Radio Frequency Identification Systems
New Ideas and Algorithms

Leonid Bolotnyy and Gabriel Robins
lb9xk@cs.virginia.edu, robins@cs.virginia.edu

Department of
Computer Science
University of Virginia
School of Engineering & Applied Science

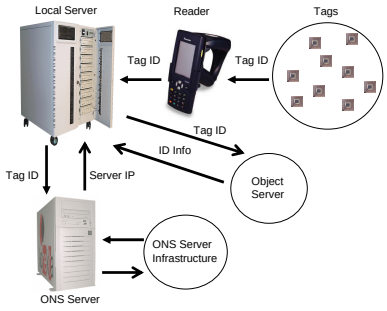
Introduction to Radio Frequency Identification (RFID) Systems

www.cs.virginia.edu/robins

RFID Primer

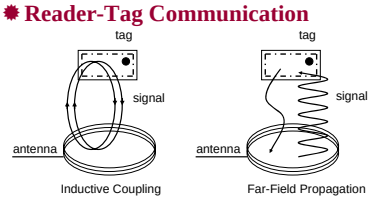
- Three types of RFID tags
  - Passive
  - Active
  - Semi-Active
- Operational Frequencies
  - 125KHz - 5.8GHz
- Operational Range
  - 5mm - 15m
- Standardization Bodies
  - International Organization for Standardization
  - EPCglobal, Inc

EPC System Architecture



Applications

Reader-Tag Communication



Major Research Issues

- Reducing the cost of tags
- Providing security and privacy
- Standardizing the technology

Multi-Tag RFID Systems

Attach more than one tag to an object

- Redundant Tags
- Dual-Tags
  - Private memory only
  - Shared memory only
  - Shared and private memory
- n-Tags

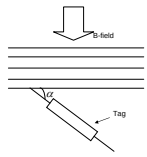
Benefits of Multi-tags

- Increased expected voltage on a tag
- Increased expected communication range
- Increased memory
- Increased reliability
- Increased durability

Reliability and Dependability

- Object's detection is more likely
- Failure of a redundant tag
  - leaves the system functional
  - is detectable in some systems

Voltage on a tag

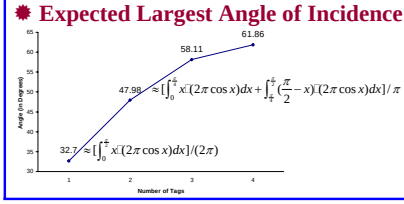


$V_0 = 2\pi f N S B_0 \cos\alpha$

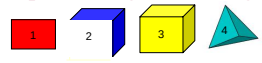
where:
f = frequency of the arrival signal
N = number of turns of coil in the loop
S = area in the loop in meters (m^2)
B_0 = strength of the arrival signal
α = angle of the arrival signal

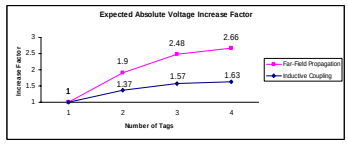
Inductive Coupling: $Voltage \propto \sin(90-\alpha)$
Far-Field Propagation: $Voltage \propto \sin^2(90-\alpha)$

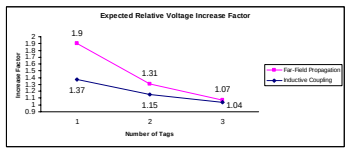
Expected Largest Angle of Incidence



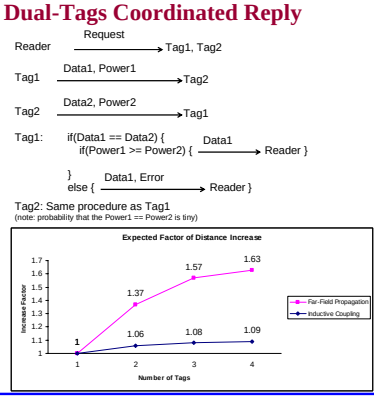
Optimal Tag Positioning







Dual-Tags Coordinated Reply



Applications of Multi-Tags

- Supply chain management
  - to increase chances of object detection
- Luggage tracking
  - regulations require different algorithms
- Preventing illegal deforestation
  - tagging of trees to prevent illegal logging

Effect on Singulation Algorithms

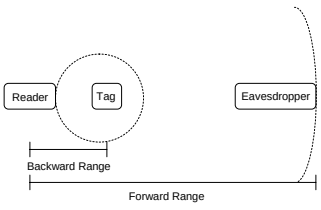
Algorithm	Redundant Tags	Dual-Tags
Binary	No Effect	No Effect
Binary Variant	No Effect	No Effect
Randomized	Doubles Time**	No Effect*
STAC	Causes DOS	No Effect*
Slotted Aloha	Doubles Time**	No Effect*

* If Dual-Tags communicate to form a single response
** Assuming an object is tagged with two tags

Security Enhancement

- n-Tags send "chaff" hiding the real IDs
- Recycled IDs are good "chaff" source
- "Chaffing and winnowing" has a cost
  - extra tag functionality
  - overhead to create and filter "chaff"

Randomized Tree Walking Algorithm




```
Traverse(i, count)
    b_i := Read random bit i
    if collision on b_i detected:
        Suspend all tags with b_i == 1
        Each suspended tag stores i
        Traverse(i+1, 0)
        Wake up tags suspended on bit i
        Traverse(i+1, 0)
    else if no collision on b_i detected:
        if(count > threshold)
            Tree-Walk remaining tags
        else Traverse(i+1, count+1)
```

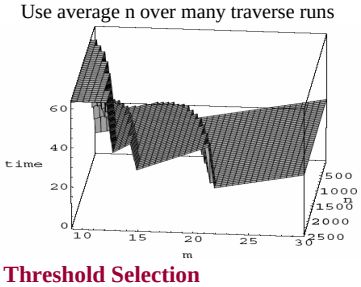
Secure Binary Tree-Walking
i. Each tag generates a random number
ii. Reader performs a tree-walk
iii. Selected tag transmits its real ID

Major questions:

- **How to deal with collisions on the tags' real-IDs?**
- **How to choose the optimal length for random numbers?**
- **How to select the threshold?**

Optimal Random Number Length

Use average n over many traverse runs



Threshold Selection

- **Start the threshold at 2**
- **Increase threshold by 1 if a collision occurs**
- **Decrease threshold by 1 if no collisions occur for entire traversal**

Randomized PRF Tree Walking Algorithm

Goal: Efficiently solve reader-tag authentication problem in the presence of many tags

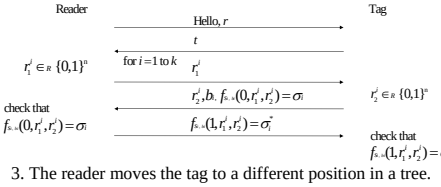
Steps of the algorithm

1. Each tag generates a random number, and the reader performs a tree-walk on these numbers

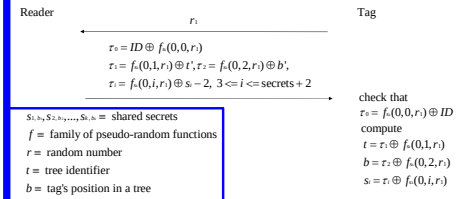
```
Traverse(i, count)
    b_i := Read random bit i
    if collision on b_i detected:
        Suspend all tags with b_i == 1
        Each suspended tag stores i
        Traverse(i+1, 0)
        Wake up tags suspended on bit i
        Traverse(i+1, 0)
    else if no collision on b_i detected:
        if(count > threshold)
            Proceed to step 2 with r = b_1,...,b_i
            Tree-Walk remaining tags
        else Traverse(i+1, count+1)
```

2. Once a tag is selected, the reader and the tag engage in a tree-waking private authentication protocol

$s_{1,0}, s_{2,0}, \ldots, s_{k,0} \in \{0,1\}^n$



3. The reader moves the tag to a different position in a tree.



$s_{1,0}, s_{2,0}, \ldots, s_{k,0}$ = shared secrets
f = family of pseudo-random functions
r = random number
t = tree identifier
b = tag's position in a tree

Properties

- Allows tags addition and removal from the system
- Provides security against active eavesdroppers
- Offers security against active readers
- Enables dynamic tradeoff between security, privacy, and singulation time
- Effective against active attacks:
  - stealing a tag
  - tracking and hotlisting

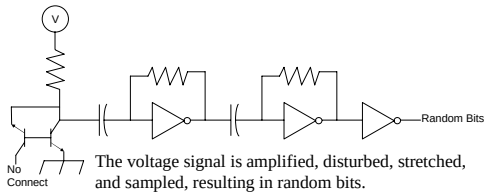
Time and Space Complexity

n is the total number of tags in the system

$O(n) \rightarrow O(\log n) \rightarrow O(depth_{tree}) \rightarrow o(depth_{tree}) \rightarrow O(1)$

→ : represents related work improvement
→ : represents our improvement as shown
→ : represents our improvement with some modifications

Random Number Generation Hardware



The voltage signal is amplified, disturbed, stretched, and sampled, resulting in random bits.

Future Work

- **Field testing of Multi-tags**
- **Identifying new applications of Multi-tags**
- **Improving hardware complexity of the algorithm**
- **Developing new efficient authentication algorithms**