International Journal of Information Technology and Knowledge Management
July-December 2009, Volume 2, No. 2, pp. 425-429

PERFORMANCE ANALYSIS OF ANTI COLLISION ALGORITHMS FOR TRACKING OBJECTS CONTAINING PASSIVE RFID TAGS

Indrajit Bhattacharya[*] & Uttam Kumar Roy[**]

Radio frequency identification (RFID) systems have emerged as an efficient and cost effective solution for tracking objects containing passive RFID tags. The choice of passive RFID tag is due to its low cost and simplicity of implementation. The problem is to identify objects containing passive RFID tags. If there are multiple objects in the range of a tag reader, they all send their ID to the reader simultaneously in response to the readers query. This leads to collision at the reader end and no tag is identified leading to the wastage of bandwidth and increase in total delay in identifying all objects. This paper presents some popular and efficient collision resolution protocols and one variation of that for the tag identification problem where a tag reader attempts to obtain the unique identity of the tags in its read range within finite time and significant delay. This paper also concentrates on different performance parameters regarding the singulation process and analyzes the performance of different anti collision algorithms on the basis of different performance parameters like total number of messages sent between tags and readers during the singulation process and the total amount of time for singulation.

Keynote: RFID, Anti-collision, Singulation.

1. Introduction

Radio Frequency Identification (RFID) [10] is so named because it relates to the identification of objects using EM radiation at radio frequencies. RFID system consists of a master device called an interrogator or reader and a number of slave device called tags. The interrogator communicates with tags via an RF link, so that all interrogator transmissions are heard (simultaneously) by all tags. RFID system is an automatic identification system. A RFID reader recognizes objects containing unique ID through wireless communication channel.

If there are multiple objects within the range of the tag reader then all objects send their identification to the tag reader and no tag is identified, leading to retransmission of tag Ids, which in effect leads to wastage of bandwidth and increase the total delay in identifying all the objects. Hence protocols need to be devised between the tags and tag reader to avoid or minimize collisions. The most important measure of system performance in RFID arbitration are the total time required to arbitrate total set of tags and the total power consumed by the tags during the process. This is in contrast to the measures of throughput and packet delay for typical multi access systems.

Tags are attached to the objects that need to be tracked or identified. Tags could be either passive or active. Active tags are those that are partly or fully battery powered, have the capability to communicate with other tags and can initiate a query of their own with the tag reader. Passive tags, on the other hand, do not need any internal power source, but are powered up by the tag reader. RFID systems have emerged as a promising solution for object identification. The most popular and widely used technology for object identification so far was magnetic bar code. But it had a very small line of sight and is difficult to adopt for some applications like supply chain management. The RFID system is cheap and error proof alternative to traditional object identification techniques.

2. Problem of Identifying Objects with Passive RFID Tags

The problem is to identify objects attached with passive tags. This problem is an extension of the multi-access channel problem [5] [6]. If there are multiple objects with in the range of the tag reader, then all objects will send their identification to the reader at about the same time in response to the reader query, which will lead to collision. These collisions at the tag reader can cause false negative and false positive effect in the tag identification.

i) False negative reads, where RFID tags might not be read at all, leading to the mistaken belief that the object is not present.

ii) False positive reads, where RFID tags might be read when they are outside the region normally associated with the location of the RFID reader, leading to a mistaken belief that the object is present.

3. Anti Collision Algorithms

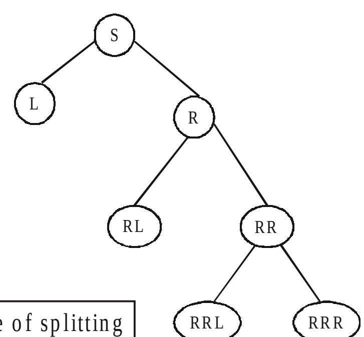
A variety of anti-collision algorithms [7] [8] [9] has been

[*] Kalyani Government Engineering College, Kalyani, Nadia, WB *E-mail: indra51276@yahoo.com*
[**] Jadavpur University, WB. *E-mail: u_roy@it.jusl.ac.in*

proposed over the time in this context to solve the problem of uniquely identify the tags in the vicinity of a reader and is discussed in the later section with their relative merits and demerits.

3.1. Tree-Search Algorithm

In conventional multi-access systems, Tree-search algorithm introduced by Capatanakis [1] [3] can be used for RFID arbitration. Nodes transmit packets in time slots when queried by the receiver or reader. If more than one node choose same time slot then a collision occurs at the reader and no response is obtained. In this algorithm collision is resolved by splitting the set of colliding nodes into two subsets. Nodes can be divided into subsets in three different ways [2] when a tag reader senses collision. In simplest way when required each station tosses a two sided coin and joins the first subset with probability *p* or the second subset with probability *p*. In other way one can assume that all stations have binary addresses preassigned to them. Each time there is a conflict, the station uses the corresponding bit of its address to decide which sub-tree to join. Another approach is to use the arrival time of the packet to decide which sub-tree to join. The packets must therefore have been time stamped with their arrival time. The channel access algorithm specifies an initial time interval at the beginning of the conflict resolution interval, and all packets with the timestamp in the specified interval get transmitted. After a conflict, the station joins the left sub-tree if its packet arrived during the first half of the last tried interval and transmits in the next slot; otherwise it joins the right interval and defers transmission. Nodes in the first subset transmit in the first time slot. Nodes in the other subset wait till the collision between the first subset of nodes is completely resolved. If the first subset of nodes encounters another collision, then further splitting is done. This is done recursively till all the collisions have been resolved. Once all the collisions in the first subset of nodes are resolved, then a similar procedure is followed for the second subset.



Example of splitting algorithm

Advantages

No clocking circuitry is required at the tags. The complexity at the tag is significantly minimized to reduce the price. The number of time slots required to identify the tags is linear in number of tags (*m*) and the number of tag replies is of the order *mlogm*.

Disadvantages

This protocol expects the tag to maintain a counter other than storing the unique identifier. If the tags discharged, then the counter value is lost. The computation required at the tags is considerable, as the tags have to generate a random number and then split themselves into subsets.

3.2. Memory Less Protocol

In memory less protocol [4] [5], each tag is uniquely identified by a binary string of *k*-bits. There are 2^k number of objects can be identified by the tag reader. In this protocol, the tag reader sends out string prefixes or query via the common communication channel. The tag reader tries to explore all the possible values of the *k*-bit string. The process of identification of the tag is hierarchical and is depicted in the form of a query tree, hence named as query tree protocol. In this protocol time need not be slotted and tags do not need to maintain any state information and they never communicate with each other. They just respond with the reader's query.

Process of Identification of four objects (010, 011, 101, 111) using Memory Less Protocol

Steps	*Query*	*Response*
1	Empty string	Collision
2	0	Collision
3	1	Collision
4	00	No Response
5	01	Collision
6	10	101
7	11	111
8	010	010
9	011	011

Advantages

The tag is memory less with minimal computational power. Hence cost of the tag can be reduced.

Disadvantages

The tag reader need to go through all the values of the *k*-bit string hence the search space is fairly large. The time to identify tags is $O(2^k)$, where *k* is the length of the string.

The basic query tree protocol can be improved by skipping internal nodes where collision is bound to happen. For example, suppose a collision occur for the sting of query *p*. Again for query prefix *p*0 collision did not occur in the next cycle. Then for the query string *p*1 collision is guaranteed in the next cycle. So instead of sending the query string *p*1 it is wised to send *p*10 or *p*11 in the next cycle. Hence it reduces the number of message sent by the tag reader. The algorithm can be further improved if the tag reader has the prior knowledge about maximum number of tags. And in this process the query string can be advanced by more than one bits at some cycles.

4. PROPOSED ALGORITHM

We consider a tag identification system, which consists of one reader and n tags. The reader is a powerful entity with lot of memory and computational power. But the tags have limited memory and activated by the tag readers.

There is a single communication channel between the reader and tags. Tags are not able to exchange messages among each other. The reader can broadcast messages to the tags. In the response the tags can send their identity to the reader. If only one tag responds to the query of the tag reader then it gets identified by the reader, otherwise a collision occurs and no tag gets identified.

Each tag has a unique ID string in $\{0, 1\}^k$, where k is the length of the ID string and a counter which is initialized by the ID key value of the tag. At the beginning, the reader does not know anything about the tags. The tag identification protocol specifies the algorithm for the reader so that it can identify all the tags with in finite time.

4.1. Algorithm

The algorithm can be described in a step-wise fashion as follows:

a) The tag reader sends out a small signal for any tag present in its zone.

b) Two possible cases can arise based on tag response.

   i) only one tag is present and gets identified.

   ii) multiple tags are present leading to collision at the reader.

c) If there is a collision tag reader broadcasts a query to the tags 'whose counter value is 0?'

d) Two possible cases can arise based on tag response.

   i) there is a tag with counter value 0 and gets identified by the reader.

   ii) no tag has counter value zero and the respective counter value of the tags are decremented by one.

e) Repeat steps a. to d. until all tags are identified.

4.2. Analysis

To implement the algorithm no clocking circuitry is required at the tag. The complexity at the tag is significantly minimized as minimum amount of computation is required at the tag end. The algorithm requires no probabilistic measure and partitioning of colliding nodes is not required. The number of cycles required to identify the tags is bounded by $O(n)$ where $n = 2^k$.

The number of cycles required to identify the tags can be further reduced by modifying the query of the reader in one/more cycle. After a collision is detected by the reader in a cycle and the query send by the reader results in 'no response' from the tags. Then in the next cycle a collision is going to occur again. We can avoid/minimize this by changing the query pattern of the reader. Reader is going to send the modified query as 'is there any tag with counter value m–1', where m is the ID key value of the last tag identified by the reader.

5. IMPLEMENTATION AND RESULTS

We have implemented the proposed algorithm [algorithm 3] in JAVA platform and analyzed the performance of the algorithm and compare them with existing Tree based [algorithm 2] and Memory less algorithm [algorithm 3] according to several performance parameters [4].

The critical parameters that are taken into consideration are:

NO OF TAGS: Total number of tags under the reader in the simulation environment. We have tested the system with varying number of tags like 8, 16, 32, 64 & 128.

NO OF BITS: The total number of bits required to represented unique tag identification. In our simulation we have tested the system with varying tag id like 8, 10, 12, 14, & 16.

RUNNING TIME: The total amount of time spent in singulation of tags and is measured in nanosecond precision.

NO OF MESSAGES: The total number of messages that is transferred between the reader and the tags during the singulation process.

We have carried out our experiment and compare the performance of the algorithms on the basis of followings:

1. How the number of messages in the system varies with increasing number of tags, with fixed no of bits in tag ID field; separate test cases in varying tag ID bits. (Figure 1 & Figure 2).

2. How the time for tag identification varies in the system with increasing number of tags, where number of bits in tag ID is fixed, separate test cases in varying tag ID bits. (Figure 3 & Figure 4).

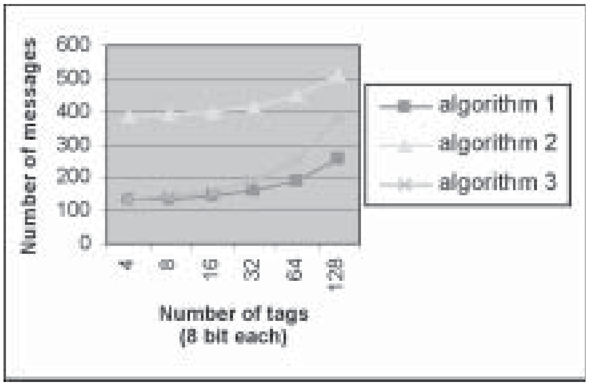


Figure 1: Number of Tags v/s Number of Messages

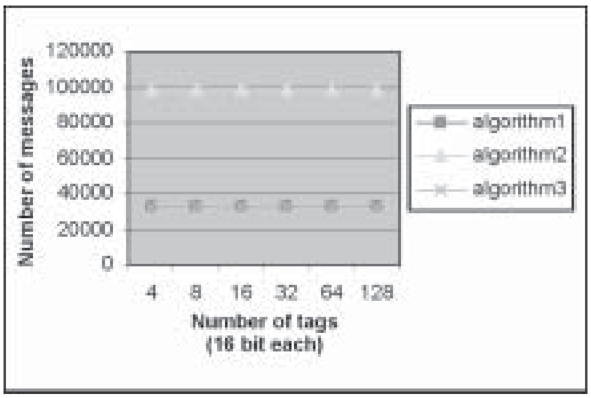


Figure 2: Number of Tags v/s Number of Messages

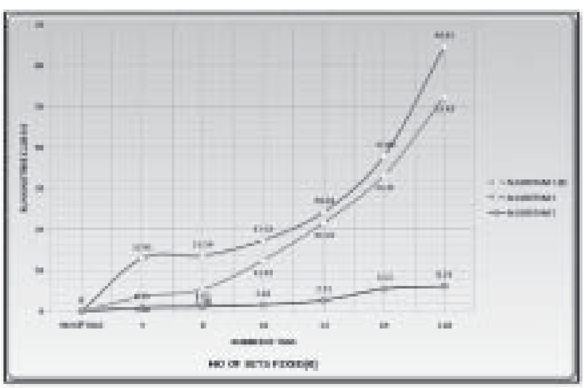


Figure 3: Number of Tags v/s Time

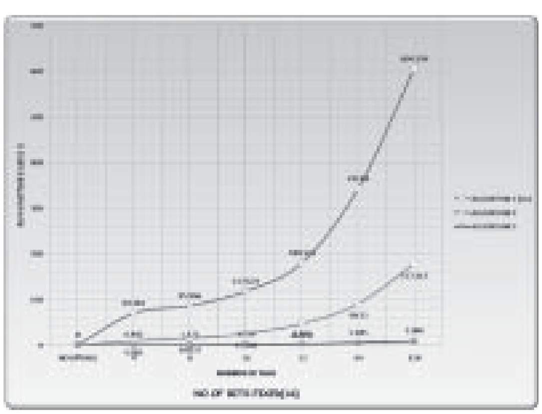


Figure 4: Number of Tags v/s Time

6. CONCLUSION

In Figure 1 and 2 we analyzed the system in terms of number of message send between the tags and readers varying the number of tags in the singulation process. In first case the number of bits of each tag is considered as 8 bits and in the second case the number of bits of each tag is considered as 16 bits.

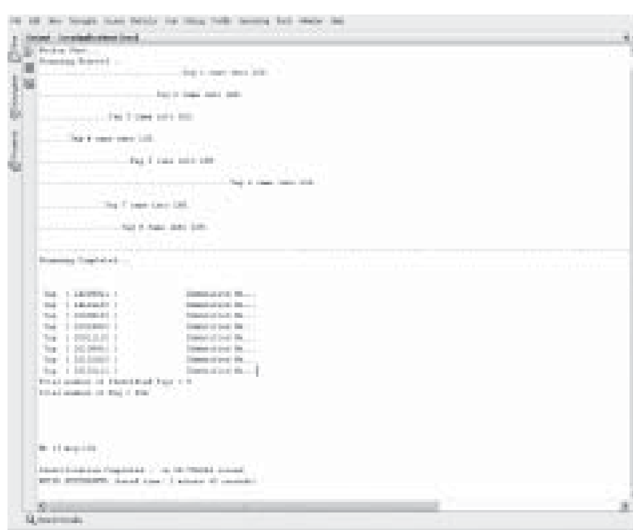
From the above comparisons it is clear that the algorithm 3 and the algorithm 1 gives more or less similar performance when we compare them in terms of total number of messages transmitted between the tags and the reader in the process of singulation, where as algorithm 2 performs worst in this context. This is because in the memory less algorithm (algorithm 2) the Reader has to send all combination of string patterns to uniquely identify the tags in its range, and hence a large number of messages has been send between reader and tag which results in high message overhead in the system.

In Figure 3 and 4 we analyzed the system in terms of amount of time needed for singulation varying the number of tags in the system. In first case the number of bits of each tag is considered as 8 bits and in the second case the number of tags of each tag is considered as 16 bits.

From the above comparisons it is clear that Algorithm 3 outperforms other two algorithms in terms of the total time required to singulate the tags under the reader. This is because depending upon the outcome of the Reader's query in each round, we can skip some bits and reader can send the next query accordingly.

Hence we can conclude that our proposed algorithm i.e. algorithm3 is the best choice to singulate the tags in a single reader multi tag environment as the singulation process is relatively fast than other existing algorithms.

Snap-shot of the Simulation:



REFERENCES

[1] Capetanakis, J. I. Tree Algorithms for Packet Broadcast Channels. IEEE Transactions on Information Theory. *IT-* **25**(5) (1979) 505-515.

[2] L. Molle, George C. Polyzos. Conflict Resolution Algorithms and their Performance Analysis. Mart Technical Report Number CS93-300, (1993). *http://citeseer.ist.pdu.edu/molle93Conflict.html.*

[3] Hush, Don R. and Wood, Cliff. Analysis of Tree Algorithms for RFID Arbitration. IEEE International Symposium on Information Theory, *IEEE*, (1998) 107.

[4] Law, Ching, Lee, Kayi and Siu, Kai-Yeng. Efficient Memory Less Protocol for Tag Identification. In Proceedings of the 4^{th} International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, *ACM*, (2000) 75-84.

[5] MIT Auto-ID centre. *http://auto-id.mit.edu/.*

[6] Robert G. Gallager. A Perspective on Multi-Access Channels. IEEE Transactions on Information Theory, IT-**31**(2) (1985) 124-142.

[7] M. Jaoccurt, A. Ehrsam and U. Gehrig, Contactless Identification Devices with Anti Collision Algorithm, IEEE Computer Society CSCC 99, Conference on Circuits, Systems, Computers and Communications, Athens, (1999).

[8] Songsen Yu, Yiju Zhan, Zhiping Wang and Zhongping Tang, Anticollision Algorithms Based on Jumping and Dynamic Searching and its Analysis, Computer Engineering, **31** (2005) 19-20.

[9] Ho-Seung Choi, Jae-Ryon Cha and Jae-Hyun Kim, Fast Wireless Anti-Ollision Algorithm in Ubiquitous ID Systems, Vehicular Technology Conference, 2004. VTC 2004-Fall 2004, *IEEE* 60^{th} **6** 4589-4592.

[10] K. Finkenzeller, RFID Handbook: Radio-Frequency Identification Fundamentals and Applications, John Wiley and Sons Ltd., ’99.