Performance evaluation of the TIANC protocol

Elvis Angelaccio

Sapienza University of Rome

2nd December 2015

Based on *Inducing Collisions for Fast RFID Tag Identification* by Gianluca Carroccia and Gaia Maselli

Introduction

Recap: RFID systems

- Goal: identification of RFID tags
- One reader queries many tags
- Tags are passive: powered by the reader, no collision detection
- Reader arbitrates the channel using some MAC protocol
- Traditional protocols try to minimize collisions
    - Anti-collision protocols

Recap: Analog Network Coding (ANC)

- Since we can't really avoid collisions, let's exploit them!
- By using ANC we can recover the original transmission, even if there was a collision
- Basic assumption:
  - Reader has multiple antennas
  - Each antenna receives a signal with some attenuation coefficients
  - Reader can create and solve a system of equations, by combining all received signals.

Description of the TIANC protocol

TIANC: Tag Identification through ANC

- Reader has a number $a > 1$ of antennas:
  - Only a single antenna queries the tags
  - All the antennas receive tags tranmissions
  - Their interrogation zones overlap

TIANC: Tag Identification through ANC

- Reader has a number $a > 1$ of antennas:
  - Only a single antenna queries the tags
  - All the antennas receive tags tranmissions
  - Their interrogation zones overlap
- Number n of tags is supposed to be known

TIANC: Tag Identification through ANC

- Reader has a number $a > 1$ of antennas:
    - Only a single antenna queries the tags
    - All the antennas receive tags tranmissions
    - Their interrogation zones overlap
- Number n of tags is supposed to be known
- If there are $k \leq a$ collisions:

$$\begin{cases} a_1 = b_{1,1} \cdot s_1 + b_{1,2} \cdot s_2 + \cdots + b_{1,k} \cdot s_k \\ \quad \vdots \\ a_a = b_{a,1} \cdot s_1 + b_{a,2} \cdot s_2 + \cdots + b_{a,k} \cdot s_k \end{cases}$$

- The reader can recover the signals $s_1, \ldots, s_k$ and demodulate them into the k tags ID.

TIANC frames

- TIANC is based on Tree Slotted Aloha (TSA)
- Each frame is divided into *Training* and *Identification* subframes
- Tags transmit in both subframes

Training subframe

Two goals:

- Goal 1: *estimate channel coefficients*
  - Tags send one bit with content 1 in a random slot
  - Reader can estimate and store the attenuation coefficients

Training subframe

Two goals:

- Goal 1: *estimate channel coefficients*
  - Tags send one bit with content 1 in a random slot
  - Reader can estimate and store the attenuation coefficients
- Goal 2: *group tags into colliding sets*

Training subframe

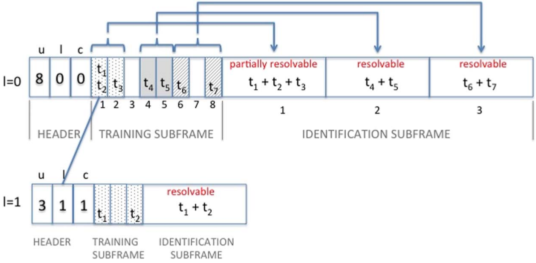
Two goals:

- Goal 1: *estimate channel coefficients*
  - Tags send one bit with content 1 in a random slot
  - Reader can estimate and store the attenuation coefficients
- Goal 2: *group tags into colliding sets*
- At the end of each slot, the reader replies with another bit:
  - ACK = singleton or collision slot (reader cannot distinguish!)
  - NACK = idle slot

Identification subframe

- Tags in the same group transmit in the same identification slot
- There are only colliding slots! Either *resolvable* or *unresolvable*
- Resolvable slots:
  - Exactly a tags transmit
  - Reader can solve a system of a equations and recover all the a IDs
- Unresolvable (or partially resolvable) slots
  - More than a tags transmit
  - Reader cannot decode all the ID
- For each unresolvable slot, a new child frame has to be issued by the reader

Example of execution ($a = 2$)



- Idea: map $a = 2$ consecutive non-idle slots in the training subframe into the same slot of the identification subframe
- Mapping from slot i to slot j:

$$j = \left\lceil \frac{i - s_{idle}}{a} \right\rceil$$

Performance Evaluation

Services offered by the system

- Service = retrieve the 96 bit ID of all the tags within the reader interrogation zone
- Possible outcomes:
    - Zero ID retrieved, if there are no tags
    - All IDs retrieved otherwise (assuming no hardware faults)

System-specific performance metrics

- Latency: How much time it takes to identify all tags?
  - A.k.a. responsiveness or response time
- System Efficiency: How much time is spent in identification
  - Not covered by the article

General Metrics

- Speed metrics:
    - Responsiveness or response time ⇒ Latency
    - Throughput: not covered
    - Utilization: not really the case, as there is only one service
- Reliability metrics
    - The system is not supposed to execute the service incorrectly
- Availabilty metrics:
    - The system is not supposed to be unavailable

Latency Evaluation

- Time = duration of training and identification slots
- EPCglobal standard: to transmit one bit it takes:
  - 25 μs (40 kbps bit rate) + physical overhead ($T_{phy} = 275.06\ \mu s$)
- In general, to compute the time of a slot s:

$$T_s = T_{phy} + T(\#\text{bits in } s)$$

- Example: trainings slots, only 2 bits:

$$T_{tr} = T_{phy} + T_{2bits} = 325.06\ \mu s$$

How many slots?

- Goal: we want to compute the time to identify n tags:

$$T(n) = H(n) \cdot T_H + X_{tr}(n) \cdot T_{tr} + X_{id}(n) \cdot T_{id}$$

- where:
  - $H(n) = \#$ frame headers
  - $X_{tr}(n) = \#$ training slots
  - $X_{id}(n) = \#$ identification slots
- Analytic method is used:
  - Due to e.g. time and cost criterions
  - Simplifying assumptions, e.g. error-free channel estimation

Probability of collisions

- Let $u = \alpha \cdot n$ be the number of slots in a frame ($\alpha > 0$)
- Tags choose slots randomly and independently
- Probabability of collision given by binomial distribution:

$$p(u, n, k) = \binom{n}{k} \left(\frac{1}{u}\right)^{k} \left(1 - \frac{1}{u}\right)^{n-k}$$

How many training slots?

- Reader recursively allocates a new frame for each collision of $k \geq 2$ tags:

$$X_{tr}(n) = \begin{cases} 1 & \text{if } n = 1 \\ u + \sum_{k=2}^{n} u \cdot p(u, n, k) \cdot X_{tr}(k) & \text{if } n > 1 \end{cases}$$

- $X_{tr}(k)$ is the # of training slots in a child subframe
- How many children? One for each collision!
- $u \cdot p(u, n, k)$ is the expected # of slots when k tags collide

How many frame headers?

- Each frame requires an header:

$$H(n) = \begin{cases} 1 & \text{if } n = 1 \\ 1 + \sum_{k=2}^{n} u \cdot p(u, n, k) \cdot H(k) & \text{if } n > 1 \end{cases}$$

How many identification slots?

- Size of identification subframe depends on the outcome of the previous training subframe:

$$X_{tr}(n) = \begin{cases} 1 & \text{if } n = 1 \\ v(u) + \sum_{k=2}^{n} u \cdot p(u, n, k) \cdot X_{id}(k) & \text{if } n > 1 \end{cases}$$

- $u \cdot p(u, n, 0)$ is the # of idle slots
- $v(u) = \frac{u - u \cdot p(u,n,0)}{a}$ is the # of identification slots in the first subframe

Test workload

- Workload = A set of n tags to be identified
- Workload is synthetic
- The value of n does not impact on protocol performance

Protocol performance

- Depends on the outcome of training slots
- The more collisions in training subframes, the more child frames
- The bigger the training subframe, the fewer the collisions

Results of the analysis (1)

Goal:

Find the optimal value of α such that the training frame size $u = \alpha \cdot n$ leads to the best latency

Results of the analysis (2)

$u = \alpha \cdot n$	a
$2.3n$	2
$2.1n$	3
$2.0n$	4
$2.0n$	5

Table 1: Optimal training frame size u for each number of antennas

Results of the analysis (3)

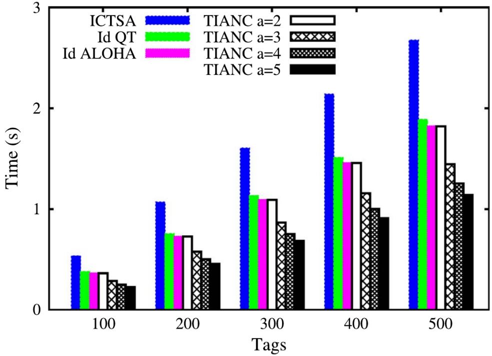


Figure 1: Comparison with ICTSA and ideal protocols (1 slot/query per tag)

Results of the analysis (4)

Speedup	a
31%	2
46%	3
53%	4
57%	5

Table 2: Speedup over ICTSA protocol

What about Time System Efficiency?

- Recap: $TSE = \frac{n}{X(n) - Y(n) + \beta \cdot Y(n)}$
  - $X(n) = \#$ of slots in TSA tree
  - $Y(n) = \#$ of idle slots in TSA tree
  - idle slots last a β fraction of non-idle slots
- TSE should be defined differently for training and identification slots
  - Identification slots cannot be idle!
- With training slots:
  - Similar to the TSA one, but...
  - β not so small, since tags transmit only 1 bit!

Conclusions

- TIANC features innovative elements such as its training mechanism
- Performance analysis shows significant speed improvements over previous work
- Performance not affected by large number of tags

Thanks for your attention!
Questions?