the insert, delete and search operations which can be executed simultaneously. Meanwhile, it requires no periodic index reorganization. In R-Tree, spatial objects are partitioned according to the region. And each node corresponds to a certain region and a disk page. The disk page of each non-leaf node stores all the area regions of all its child nodes, which means the regions of all its child node fall into its own scope. The disk page of each leaf node stores the bounding rectangles of all the spatial objects within its scope. Each node has an upper bound and a lower bound for the number of child nodes. The lower bound ensures the effective utilization of the disk space, and the upper bound ensures that each node corresponds to a certain disk page. When the required space after inserting a new node exceeds a disk page, the original node will be divides into two nodes. An example of R-Tree is shown in Fig. 2.

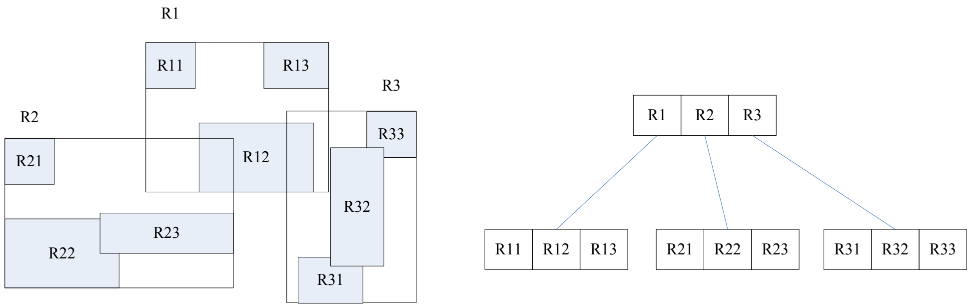

Figure 2. The Structure of R-Tree

Further, we combine R-Tree (the spatial object index mechanism) with TSB-Tree (Time-split B Tree, the temporal index method). Both the spatial state and the temporal interval of the label objects will be stored in the index entry. When the spatial state changes, a new generated data item will be inserted into the index. In order to support the trajectory query and the moving pattern discovery much more efficiently, we also optimize the structure of RT-Tree by the technology of the hierarchical tree. We utilize the density-based clustering algorithm and cluster the regions of the label objects. In this way, we could get a hierarchical RT-tree by grouping the close regions into the same cluster in different geographical scales. As shown in Fig. 3, the nodes in the tree represent different region clusters and different levels represent different geographical scales. A deeper level indicates a finer granularity and smaller space for the nodes.

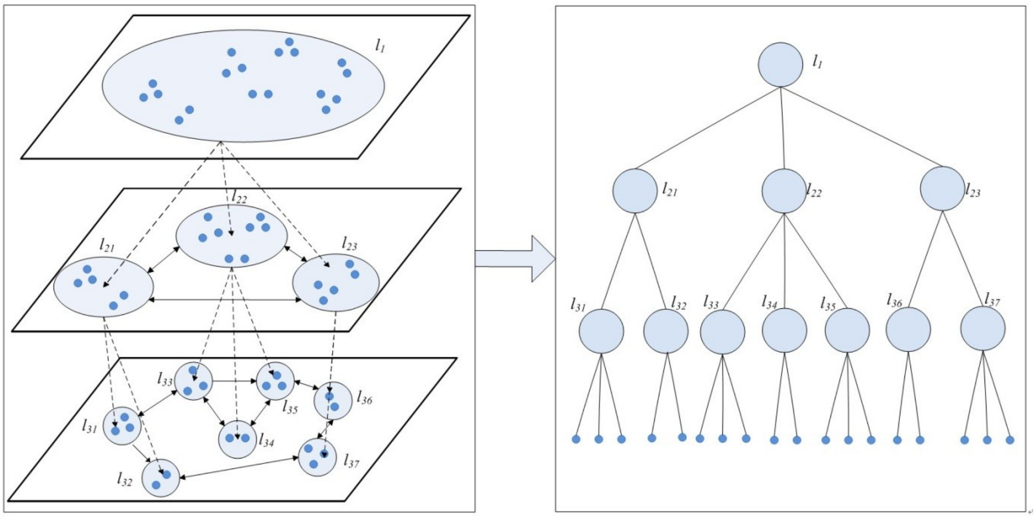

Figure 3. Hierarchical Clustering in RT-tree

V. Implementation

RFIDSim is designed to rely on a discrete event simulator. A discrete simulation is a form of simulation where all actions within the simulation can be modeled in discrete points in time. For example, a signal is broadcasted from the reader at time t, and is received at time t+k by a tag. Though the movements of tags are continuous in real world, we can simulate all the movements as discrete events at a proper predefined frequency to ensure the validity of the simulation.

To implement RFIDSim, we chose to use Eclipse RCP which is a platform for building and deploying rich client applications. It includes the ability to deploy native GUI applications to a variety of desktop operating systems, such as Windows, Linux and Mac OSX. It also poses an integrated update mechanism for deploying desktop applications from a central server. This plug-in based development produces a modular structure and allows selected modules to be put integrated together in different software versions.

This RFIDSim consists of three plug-ins, each of which implements a special function. The first plug-in is the core layer which builds the abstract model and defines the extension points of reader, tag, obstruction, signal propagation and etc. The second plug-in is the implementation layer which implements multiple entity models by loading the extension points from the core layer. The third plug-in is the GUI layer which provides users a visual developing platform by using GEF (Graphical Editing Framework).

GEF allows us to easily develop graphical representations for existing models. Thus, we choose to use GEF to develop a feature rich graphical editor here. With this editor, users can modify models, like changing element properties, by using very common functions like the drag and drop, copy and paste, and actions invoked from menus or toolbars.

The relationship between core layer and implementation layer is shown in Fig. 4. By loading the extension points from the core layer, users can even develop their own plug-in to implement new types of entities. Thus, this RFIDSim has a strong expansibility.

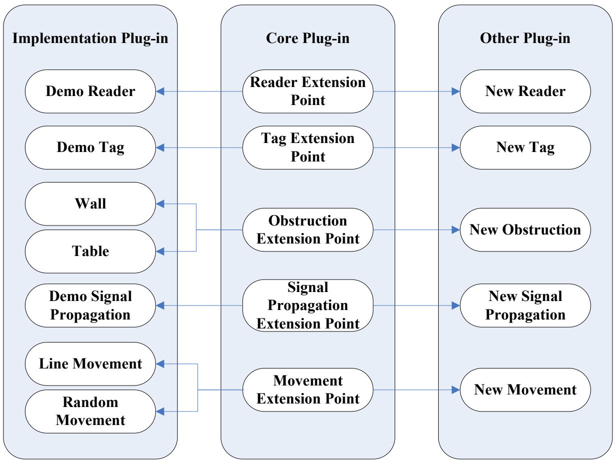

Figure 4. Expansibility of RFIDSim

TABLE II
ENTITIES IMPLEMENTED IN RFIDSIM

Entity	Description
Application	initiates readers and generates proper command sequences according to the pre-input programme and stores all the data collected during a simulation process and presents it to the user on the console
Reader Logical Layer	generates the main ISO 18000-6C commands
Reader Physical Layer	implements capture model
Tag Logical Layer	receives reader commands, updates state and generates appropriate replies
Tag Physical Layer	implements capture and backscatter model
Scenario Obstruction	obstructs tag movement and signal transmittance
Signal Propagation	implement the path loss model
Movement	implement different types of tag movements
Radio Field	changes location of RFID tags and deliver datapackages between readers and tags

The RFIDSim implements the entities shown in Table II. At the beginning of a simulation, users need to build their own scenario or choose a default scenario (such as an intellectual museum) offered by the system. The default programme implements the basic tag identification process. If users want to implement other functions such as tag memory access, they can write a special programme to control the simulation process. Then, the application initiates readers and generates proper command sequences according to the pre-input programme to control the behavior of readers. And RFID readers and tags communicate with each other by datapackages delivered by the Radio Field. Whenever a simulation finishes, the application would store the data collected during the simulation for further processing. The architecture of RFIDSim is shown in Fig. 5.

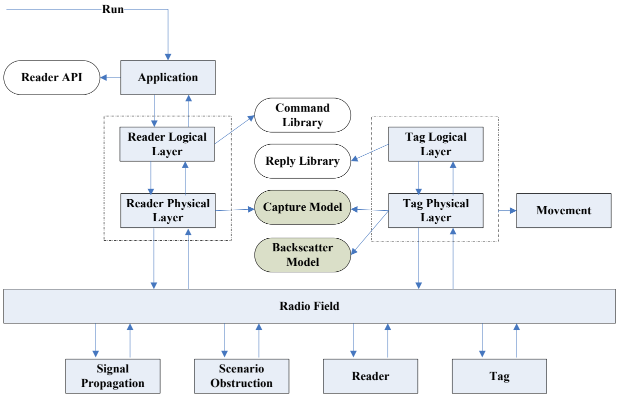

Figure 5. Architecture of RFIDSim

In order to control the behavior of readers as the real ones, we abstract some common reader APIs (cf. Table III) by which users can write a programme to control the simulation process.

TABLE III
READER APIS AND THEIR DESCRIPTION

Category	API	Decription
Reader Administration	OpenReader	to open a reader
	CloseReader	to close a reader
	BeepControl	to control a reader buzzer
	Power	to set a reader's transmiting power
Tag Manipulation	SingleTagIdentify	to identify a single tag
	MultipleTagIdentify	to identify multiple tags
	ReadTag	to read a tag's memory
	ReadTag0	to read a certain tag's memory
	WriteTag	to write a tag's memory
	WriteTag0	to write a certain tag's memory
	LockTag	to lock a tag
	LockTag0	to lock a certain tag

VI. EXPERIMENTS OF RFIDSIM

The objective we implement RFIDSim is to provide users a visual developing platform for RFID facility deployment in multiple scenarios. To evaluate the effectiveness of this system, we carried out a series of experiments. In the next section, we present one of basic experiments with the purpose of figuring out a proper deployment in an intellectual museum.

In an intellectual museum scenario, tags appear as exhibits and visitors. The exhibits are displayed in different rooms. Several rooms compose an exhibition hall. Different exhibition halls are connected by channels. Readers are placed next to exhibits for the use of supervisory control. Visitors can walk around in the intellectual museum. Once a visitor walks into the accessible range of a reader, it would be read immediately. Information of the visitor such as the identification number, the location area and the exact time of this identification process would be generated. Based on this information, we can track visitors, analyze their interests, optimize the exhibition deployment and even deal with the abnormal conditions.

Table IV shows the use case for the experiment. And Table V shows the simulation results. As all the exhibits remain static in this experiment, here we only record the data which are related to the visitors in Table V.

TABLE IV
USES CASE OF RFIDSIM

Use Case	Test of RFIDSim
Level	High
Input	(a) Choose the default scenario of intellectual museum. (b) Deploy readers and tags. (c) Use the default programme with the function of tag identification.
Steps	(a) Run the system.
	(b) Choose the intellectual museum as the simulation scenario.
	(c) Deploy tag0 to tag 5 as the exhibit0 to exhibit5.
	(d) Deploy tag6 to tag8 as the visitor0 to visitor2.
	(e) Deploy reader0 to reader5 next to the exhibits.
	(f)) Add random motion to tag 6.
	(g) Add rectilinear motion to tag7 and tag8.
	(h) Choose the default programme with the function of tag identification.
	(i) Open a console and run the simulation.
	(j) Use the simulation results to analyze the rationality of the deployment.
	(k) If the deployment is irrational, go to (c); else, finish the simulation.
Expected Results	When a tag is in the accessible range of a reader, it would be read by the reader immediately. The data collected during simulation would be stored in the database for further processing, and also be outputted on the console to users.

Using a proper location algorithm, users could achieve the target tracking with the data collected during the simulation. They can also figure out whether all the RFID facilities are properly deployed at the right places and make the corresponding adjustment.

Further, RFIDSim supports the users to select different location scales according to their different simulation requirements for the further data cleaning. The cleaning method is as follows: the system partitions the simulation space into equal square regions by using the scale as the side length of the squares. All the coordinates in a square region are substituted by the coordinate of the central point of the square, which means the smaller the scale is, the finer the grain of the location information will be. On the contrary, the larger the scale is, the coarser the grain of the location information will be. In some outdoor RFID simulation scenarios, it’s necessary to select a larger scale.

After executing the location algorithm and the data cleaning, we could obtain a series of trajectories composed of the coordinate points with timestamp for each label in the simulation scenario for supporting the further trajectory analysis. By adopting the adaptive clustering algorithm, the trajectories of different targets could be mapped into different levels in RT-tree. And we could get different graph models by connecting different clusters accordingly.

TABLE V
SIMULATION RESULTS

Time	Data	Location of Tag
1-32	No data.	Visitor0 wanders between exhibit0 and exhibit1. Visitor1 is on the way to the room where exhibit5 is displayed. Visitor2 is on the way to the room where exhibit3 is displayed.
33-56	Visitor0 is read by reader0.	Visitor0 is at exhibit0. Visitor1 is on the way to the room where exhibit5 is displayed. Visitor2 enters the room where exhibit3 is displayed.
57-136	No data.	Visitor0 wanders off exhibit0. Visitor1 enters the room where exhibit5 is displayed. Visitor2 is in the room where exhibit3 is displayed.
137-148	Visitor2 is read by reader3.	Visitor0 wanders between exhibit0 and exhibit1. Visitor1 is in the room where exhibit5 is displayed. Visitor2 is at exhibit3.
149-160	Visitor1 is read by reader5. Visitor2 is read by reader3.	Visitor0 wanders between exhibit0 and exhibit1. Visitor1 is at exhibit5. Visitor2 is at exhibit3 .
161-188	Visitor0 is read by reader1. Visitor1 is read by reader5. Visitor2 is read by reader3.	Visitor0 is at exhibit1. Visitor1 is at exhibit5. Visitor2 is at exhibit3.
189-200	Visitor1 is read by reader5. Visitor2 is read by reader3.	Visitor0 wanders off exhibit1. Visitor1 is at exhibit5. Visitor2 is at exhibit3.

VII. CONCLUSION

In traditional RFID applications such as keyless entry badges, all the readers are sparsely deployed. However, today's RFID applications usually require readers to be densely deployed. Thus, the deployment problem of the RFID devices arises in real scenario. To solve this problem, we developed the RFIDSim.

RFIDSim is an improved simulation platform for the RFID application deployment which implements part of ISO 18000-6C communication protocol and supports the path loss, backscatter, capture and tag mobility models as well. It is driven by a discrete event simulator and has strong expansibility. To control readers' performances, we also abstract some common reader APIs. From the test results, we can conclude that the RFIDSim provides users a visual developing platform for RFID facility deployment in multiple scenarios. And from the data collected during the simulation, users can judge whether a certain deployment is fairly appropriate or not. Further more, in the simulation platform, we store the data by combining R-Tree with TSB-Tree to support the RFID object location and the trajectory analysis.

For future work, we will try to perfect reader commands, tag replies and tag states, making them perfectly accord with ISO 18000-6C protocol. We will also impove the reader APIs to make models work just as the real ones. Limited by the experiment condition, we didn't comparison testing between the models and the real devices. Thus, more tests on this system would be performed in the near future.

REFERENCES

[1] S. Lewis, "A basic introduction to RFID technology and its use in the supply chain" Jan 2004. [Online]. Available: http:// www.idii.com/wp/LaranRFID.pdf.

[2] Philips Semiconductor and TAGSYS and Texas Instruments Inc., "Item-level visibility in the pharmaceutical supply chain: a comparison of HF and UHF RFID technology" July 2004. [Online]. Available: http://www.hibcc-au.com.au/RFID%20Website/RFID%20 Reference%20 Files/jointPharma.pdf.

[3] C. Floerkemeier and S. Sarma, "RFIDSim-a physical and logical layer simulation engine for passive RFID" IEEE Transactions On Automation Science and Engineering, vol. 6, no. 1, pp. 33-43, Jan 2009.

[4] XIA Tian, ZENG Juan-fang and LIU Yu. "A configurable simulation platform for RFID application deployment" Microcomputer Information, vol. 25, no. 1-2, pp. 224-226, 2009.

[5] Gilles Fritz, Vincent Beroulle, et, al. "SystemC Modeling of RFID Systems for Robustness Analysis" in Proceeding of the 19th International Conference on Software, Telecommunications and Computer Network(SoftCOM), pp. 1-5, 2011

[6] Matthieu-P. Schapranow, Cindy Faehnrich, Alexander Zeier, et, al. "Simulation of RFID-aided Supply Chains: Case Study of the Pharmaceutical Industry" in Proceeding of the Third International Conference on Computational Intelligence, Modelling & Simulation, pp. 340-345, 2011

[7] M. Zorzi and R. Rao, "Capture and retransmission control in mobile radio" IEEE J.Sel. Areas Commun, vol. 12, no. 8, pp. 1289-1298, Oct. 1994.

[8] "Class 1 generation 2 UHF air interface protocol standard version 1.0.9, EPCglobal, 2005". [Online]. Avaliable: http:// www.epcglobalus. org.

[9] H. Hashemi, "The indoor radio propagation channel" Proceedings of IEEE, vol. 81, no. 7, pp. 943-968, Jul. 1993.

[10] K. S. Leong, M L Ng and P. H. Cole, "Operational considerations in simulation and deployment of RFID systems" in Proceeding of 17th Int. Zurich Symp. Electromagnetic Compatibility, pp. 521-524, 2006

[11] Yu Zheng, Lizhu Zhang, Xing Xie, Wei-Ying Ma. Mining interesting locations and travel sequences from GPS trajectories. In Proceedings of International conference on World Wild Web (WWW 2009), Madrid Spain. ACM Press: 791-800.

Tiancheng Zhang received his Ph.D. degree in computer science from the Northeastern University (China) in 2008. He is currently an associate professor of computer science at Northeastern University. His major interests include RFID data management and data stream analysis.

Yifang Yin received her B.S. degree in computer science from Northeastern University(China) in July 2011. She is the major co-auther of this work. She is currently a Ph.D candidate at the National University of Singapore. Her main research interest is multimedia data analysis.

Dejun Yue received his B.S and M.S. degrees both in computer science from the Northeastern University(China) in July 2005 and March 2008. He is currently a Ph.D candidate at the Northeastern University. His major interests include data stream analysis and XML data management.

Xirui Wang received his B.S degree in computer science from the Fuzhou University in July 2011. He is currently a graduate student of computer science at Northeastern University(China). His major interest is RFID data analysis.

Ge Yu received his Ph.D. degree in information science from the department of engineering at Kyushu University (Japan) in 1996. He is currently an professor of computer science at Northeastern University(China). His major interests include database theory and applications, data mining, deep web, wireless sensor network.