Cloud-Based Iterative RFID Tag Search Protocol Using Bloom Filters

Yuming Qian[1], Daqiang Zhang[1(✉)], Shengjie Zhao[2], Xiaopeng Fan[3], and Ke Fan[1]

[1] School of Software Engineering, Tongji University, Shanghai, China
{lincolnmill08, dqzhang, kefan}@tongji.edu.cn
[2] The Key Lab of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai, China
shengjiezhao@tongji.edu.cn
[3] Shenzhen Institute of Advanced Technology, CAS, Shenzhen, China
xp.fan@siat.ac.cn

Abstract. RFID (Radio Frequency Identification) has achieved widespread success in supply chain management, object identification, and assets tracking. In these applications, we often need to search for a particular set of products in a large-scale collection of products. Existing schemes have been proposed, but they are limited by a couple of problems. Firstly, they fail to work under the situation when the cardinality of X is much larger than that of Y. Secondly, they implicitly assume the RFID readers are adequate powerful so that readers can handles a large number of query requests in a short time. To this end, we propose ITSP – Cloud-based Iterative Tag Search Protocol, which can locate the tags in a secure and efficient manner. To be secure, ITSP authenticates the communications between RFID readers and tags to in advance. To be efficient, ITSP reformats the single-round communication model to a multi-round communication one between readers and tags. Furthermore, ITSP employs a cloud-based service to rapidly conduct the searching process. Extensive experimental results show that the proposed scheme achieves high-levels of searching efficiency with the improvement at least 19 %.

Keywords: RFID · Cloud computing · Tag query · Iterative searching

1 Introduction

RFID (Radio Frequency Identification) is a wireless communication technology [1], which applies radio signals to identify specific targets without touching objects mechanically and optically. Usually, An RFID system consists of three parts: readers, tags, and antennae. The readers transmit encoded radio signal to interrogate the tags and gets stored information within tags. Readers have relatively powerful computation and storage capability [11]. Each tag has a unique ID according to the EPC global Class-1 Gen-2 standard [12]. It receives the reader message and responds with required information. RFID antenna receives and transmits the radio signal between tags and readers. Owing to its low cost, non-contact sensing, and miniaturization, RFID has been widely

V.C.M. Leung et al. (Eds.): CloudComp 2014, LNICST 142, pp. 145–154, 2015.
DOI: 10.1007/978-3-319-16050-4_14

adopted in inventory management [2], product tracking, intelligent transportation systems [3–5], animal identification [6] and healthcare systems [7, 8, 17].

In RFID applications, we may encounter the following scenario. A batch of new products are stacked in several warehouses that are physically remote. Sometimes, the manufacturer needs to detect inferior products by counting all tags attached to the products. Given the goal set of tags X, the problem is how to search them in the coverage area of all the readers, which have the set of the tags Y. The objective of the RFID search problem is to find the set of X ∩ Y. Note that X may be totally exclusive to Y, i.e., X ∩ Y = Φ. For example, all of the products in some warehouse are qualified. On the other hand, X may be totally included by Y, i.e., X ∩ Y = X. For example, all of the defective products are placed in some warehouse. Moreover, the RFID readers suffer from limited computation and computing capabilities. Each round detection, they will spend a second. Therefore, they cannot continuously check the large number of tags without break. Table 1 lists our notations used in this paper.

Table 1. Key notations used in the design of the proposed scheme

Symbols	Description
X	The set of the goal tags
Y	The set of the tags in the coverage zone
Y_C	The candidate set of the tags
X ∩ Y	The intersection set of X and Y
$\lvert\cdot\rvert$	Cardinality of the set
BF(·)	The bloom filter for a set
A ∩ BF(B)	The subset of A that is filtered by the bloom filter for B
h(·)	A uniform hash function
α	The transmission rate from a reader to a tag
β	The transmission rate from a tag to a reader
P_{req}	The false positive rate
K	The number of the hash functions
S	The random seed
t	The time for transmitting one bit

Many schemes have been proposed. However, they cannot deal with some problems. On the one hand, when the cardinality of X is larger than Y, some protocols cannot work. On the other hand, with the number of tags getting larger, the search efficiency drops due to the limited computing capabilities of readers. In this paper, we propose ITSP – Cloud-based Iterative Tag Search Protocol. In the given protocol, we reformats the single-round communication model to a multi-round communication one between readers and tags. Besides, we put the process of computing into the cloud which can make up the defects of readers in computing. To summarize, the contributions of this paper are two-fold. First, the protocol can still work well when $|X| > |Y|$ and we can choose the best search method according to the relationship of $|X|$ and $|Y|$ for the best efficiency. Secondly it uses the cloud-based service to get the efficient search performance.

The rest of the paper is organized as follows. Section 2 introduces the related works. Section 3 describes the details of the Cloud-based Iterative Tag Search Protocol. Section 4 evaluates the performance of our protocol. Section 5 draws the conclusion.

2 Related Works

Zheng and Li propose the Compact Approximator based Tag Searching [13] (CATS) protocol that improves the search efficiency dramatically. It is a two-phase protocol which considers all the readers as a whole. In the first phase, a bloom filter BF_1 is constructed with K_1 hash functions and the seed S_1. BF_1 and S_1 will be broadcast to all of the tags in the system. Once a tag receives BF_1, it checks whether it can be filtered by this bloom filter with parameter K_1 and S_1 or not. The tags that are not filtered will keep silent in the next phase and do not response to the reader any more. On the other hand, other tags will keep active as the candidate tags. In the second phase, the reader broadcasts parameter K_2 and S_2 as hash function and the seed respectively to the tags. The reader will build up a new bloom filter BF_2 according to the replies from those candidate tags. If the time slot is not empty, the corresponding element in the bloom filter will be assigned as '1', otherwise as '0'. According to BF_2, the target tags also are checked. The tags that keep active will be considered as the search results. Thus CATS can reduce the search delay by tolerating some false positive cases. However, it cannot work under the environment that the cardinality of X is much larger than that of Y [14]. Though the Iterative Tag Search Protocol (ITSP) protocol overcomes the disadvantage above, we can still decrease the search delay by adjusting the number of the filtering vector on the basis of the cardinalities of X and Y.

Both CATS and ITSP consider the readers as a whole, which results in the interferences among the readers in practice. Some existing works have been done to solve the interference problem. Waldrop et al. [9] proposed the colorwave algorithm that colors the readers in the system so that there is not any interference among the readers with the same color. Tang et al. [10] proposed the RASPberry that makes the system work in a stable way in the long term when the arrival rate of the tags are in the range of the readers.

The security issue in RFID search is also another focus in previous work. Based on the analysis work from Chun and Hwang [15], we found that the main reason for unsecurity in RFID search efficiency is the loss of strong authentication between readers and tags. If we have a sound mechanism for authentication, we can ensure the security during the RFID search.

3 ITSP: Cloud-Based Iterative Tag Search Protocol

In this section, we mainly talk about how to enhance the search efficiency based on the cardinality of X and Y and how to leverage the cloud-based service to improve the computing capacities of readers.

Figure 1 shows the architecture of RFID search in Cloud. First, we input the set of wanted tags into the readers. Then the readers request the cloud-based service for computing. After the cloud return the result, readers broadcast the computing results to the tags. Tags will response to the request from readers. The communication between readers and all tags may take several rounds and in each round, readers leverage the cloud service to do computation.

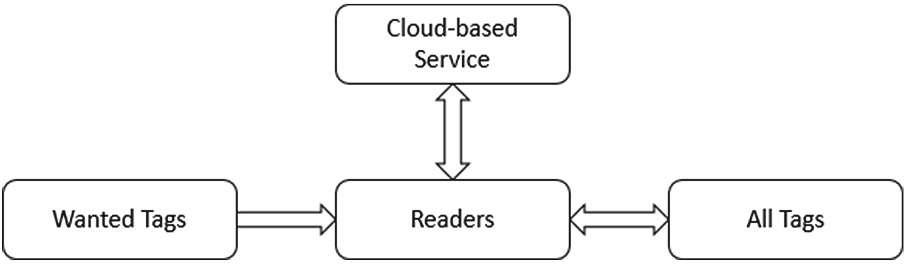

Fig. 1. The architecture of RFID Search on cloud-based service

3.1 Large-X-Query Algorithm

CATS cannot work when the cardinality of X is much larger than Y. The only one round interaction mainly results in the problem. Because in the communication process, CATS should meet both the false positive rate and the efficiency requirements. We increase the number of the interaction round to solve this problem based on the idea from ITSP [14].

We apply a partial bloom filter in Large-X-Query algorithm rather than a standard bloom filter. We define some variables. Let total number of the elements as m and the length of bloom filter as L. We define the hash function number as K. In the *i*th round, the candidate tags in X and Y are X_i and Y_i respectively. There are some other definitions, such as $W = X \cap Y$, $U_i = X_i - W$, $V_i = Y_i - W$. U_i and V_i represent the remaining tags that do not belong to the set X and Y respectively. $X_1 = X$, $Y_1 = Y$. In a partial bloom filter, the false positive rate obtains the minimum value 1/2 in each vector when $L/K = m \times \ln 2$ [16]. Thus, in each round, the reader transmits m_i vector to the tags and the tags transmit one vector to the reader. After one-round transmit, X and Y will change to:

$$Y_{i+1} = V_i \times (1/2)^{m_i} + W \tag{1}$$

$$X_{i+1} = U_i \times 1/2 + W = (X_i + W)/2 \tag{2}$$

The time of this round is:

$$f(m_i) = 1/\ln 2 \times m_i \times X_i \times t + 1/\ln 2 \times Y_{i+1} \times t \tag{3}$$

$$= t/\ln 2(m_i \times X_i + V_i/2^{m_i} + W) \tag{4}$$

When

$$m_i = \frac{\ln(\ln 2 \times V_i/X_i)}{\ln 2} \tag{5}$$

The transmission delay of each round obtains the minimum value. Because there is not any collision among vectors, the false positive rate is with no relation to the number of transmitted vector. The false positive rate of each round is always 1/2. If we assume there are b rounds of communication between the readers and the tags, the readers totally receive b vectors. To satisfy the false positive rate, b should meet the requirement $(1/2)^b < P_{req}$, i.e.,

$$b < -\log_2 \mathrm{P}_{req} \tag{6}$$

Thus, total search time is:

$$T_{lxq} = \sum_{i=1}^{b} \left(m_i \times \frac{X_i}{\ln 2} + \frac{Y_i}{\ln 2} \right) \times t \tag{7}$$

In Large-X-Query, the reader transmits m_i vectors to the tags and the tags transmit one vector to the reader in each round (Fig. 2).

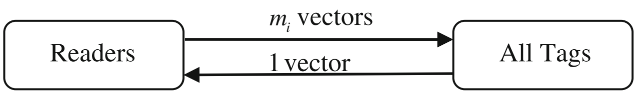

Fig. 2. The *i*th round in Large-X-Query algorithm

3.2 Large-Y-Query Algorithm

In Large-X-Query algorithm, the tags only transmit one vector to the reader. However, transmitting multiple vectors to the reader may improve the efficiency, especially when the cardinality of Y is much larger than that of X (Fig. 3).

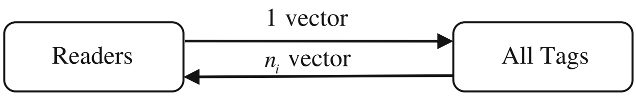

Fig. 3. The *i*th round in Large-Y-Query algorithm

The reader transmits one vector to the tags and the tags transmit n_i vectors to the reader. We name this algorithm as Large-Y-Query. After each round, X and Y will change to:

$$Y_{i+1} = V_i \times 1/2 + W = (Y_i + W)/2 \tag{8}$$

$$X_{i+1} = U_i \times (1/2)^{n_i} + W \tag{9}$$

The time of this round is:

$$f(n_i) = 1/\ln 2 \times X_i \times t + 1/\ln 2 \times n_i \times Y_{i+1} \times t \tag{10}$$

$$= t/\ln 2(n_i \times Y_{i+1} + U_i/2^{n_i} + W) \tag{11}$$

When

$$n_i = \frac{\ln(\ln 2 \times U_i/Y_{i+1})}{\ln 2} \tag{12}$$

The time of this round can obtain the minimum value and total time is:

$$T_{lyq} = \sum_{i=1}^{b} \left(\frac{X_i}{\ln 2} + n_i \times \frac{Y_i}{\ln 2} \right) \times t \tag{13}$$

If there are b rounds of communication between the readers and the tags, the tags totally receive b vectors. To satisfy the false positive rate, b should meet the requirement $(1/2)^b < P_{req}$, i.e.,

$$b < -\log_2 \mathrm{P}_{req} \tag{14}$$

3.3 Large-XY-Query Algorithm

We generalize the above algorithm to present Large-XY-Query. In each round the readers transmit m_i vectors to the tags and the tags transmit n_i vectors to the readers. After each round, X and Y will change to (Fig. 4):

$$X_{i+1} = U_i \times (1/2)^{n_i} + W \tag{15}$$

$$Y_{i+1} = V_i \times (1/2)^{m_i} + W \tag{16}$$

The time of this round is:

$$f(m_i, n_i) = t/\ln 2 \times (m_i \times X_i + n_i \times Y_{i+1}) \tag{17}$$

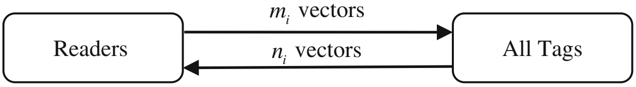

Fig. 4. The *i*th round in Large-XY-Query algorithm

This is an equation with two unknowns which change constantly so that we cannot obtain the optimal solution. Instead, we use the m_i and n_i in the two special conditions. Total time is obtained by the following equation:

$$T_{lxyq} = \sum_{i=1}^{b} \left(m_i \times \frac{X_i}{\ln 2} + n_i \times \frac{Y_i}{\ln 2} \right) \times t \tag{18}$$

The readers and the tags do not stop communicating until X_i or Y_i meets the following conditions.

$$\frac{|Y_i| - |W|}{|W|} < P_{req} \tag{19}$$

$$\frac{|X_i| - |W|}{|W|} < P_{req} \tag{20}$$

3.4 ITSP Algorithm

According to the experimental results in Sect. 4, we can get the final search protocol ITSP which can help us to choose the best algorithm based on the cardinality of X and Y.

Algorithm 1. ITSP algorithm

if |W|/min(|X|,|Y|)<=0.45 **then**
    return Large-XY-Query
else if |X|>|Y| **then**
    return Large-X-Query
else
    return Large-Y-Query
end if

4 Evaluation

In this section we evaluate the performance of the proposed ITSP algorithm in our paper. We mainly focus on the search efficiency and compare it with other protocols. During the simulation process, we assume that there is no transmission loss between RFID tags and the reader. In each frame the reader initiates the communication by sending commands to the tags and waits for tag's response. The RFID reader is capable of detecting and distinguishing empty slots from nonempty slots. In the ITSP series algorithms, we mainly focus on the size of transmitting vectors while other operation in RFID search is negligible, such as estimating the size of Y and the cost of transmitting hash seeds. In our simulation, we set both the R $\Rightarrow$ T transmission rate and the T $\Rightarrow$ R transmission to be 100 kbps. Accordingly, $t = \frac{1\,bit}{100\,kps} = 10^{-5}$sec.

In our simulation, we will assume that |W| and |Y| is known. Because CATS protocol cannot work when |X| > |Y| and we still need to take the variety of |W| in account. We will mainly talk about four cases. Table 2 shows the parameters of the four cases. We set $P_{req} = 0.001$ among all of these four cases. Figures 5, 6, and 7 show the result of these cases respectively. From Fig. 5, we can observe that the performance of ITSP series algorithm is better than that of CATS in terms of search time. The larger the ratio is, the better ITSP series algorithms perform. For example, when $ratio = 0.9$, the Large-XY-Query, Large-X-Query, Large-Y-Query algorithm reduce the search time of CATS as much as 19.3 %, 22.6 % and 23.7 % respectively. As we decrease the ratio, the gap between CATS and ITSP series algorithms gradually shrinks. At the same time, we can find that Large-Y-Query is the best algorithm when |X| > |Y|. Since the cardinality of Y is larger than X, the tags transmitting n_i vectors will bring more valid information in order to decrease the search time. CATS perform poorly when |X| > |Y|, and even cannot work when |X| ≫ |Y|. Figure 6 only analyzes the performance of ITSP series algorithms. In this case, ITSP series algorithms can work efficiently. We observe that Large-X-Query is the best algorithm in search efficiency. Because the cardinality of X is larger than Y, the readers transmitting m_i vectors will bring more valid information to tags.

Table 2. Parameter settings of four cases

Case	|X|	|Y|	|W|
1	$\|Y\| \times$ ratio	1000000	$\|X\|/2$
2	1000000	$\|X\| \times$ ratio	$\|Y\|/2$
3	500000	1000000	$\|X\| \times$ ratio
4	1000000	500000	$\|Y\| \times$ ratio

Next, we will focus the relationship between the search time and |W|. Based on this simulation, we will draw a conclusion that which is the best algorithm in ITSP series algorithms. Figure 7 shows that when $|W|/\min(|X|, |Y|) \leq 0.45$, Large-XY-Query is the best algorithm. While $|W|/\min(|X|, |Y|) > 0.45$, the best search algorithm was chosen depending on the relation of |X| and |Y|.

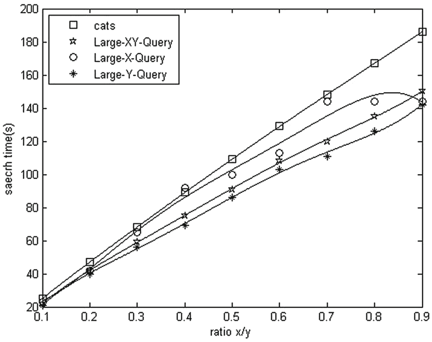

Fig. 5. The relation between search time and |X|

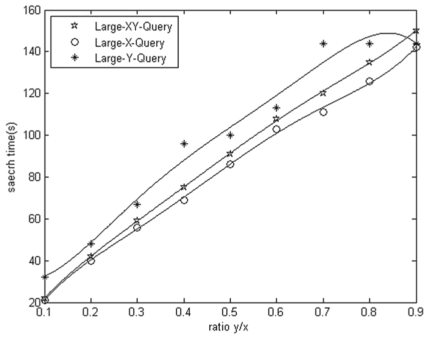

Fig. 6. The relation between search time and |Y|

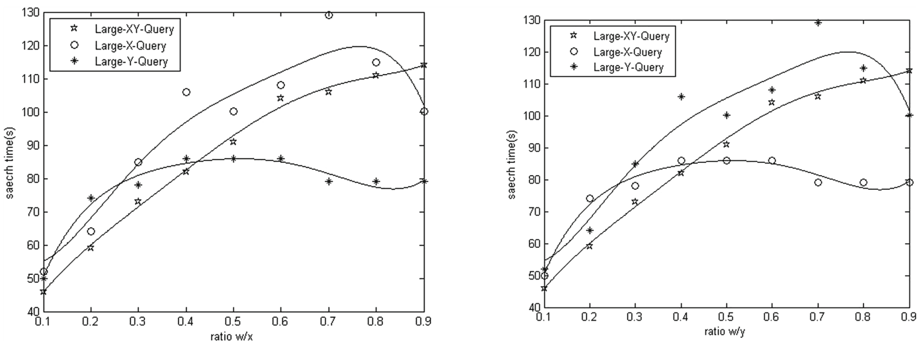

Fig. 7. The relation between search time and |W|

Another performance issue we want to investigate is relationship between the search time and P_{req}. Similarly, we will consider $|X| > |Y|$ and $|X| < |Y|$ these two cases. We use 3rd and 4th parameter settings in Table 2 and varying P_{req} from 10^{-1} to 10^{-5}. Figure 8 compares the search efficiency under different false positive rate circumstances. It is easy to find that the less false positive rate is, the longer search time is. Generally speaking, the gap between the search time required by the ITSP series algorithms and that by the CATS keeps getting larger with the decrease of P_{req}, particularly when P_{req} is small. Thus, ITSP series algorithm is more applicable for those application in which strict false positive rate is necessary. This also proves the validation of ITSP series algorithm in search efficiency.

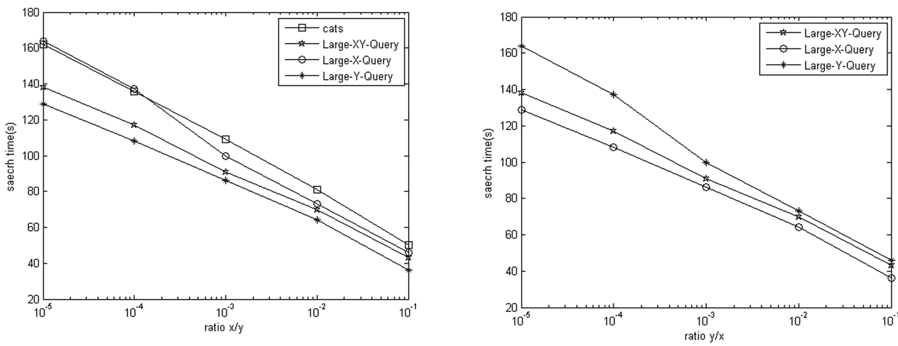

Fig. 8. The relation between search time and P_{req}

5 Conclusion

This paper have investigated the tag search problem in a large-scale RFID system and proposed ITSP – Cloud-based Iterative Tag Search Protocol. ITSP achieves high-levels of accuracy in a secure and efficient manner, no matter the cardinality of the wanted tags is bigger than all tags.

Acknowledgments. Dr. Zhang and Dr. Zhao are the corresponding author. This work was supported in part by the National Natural Science Foundation of China under Grant 61103185 and 61472283, the Fok YingTong Education Foundation of China under Grant 142006, the Research Funds for the Central Universities under Grant 2100219043, and the National Basic Research Program of China (973 Program) under Grant 2014CB340404.

References

1. Zhang, D., Zhou, J., Guo, M., Cao, J., Li, T.: TASA: tag-free activity sensing using RFID tag arrays. IEEE Trans. Parallel Distrib. Syst. **22**(4), 558–570 (2011)
2. Tan, C.C., Sheng, B., Li, Q.: Efficient techniques for monitoring missing RFID tags. IEEE Trans. Wirel. Commun. **9**(6), 1882–1889 (2010)
3. Chen, S., Zhang, M., Xiao, B.: Efficient information collection protocols for sensor-augmented RFID networks. In: INFOCOM 2011, pp. 3101–3109 (2011)
4. Lee, C.-H., Chung, C.-W.: Efficient storage scheme and query processing for supply chain management using RFID. In: SIGMOD Conference, pp. 291–302 (2008)
5. Qiao, Y., Chen, S., Li, T., Chen, S.: Energy-efficient polling protocols in RFID systems. In: MobiHoc 2011, p. 25 (2011)
6. Ni, L.M., Liu, Y., Lau, Y.C., Patil, A.P.: LANDMARC: indoor location sensing using active RFID. Wirel. Netw. **10**, 701–710 (2004)
7. Chang, J.-C., Wu, H.-L., Zhang, D.: On constant-time-identification and privacy-preserving RFID protocols: trade-off between time and memory. In: UIC/ATC 2013, pp. 613–618 (2013)
8. Li, Y., Ding, X.: Protecting RFID communications in supply chains. In: ASIACCS 2007, pp. 234–241 (2007)
9. Waldrop, J., Engels, D.W., Sarma, S.E.: Colorwave: an anticollision algorithm for the reader collision problem. In: ICC 2003, pp. 1206–1210 (2003)
10. Zhou, Z., Gupta, H., Das, S.R., Zhu, X.: Slotted scheduled tag access in multi-reader RFID systems. In: ICNP 2007, pp. 61–70 (2007)
11. Tang, S., Yuan, J., Li, X.-Y., Chen, G.: RASPberry: a stable reader activation scheduling protocol in multi-reader RFID systems. In: ICNP 2009, pp. 304–313 (2009)
12. Yoon, W., Vaidya, N.H.: RFID reader collision problem: performance analysis and medium access. Wirel. Commun. Mob. Comput. **12**(5), 420–430 (2012)
13. Zheng, Y., Li, M.: Fast tag searching protocol for large-scale RFID systems. IEEE/ACM Trans. Netw. **21**(3), 924–934 (2014)
14. Chen, M., Luo, W., Mo, Z., Fang, Y.: An efficient tag search protocol in large-scale RFID systems. In: IEEE INFOCOM, 2014, pp. 899–907 (2014)
15. Chun, J.Y., Hwang, J.Y., Lee, D.H.: Privacy-enhanced RFID Tag Search System (Chapter 9). In: Preradovic, S. (ed.) Advanced Radio Frequency Identification Design and Applications. InTech, Rijeka (2011). ISBN 978-953-307-168-8
16. Cha, J.-R., Kim, J.-H.: Dynamic framed slotted ALOHA algorithms using fast tag estimation method for RFID system. In: IEEE CCNC, 2006, pp. 768–772 (2006)
17. Chen, M., Gonzalez, S., Leung, V., et al.: A 2G-RFID-based e-healthcare system. IEEE Wireless Commun.**17**(1), 37–43 (2010)