TagTone: Scalable RFID Communication through Multi-Frequency Analysis

Tanmoy Das, Gopi Krishna Tummala, and Prasun Sinha
The Ohio State University
{das.167, tummala.10, sinha.43}@osu.edu

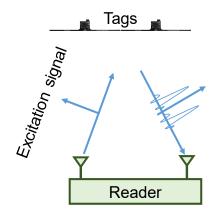
***Abstract*—RFID tags are inexpensive wireless sensors that harvest energy from the excitation signal sent by a reader. These tags are widely used in industrial settings for several applications. These applications require large-scale dense deployment where packet collisions are unavoidable. The major hurdle limiting these applications from scaling is the lack of suitable protocols that can decode collided packets from a large number of tags. We present TagTone, a new protocol that utilizes the wide bandwidth for RFID communication and the independence of the channels to decode colliding packets from a large number of tags. TagTone is a scalable protocol that can decode more tags as the bandwidth or the number of antennas increases making it an ideal solution for dense deployment. In this paper, we present a thorough analysis to show that RFID channels are independent across frequencies and antennas. Packets from the tags are collected using an excitation signal comprising of multiple complex sinusoids (tones) at different frequencies. TagTone includes a novel Moving Window Packet Separation (`MWPS`) algorithm which modifies traditional Independent Component Analysis (ICA) for superior decoding. TagTone implemented with USRP N210s can decode $7\times$ more colliding tags and provides $7\times$ better throughput than the existing protocols.**

I. Introduction

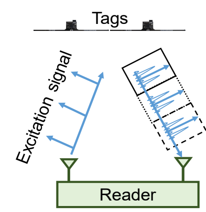
Radio-Frequency Identification (RFID) tags are inexpensive and batteryless wireless sensors that harvest energy from the excitation signal generated by a RFID tag reader. After the tags harvest sufficient amount of energy, they transmit data by reflecting the same excitation signal. These tags are widely deployed to support the needs of several industrial applications like inventory management [6], [7], supply chain management [2], [4], object tracking [12], [19], and robotic navigation [17], [9]. Due to low power design, RFIDs cannot communicate with each other and hence multiple tags transmit their data at the same time, resulting in packet collisions, especially in dense settings. The capability of decoding the collided packets would facilitate the development of new applications suitable for dense deployment.

Some existing protocols [18], [10], [15] fail to decode colliding packets when transmissions from the tags are affected by hardware heterogeneity in the tags. Hardware heterogeneity is the result of manufacturing inaccuracies. To overcome this difficulty, I-Q clustering based protocol [14] was proposed. The number of I-Q clusters increases exponentially with the number of tags and all the clusters must be present in the received data for correct decoding. But packets from the tags contain a small number of bits (few hundreds) due to the low power constraint. So, the number of clusters present in the received data is very small. For a large number of colliding tags, most of the clusters are not present in the received data. Due to this, the I-Q clustering method fails when the number of colliding tags is large.

This work was partially supported by NSF Grant #CNS-1302620.



(a) Current protocols do not use the whole bandwidth because the excitation signal contains a single tone.



(b) TagTone utilizes the whole bandwidth to decode a large number of tags.

Fig. 1: The difference between existing protocols and TagTone.

Unlike existing protocols, TagTone leverages the large bandwidth (tens of MHz) allocated for RFID communication. Tags use much smaller bandwidth (hundreds of KHz) due to power constraints. By utilizing the large bandwidth, TagTone decodes data from a large number of tags. Existing schemes cannot use the large bandwidth because the effect of hardware heterogeneity and the number of I-Q clusters do not change with the bandwidth.

Current schemes use a single tone as an excitation signal. Tags reflect the excitation signal to send their data. The data from the tags is centered around this tone (Figure 1(a)) and rest of the bandwidth remains unused. In contrast, TagTone uses an excitation signal which is the summation of tones at multiple frequencies to utilize the large bandwidth (Figure 1(b)).

Use of multiple tones divides the total transmit power among the frequencies. The received signal power depends on the transmit power. So, the received signal power reduces at a single frequency. But the data remains the same across all the frequency. By combining the data from multiple frequencies, the decoding performance is improved. To combine data from multiple frequencies, TagTone needs to estimate the channels, which include phase and attenuation of the signal, of the tags at multiple frequencies.

Channel estimation is challenging because the tags cannot communicate with each other and the number of tags present is not known beforehand. TagTone exploits two properties of RFID communication for channel estimation. *1) Independence*

978-1-5386-5979-3/18/$31.00 ©2018 IEEE

of the channels: Our observation (§ VI-E) suggests that the channels of the RFID tags are independent across frequencies and antennas; and *2) Independence of the samples:* The samples received from different tags are independent due to hardware heterogeneity which has two effects: *a) Random starting delay*: RFIDs harvest energy from the excitation signal and store it in a capacitor. A packet is transmitted when the capacitor is fully charged. The variability in capacitor charging time causes a random initial delay at the start of packet transmission [10]. *b) Variable bit duration*: RFIDs use a low energy RC-circuit based clock with high clock drift ($\pm 22\%$ [16], [14]). The bit duration of a tag is variable due to this clock drift.

TagTone estimates the channels that are used to separate the colliding packets. When the available bandwidth is W, K is the number of receive antennas and a tag requires bandwidth of B for communication, TagTone can estimate channels from $\frac{KW}{B}$ tags (§ V-A). This channel information is used to decode data from $\frac{KW}{B}$ tags.

TagTone leverages an existing technique called Independent Component Analysis (ICA) (§ IV-B) for channel estimation and packet separation.

Challenges with vanilla ICA: TagTone faces a significant challenge in employing ICA which requires good statistical independence among the samples from different tags. Even though samples from different tags are independent due to hardware heterogeneity, the independence is not uniform across all the received samples. For some samples, the independence can be poor due to variable size and misalignment of the packets. Hence, direct application of ICA on the whole received data fails.

Moving Window Packet Separation (`MWPS`) algorithm: `MWPS` algorithm chooses an appropriate subset (window) of samples where the statistical independence is high and applies ICA on those samples. However, finding an appropriate window size is a nontrivial problem. If a small window is chosen, only a fragment of each packet will be decoded, leading to an exponential number of combinations of decoded packets. On the other hand, a large window contains some portions that have poor statistical independence.

Contributions: The following are contributions by TagTone:

1) A thorough study shows that RFID channels are independent across frequencies and antennas.

2) TagTone is the first scalable protocol that can decode more tags as the bandwidth and the number of receive antennas increases making it an ideal solution for dense deployment of RFIDs.

3) TagTone decodes $7\times$ more colliding tags and provides $7\times$ better throughput than the existing protocols in experiments with USRPs and WISP 5 tags.

II. Related Work

Existing protocols to read RFID tags can be divided into two categories: 1) Collision avoiding MAC protocols [3] require a large amount of time (grows exponentially with the number of tags [8]) to read multiple RFID tags; and 2) Collision embracing protocols can decode colliding packets and suitable for dense deployment. But their performance is adversely affected by hardware heterogeneity. These protocols can be divided into three groups: **a) No starting delay and clock drift:** Buzz [18] assumes there is no clock drift or starting delay because it requires symbol level synchronization. Buzz first estimates the channels from the tags which requires a large amount of time as the tags cannot communicate with each other and the number of tags is not known. In addition, a tag needs to transmit a packet multiple times before the reader can decode it which consumes a significant amount of time; **b) No clock drift:** LFB [10] and BST [15] assume that there is no clock drift and decode the colliding packets by leveraging the random starting delay of the tags. To mitigate the effect of clock drifts, LFB uses an external high accuracy clock [5] that itself consumes 10μW of power which is much larger than the power consumption of the whole RFID circuitry (1.05μW) [16]. Commercial RFID tags use low accuracy clocks to reduce power consumption; and **c) With clock drift and staring delay:** BiGroup [14] is a protocol that can handle both starting delay and clock drift. It relies on an I-Q clustering method to decode data from the tags. Due to low SNR and a large number of clusters (for C tags there are 2^C clusters), the inter-cluster distance becomes very small and it is difficult to distinguish different clusters. Clustering errors reduce the performance when the number of tags is large (> 4).

III. Overview

TagTone uses summation of multiple uniformly separated tones as an *excitation signal* that is reflected by the RFID tags. The received signal is processed using an *FFT based filtering* module to separate out the signals at multiple frequencies. These filtered signals are processed by the `MWPS` algorithm to isolate the samples of the tags. These samples are further analyzed by the *packet decoding module* to decode data from different tags. The key contribution of the paper is the `MWPS` algorithm that separates colliding packets from the tags by processing the filtered signals. In the next section, we present the `MWPS` and rest of the modules are presented in the succeeding sections.

IV. `MWPS` Algorithm

In this section, we describe the ICA technique, the problems of using vanilla ICA on the received samples, and the design of `MWPS` algorithm.

A. Problem formulation

After FFT based filtering, the received data contains samples collected at F frequencies. The samples are complex numbers that have real and imaginary parts. These two parts of the samples are collected using two independent modules (known as I and Q). As a result, the channels for the real and imaginary parts are totally independent [14], [10]. So, F frequencies are converted into $2F$ independent subfrequencies. The received signals from $2F$ subfrequencies are modeled as $\mathbf{Y} = \mathbf{A}\mathbf{X}+\mathbf{N}$.

Here $\mathbf{Y}$ is a matrix ($2F \times S$) containing the received samples from $2F$ subfrequencies. S is the number of received samples in each subfrequency. The channel matrix is represented by $\mathbf{A}$ ($2F \times P$) and $\mathbf{X}$ is the matrix ($P \times S$) where j^{th} row contains the samples from j^{th} tag. $\mathbf{N}$ ($2F \times S$) is the noise matrix. Here P is the number of colliding tags.

B. Traditional ICA

ICA is suitable for solving abovementioned problem because the samples generated by the tags are independent of each other due to hardware heterogeneity and channels of different tags across different subfrequencies are independent [1]. ICA estimates a $P \times 2F$ matrix, $\mathbf{W}$ that is the inverse of the channel matrix $\mathbf{A}$ so $\mathbf{A} = \mathbf{W}^{-1}$. ICA can estimate channels from as many as $2F$ tags [1]. ICA estimates the number of colliding packets using Principal Component Analysis (PCA).

C. Problems with traditional ICA

Direct application of ICA on the received samples faces a significant problem. ICA requires high statistical independence among the samples from different tags. But due to packet misalignment (caused by random starting delay and variable packet size), when some tags are sending data other tags might have stopped their transmissions (effectively they are sending samples of value zero). When multiple tags send the same samples for a long period of time the statistical independence among the samples reduces. Due to this, the statistical independence can be very low in some portion of the received data. As a result, ICA fails to separate out the samples when it is applied to the whole received data.

D. MWPS algorithm

MWPS carefully chooses a window to avoid those parts that have poor statistical independence as much as possible. But it is difficult to find such a window without knowing when the transmission of a packet starts and stops from multiple tags. Determining the start and end of a packet is non-trivial when multiple packets are colliding. One option is to use a small fixed size window. In a small window, the probability that all the tags are sending data is very high. As a result, it has good statistical independence. But a small window contains a portion of the collided packets and it is difficult to construct a whole packet from those smaller portions. To construct a whole packet, we need to consider all possible combinations of those small portions across all the windows. The number of combinations grows exponentially with the number of windows used to process the received samples. In contrast, in a large window, the statistical independence among the samples could be very low in some parts.

1) Design of MWPS: MWPS dynamically controls the window size. It starts with an initial window size and if no packet is correctly decoded, the window size is increased to ensure that current window includes at least one packet. If any packet is correctly decoded then it is subtracted out from the received ensemble and the window is moved to next data sample. To check the validity of a decoded packet CRC is used. There are three major components in MWPS that are described below.

1) Controlling the window size: MWPS uses a window that is large enough to include at least one packet so that we can avoid the combination of smaller portions across multiple windows. MWPS starts with an initial window size (line 5). If the initial window does not contain a whole packet, the decoding fails. When no packet is decoded from the current window, the window size is increased by a fixed amount (line 23). A fixed increment is used because it is not known when packet transmissions from different tags end.

Tags can transmit data at different rates and the number of bits in a packet can be different. For maximum data rate D and packet length of b bits, the number of samples within a packet is $\frac{b}{DT}$ without clock drift. Here T is the sampling interval. FFT based filtering reduces the number of samples by a factor of F. We heuristically use $\frac{b}{DTF}$ as the initial window.

2) Subtracting the decoded packet from received samples: When a packet is successfully decoded it is subtracted out from the received samples to reduce the interference for the undecoded packets (line 13). But caution must be exercised at the time of subtraction. As the estimated samples are corrupted by noise, simply subtracting them from the received samples can increase the noise power which makes the decoding difficult for the undecoded packets. We need to decrease the noise in the estimated samples before subtracting them. A median filter is used to reduce the noise in the estimated samples. This is the best choice because, within a bit duration, all the samples have the same value and median filter can easily cancel out the outliers (caused by noise) from these samples. In addition, the estimated channel values (§ IV-B) are used for near perfect reconstruction of the received samples.

3) Moving the window: We also need to move the window to ensure that it includes relevant samples. After a packet is decoded and subtracted from the received samples, the first few samples in the current window may not contain any meaningful data but noise. We need to move the window to exclude these samples. The next window should start from the first sample of the next undecoded packet. To detect that sample, the noise power and the power of the samples inside the window are compared and the first sample that has more power than the noise is chosen as the start of the window (lines 18-21).

V. Detailed System Description

In this section, we provide a detailed description of the rest of our system including *excitation signal*, *FFT based filtering*, and *packet decoding*.

A. Excitation Signal

The reader sends an excitation signal which is the sum of tones at F equispaced frequencies. This signal is reflected by a tag. The tags operate at a low data-rate due to power constraint, therefore they require small bandwidth (B). During the reflections, the tag multiplies its own signal with the excitation signal [11]. This multiplication creates two mirror images [11] of bandwidth B around every frequency as shown

Algorithm 1: MWPS algorithm.

Input: **(a)** received samples, $\mathbf{Y}_{2F\times S}$; **(b)** initial window size, w_init; **(c)** window increment size, w_incr; **(d)** the noise power at the RFID reader, P_N.
Output: **(a)** number of decoded tags, n; **(b)** decoded packets from the tags, $\mathbf{M}_{n\times b}$. Packets are at most b bits long.

$n \leftarrow 0$
$s \leftarrow 0$
$w \leftarrow w\_init$
while $s + w \leq S$ **do**
    // Process s^{th} to $(s+w)^{th}$ columns.
    $\mathbf{W}_{P\times 2F} \leftarrow ICA(\mathbf{Y}_{2F\times(s:s+w)})$
    $\hat{\mathbf{X}}_{P\times w} \leftarrow \mathbf{W}_{P\times 2F}\mathbf{Y}_{2F\times(s:s+w)}$
    $decoded \leftarrow false$
    for $i \leftarrow 1$ *to* P **do**
        // process each ICA output § V-C
        $msg \leftarrow process\_output(i^{th}$ *row of* $\hat{\mathbf{X}}_{P\times w})$
        if *packet is correctly decoded* **then**
            subtract the packet from received samples
            $decoded \leftarrow true$
            $n \leftarrow n + 1$
            store the packet in $\mathbf{M}$
    // Packets are decoded so move window
    if $decoded = true$ **then**
        // Find sample with more power than noise
        for $i \leftarrow s$ *to* $s + w$ **do**
            if *mean power of* i^{th} *column of* $\mathbf{Y}_{2F\times S} > P_N$ **then**
                $break$
        $s \leftarrow i$
    // Decoding failed, increase window size
    else
        $w \leftarrow w + w\_incr$
return n and $\mathbf{M}_{n\times b}$

in Figure 1(b). Due to this, the separation between two adjacent frequencies must be at least $2B$ so that there is no interference between the mirror images at two frequencies. We can estimate channels from more number of tags as the number of frequencies in the excitation signal increases.

Maximum number of frequency that can be used is $\frac{W}{2B}$ (W is available bandwidth). A frequency is divided into two independent subfrequencies. The number of independent subfrequencies is $\frac{W}{B}$. The channels are independent across the receive antennas. Hence, for K receive antennas the number of independent subfrequencies is $\frac{KW}{B}$ and ICA can estimate channels from $\frac{KW}{B}$ colliding tags.

B. FFT based filtering

We have adopted FFT-based filtering technique because of the following reasons. *1) Equal separation of frequencies*: the minimum separation between two adjacent frequencies should be $2B$. We use F equispaced frequencies. The uniform separation makes the signals at different frequencies orthogonal to each other. So, an FFT can be used to separate out the signals at different frequencies; *2) No clock offset*: clock offset between transmitter and receiver destroys the orthogonality between the frequencies. But the transmitter and the receiver of the signal are in fact the same node using the same clock. So, there is no clock offset; and *3) Computation efficiency*: FFT based filtering is considerably faster than existing FIR and IIR based filters [13] because a single FFT operation is good enough to separate out the signals from F frequencies but F different IIR or FIR filters are required for the separation of the signals. The runtime complexity of FFT based filtering for S samples is $O(S\log F)$ [13]. But the runtime complexity for FIR and IIR based filterings are $O(FS\log S)$ [13] and $O(FSL)$ (L is the length of the IIR filter) [13], respectively. FFT based filtering is $O(F\frac{\log S}{\log F})$ times faster than FIR based filtering and $O(\frac{FL}{\log F})$ times faster than IIR based filtering.

After FFT based filtering, the total signal power remains the same according to Parseval's theorem. The signal power is distributed across multiple frequencies which are combined during the packet separation.

C. Packet decoding

The output of ICA has ambiguities [1] that could flip (multiplied by -1) the sign of the estimated samples. But RFID tags transmit data by changing the power of the reflected signal i.e., high power is the bit '1' and low power is the bit '0'. The power of a sample is measured by the squared of its amplitude. If the sign of a sample is flipped, the power of the sample remains same. The data packet can be correctly decoded if all the transitions (change from high to low power or low to high power) are correctly detected. The positions of all the transitions in the output samples are determined by comparing the power of consecutive samples. But due to noise, there could be some erroneous transitions.

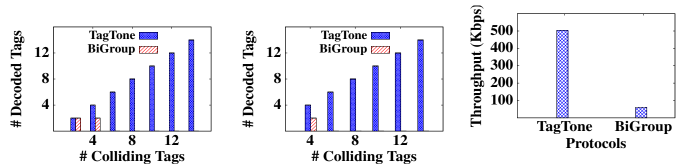
The bit duration is used to filter out the erroneous transitions. But bit durations of the tags are different due to clock drift. To correctly estimate the bit duration, a preamble of three consecutive '1's is added before every packet. The average bit duration of first three bits is used as an estimate for the bit duration.

VI. Experiments

Prototype: A RFID reader is built using USRP-N210 software defined radio platform and ALR-8696-C antennas. The reader uses TagTone to decode packets. We use programmable Wireless Identification and Sensing Platform 5 (WISP 5) tags. The excitation signal is the summation of multiple tones at different frequencies. The separation between two adjacent frequencies is 2MHz. The packets from the tags are 200 bits long including 16 bits CRC. The tags transmit their packets as soon as they have harvested enough energy for transmission. We have compared the performance of TagTone with BiGroup because it is the only protocol that can handle both starting delays and clock drifts in the colliding packets.

A. Decoding of colliding tags

TagTone: Performance of TagTone to decode colliding tags is evaluated in this experiment. There are 30 tags placed in front of the antennas. In Figure 2(a), all the tags transmit using the datarate of 100Kbps. In Figure 2(b), the colliding tags use two different datarates 50Kbps and 100Kbps. The excitation signal contains 10 frequencies. The separation between two adjacent frequencies is 2MHz so total bandwidth is 20MHz. TagTone can decode all the colliding tags most of the time.

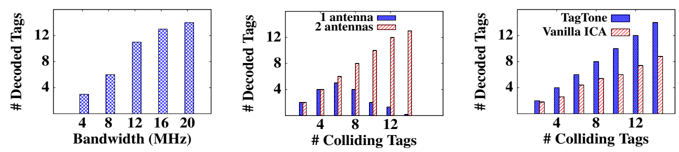


(a) Homogeneous data rates.

(b) Heterogeneous data rates.

(c) Throughput for TagTone and BiGroup.

Fig. 2: TagTone vs BiGroup



(a) # decode tags vs bandwidth.

(b) # decode tags vs # receive antennas. 8MHz bandwidth.

(c) MWPS vs vanilla ICA.

Fig. 3: Performance of TagTone and MWPS.

As the number of frequencies is 10, so TagTone can decode at most 20 colliding tags as explained in § V-A.

BiGroup: Performance of BiGroup is also presented in Figure 2(a). BiGroup can successfully decode data from 2 colliding tags. BiGroup uses I-Q clustering to decode data from the tags. But due to low SNR and large number of clusters (C colliding tags create 2^C clusters), the inter-cluster distance reduces and it becomes difficult to distinguish different clusters. Errors in clustering cause a large number of bit errors and reduce the performance.

The average throughput for TagTone and BiGroup across 300 iterations are presented in Figure 2(c). The throughput of TagTone is $\approx 7\times$ better than BiGroup. BiGroup cannot decode any packet when the number of colliding tags is more than 4, so the BiGroup results are omitted for the rest of this section.

B. Increasing the bandwidth

The bandwidth of the excitation signal is varied by changing the number of frequencies. Bandwidth is varied from 4MHz (for 2 frequencies) to 20MHz (for 10 frequencies). 30 tags are placed in front of the antennas. In Figure 3(a), the maximum number of decoded tags is shown for each setting. As the bandwidth increases, the maximum number of decoded tags increases almost linearly which means TagTone can decode more number of tags as the bandwidth increases. As the bandwidth increases the number of frequencies also increases which improve the performance.

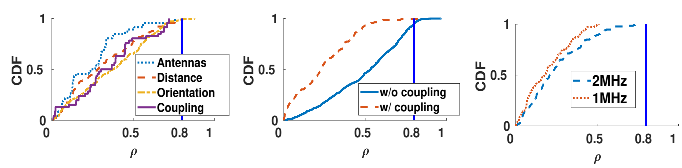
C. Using multiple receive antennas

The performance of TagTone as the number of receive antennas increases is shown in Figure 3(b). The excitation signal contains 4 frequencies so the total bandwidth used is 8MHz. For a single receive antenna, channel from at most 8 colliding tags can be estimated and decoded (§ V-A) and beyond 8 tags the decoding performance degrades sharply. For 2 receive antennas, it is possible to decode 14 colliding tags. In this experiment, the total bandwidth remains same but TagTone can decode more number of tags by increasing the number of receive antennas because channels are independent across antennas.

D. Detailed performance analysis of MWPS

The performance of MWPS and vanilla ICA are shown in Figure 3(c). Due to the misalignment of the packets, the statistical independence in some portion of the received samples can be very low. As a result, vanilla ICA performs poorly when directly applied on the whole received samples. But MWPS uses a moving and adaptable window to select a subset of samples that have high statistical independence and uses ICA on the samples contained within the window. Hence, it provides better performance $(1.5\times)$ than vanilla ICA.

E. Understanding channel independence



(a) ρ for same frequency but different tags.

(b) ρ for same tag but different frequencies.

(c) ρ for dense deployment.

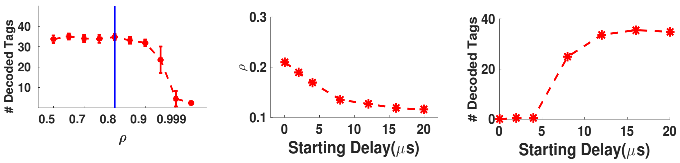
Fig. 4: Absolute correlation coefficient (ρ) of RFID channels.

The results presented in this section show that the channels from the RFID tags are: 1) independent across frequencies for a tag; and 2) independent across different tags for the same frequency in a dense deployment. The independence of the channels is measured by the absolute correlation coefficient (ρ). According to the results presented in Figure 5(a), *the decoding method works very well as long as ρ between channels is smaller than 0.8.*

The channel values of the tags are measured from the packets that are sent by the tags. The excitation signal contains 12 frequencies. The distance between the tags and antennas is $18''$ (inches).

1. Channels of different tags at same frequency: The ρ values for the channels across different tags for the same frequency are depicted in Figure 4(a). Different parameters are changed and ρ values are plotted for each one of them. *a) The orientation of a tag:* A single tag is used. The tag and the antennas are parallel to each other. The orientation of the tag is changed by rotating it in a circle. ρ values are less than 0.8 for $\approx 98\%$ of the time; *b) Distance:* A tag is placed in different positions that are at least $4''$ inches apart. ρ values are less than 0.8 for $\approx 97\%$ of the time; *c) Different receive antennas:* Two different receive antennas are used to measure the independence of the channels from a single tag. ρ values are less than 0.8 all the time; and *d) Mutual coupling of antennas of the tags:* The independence of the channels due to the coupling of antennas of the tags is measured by using two tags that are less than $4''$ apart. ρ values are less than 0.8 for $\approx 99\%$ of the time.

2. Channels of different frequencies at the same tag: The



(a) # decoded tags vs ρ among the channels.

(b) ρ among the samples from different tags.

(c) # decoded tags vs starting delay.

Fig. 5: # decoded tags vs channel and sample independence.

ρ values for the channels across different frequencies for the same tag are depicted in Figure 4(b). ρ values do not change significantly for orientation, distance or antennas. But ρ values change notably due to coupling. Without coupling, ρ is less than 0.8 for $\approx 94\%$ of the time while with coupling ρ is less than 0.8 all the time.

3. Channels of multiple tags in dense deployment: To analyze the combined effect of all the parameters, multiple tags are used. The tags are placed within a rectangular box of $18'' \times 12''$. The separation between two adjacent frequencies is varied. Two different values (1MHz and 2MHz) are used. In Figure 4(c), ρ values are less than 0.8 all the time.

This observation suggests that RFID channels satisfy the requirement of statistical independence which is necessary for our decoding process.

VII. Trace-Driven Simulations

In this section, the performance of TagTone is evaluated by varying certain parameters (hardware heterogeneity, channel independence, etc.) that are difficult to control in experiments. Following traces from the experiments are used: 1) channel values; 2) SNR; 3) starting delay; and 4) clock drift. 25 tones are used to collect data from 40 colliding tags and the data rate of the tags is set to 100Kbps.

1) Channel independence: The performance of TagTone for different channel independence is shown in Figure 5(a). The channels are modeled using a multivariate Gaussian random variable. When ρ is less than 0.8, the performance of TagTone remains stable but beyond 0.8 performance degrades sharply. High ρ values indicate low statistical independence. So, TagTone works perfectly even when the channel independence is low.

2) Starting delay: In Figure 5(b), the ρ value among the samples from different tags decreases as the starting delay increases which means the independence among the samples increases as the starting delay increases. The starting delay can be as large as $120\mu s$ [14]. The channel estimation accuracy improves as the statistical independence among the samples increases. As a result, the performance of TagTone improves (Figure 5(c)). Performance of TagTone stabilizes when ρ is less than 0.13. Low ρ values indicate high statistical independence which implies that TagTone requires high statistical independence among the samples. The same phenomenon is observed for clock drift but due to lack of space, those results are omitted.

VIII. Conclusion

In this paper, a novel solution is presented for RFID communication called TagTone. The proposed solution decode collided packets from a large number of RFID tags by leveraging the large bandwidth available for RFID communication. TagTone can decode $7\times$ and $20\times$ more tags than the existing protocols in experiments and simulations, respectively.

IX. Future Work

Currently, we compare the power of the samples with the noise to detect the presence of a packet. But the transmit power is divided among the frequencies causing a reduction in transmit power per frequency. Consequently, for the tags that are far away the received signal power can be smaller than the noise power and it is difficult to detect those packets. A new technique is required that can detect the signals from the tags that are far away. Developing such a technique and its evaluation are left as future work.

References

[1] Independent component analysis: algorithms and applications. *Neural Networks 13*, 45 (2000), 411 – 430.

[2] Amazon Announces Program With Auburn RFID Lab. http://www.rfidjournal.com/articles/view?13091, 2017.

[3] EPC Radio-Frequency Identity Protocols Generation-2 UHF RFID. https://www.gs1.org/epcrfid/epc-rfid-uhf-air-interface-protocol/2-0-1, 2017.

[4] Inditex to speed up supply chain management with RFID. http://www.rfidjournal.com/articles/view?7756, 2017.

[5] NX3225GD: 8MHz crystal clock. https://goo.gl/bC2KZe, 2017.

[6] River Island to Deploy RFID Across All 280 Stores. http://www.rfidjournal.com/articles/view?15196/1, 2017.

[7] Wal-Mart Takes a New Approach to RFID. http://www.rfidjournal.com/articles/view?7756, 2017.

[8] Davis, D., and Gronemeyer, S. Performance of Slotted ALOHA Random Access with Delay Capture and Randomized Time of Arrival. *IEEE Transactions on Communications 28*, 5 (1980), 703–710.

[9] Gueaieb, W., and Miah, M. S. An Intelligent Mobile Robot Navigation Technique Using RFID Technology. *IEEE Transactions on Instrumentation and Measurement* (2008).

[10] Hu, P., Zhang, P., and Ganesan, D. Laissez-Faire: Fully Asymmetric Backscatter Communication. In *Proc. of ACM SIGCOMM* (2015).

[11] Kellogg, B., Talla, V., Gollakota, S., and Smith, J. R. Passive Wi-Fi: Bringing Low Power to Wi-Fi Transmissions. In *NSDI* (Santa Clara, CA, 2016).

[12] L. Shangguan, Z. Li, Z. Y. M. L., and Liu, Y. OTrack:Order Tracking for Luggage in Mobile RFID Systems. In *IEEE INFOCOM* (2013).

[13] Oppenheim, A. V., and Schafer, R. W. *Discrete-Time Signal Processing*, 3rd ed. Prentice Hall Press, Upper Saddle River, NJ, USA, 2009.

[14] Ou, J., Li, M., and Zheng, Y. Come and Be Served: Parallel Decoding for COTS RFID Tags. In *Proc. of ACM MobiCom* (2015).

[15] P. Hu, P. Z., and Ganesan, D. Leveraging Interleaved Signal Edges for Concurrent Backscatter. In *Proc of ACM HotWireless* (2014).

[16] Pillai, V., et al. An Ultra-Low-Power Long Range Battery/Passive RFID Tag for UHF and Microwave Bands With a Current Consumption of 700 nA at 1.5 V. *IEEE Transactions on Circuits and Systems I: Regular Papers* (2007).

[17] Wang, J., Adib, F., Knepper, R., Katabi, D., and Rus, D. RF-compass: Robot Object Manipulation Using RFIDs. In *In Proc. of ACM MobiCom '13*.

[18] Wang, J., Hassanieh, H., Katabi, D., and Indyk, P. Efficient and Reliable Low-power Backscatter Networks. In *Proc. of ACM SIGCOMM* (2012).

[19] Wang, J., and Katabi, D. Dude, Where's My Card?: RFID Positioning That Works with Multipath and Non-line of Sight. In *Proc of ACM SIGCOMM* (2013).