

# RFID: Review & Comparison of Tree Based Tag Collision Resolution Algorithms

Shilpa Bhatt<sup>1</sup>, Deepak Dhadwal<sup>2</sup>, Ayush Bhatt<sup>3</sup>

<sup>1</sup>Student, ECE, Maharishi Markandeshwar University, India

<sup>2</sup>Associate Professor, ECE, Maharishi Markandeshwar University, India

<sup>3</sup>Engineer, Engineers India Limited, India

## Abstract

In RFID, large number of tags communicates with small number of readers by simultaneous transmission. This situation leads to collisions for which several algorithms have been developed keeping in mind the goal of its avoidance. This being the major contributor of performance in a RFID system, it has attracted a lot of researchers for developing fast algorithms. Some of such widely used tree based algorithms includes binary search tree, back tracking based, and, matrix based algorithms are discussed, and, there MATLAB simulation is performed. Binary search algorithm uses NRZ encoding to find the collision bits and on its basis, it finds the least valued tag. In back track algorithm the process of finding the least value tag is backtracked to find the next least value tag. In matrix based algorithm, the tags are divided into groups to reduce the chances of collision. All these algorithms are simulated using MATLAB, and, simulation results are compared and displayed using line graph.

**Keywords-** Anti-Collision, Algorithm, RFID, Binary tree, Backtrack Binary Tree, Matrix

## I. INTRODUCTION

RFID systems are a better approach over barcode scanner as it can read multiple tags simultaneously and also it can store more data with a facility to modify it at a later stage. RFID stands for “Radio Frequency Identification” which is a non-light sight communication, and, makes use of radio frequencies to transmit or receive signals.

Radio Frequency Identification (RFID) tag commonly known as electronic tag, is a non-contact automatic identification technology, which takes the RF signal as the transfer medium of information and energy, so as to complete the information exchange with the measured objects [1].

An RFID system is always made up of two components: The transponder located on the object to be identified, and, the interrogator or reader, which, depending upon the design and the technology used, may be a read or write/read.

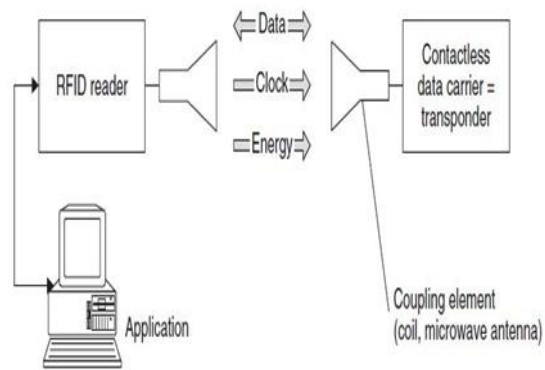


Figure 1: A Broad View of Rfid System

A reader typically contains a radio frequency module (transmitter and receiver), a control unit and a coupling element to the transponder. In addition, many readers are fitted with an additional interface (RS 232, RS 485, etc.) to enable them to forward the data received to another system (PC, robot control system, etc.). The transponder, which represents the actual data-carrying device of an RFID system, normally consists of a coupling element and an electronic microchip. When the transponder, which does not usually possess its own voltage supply (battery), is not within the interrogation zone of a reader is totally passive. The transponder is only activated when it is within the interrogation zone of a reader. The power required to activate the transponder is supplied to the transponder through the coupling unit (contactless), as are the timing pulse and data [2].

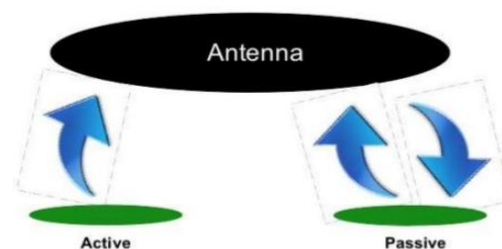


Figure 2: Active and Passive Tags

RFID tags fall into two categories: Active and Passive (Memory less). An active tag is equipped with a battery and relatively expensive circuit. Thus it can perform complicated processing, but its size is

large. On the other hand, a passive tag is cheap, small, and powered by the radio from reader, but it has no storage for processing history, except for the ID and some embedded information. In usual active tags are used for large goods such as container in port or cars in parking area, while passive tags for small items such as goods in a retail warehouse [3].

In past, there have been several technical obstacles that have kept RFID aloof from gaining full acceptance in consumer applications as well as in enterprises. One of such important performance issues have been the RFID collision problem. This problem typically results in a failed transmission. It occurs in signal transmission of the readers or the tags, which hardly leads to fast identification. Therefore, it becomes a key issue to develop an efficient anti-collision protocol so as to reduce collisions in the interrogation zone of an RFID reader. The RFID collision problems are classified into Tag collisions and Reader Collisions.

Collisions are divided into reader collisions and tag collisions. Reader collision problems arise when multiple readers are simultaneously used. The other, most important, collision problem (approached in this paper) is the tag collision that occurs when several tags try to answer to a reader query at the same time [6]. The RFID tag collision occurs when multiple tags respond simultaneously to a reader's signal and their communication signals interfere with one another confusing the reader. The Reader-Tag collision happens when signal from a neighboring reader interferes with tag response being received at another reader. It also happens when tag hears multiple readers at the same time. The tag collision problems are further subdivided into active tag collisions and passive tag collisions, where passive tag collisions are more complicated. Keeping in view, that the problem occurs more often in practical systems than the other two counter parts and since most of the applications involve tag collision, and in that too the passive tag collision problems are more challenging, a tag anti-collision deterministic algorithm from the former category has been designed for passive tags. Currently in industry, one solution to the above problem is found by establishing a two-way communication link between the tag and the interrogator, which is the Interrogator-Talks First (ITF) protocol. A good tag collision arbitration protocol for RFID tags should have the following characteristics: First, a interrogator ought to identify all the tags inside its own reading range. Since the interrogator cannot estimate the number of tags precisely, the guarantee of recognizing all tags must be taken into consideration in the design of the tag hard system and anti-collision protocol. Second, a tag should be identified while consuming a small amount of resource, since the tag has low power.

Thus, the tag anti-collision protocol must load the tag with the least possible communication time [7].

## II. AVIALABLE SOLUTIONS

Conventionally the problem of tag collision in RFID was addressed by ALOHA systems. The random access method employed by The ALOHA systems were based on the use of error detecting code. Each user at a console transmits packets to the reader over the same high data rate channel in a completely unsynchronized (from one user to another) manner. If and only if a packet is received without error it is acknowledged by the reader. After transmitting a packet the transmitting console waits a given amount of time for an acknowledgement; if none is received the packet is retransmitted. This process is repeated until a successful transmission and acknowledgement occurs or until the process is terminated by the user's console in case of collision (through the use of jam signal). On receiving jam signal all the transmitters enter into wait mode for a self-picked random time, and, then retransmits. This algorithm has a serious problem of tag starvation. Then to overcome this problem tree based algorithms of collision resolution were invented. These were binary tree protocol & query tree based protocol. Although they do not have the problem of starvation but could cause incur delays. One solution could be splitting of colliding tags into subsets and try to recursively do this until a subset has only a single tag.

### A. Binary search Algorithm

This algorithm considers every source as a leaf on a binary tree. Representation of these sources could be considered as Binary addressing schemes. For this, it uses NRZ binary encoding in which zero is an invalid state. So whenever it experiences an invalid state in at least one bit of received inputs, it follows certain steps: Firstly, the reader identifies the position of bits where collision has occurred.

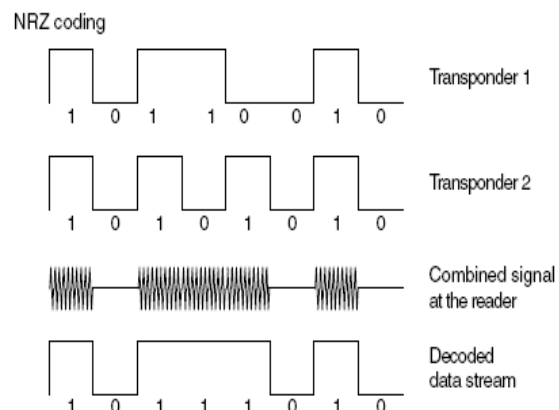


Figure 3: NRZ Coding Used for Bit Collision Detection

After the completion of the read/write operations, transponder can be fully deactivated by an UNSELECT command, so that is no longer responds to the next REQUEST command. Now, reader will

request all those transmitters to resend their tag ID who's MSB (in which collision has occurred) is zero. This process continues till the collision is resolved & then the data is read from selected tag. The reader iterates through above listed tags till all the tags are read.

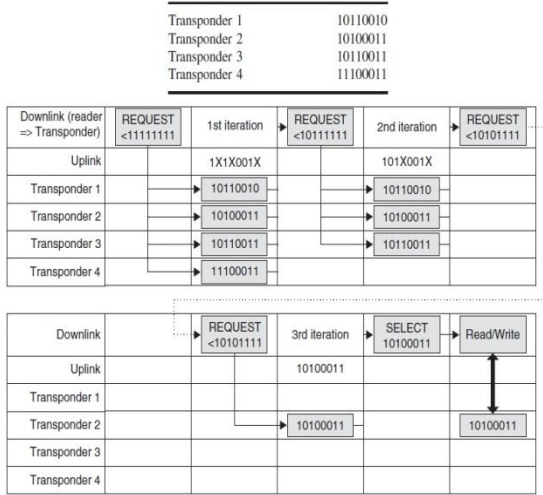


Figure 4: Binary Search Algorithm

A modification was made on traditional binary search algorithm, which optimizes the worst case of the binary search algorithm by comparing the input element with the first & last element of the data set along with the middle element and also checks the input number belongs to the range of numbers present in the given data set at each iteration there by reducing the time taken by the worst cases of binary search algorithm [11].

**B. Back Tracking Binary Search**

In binary search, say there are n parallel transmitters, the algorithm finds the tag with least numeric value, reads its data & then iterates the entire process to find the next least valued tag. This entire process is repeated n number of times to read the data from entire available tags. So this algorithm was enhanced to backtrack this process to find least valued tag ID .In this a depth counter is maintained which is by default initializes to zero and each step when a receiver selects some tags on the basis of a bit .All the selected counters incremented their depth counters by one. Once a tag is read the receiver sends an active signal to all the transmitters which ask them to decrement their depth counter by one making the one with zero depth active. So in this way the depth counters helps in back tracking the previous unselected tags thereby reducing the collision dramatically.

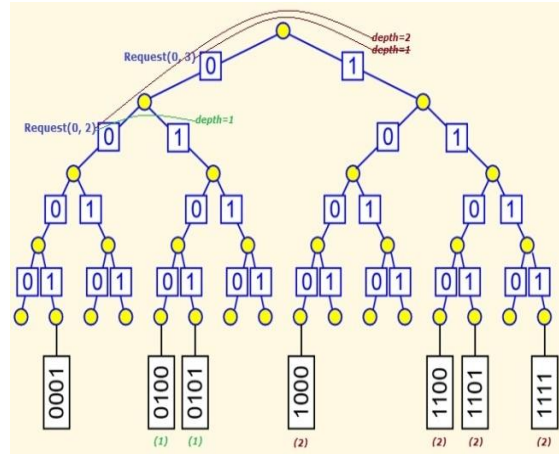


Figure 5: Backtrack Binary Search

On the base of the binary search algorithm of backtracking, an enhanced binary anti-collision search algorithm for radio frequency identification (RFID) system was also proposed. With the method of transferring the collision bit in place of the ID of the tag, the proposed algorithm improved identification efficiency significantly. As the amount of data transfer was reduced, it also reduced the need of high bandwidth and high baud-rate [8].

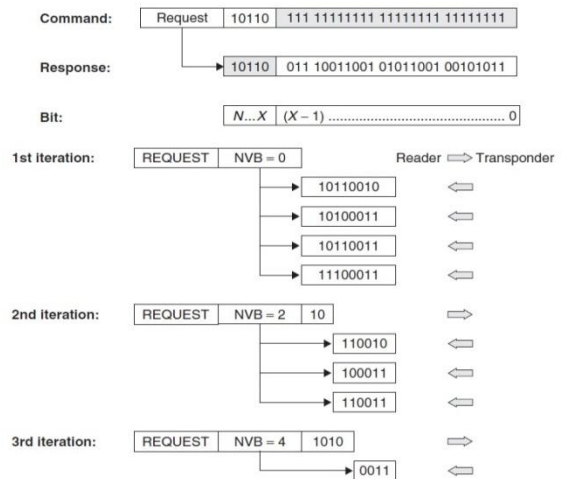


Figure 6: Enhancement over backtracking Binary Search Algorithm

**C. Matrix Based Binary Search**

Till now, all the research ware made for optimization of collision resolution, but then came the need to optimize the collision itself, i.e. to reduce chances of collision. So, an algorithm was evolved in which the tags were divided into groups, hence reducing the collision as depicted in the diagram.

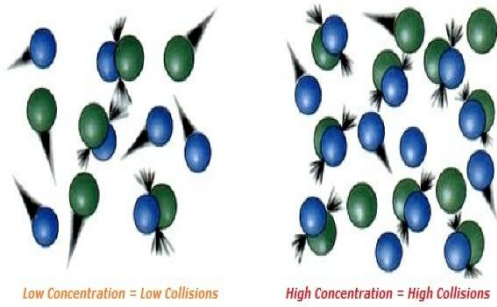


Figure 7: Reduced Collisions due to Reduced Population

In this they introduced the concept of grouping among the tags. The purpose of grouping is to reduce the search time of identifying tags by using the BST algorithm. In result, it performs multiple searches to continuously identify the tags. There needs to be pointed out that the number of tags of each group is not as little as possible while grouping. It should make reasonable grouping according to the current number of tags. Therefore, they proposed the idea of a matrix-based tag ID grouping. For the BSTM algorithm, the key is how to reasonably group tags. The method was to calculate the value of square root according to the number of tags. Then it was selected as the row value of the matrix with the smallest integer which was greater than the value of square root, that is:  $Line = \sqrt{N}$ . On this basis, they calculated the column value of the matrix, that is:  $Column = N / \sqrt{N}$ . The time which the BSTM algorithm spends includes two aspects in theory: the time of the matrix-based grouping; the time of searching for the BST algorithm. This algorithm reduced the chances of tag collisions, that was very much needed. However the grouping can be made multidimensional, thereby further reducing the probability of collision to next level.

III. SIMULATION

To compare the anti-collision algorithms, the simulation is achieved by using MATLAB. The simulation can process the anti-collision procedure between reader and tags, and use different algorithms such as Binary Tree, Back track binary tree & matrix algorithm. The simulation is achieved by considering a configurable parameter, i.e. the number of tag resulting in collision.

A. Simulation of Binary Search Tree

The simulation of binary search is based on NRZ encoding which is shown in table below. It shows the number of tags read by the reader & total number of iterations consumed for reading all tags. In this simulation, MATLAB is used which takes the input from excel file and a receiver class generates the corresponding NRZ Code.

Tags	64	128	256	512	1024
Iterations	192	448	1024	2304	5120

Table 1: Binary Search Results

B. Simulation of Back Track Binary Search Tree

For simulating back tracked binary, same receiver class is used which is inherited by back tracked binary class. This class along with tag matrix maintains a corresponding depth matrix which is updated in every iteration. Once the first iteration is complete, the depth matrix is fully loaded & hence with its help the whole process of iteration is back tracked for which results are shown in below table:

Tags	64	128	256	512	1024
Iterations	69	134	263	520	1033

Table 2: Backtrack Binary Search

C. Simulation of Matrix Based Algorithm

The simulation is done for matrix algorithm by inheriting the back tracked binary class. On the basis of number of group to be formed is identified by considering the integer value of square root of total number of tags. This value is then divided by total number of tags to identify the number of tags in each group. As a part of pre-processing, the class generates depth matrix by assigning each group with incremental depth. As a result of this pre-computed depth matrix, the tags are iterated group wise. The table shown below comprises of its results:

Tags	64	128	256	512	1024
Iterations	66	131	259	515	1028

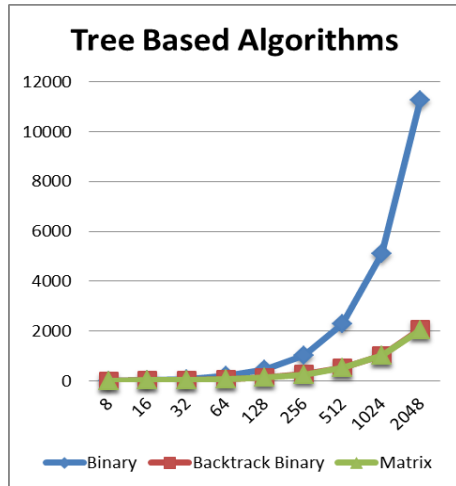
Table 3: Matrix Based Binary Search Results

IV. RESULTS

After all algorithms were performed in MATLAB, it was observed that initially there were a lot of collisions happening during binary search. But, when backtracking was applied over binary, the collisions exponentially reduced. Then the grouping reduced the collisions further to some extent.

No Tags	Binary	Backtrack Binary	Matrix
2048	11264	2058	2053
1024	5120	1033	1028
512	2304	520	515
256	1024	263	259
128	448	134	131
64	192	69	66
32	80	36	33
16	32	19	17
8	12	10	9

Table 4: Comparison of Tree Based Algorithms



Graph 1: Tag – Iteration Comparison Graph of All Three Tree Based Algorithms

### V. CONCLUSION

Collision detection is a crucial task in RFID systems. In this paper, different algorithms are shown which have their own capability to read the number of tags. The matrix based is a better approach over binary & back tracking as it uses benefits of both algorithm & also, it uses grouping technique which enhances its application. An intelligent algorithm can be developed to resolve the collision.

### REFERENCES

[1] Mingsheng Hu, Zhijuan Jia, Xiaoyu Ji and Liu Hong, “RFID Anti-Collision Algorithm Based on Occurrence Position”, Research Journal of Applied Sciences, Engineering and Technology, vol. 5, pp. 23- 29, January 2013.

[2] Klaus Finkenzeller [2010]. “RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification”, 3rd edition, John Wiley & Sons.

[3] Haosong Gou, Younghwan Yoo, “Bit Collision Detection Based Query Tree Protocol For Anti Collision In RFID System”, International Journal of Innovation Computing, Information & Control , vol. 8, Number 5(A) , pp. 3081-3102, May 2012.

[4] Wang Jianfang, “A Novel Anti-collision Backtracking Algorithm Based on Binary-tree Search in UHF”, Proceedings of the Third International Symposium on Computer Science and Computational Technology, ISCSCT 2010, pp. 212-214, August 2010.

[5] Lvqing Yang, Caili Wang, Wenhua Zeng, “Research of Matrix-based Grouping Method on Anti-collision Algorithm for RFID Tag Identification”, Advances in information Sciences and Service Sciences (AISS), Volume5, Number13, pp. 126-123, August 2013.

[6] Usama S. MOHAMMED, Mostafa SALAH , “Tag Anti-collision Algorithm for RFID Systems with Minimum Overhead Information in the Identification Process”, RADIOENGINEERING, VOL. 20, NO. 1, pp. 61-68 APRIL 2011.

[7] Geng Shu-qin, Wu Wu-chen, Hou Li-gang and Zhang Wang (2010). “Anti-Collision Algorithms for Multi-Tag RFID”, Radio Frequency Identification Fundamentals and Applications Bringing Research to Practice, Cristina Turcu (Ed.), ISBN: 978-953-7619-73-2, InTech, Available from: <http://www.intechopen.com/books/radio-frequency-identification-fundamentals-and-applications-bringing-research-to-practice/anti-collision-algorithms-for-multi-tag-rfid>

[8] X.-L. Shi, X.-W. Shi, Q.-L. Huang, and F. Wei, “An enhanced Binary Anti-Collision algorithm of backtracking in RFID System”, Progress In Electromagnetics Research B, Vol. 4, pp. 263–271, 2008.

[9] Jihoon Myung, Student Member, IEEE, Wonjun Lee, Senior Member, IEEE, and Jaideep Srivastava, Fellow, IEEE, “Adaptive Binary Splitting for Efficient RFID Tag Anti-Collision”, IEEE COMMUNICATIONS LETTERS, VOL. 10, NO. 3, pp 144-146, MARCH 2006.

[10] Anchala Kumari, Rama Tripathi, Mita Pal and Soubhik Chakraborty, “Linear Search versus Binary Search: A Statistical comparison for binomial inputs”, International Journal of Computer Science, Engineering and Applications (IJCSA) Vol.2, No.2, pp. 29-39, April 2012.

[11] Ankit R. Chadha, Rishikesh Misal, and Tanaya Mokashi, “Modified Binary Search Algorithm”, International Journal of Applied Information Systems (IJ AIS) – ISSN : 2249-0868 Volume 7– No. 2, April 2014.

[12] Parveen Kumar, “Quadratic Search: A New and Fast Searching Algorithm (An extension of classical Binary search strategy)”, International Journal of Computer Applications (0975 – 8887), Volume 65– No.14, March 2013.

[13] Ajit Singh and Dr. Deepak Garg, “Implementation and Performance Analysis of Exponential Tree Sorting”, International Journal of Computer Applications (0975 – 8887), Volume 24– No.3, pp. 34-38, June 2011.

[14] <http://www.slideshare.net/KALPNA06/rfid-presentation-37643386>

[15] <http://lbi.ro/teachinginnovatively/store/soft.php?soft=47>

[16] <http://wikipedia.unicefuganda.org/latest/A/Chemical%20kinetics.html>