Empowering Low-Power Wide Area Networks in Urban Settings

Rashad Eletreby
Carnegie Mellon University
Pittsburgh, PA 15213
reletreby@cmu.edu

Diana Zhang
Carnegie Mellon University
Pittsburgh, PA 15213
dianaz1@andrew.cmu.edu

Swarun Kumar
Carnegie Mellon University
Pittsburgh, PA 15213
swarun@cmu.edu

Osman Yağan
Carnegie Mellon University
Pittsburgh, PA 15213
oyagan@andrew.cmu.edu

ABSTRACT

Low-Power Wide Area Networks (LP-WANs) are an attractive emerging platform to connect the Internet-of-things. LP-WANs enable low-cost devices with a 10-year battery to communicate at few kbps to a base station, kilometers away. But deploying LP-WANs in large urban environments is challenging, given the sheer density of nodes that causes interference, coupled with attenuation from buildings that limits signal range. Yet, state-of-the-art techniques to address these limitations demand inordinate hardware complexity at the base stations or clients, increasing their size and cost.

This paper presents Choir, a system that overcomes challenges pertaining to density and range of urban LP-WANs despite the limited capabilities of base station and client hardware. First, Choir proposes a novel technique that aims to disentangle and decode large numbers of interfering transmissions at a simple, single-antenna LP-WAN base station. It does so, perhaps counter-intuitively, by taking the hardware imperfections of low-cost LP-WAN clients to its advantage. Second, Choir exploits the correlation of sensed data collected by LP-WAN nodes to collaboratively reach a far-away base station, even if individual clients are beyond its range. We implement and evaluate Choir on USRP N210 base stations serving a 10 square kilometer area surrounding Carnegie Mellon University campus. Our results reveal that Choir improves network throughput of commodity LP-WAN clients by 6.84 × and expands communication range by 2.65 ×.

KEYWORDS

Internet-of-things, Low power wide area networks, frequency offsets, LoRa, Chirp spread spectrum

ACM Reference format:
Rashad Eletreby, Diana Zhang, Swarun Kumar, and Osman Yağan. 2017. Empowering Low-Power Wide Area Networks in Urban Settings. In *Proceedings of ACM Sigcomm conference, Los Angeles, California USA, August 2017 (SIGCOMM'2017),* 13 pages.
https://doi.org/10.475/123_4

SIGCOMM'2017, August 2017, Los Angeles, California USA

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

1 INTRODUCTION

Recent years have witnessed Low-Power Wide Area Networks (LP-WANs) emerge as an attractive communication platform for the Internet of Things (IoT) [37]. LP-WANs enable low-power devices (milliwatts) to transmit at low data rates (kilobits per second) over long distances (several kilometers). LP-WANs are an ideal vehicle for cheap, low-power IoT devices such as sensors that have limited power budget (e.g. a ten-year lithium ion battery) but also send few kilobits per second of sensed data to the cloud. Consider future smart cities where a few LP-WAN towers gather sensor data from a large number of low-power devices in the city. Such devices can exploit this reliable infrastructure to communicate, no matter where they are placed, without ever being charged during their lifetimes. Several LP-WAN proposals have emerged in the past few years, including commercial technology for the unlicensed 900 MHz band (LoRaWAN [28], SigFox [31]) as well as open standards for reusing cellular infrastructure (LTE-M [25], NB-IoT [34]).

Yet, deploying city-scale LP-WAN networks is challenging for two reasons: the density of deployment and the nature of urban environments. First, the sheer density of deployment of LP-WAN nodes means that transmissions from a large number of radios will often collide. Such collisions adversely impact LP-WANs, draining battery life and wasting precious air time and spectrum in a dense network. Second, deployments in urban areas cause the already weak signals of low-power nodes to be further attenuated by buildings and other obstacles before reaching the base station. This greatly reduces the range of LP-WAN sensors from over 10 km in rural areas to 1-2 km or less in urban settings [2, 4, 41].

At the root of these challenges is the limited capability of LP-WAN hardware, both at the base station and clients. On one hand, the limited power budget and low cost of LP-WAN clients make it challenging to deploy sophisticated MAC and PHY-layer schemes to avoid collisions [12, 13]. On the other hand, LP-WAN base stations struggle to resolve a large number of such collisions. Indeed, state-of-the-art techniques such as uplink MU-MIMO [6, 26, 40] can at best separate as many sensor nodes as there are base station antennas (atmost 3-4 today due to limits on the size and cost [16, 27, 39]). As a result, there remains a fundamental disconnect between the vision of dense, city-wide LP-WANs and the capabilities of state-of-the-art LP-WAN hardware.

This paper aims to bridge this disconnect – it builds Choir, a solution to overcome the challenges of dense, city-scale LP-WANs

despite the limited capability of client sensor nodes and base stations. First, we resolve collided transmissions from dense deployments of LP-WAN clients at an LP-WAN base station, even if it is not MIMO-capable. We demonstrate how Choir improves the throughput, latency and battery life of LP-WAN nodes. Second, Choir improves the range of teams of low-power LP-WAN sensors in urban environments. We demonstrate how teams of co-located sensors can communicate together with an LP-WAN base station, even if they are individually beyond its reception range. Choir is fully implemented at the LP-WAN base station without requiring hardware modifications to LP-WAN sensors. We integrate and evaluate Choir with low-power embedded sensing hardware and demonstrate end-to-end performance in a large neighborhood surrounding Carnegie Mellon University (CMU) campus.

At the heart of our approach to disentangle collisions at the base station is a strategy that exploits hardware imperfections of low-cost components in LP-WAN radios. Specifically, the signals transmitted by such hardware produces offsets in time, frequency, and phase. Choir proposes algorithms that use these offsets to separate and decode collisions from users. It achieves this by leveraging properties of the physical layer of LoRaWAN LP-WAN radios that transmits signals in the form of chirps, i.e., signals whose frequency varies linearly in time. We show how hardware offsets, whether in time, frequency or phase manifest as distinct aggregate frequency shifts in chirps from each transmitter. We then filter the received signal using these shifts to separate signals from different transmitters. Choir then overcomes multiple challenges to decode useful data packets from each filtered signal component. First, it develops novel algorithms to separate bits of data from hardware offsets, both of which are embedded in frequency shifts of chirps. Second, it uses the precise values of the offsets of the separated signals to identify which bits of decoded data belongs to which client to reconstruct the packet over time. Given that Choir disentangles sensors in the time and frequency domain as opposed to the antenna domain (i.e., MIMO), it can be implemented on a single-antenna base station. As a result, Choir directly improves the throughput of dense urban LP-WANs by decoding transmissions from multiple nodes simultaneously with minimal coordination overhead. It further enhances both latency and battery life of LP-WAN clients by removing the need for retransmissions.

Beyond dealing with density, we show how hardware offsets between transmissions can boost the range of LP-WANs. Specifically, we consider transmissions from teams of LP-WAN sensors that are individually beyond the range of the base stations, but are physically co-located. Such sensors are likely to record similar readings resulting in overlapping values for the most-significant bits of sensed data. Choir devises a mechanism for such overlapping most-significant bits to be recovered to help obtain a coarse view of sensed data in a given area. We propose a simple modification of the LP-WAN PHY that allows overlapping chunks of bits collected by sensor nodes to be transmitted concurrently as overlapping chunks of signals that are received at higher aggregate power. Choir develops a novel algorithm to achieve this in software without requiring expensive hardware modifications at the LP-WAN clients to tightly synchronize their transmissions. We generalize our approach to build a system that provides a coarse-grained view

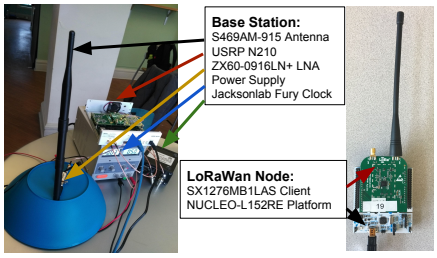


Figure 1: LP-WAN Setup: Depicts Choir's USRP N210 based LP-WAN base station and commodity LoRaWAN clients.

of sensors further away, while improving throughput and providing a fine-grained view of sensors near the base station. We further discuss how the concepts in this paper apply to emerging and future LP-WAN standards such as NB-IoT [34] and SIGFOX [31].

We implement Choir on a testbed of LoRaWAN LP-WAN radios. We deploy LP-WAN base stations (see Fig. 1) on the top floors of buildings covering a large area spanning 10 square kilometers around CMU campus. We emulate LP-WAN base stations using USRP N210 software radios. Our commodity LP-WAN client devices transmit measured temperature and humidity data, and are spread across CMU campus. We compare our system with a baseline that employs uplink MU-MIMO [38] as well as different modes of the standard LoRaWAN PHY and MAC [28]. Our results reveal the following:

- **Density:** For 30 nodes placed over 100 randomly chosen locations, with as many as 10 nodes transmitting data at any given time, Choir achieves a throughput gain of 6.84 × over standard LoRaWAN. It further achieves a 4.88 × reduction in latency and 4.54 × reduction in number of transmissions per decoded data packet.
- **Range:** For a team of up to 30 Choir nodes integrated with temperature sensors across four floors in a large building, we retrieve sensor data from distances as much as 2.65 km with loss of resolution of 13.2 %, despite the fact that each sensor can be heard individually no further than 1 km away (a gain of 2.65 ×).

Contributions: This paper presents Choir, a novel system that exploits the natural hardware offsets of LP-WAN clients to both disentangle and decode their collided transmissions using a single-antenna LP-WAN base station. Choir allows teams of LP-WAN sensor nodes transmitting correlated data to reach an LP-WAN base station, despite being individually beyond communication range. Our system is fully implemented and deployed on a large outdoor testbed spanning 10 square kilometers.

2 RELATED WORK

Low-Power Wide-Area Networks: Private enterprises such as LoRaWAN [28] and SigFox [37] have developed LP-WAN chips that use extremely narrow bands of unlicensed spectrum for diverse set of applications. 3GPP has also developed two LP-WAN standards for cellular base stations, namely, LTE-M [25] and NB-IOT [34]. Common to all LP-WAN technologies is the limited power budget and bandwidth, hardware simplicity and low cost of client nodes.

Multiple deployment efforts recognize the challenge of limited range of LP-WAN radios in urban environments [2, 8] as well as collisions in large-scale settings [2, 9, 11, 24].

LoRaWAN in particular uses chirp spread spectrum (CSS) for transmitting data due to its low power requirements, hardware simplicity, and performance under multipath and narrowband interference. However, we emphasize that LoRaWAN, like other CSS-based technologies [14, 42, 46] in radar or 802.15.4a divides chirps in time using TDMA, CSMA or Aloha and does not decode collisions [8], unlike say CDMA[1] [43]. This is because decoupling data from collisions of chirps in the presence of frequency and timing offsets is fundamentally challenging (we elaborate on these challenges and our solutions to overcome them in Sec. 4). Indeed, there has been much work on collision avoidance and MAC protocols [14, 46] for CSS for precisely this reason. However, our approach aims to directly leverage hardware offsets to decode CSS collided transmissions, as well as extend communication range.

Decoding Collisions in Wireless Networks: There has been much past work on decoding collisions in wireless networks, particularly for cellular networks [15, 47], RFIDs [44] and wireless LANs [7, 18, 40]. Much of this work relies on using multiple antennas on wireless nodes be it MU-MIMO on the downlink [6], uplink [40] or fully distributed MIMO [33]. Unfortunately, the maximum gain of these systems is limited by the number of antennas on the base station (at best 3-4 today due to limits on size and cost [16, 27, 39]). However, our system is able to separate collisions even with a single-antenna base station.

Our proposed research perhaps is most closely related to systems that were designed to decouple collisions across time and frequency. ZigZag [19] decodes multiple collisions by intelligently separating them in time. It requires multiple collisions from the same end-user devices to decode data, unlike our work which strives to separate data from a single collision. Recent work has also proposed the use of carrier frequency offset to count transmissions from active RFID tags [1]. They require the frequency offsets to be much larger than bandwidth to separate simultaneous transmissions from different narrow-band users, which, while true for active RFIDs, does not hold for LP-WAN radios [28]. In contrast to these systems, our approach separates even a single collision of transmissions from multiple nodes that overlap in both time and frequency. It achieves this by exploiting both timing and frequency offsets between the nodes as well as properties of the LP-WAN PHY-layer.

Wide-Area Wireless Sensor Networks: Several proposals have been made for better MAC protocols to avoid collisions in sensor networks, including improved TDMA based methods [22, 36] and collision-recovery methods [19, 23, 45]. Such schemes utilize either customized hardware [19, 23] or improved sensor coordination [36, 45] to recover from or avoid collisions.

Our work is also related to systems that exploit correlation of sensor data to improve performance. Glossy [17] develops hardware and software at sensors to improve time-synchronization and exploit constructive interference of sensed data from different 802.15.4 transmitters. Past research has also proposed modifications to the PHY-layer protocols such as the use of compressed sensing [30] to exploit correlation of sensor data and improve performance. Our work, while building on these systems, differs in that it seeks to exploit correlation of sensor data without requiring hardware modification to LP-WAN sensor radios or the LP-WAN PHY layer. In doing so, it achieves the gain of exploiting correlated sensor data without introducing complexity in PHY-layer hardware and protocols.

[1] A LoRaWAN base station can decode collisions occurring between nodes that utilize different data rates (there are only 4 different data rates in the uplink in the USA). However, it can not handle collisions occurring on a given data rate.

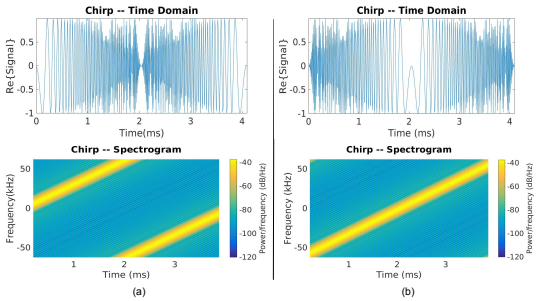


Figure 2: LP-WAN PHY: LoRaWAN uses chirps to depict transmitted bits.

3 A PRIMER ON LP-WANS

This section provides a brief primer on the LoRaWAN LP-WAN protocol. LoRaWANs operate in the unlicensed 900 MHz band with bandwidths of up to 500 kHz. LoRaWAN base stations transmit at powers up to 1 Watt while clients transmit few milliwatts at best. The PHY and MAC layers are designed with this power asymmetry in mind.

Physical Layer: The LP-WAN PHY encodes information in the form of multiple "chirps" that are signals whose frequency varies linearly in time over the available bandwidth. Fig. 2(a)-(b) illustrate two such chirps depicting bits "0" and "1" in the time domain and the corresponding spectrogram. Different bits are encoded by initiating the chirps at different frequencies, for instance "0" at −62.5 kHz and "1" at 0 kHz over a bandwidth of 125 kHz. LoRaWAN uses chirps, as they occupy limited instantaneous bandwidth and therefore consume very little power in communicating bits over long distances. Further, they are robust to narrowband interferers.

Rate Adaptation: While Fig 2 encodes one bit per chirp, LoRaWAN supports larger data rates by increasing the number of possible starting frequencies of a chirp to pack in more bits. For instance, a transmission with 3-bits per chirp would choose from one of 2^3 possible starting frequencies. The LoRaWAN standard allows as many as 12-bits encoded in a chirp. LoRaWAN base stations program each clients to operate on a suitable data rate based on its received signal-quality.

MAC Layer: As described in Sec. 2, the LoRaWAN MAC is designed to avoid collisions and divide air time between competing users. LoRaWAN typically employs two modes to do this for low-power nodes [28]: (1) An Aloha MAC that allows nodes to transmit as soon as they wake up and apply random exponential back-off, when faced with a collision. While simple, Aloha scales poorly in dense networks due to frequent collisions [32]. (2) A TDMA scheduler where the base station allots predetermined slots to clients. The choice of scheme depends on the application (e.g. whether sensed data is bursty) and client power constraints.

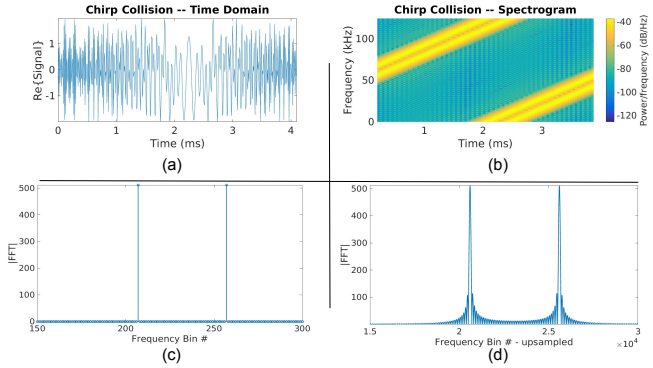


Figure 3: Decoding collisions: Spectrogram of two collided chirps, and the corresponding Fourier transform peaks.

4 CHOIR IN A NUTSHELL

In this section, we provide a brief overview of Choir's core algorithm. At a high level, our goal is to disentangle wireless signal measurements from colliding commodity LP-WAN transmitters with a single-antenna LP-WAN base station.

The core concepts behind Choir are best understood with an example. Consider two LP-WAN radios, both transmitting the same sequence of n bits to an LP-WAN base station. We assume these n bits are encoded in a single chirp as in Fig. 2 by each transmitter. Suppose the two transmissions are aligned perfectly in time, inducing a collision between their chirps. Given that the two LP-WAN radios encode their bits in the exact same way, the resulting chirps would be identical. At first blush, one would assume that these chirps would combine either constructively or destructively upon colliding. This would be problematic for two reasons: First, the combined signal would be indistinguishable from a single transmitter with higher power, rendering the two chirps from the two transmitters impossible to be separated. Second, if the signals add up destructively, one would not be able to recover either of their transmissions.

Choir recognizes that in practice, however, the two signals can be separated by exploiting the natural hardware imperfections of the two radios. Specifically, signals from the two transmitters are likely to experience a small frequency offset, due to a difference in the frequency of their oscillators. This would result in the two chirps being slightly offset in frequency. Fig. 3 depicts the spectrogram of two collided chirps from two commodity LP-WAN radios gathered by a software radio. Note that one can observe two distinct chirps that are shifted in frequency, despite the fact that they both convey the same information. At this point, we can separate the two chirps using a simple process: (1) We first multiply the received signal by a down-chirp[2] that would result in two tones at two frequencies. (2) We then apply a Fourier transform of size 2^n, which results in two peaks corresponding to the two transmissions. In Fig. 3(c), we observe two peaks at two distinct bins, corresponding to the two transmissions. One can then repeat this process for subsequent received chirps to disentangle transmissions from the two users.

While the above approach succeeds in separating the two transmitters, it fails to decode useful data. To see why, recall that LoRaWAN encodes data by shifting chirps in frequency. Specifically, n bits are encoded as 2^n distinct chirps, each starting at a unique frequency. Consequently, the location of the peak corresponding to each transmitter is given by the sum of this frequency offset and the underlying data transmitted by the user. To illustrate the problem, observe that Fig. 3, has two peaks at bins 207 and 257. Such a collision could both be interpreted as identical data and a frequency offset of 50 bins, or zero frequency offset and encoded data differing by 50 bins, or any of the many options in between.

Choir overcomes this problem by relying on the fact that while frequency offset remains constant over a packet between chirps, data does not. To see how this is useful, consider two packets consisting of three symbols (i.e., three chirps) that collide from two users. We assume that the first symbol is a known preamble shared by all users, while the second and third carry useful data. As a result, peak locations from the first symbol can be used to estimate the frequency offset of the first and second user respectively. These frequency offsets can now be subtracted from peaks in subsequent users to capture data corresponding to the first and second user respectively.

An important question still remains – How do we know who is the first and second user in each data symbol? Knowing this is necessary to map the correct frequency offset to the correct peak. More importantly, it is required to avoid mixing up the data bits of the two transmitters when reporting the decoded data.

Our solution to resolve this challenge relies on the fact that data bits occur on integer peak locations in the Fourier transform, while frequency offsets need not. Put differently, frequency offset is a physical phenomenon and does not need to be a perfect multiple of the size of a Fourier transform bin. As a result, the peak locations can be an arbitrary fraction of a Fourier transform bin. To illustrate, suppose we observe two data symbols where the peaks are at 207.2, 257.6 for the first symbol and 81.6, 200.2 for the second symbol. While the integer parts of these peak locations depend on both data and frequency offset, the fractional part depends only on frequency offset, which remains consistent across symbols. Consequently 207.2 and 200.2 must map to one user while 257.6 and 81.6 belong to the other. Choir therefore can use the fractional part of peak locations to distinguish between peaks corresponding to the different users in each symbol, prior to decoding their data bits.

The rest of this paper focuses on achieving three important objectives to realize the above design:

- *Separating Multiple Users:* First, we must estimate frequency offsets accurately to within a fraction of each bin of the Fourier transform. In doing so, we must account for and actively leverage leakage between peaks that is produced due to the frequency offsets that are non-integer multiples of a Fourier transform bin. We then use these frequency offsets to separate collisions of multiple users. Sec. 5 describes our approach in greater detail.
- *Tracking Users using Time and Frequency Offsets:* While our discussion so far assumes that signals collide in a perfectly synchronized manner in time, collisions can occur with arbitrary timing offsets in practice. We overcome this by exploiting the duality between time and frequency in chirps: an offset in time manifests as an equivalent offset in frequency. Our approach in Sec. 6 describes how we exploit this

[2] A down-chirp, i.e., a chirp whose frequency decreases with time, is merely the complex conjugate of the corresponding up-chirp that was used for CSS modulation.

property to account for timing offsets. We also explain how tracking timing offset, frequency offset as well as channels across symbols help us identify which user is which between collisions.

- *Exploiting Correlated Data:* Our discussion thus far has focused on decoding uncorrelated data bits transmitted by two users. However, sensor data is often correlated resulting in nodes transmitting identical chunks of bits. In Sec. 7 we design algorithms to recognize and exploit such scenarios to boost the range of sensors that are otherwise beyond the communication range of the base station.

5 SEPARATING COLLISIONS

This section describes how Choir can separate transmissions from multiple client nodes that utilize the same spreading factor[3] (we discuss the case of different spreading factors in the concluding remarks of Sec. 5.2) and whose transmissions are synchronized perfectly in time (we discuss the effect of timing offset in Sec. 6) . As explained earlier, our approach relies on accurately estimating and exploiting the frequency offset of individual clients. However, estimating frequency offsets accurately in the presence of noise remains a challenge. To illustrate the need for this, we revisit our example from Sec. 4. As shown in Fig. 3(c), signals from the two transmitters manifest as two distinct peaks shifted owing to frequency offset. At this point, we can directly read-off the locations of the peaks to estimate the respective frequency offset of the two nodes. Further, we can filter out the signals around each of the two peaks to obtain the signals from the respective transmitters.

While the above approach is simple, it is prone to inaccuracies and vulnerable to interference. To see why, recall that the estimate of frequency offsets from peak locations is only accurate to within one FFT bin. However, frequency offset is a physical phenomenon that need not be an integer multiple of an FFT bin. This means that the above method loses any information pertaining to frequency offset that is a fraction of one FFT bin. Failing to account for fractional frequency offsets has two important implications: (1) First, as explained in Sec. 4, the fractional part of the frequency offset is extremely useful in identifying which user is which across symbols; without this, the data corresponding to a single user cannot be tracked over time. (2) Second and more fundamentally, having an inaccurate estimate of frequency offset leads to an inaccurate estimate of wireless channels and thus, of the decoded data. Specifically, it causes peaks corresponding to one transmitter to "leak" into others, causing interference. Indeed, this leakage is particularly acute when one transmitter is significantly closer to the receiver compared to the other. This causes one peak to be buried due to noise stemming from another (the so-called near-far effect [20]). Below, we detail our solution to both of these challenges to estimate frequency offsets as well as separate and decode data.

5.1 Measuring Accurate Frequency Offset

Our solution to accurately measure the fractional part of the frequency offset is based on exploiting the leakage of one peak to the other. In order to illustrate this with an example, let us revisit the scenario in Fig. 3 where the two transmitters are separated by a fractional value of frequency offset corresponding to 50.4 bins. The figure however depicts two clear peaks separated by exactly 50 bins. Indeed the remaining separation of "0.4" is encoded in the smaller peaks that leak around the surrounding bins. To better understand and analyze this leakage, let us perform a Fourier transform of the collision between the two transmitters over a wider window (10× larger) by zero-padding the signal. Fig. 3(d) plots the resulting Fourier transform output. Observe that we now have "sinc" functions centered around each peak, a property that stems from the Nyquist sampling theorem. We now observe that the smaller peaks around the main two peaks are produced due to the side-lobes of these sinc functions. Notice that these side lobes are periodic, at an interval of exactly one FFT bin. Indeed, if the two main peaks were apart by an integer multiple, the zeros of the side lobes would overlap perfectly with the main peaks, ensuring zero leakage. In contrast, a fractional separation between peaks causes the side lobes of one peak to interfere with the main lobe of another peak, distorting its shape and location. Consequently, identifying the location of the maxima of the two sinc main peaks provides only a coarse estimate of the frequency offset. In the above example, we observe the two peaks separated by 50.3 bins – an improvement over the previous estimate of "50" bins, but a value that is still erroneous.

To obtain a more fine-grained estimate of the frequency offset, we explicitly model the leakage of the sinc function of one client's signal into the other. First, we estimate the wireless channels of each transmitter, given our coarse estimate of its frequency offset. We then re-construct the received signal using the obtained wireless channels and our frequency offsets. We subtract the reconstructed wireless signal from the actual one to obtain the residual signal. The power of this residual function is an estimate of the goodness of our current frequency offset estimates. We then jitter our estimates of frequency offset and repeat the process, until the power of the residual is minimized. We show that the power of residuals across frequency offset values is locally convex, allowing us to search over the space of frequency offsets efficiently.

Analysis: We now illustrate our approach to mathematically estimate frequency offsets from the collision of two transmitters sending an identical symbol, e.g. a preamble sequence. Let h_1 and h_2 denote the wireless channels and f_1 and f_2 denote the frequency offsets of two transmitters whose chirps collide in time. Let C denote the chirp in the preamble transmitted by both clients that spans a bandwidth of B. Then, we can write the time domain representation of the collision as:

$$y(t) = h_1 e^{j2\pi f_1 t} C + h_2 e^{j2\pi f_2 t} C \implies yC^{-1} = h_1 e^{j2\pi f_1 t} + h_2 e^{j2\pi f_2 t}, \quad (1)$$

where C^{-1} denotes the down-chirp corresponding to the up-chirp C. Ideally, the frequency domain representation of the above signal (obtained via FFT), denoted $\mathcal{F}(yC^{-1})$, should result in two peaks at frequencies f_1 and f_2. However, in practice f_1 and f_2 are unlikely to be at integer boundaries of the FFT bin, as explained earlier. As a result, the peaks of the Fourier transform will likely be close, but not equal, to f_1 and f_2. Let us denote the observed peak locations as $\tilde{f}_1$ and $\tilde{f}_2$, respectively. One can then estimate the approximate wireless channels $\tilde{h}_1$ and $\tilde{h}_2$ that best fit Eqn. 1. Fortunately, given that Eqn. 1 is linear, this can be obtained using a least-squares closed

[3]The spreading factor denotes the number of bits that can be encoded per symbol. Each spreading factor maps to a particular chirp used for CSS modulation and demodulation and it determines the data rate.

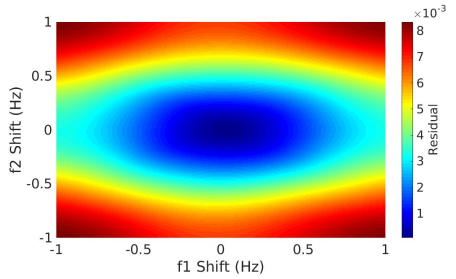


Figure 4: Residual Function: Depicts the residual function for a representative trace with two colliding clients. The function is locally convex.

form as shown below:

$$[\tilde{h}_1\ \tilde{h}_2] = (E^T E)^{-1} E^T y C^{-1} \text{ ,where, } E = [e^{j2\pi\tilde{f}_1 t}\ e^{j2\pi\tilde{f}_2 t}] \quad (2)$$

We can then estimate the goodness of fit of the estimated frequency offset from the observed values by capturing how well the above channels fit the received signal. Specifically, we measure the difference between the received signal and the reconstructed signal based on $\tilde{f}_1, \tilde{f}_2$ as

$$R(\tilde{f}_1, \tilde{f}_2) = ||yC^{-1} - (\tilde{h}_1 e^{j2\pi\tilde{f}_1 t} + \tilde{h}_2 e^{j2\pi\tilde{f}_2 t})||^2 \quad (3)$$

Clearly, from Eqn. 1 and Eqn. 3 above, the residual $R(\tilde{f}_1, \tilde{f}_2)$ will be *minimized* when $\tilde{f}_1 = f_1$ and $\tilde{f}_2 = f_2$. Our solution therefore re-estimates the above residual for frequency offsets in the neighborhood of $\tilde{f}_1$ and $\tilde{f}_2$, and identifies the offsets at which the residual is minimized. Namely, we compute the updated frequency offsets $\bar{f}_1$ and $\bar{f}_2$ via

$$(\bar{f}_1, \bar{f}_2) = \arg \min_{(f_1 \in (\tilde{f}_1 - \Delta, \tilde{f}_1 + \Delta), f_2 \in (\tilde{f}_1 - \Delta, \tilde{f}_2 + \Delta))} R(f_1, f_2) \quad (4)$$

where Δ is the bin-size of the FFT. Repeating the above steps exhaustively over all frequency offsets can be computationally expensive. However, in practice, the residual function $R(f_1, f_2)$ is locally convex, allowing for more efficient search strategies. Intuitively, the local convexity stems from the fact that wireless channels themselves are physical phenomena and therefore tend to be continuous and differentiable. To illustrate, Fig. 4 plots a representative example of the residual function for a collision of two LoRaWAN transmitters from our experiments in Sec. 9. This allows us to apply *stochastic gradient-descent* algorithms [29] on the residual function with randomly chosen initial points that are likely to converge to the global minimum. Algm. 1 provides the pseudo-code of our approach. We note that while the discussion above focuses on two colliding transmitters, it can be readily generalized to multiple collisions.

5.2 Accounting for the Near-Far Effect

While the previous discussion assumes that one can obtain a coarse estimate of the frequency offset by detecting peaks in the Fourier transform, this is often not the case. Consider teams of colliding clients where some are physically closer to the base station compared to others. The nearby users will have clear peaks that are readily discernible from the Fourier transform. The further away users, however, may have significantly weaker peaks that are comparable to the side-lobes of the nearby transmitters. Indeed, it is quite possible that transmissions from users are missed altogether. This is essentially a near-far problem where strong receptions from nearby transmitters overwhelm weaker transmitters [20].

At first blush, one might consider directly employing successive interference cancellation [21] as a solution to overcome this problem. This method estimates and extracts signals of the strongest transmitter from the collision, and repeats this process for the second strongest transmitter, and so on, until no transmitters remain. However, this approach fails to eliminate leakage between a set of transmitters of similar power levels. In contrast, our approach to model and eliminate leakage as in Sec. 5.1 above gets rid of interference between transmitters, but is susceptible to missing weak clients altogether.

Our approach therefore strives to strike a balance between modeling leakage and recovering weak clients to get the best of both worlds. We rely on the fact that while interference from strong transmitters to weaker ones (and to each other) is likely to be high, the opposite is unlikely to be true. This leads us to apply successive interference cancellation in phases, as opposed to one transmitter at a time. Specifically, our approach first measures frequency offset and channels of all the strong transmitters whose peaks are discernible simultaneously, as explained in Sec. 5.1 above. We then subtract the signals of these transmitters from our received signal to eliminate interference to any weaker clients whose peaks were overwhelmed by these transmitters.

We note that our approach, like traditional outdoor networks, is always limited by the resolution of the analog-to-digital converter. As a result, extremely weak transmitters are likely to be missed if they are not registered by the analog components. We discuss extending communication range for such transmitters in Sec. 7.

Decoding Data from Collisions: Next, we note that once the wireless channels and frequency offsets are estimated, decoding data is extremely simple. Specifically, consider collisions of two transmitters synchronized in time whose data as well as preamble symbols collide. We first estimate the peak locations, i.e. frequency offsets, $\tilde{f}_1$ and $\tilde{f}_2$ averaged across each symbol of the preamble. We then repeat this process for the data symbol, where peak locations are given by $d_1 + \tilde{f}_1$ and $d_2 + \tilde{f}_2$, a sum of both the frequency offsets and the data (d_1, d_2). One can then subtract the known frequency offset from these values to obtain the data. Further, one can use the fractional part of the frequency offset (see Sec. 4) to infer which of these data bits maps to which user across symbols. In Sec. 6 below, we elaborate how timing and phase offsets can further be used to achieve this mapping accurately.

Finally, five additional points are worth noting: (1) Our system relies on frequency offsets of LP-WAN radios to remain stable within a packet ($\sim$ 10 ms) but diverse across clients, owing to hardware differences. Our results in Sec. 9.1 show that this is indeed the case across a large number of LoRaWAN boards. While LoRaWAN is the only available LP-WAN platform in the U.S. today, we expect competing technologies to have similar characteristics, given that they use similar inexpensive components [31]. (2) While the above approach is tied to LoRaWAN's chirp-based PHY, the notion of using frequency offsets to separate transmissions broadly applies to other LP-WAN technologies such as NB-IoT [34] and SigFox [31]. Indeed, given that these technologies use an ultra-narrowband PHY,

we expect their bandwidth to be far lower than frequency offset that allows filtering their transmissions based on hardware offsets significantly simpler. It is worth noting, however, that timing offsets do no necessarily map to frequency offsets in these technologies, thus Choir would have to be modified in accordance. (3) While our system allows collisions from multiple transmitters to be decoded, its gains are not unbounded. Gains are limited by noise and the possibility of overlapping frequency offsets that increases with collisions from a larger number of transmitters. Our results in Sec. 9.2 measures the scaling limits of Choir. (4) The current implementation of LoRaWAN allows simultaneous decoding of collided data packets that were transmitted across *different* orthogonal spreading factors. Such a feature is made possible due to the orthogonality of the chirps associated with the different spreading factors. In particular, a packet transmitted at a given spreading factor can *only* be demodulated with a *unique* chirp associated with this spreading factor. This form of diversity improves the overall data rate as it allows parallel decoding of collided data packets. Choir considers the *extreme* case of a congested network utilizing a *single* spreading factor, and a *single* antenna. However, Choir can indeed benefit from i) a base station utilizing multiple antennas (see Sec. 9.5), or ii) a network employing different spreading factors. In particular, consider 5 LP-WAN sensors simultaneously transmitting data packets with spreading factors 7, 7, 8, 8, and 9, respectively. The base station demodulates the received data stream using the unique chirps associated with each spreading factor. In particular, let C_7, C_8, and C_9 be the chirps associated with spreading factors 7, 8, and 9 respectively, and let y denote the received signal. Consider $y_7 = yC_7^{-1}$, $y_8 = yC_8^{-1}$, and $y_9 = yC_9^{-1}$. By the orthogonality of C_7, C_8, and C_9, it is clear that y_7, y_8, and y_9 would *only* contain the part of the data transmitted with spreading factors 7, 8, and 9, respectively[4]. At this point, the base station can run Choir to the three *independent* data streams y_7, y_8, and y_9 to disentangle possible collisions occurring at each of these different spreading factors. Clearly, the orthogonality of the chirps resulting from utilizing different spreading factors alleviates the collisions, and avoids the complexity and scalability issues associated with having all collisions occurring on a single spreading factor. (5) The objective of Choir is to handle unintended collisions among LoRaWAN nodes. As for potential collisions between LoRa and other technologies, we rely on the fact that LoRa utilizes CSS as a form of spread spectrum that makes it robust to cross-technology interference similar to CDMA systems.

6 MITIGATING TIMING OFFSETS

Our discussion so far assumes that clients transmit their packets coherently in time. In this section, we describe our approach to estimate and actively exploit the natural timing offsets between transmissions. We then use timing and frequency offset estimates, along with wireless channels to map which bits belong to which users within a packet.

6.1 Exploiting Timing Offsets

We exploit timing offsets by leveraging the properties of the chirp spread spectrum used by LoRaWAN radios. In particular, we use

[4]This is similar to how LoRaWAN currently demodulates collided data transmitted with multiple spreading factors.

Algorithm 1 Decoding collisions using Choir

$PreambleLen = \ell$, $SpreadingFactor = SF$
$SymSize = 2^{SF}$
$FFTLen = 10 \times SymSize$
while $SymCount < \ell$ **do**
  $DemodSym = Symbols(SymCount). * DownChirp$
  $n = \text{NumPeaks}(FFT(DemodSym, FFTLen))$
  $[\tilde{f}_1, \ldots, \tilde{f}_n] = \text{FindPeaks}(FFT(DemodSym, FFTLen))$
  $[\tilde{h}_1, \ldots, \tilde{h}_n] = \text{LeastSquares}([\tilde{f}_1, \ldots, \tilde{f}_n])$
  $[\overline{f}_1, \ldots, \overline{f}_n] = \arg\min_{\left(f_1 \in \left(\tilde{f}_1 - \Delta, \tilde{f}_1 + \Delta\right), \ldots, f_n \in \left(\tilde{f}_n - \Delta, \tilde{f}_n + \Delta\right)\right)} \text{R}([\tilde{f}_1, \ldots, \tilde{f}_n])$
  $SymCount = SymCount + 1$
end while

the fact that chirps, by definition, are signals whose frequency increases linearly with time. This means that any offset in time of a received signal chirp can be equivalently interpreted as a corresponding offset in frequency. Given that our approach corrects for and exploits frequency offsets, it remains immune to timing offsets as well.

Analysis. To demonstrate why this is the case, let us consider chirps (symbols) from two different clients that collide in time. Let us assume the two symbols in Eqn. 1 are off by a shift in time of Δt_1 and Δt_2, respectively. Then a shift in time of Δt is akin to a shift in frequency of $B\Delta t/T$. This means that, in the presence of frequency offset, the received wireless signal can be re-written as:

$$y(t) = h_1 e^{j2\pi\left(f_1 + B\frac{\Delta t_1}{T}\right)t} C + h_2 e^{j2\pi\left(f_2 + B\frac{\Delta t_2}{T}\right)t} C \tag{5}$$

In other words, the timing offset is simply absorbed into our frequency offset estimates in Sec. 5. More importantly, the timing offset between any two transmitters, just like the frequency offset, remains consistent across symbols over the duration of a packet (we validate this in Sec. 9.1).

Dealing with Inter-Symbol Interference. While the approach described above accurately accounts for timing offset within a symbol, it fails to capture the effect of inter-symbol interference. To illustrate, consider Fig. 5 where transmissions of two users collide, each sending different data symbols and each encoded by a chirp. In this case, it is quite possible that over the duration T of a chirp, one can observe as many as four distinct symbols colliding as shown. Failure to account for these collisions would lead to access points dropping or re-ordering symbols between users.

Our solution to account for inter-symbol interference explicitly tracks the peaks that result from this interference. Specifically, a collision of two shifted symbols produces at most four peaks in the Fourier Transform – two belonging to the first client and two to the second. Fig. 5 shows such a Fourier transform for two adjacent time windows, each of length T. One can then extract the locations of these peaks to obtain four distinct data values per time window. However, given that two symbols participate in both the first and second collision, the two collisions are guaranteed to share two common data values (see Fig. 5). Indeed, any pair of adjacent collisions will share at least two common data values. By ensuring that any such common values are reported only once (for e.g., the first time they appear), one can eliminate half of the observed peaks. This enables the data from all transmitters to be correctly reported in-sequence, despite inter-symbol interference.

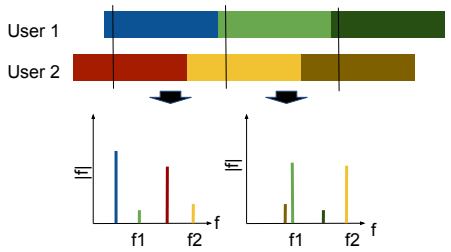


Figure 5: Inter-Symbol Interference: Spectrogram of two collided chirps, and the corresponding Fourier transform peaks.

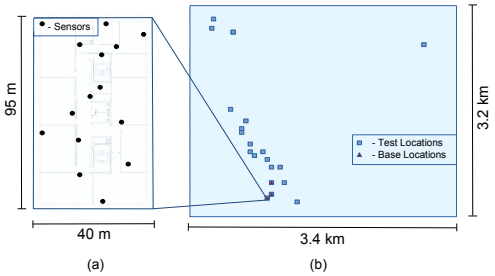


Figure 6: Testbed: (a) Sensor testbed spans four floors of a large university building; floor plan of one such floor is shown with sensor locations marked by dots. (b) Anonymized map of the neighborhood surrounding CMU campus with testbed spanning 10 square kilometers.

6.2 Mapping Symbols to Users within a Packet

In this section, we use both time and frequency offsets to map which symbols (i.e., chirps) correspond to which user within a packet (See Sec. 5.2) along with one other metric – wireless channels. Like hardware offsets, wireless channels are expected to remain consistent for a given client over a packet and vary between clients. For instance, in Fig. 5, we observe that peaks of the same user over two symbols are not only identical in frequency offset, but also in relative height. This means that channel magnitude and phase, after correcting for any phase offsets between symbols introduced by frequency offsets, serves as a feature to identify users. This allows us to build a semi-supervised clustering model (we use the HMRF-based approach in [10]) using the fractional part of peak location, channel magnitude, and phase. We give the clustering algorithm known prior relationships, e.g. multiple peaks in the same symbol map to distinct users. We then run the clustering algorithm to recover the sequence of bits corresponding to each user.

7 EXPLOITING CORRELATED TRANSMISSIONS

So far, we have explained our approach to disentangle and decode colliding transmissions from clients who are all within communication range of the base station. We now argue how this approach also provides a unique opportunity to retrieve data from clients beyond communication range. The data transmitted by an LP-WAN sensor can not be decoded by the base station if the received SINR falls below a particular threshold. Indeed, a message modulated by CSS can be recovered even if it is deeply buried in noise, but there is a particular minimum SINR below which a transmitted message will not even be detected by the base station.

Although one would expect that all LP-WAN sensors were initially deployed in the vicinity of an LP-WAN base station (and thus should always be reachable), their transmissions may not reach the base station because of the randomness of the wireless channel, interference with other technologies sharing the same bandwidth, or a change in the surrounding urban environment itself.

While individual sensors may be beyond communication range, collisions of teams of such sensors can be detected by base stations. One can then decode these collisions to recover bits transmitted by these sensors that overlap. Indeed, given that sensors geographically close to one another are likely to have several overlapping bits, one can use this information to obtain a coarse view of sensor data in a given geographical region.

The rest of this section addresses various challenges in achieving such a design. First, how do we ensure that teams of sensors transmit packets that are synchronized in time? Second, how do we detect and decode their collisions, despite the fact that individual sensors are beyond communication range? Finally, how do we choose which sensors transmit concurrently?

7.1 Coordinating Transmissions from Sensor Teams

Consider a team of sensors that are individually beyond communication range but would like to transmit identical data packets. Indeed, gathering a large team of sensors would cause the overall received power to add up, increasing signal power. However, to do so one would have to ensure that the transmissions are synchronized in time so that identical symbols across transmitters add up to reinforce received power.

Time Synchronization. We rely on the fact that Choir is immune to timing offset. Specifically, we first make the base station transmit a beacon packet that solicits a response from all sensors in a given geographic boundary. Given that the base station affords a much higher transmit power (and superior antennas) as compared to the client, its signal will be received by all these sensors, even if their signals are individually too weak to reach the base station. The sensors then respond concurrently with packets in the next time slot (i.e. after a fixed pre-agreed duration of time). However, in practice, such synchronization is never perfect and packets between different sensors will continue to have a small timing offset. Fortunately, given the relatively long symbol durations of LP-WAN ($\sim$ 10 ms), this timing offset is smaller than one symbol (see Sec. 9.1-9.3). As described in Sec. 6, such timing offsets can be interpreted as a corresponding frequency offset between the different transmitters. Recall that Choir exploits such frequency offsets to obtain distinct peaks corresponding to each client in any collision (as in Fig. 3). As a result, the coarse time-synchronization provided by the base station's beacon packet is sufficient to observe such peaks, at least for sensors above the noise-floor at the base station.

Whom do we coordinate? Now that we have a mechanism to coordinate sensors, how do we decide whom to coordinate or schedule at any given time. In practice, making this decision is a function of the spatial distribution of sensor data, which can vary between different kinds of sensors and different environments. Given that sensors are often deployed statically in buildings over long durations,

one can learn the extent of these correlations over time. Indeed, such a scheduling algorithm can estimate the signal-to-noise ratio of clients to schedule larger groups of sensors for transmitters that are further away. In effect, this leads to a system whose resolution of measured sensor data increases for sensors that are geographically closer to the base station. Our results in Sec. 9 measures the correlation of data from various subsets of temperature sensors deployed across multiple buildings in CMU campus.

7.2 Decoding Beyond Communication Range

Now that we have synchronized collisions from a desired group of clients, we next detect and decode their data.

Detecting Packets. A key challenge however is that for transmitters far away from the base station, all peaks may be buried below the noise-floor. Indeed, this makes detecting collided packets from teams of weak client transmitters particularly challenging. Our solution to overcome this problem relies on the multiplicity of clients that collide as well as preamble symbols. In particular, we coherently add the power of the Fourier representation, given by Eqn. 1 over sliding windows of n symbols, where n is the size of the preamble. Despite the fact that peaks in any one symbol are below the noise, they are unlikely to be buried in noise when averaged over a large number of symbols. This allows us to both detect the packet as well as obtain coarse estimates of frequency offsets as required by Algm 1.

Decoding Algorithm. While averaging over symbols is useful to detect the energy of the preamble, one cannot do so for the data given that each data symbol carries a unique sequence of bits. Therefore, our solution to decode data relies on a maximum-likelihood approach that exploits the knowledge of frequency offsets. Specifically, we reconstruct different possible collisions of the transmitters given their channels and frequency offsets (from Algm. 1) for each possible sequence of data bits in a symbol. We then obtain the data bits by identifying the collision that best fits the observed data symbol. Mathematically, for any received signal y, channels h_i, timing offsets t_i and frequency offsets f_i for each client i, we obtain the data symbol d as

$$d = \arg\min_{d} ||y - \sum_{i} h_i e^{j2\pi(f_i + B\frac{\Delta t_i}{T} + d)t} C||^2 \quad (6)$$

where C is the known preamble chirp that spans a bandwidth B over time T, as before. Given that the above equation models and exploits the presence of multiple clients in the collision, it provides a robust method to decode data despite each individual client's signal being below noise.

Dealing with Collisions. Despite scheduling certain teams of transmitters with a beacon from the base station, it is possible that such transmissions will experience collisions with other sensors closer to the base station. Our approach to deal with such unwarranted collisions is very similar to Choir's solution for the near-far effect in Sec. 5.2. In particular, we first measure and subtract peaks above the noise from the received signal until repeatedly using Algm. 1 until no clear peaks are visible. Finally, we apply the detection and decoding steps described above to extract scheduled transmissions from clients that are below the noise floor.

A few points are worth noting: (1) Like any other protocol, Choir may be unable to recover collisions owing to excessive interference or noise leading to some packets unacknowledged. In this scenario, Choir relies on LoRaWAN's underlying MAC protocol (ALOHA or TDMA) to identify such loses (e.g. using acknowledgments) and re-transmit. (2) To achieve gains, Choir requires that overlapping chunks of bits of sensor data lead to overlapping chunks of signals that then add up in power. However, interleaving and coding schemes may cause even data different by one bit to have few coded bits in common. Our solution to resolve this is to splice sensed data into smaller packets that carry different chunks of consecutive sensed bits so that those with most significant bits remain identical, even after coding.

8 IMPLEMENTATION

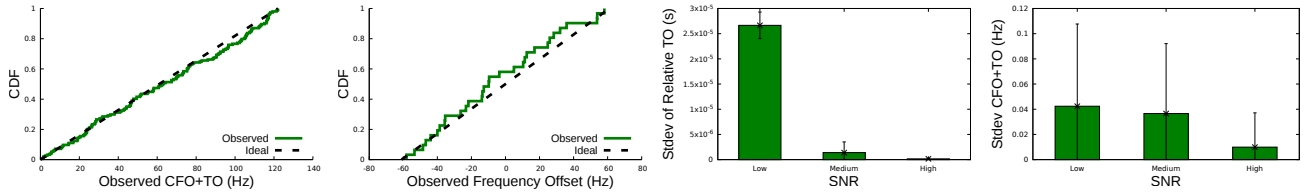
We implement Choir on a testbed of software radio base stations and clients built using commodity components and the LoRaWAN chip. Our base stations are composed of USRP N210 software radios and the WBX daughterboards operating at the 900 MHz bands[5]. We use the UHD+GnuRadio library and develop our own LoRaWAN decoder and Choir's algorithms in C++ and MATLAB to process signals. Unless specified otherwise, our base station uses a single S469AM-915 antenna and a ZX60-0916LN+ low noise amplifier. We mounted the base station on the top floors of three tall buildings on CMU campus. Our experiments using MU-MIMO deploy with up to 3 base-station antennas synchronized by a Jacksonlab Fury clock.

The clients are SX1276MB1LAS boards with an embedded LoRaWAN chip that is mBed compatible. We connect these boards with NUCLEO-L152RE boards with the mBed platform to program the LoRaWAN chips to transmit sensor data at regular time periods. The boards operate at a center frequency of 902 MHz over a bandwidth of 500 KHz or 125 KHz depending on the data rate the wireless channel supports [5]. We consider three different types of data: (1) Random sequence of bits per packet that are transmitted periodically at regular intervals (500 ms). (2) A specific known sequence of bytes at the same period. (3) Sensor data from temperature and humidity sensors placed across different buildings in the university campus, as they are observed. We leverage an open environmental sensor board platform with an Atmel Atmega32L microcontroller and on-board BME280 temperature and humidity sensors.

Evaluation: We evaluate our system in a neighborhood of CMU campus. The campus contains and is surrounded by several multi-storey buildings, trees and hilly terrain. We make up to 30 client nodes simultaneously transmit from as many as 100 locations across four floors of five different buildings in different parts of the campus as well as in buildings, roads and pedestrian walkways outside campus over an area spanning 10 square kilometers around the campus. Fig. 6 plots the scale of our testbed area with the actual roads and building shapes omitted due to anonymity. We note that we consider concurrent transmissions from multiple distributed client nodes to a single base station at any time.

Baseline: We compare our system with two baselines: (1) LoRaWAN: A standard LoRaWAN baseline that uses slotted

[5]Note that dedicated LoRa base stations can support better ADCs than the USRP given that the base station can afford to be more expensive and power hungry.



(a) (b) (c) (d)

Figure 7: Characterizing Hardware Offsets: (a)-(b) Measures the CDF of the time plus frequency offset and only the frequency offset as observed across 30 LoRaWAN LP-WAN nodes. (c)-(d) Measures the average and standard deviation of the root mean-squared error of the relative timing offset and the frequency offset plus timing offset within a packet.

ALOHA coupled with exponential backoff to retransmit; (2) LoRaWAN+Oracle: LoRaWAN with an oracle scheduler that explicitly schedules transmissions optimally to avoid collisions; (3) Choir: Our System which decouples collisions using hardware offsets. When deploying multi antenna stations (Sec. 9.5), we additionally compare our system with the state-of-the-art uplink MU-MIMO [40].

9 RESULTS

9.1 Characterizing Hardware Offsets

In this experiment, we characterize the distribution of observed frequency and timing offsets measured across different LoRaWAN hardware. We do this to characterize the diversity of these offsets with real hardware, which is crucial to separate different users. We further evaluate how stable they remain across symbols over the duration of a packet.

Method: We consider a testbed with two single-antenna LP-WAN radio which transmit a known sequence of bits concurrently. We synchronize transmissions using beacon packets as described in Sec. 6. We receive collisions from these transmitters on a USRP N210 software radio emulating a single-antenna LP-WAN base station. We repeat this experiment across multiple packets and measure the timing and frequency offset on a per-symbol basis as described in Sec. 5 and 6. We further perform this experiment for different pairs of LoRaWAN radios across 30 LP-WAN radios.

Results: Fig. 7(a) and (b) plot the cumulative distribution of time plus frequency offset, and only the frequency offset, respectively, as measured across 30 nodes. We specifically focus on the fractional component of frequency offset and sub-symbol timing offsets, given that these are the quantities that help us separate transmissions of users. We note that observed sub-symbol timing offsets and frequency offsets in the wild across nodes are equally likely to span the entire range of possible values. The diversity of these hardware offsets makes them suitable vehicles to track and separate users.

Next, we evaluate the stability of these values across symbols and our ability to measure them accurately. We plot the average and standard deviation of the root mean-squared percentage error of the relative timing offset and the frequency offset plus timing offset within a packet in fig. 7(c) and (d) across a range of SNR values. As a percentage relative to the duration of symbol and bandwidth of a subcarrier, respectively, we observe the mean error of these offsets to be just 1.84 % and .04 % respectively, attesting the stability of these values and Choir's ability to track them accurately.

9.2 Disentangling Collisions

In this experiment, we present our results from disentangling collisions from simultaneous transmissions by a large number of LP-WAN nodes.

Method: We consider a testbed, initially with two single-antenna LP-WAN radios which each transmit a randomly chosen sequence of bits concurrently. We receive collisions from these transmitters on a USRP N210 software radios emulating a single-antenna LP-WAN base station. We repeat this experiment across a range of locations of the two LP-WAN nodes where both nodes experience different levels of signal-to-noise ratio (SNR). We then progressively add nodes until the network has as many as 10 nodes transmitting concurrently at any time. We measure three metrics: (1) network throughput of all nodes; (2) latency measured between a beacon packet from the base station and the response packet from a client; (3) total number of transmissions and re-transmissions required to send one packet worth of data – a useful metric to measure energy efficiency, as that packet transmission is a major drain on battery for sensors [3].

Results: Fig. 8(a)-(c) Measures the throughput, latency and number of transmissions for Choir and the LoRaWAN baseline for two radios across different SNR regimes – low (<5 dB), medium (5-20 dB) and high (>20 dB). Nodes transmit at the fastest data rate that can be supported by the SNR. We observe that Choir experiences a 2.58×(2.113×) gain in throughput vs. LoRaWAN(+Oracle), 3.9×(1.5×) reduction in latency and 3.0549×([6]) × reduction in number of transmissions required to send a useful packet of data over standard LoRaWAN. Indeed, Choir's performance remains consistent across SNR regimes. Fig. 8(d)-(f) measures the throughput, latency and number of transmissions for Choir and the LoRaWAN baseline as the number of concurrent users colliding progressively increases. Our system's performance increases progressively as the number of users increases, given the opportunities to decode multiple users simultaneously, with 29.02×(6.84×) gain in throughput vs. LoRaWAN(+Oracle), and 19.37×(4.88×), 4.54× reduction in latency and retransmissions respectively for 10 simultaneous users. We observe that the scaling, while impressive is not unbounded. This is because at such a large number of concurrent users, the near-far effect coupled with collisions in hardware offsets become increasingly likely to limit system performance.

[6]Oracle has perfect performance in # transmissions

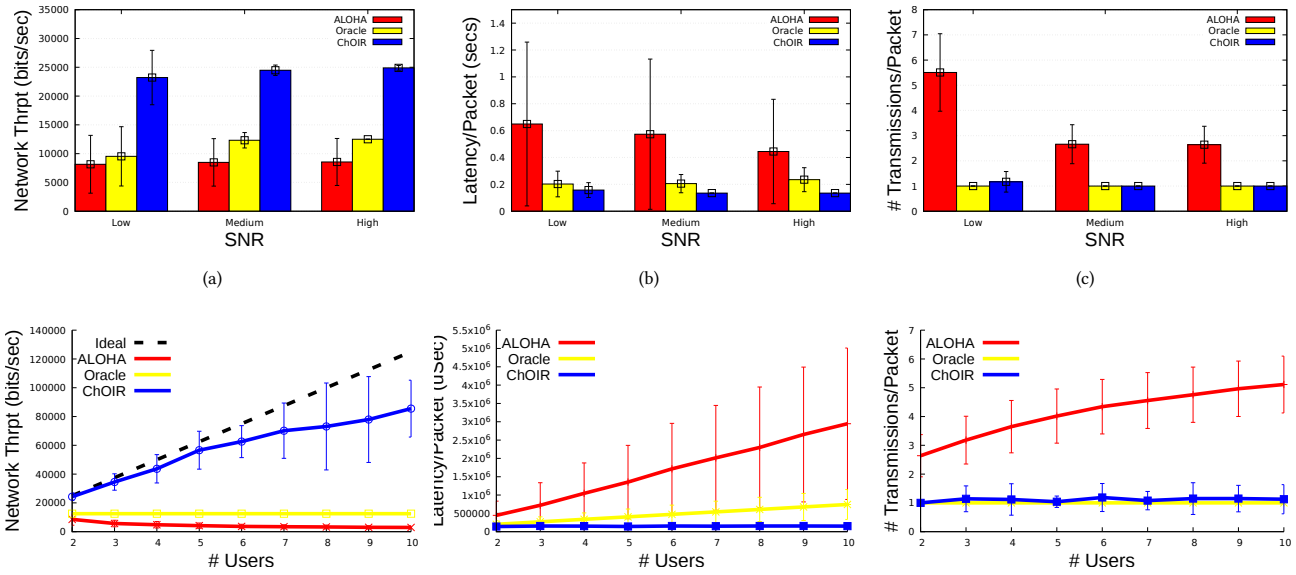


Figure 8: Disentangling Collisions: Consider concurrent transmissions from several LP-WAN nodes across a wide range of SNRs decoded at a single-antenna LP-WAN base station. (a)-(c) Measures the throughput, latency and number of transmissions for Choir and the LoRaWAN baseline across low (<5 dB), medium (5-20 dB) and high (>20 dB) SNRs. (d)-(f) Measures the throughput, latency and number of transmissions for Choir and the LoRaWAN baseline across number of users.

9.3 Extending Range of LP-WANs

In this experiment, we evaluate the promise of Choir in extending the range of low-power networks in urban settings.

Method: We consider two or more nodes in our testbed physically separated by the USRP base station by a given distance and transmitting identical data. In particular, we focus on locations of nodes from which signals cannot reach the base station, even at the minimum data rate of LoRaWAN. We then progressively increase the number of sensors situated at randomly chosen positions in the testbed that collide at any time. We group the observed data based on the minimum distance between the nodes that collide and the base station. We measure two quantities: (1) the throughput achieved by teams of these sensors in transmitting the desired data sequence, as the number of sensors broadcasting identical data increase across a range of distances (and SNR) between the base station and clients; (2) the maximum distance of the closest transmitter whose collisions were decodable when it collaborates using Choir with different number of other transmitters to reach the base station.

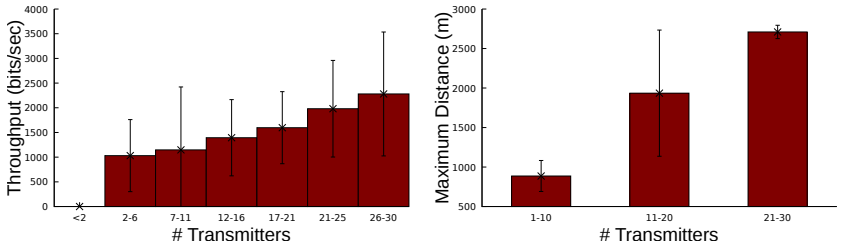
Results: Fig. 9(a) measures the throughput of various numbers of LoRaWAN clients coordinating to transmit a given data sequence to the base station. We chose these clients so that individually, their throughput to the base station is zero even at the lowest data rate. However, collectively their throughput increases substantially with teams of up to 30 nodes transmitting at data rates as high as 5470 bps. This is because as larger teams of clients collide, their signals are received at greater power, allowing these clients to transmit at higher data rates. Next, we study the impact of this on the range of the LP-WAN network. Fig. 9(b) plots the maximum of the distance of the closest transmitter to a base station, as it collaborates with teams of other transmitters to reach the base station. We observe that while one client in the network could reach at best a distance of 1 km – a fairly low quantity, in part due to the tall buildings and hilly topography of CMU campus as well as hardware limitations of the USRP's receive chain [35]. Under identical hardware constraints, teams of colliding 30 clients using Choir could reach the base station even when the closest of them was 2.65km away, an improvement of 2.65 ×.

9.4 Exploiting Correlated Sensor Data

In this experiment, we evaluate Choir's ability to exploit transmission of sensor data that is spatially correlated

Method: We leverage a testbed of sensor nodes placed in four different floors of across two large buildings in CMU campus (Fig. 6 plots the sensor locations). Each sensor measures both temperature and humidity values in the room in which they are placed. We co-locate 36 LP-WAN radios with these sensors and transmit. These sensors transmit periodically at a rate of 1 reading per minute. We then measure the network throughput and resolution of the sensor data from the base station.

Results: Fig. 11(a) plots the mean percentage error of the observed sensor data against the true values, for sensors grouped together using different strategies – randomly, by floor and by relative distance from the center of the floor. We find that the relative distance from the center of the floor to be an excellent method to group together sensors. This stems from the fact that the farther away these sensors are from the center of the building, the closer they are to the outside temperature (or humidity). Next, we evaluate the end-to-end performance of our system compared to the LoRaWAN baselines. Specifically, our system schedules transmissions from groups of sensors that are beyond the range of the LP-WAN base station, while allowing nearby sensor nodes to transmit data as



(a) (b)

Figure 9: Extending LP-WAN Range: Consider concurrent identical transmissions from several LP-WAN nodes to a single-antenna LP-WAN base station. (a) Measures throughput gain of Choir over the LoRaWAN baseline systems across SNRs. (b) Measures the physical range achieved given different number of concurrent users.

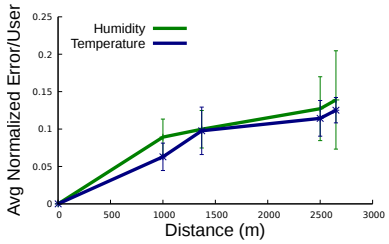
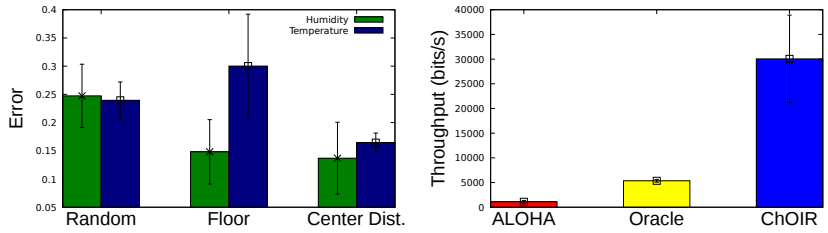


Figure 10: Resolution vs. Distance: Measures the resolution of the sensor data received from sensors in a given floor across distance.



(a) (b)

Figure 11: Exploiting Sensor Data Correlation: (a) Measures the correlation of sensor data (temperature and humidity) by considering three different ways to group them – random, by floor and by distance. (b) Measures end-to-end network throughput required to convey data over the network for a mix of sensors, some close and some far from the base station for Choir and baseline systems.

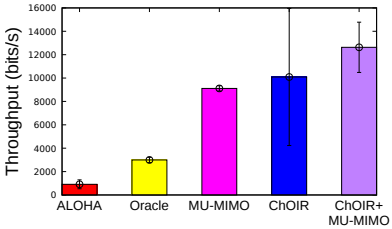


Figure 12: Comparison with MU-MIMO: Measures the throughput gain of Choir using both one and 3-antennas against MU-MIMO [40] on a 3-antenna base station.

they collect them. We then apply Choir's algorithms described in Sec. 7.2 to decode potential collisions as well as to exploit correlation in sensor data to recover overlapping bits from groups of sensors that are beyond base station's range. Fig. 11(b) measures the network throughput of observed sensor data for Choir and the baseline systems. We note that our system has a gain of 29.3377× over LoRaWAN+ALOHA and 5.609× over the LoRaWAN+Oracle baseline. Finally, one may wonder how the resolution of observed sensor data from sensors beyond communication range varies with their distance to base station. Fig. 10 plots the mean percentage error of the recovered sensor data across an increasingly large group of sensors as we vary their distance to the base station (relative to the closest sensor). As expected, we observe a gradual decrease in resolution with distance, with an error of 13.2% for teams of up to 30 sensors at a distance of at least 2.5km from the base station. We note that despite the loss in resolution, Choir is far superior to the baseline systems, where all of these nodes would be beyond communication range of the base station.

9.5 Effect of Multiple Antennas

We evaluate the performance of Choir relative to MU-MIMO for a base station with 3-antennas. Fig. 12 plots the network throughput of a team of 5 sensors transmitting data to the base station using: (1) Only one receiver antenna and LoRaWAN+ALOHA; (2) Only one receiver antenna and LoRaWAN+Oracle; (3) All three antennas and uplink MU-MIMO; (4) Choir using only one receiver antenna; (5) Choir run on all three antennas and averaging results. We observe that while MU-MIMO's gain over standard LoRaWAN is capped at 9.994 ×(3.04 ×) vs. LoRaWAN(+Oracle), Choir, with even a single antenna is at 11.07 ×(3.37 ×). Further, the presence of multiple antennas can be used to further improve Choir's gain to 13.8489 ×(4.217 ×), demonstrating that its gains are complementary to MU-MIMO.

10 CONCLUSION

This paper presents Choir, a system that improves throughput and range of low-power wide area networks in urban environments. Choir proposes a novel approach that exploits the natural hardware offsets between low-power nodes to disentangle collisions from several LP-WAN transmitters using a single-antenna LP-WAN base station. Further, Choir allows teams of LP-WAN sensor nodes with correlated data to reach the base station, despite being individually beyond communication range. Our system is implemented and deployed on a large outdoor testbed spanning 10 km^2 around CMU campus.

11 ACKNOWLEDGMENT

This work has been supported in part by National Science Foundation through grants CCF-1617934 and CNS-1657318, as well as the ARCS Foundation and the Michel, E 1968, 1976, and Kathy Doreau Graduate Fellowship in Electrical and Computer Engineering. We thank Prof. Dina Katabi, Prof. Peter Steenkiste, Prof. Srinivasan Seshan, Prof. Vyas Sekar, Prof. Anthony Rowe, Craig Hesling, Adwait Dongare, Khushboo Bhatia, and Artur Balanuta for their support and insights.

REFERENCES

[1] Omid Abari, Deepak Vasisht, Dina Katabi, and Anantha Chandrakasan. Caraoke: An e-toll transponder network for smart cities. In *Proc. of ACM SIGCOMM 2016*, Vol. 45. ACM, 297–310.

[2] Ferran Adelantado, Xavier Vilajosana, Pere Tuset-Peiro, Borja Martinez, and Joan Melia. 2016. Understanding the limits of LoRaWAN. *arXiv preprint arXiv:1607.08011* (2016).

[3] Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. 2002. Wireless sensor networks: a survey. *Computer networks* 38, 4 (2002), 393–422.

[4] LoRa Alliance. 2015. A Technical Overview of LoRa and LoRaWAN. (2015). https://www.lora-alliance.org/portals/0/documents/whitepapers/LoRaWAN101.pdf.

[5] LoRa Alliance. 2015. LoRaWAN Specification. *LoRa Alliance* (2015).

[6] Narendra Anand, Ryan E Guerra, and Edward W Knightly. The case for UHF-band MU-MIMO. In *Proc. of ACM Mobicom 2014, Maui, Hawaii, USA*. 29–40.

[7] Ehsan Aryafar, Narendra Anand, Theodoros Salonidis, and Edward W Knightly. Design and experimental evaluation of multi-user beamforming in wireless LANs. In *Proc. of ACM Mobicom 2010, Chicago, Illinois, USA*. 197–208.

[8] Aloÿs Augustin, Jiazi Yi, Thomas Clausen, and William Mark Townsley. 2016. A study of LoRa: Long range & low power networks for the internet of things. *Sensors* 16, 9 (2016), 1466.

[9] Jean-Paul Bardyn, Thierry Melly, Olivier Seller, and Nicolas Sornin. 2016. IoT: The era of LPWAN is starting now. In *European Solid-State Circuits Conference, ESSCIRC Conference 2016: 42nd*. IEEE, 25–30.

[10] Sugato Basu, Mikhail Bilenko, and Raymond J. Mooney. A Probabilistic Framework for Semi-supervised Clustering. In *Proc. of ACM SIGKDD 2004, Seattle, WA, USA*. 59–68.

[11] Martin C. Bor, Utz Roedig, Thiemo Voigt, and Juan M. Alonso. 2016. Do LoRa Low-Power Wide-Area Networks Scale?. In *Proc. of ACM MSWiM 2016, Malta, Malta*. 59–67.

[12] Alexandros-Apostolos A Bouleogeorgos, Panagiotis D Diamantoulakis, and George K Karagiannidis. 2016. Low Power Wide Area Networks (LPWANs) for Internet of Things (IoT) Applications: Research Challenges and Future Trends. *arXiv preprint arXiv:1611.07449* (2016).

[13] Marco Centenaro, Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. 2015. Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios. *arXiv preprint arXiv:1510.00620* (2015).

[14] Hyeonwoo Cho and Sang Woo Kim. 2012. An anti-collision algorithm for localization of multiple chirp-spread-spectrum nodes. *Expert Systems with Applications* 39, 10 (2012), 8690–8697.

[15] Yongjiu Du, Ehsan Aryafar, Joseph Camp, and Mung Chiang. 2014. iBeam: Intelligent client-side multi-user beamforming in wireless networks. In *Proc. of IEEE INFOCOM 2014*. IEEE, 817–825.

[16] Mahmoud Elkhodr, Seyed Shahrestani, and Hon Cheung. 2016. Emerging Wireless Technologies in the Internet of Things: a Comparative Study. *arXiv preprint arXiv:1611.00861* (2016).

[17] Federico Ferrari, Marco Zimmerling, Lothar Thiele, and Olga Saukh. 2011. Efficient network flooding and time synchronization with glossy. In *Proc. of ACM/IEEE IPSN 2011*. 73–84.

[18] Shyamnath Gollakota, Fadel Adib, Dina Katabi, and Srinivasan Seshan. 2011. Clearing the RF smog: making 802.11 n robust to cross-technology interference. In *ACM SIGCOMM Computer Communication Review*, Vol. 41. ACM, 170–181.

[19] Shyamnath Gollakota and Dina Katabi. 2008. ZigZag Decoding: Combating Hidden Terminals in Wireless Networks. In *Proc. of ACM SIGCOMM 2008,Seattle, WA, USA*. ACM, 159–170.

[20] DAVID J Goodman and Adel AM Saleh. 1987. The near/far effect in local ALOHA radio communications. *IEEE Transactions on vehicular technology* 36, 1 (1987), 19–27.

[21] Daniel Halperin, Thomas Anderson, and David Wetherall. 2008. Taking the sting out of carrier sense: Interference cancellation for wireless LANs. In *Proc. of ACM MobiCom 2008*.

[22] Ahsanul Haque, Manzur Murshed, and Mortuza Ali. Efficient Contention Resolution in MAC Protocol for Periodic Data Collection in WSNs. In *Proc. of ACM IWCMC 2010, Caen, France*. 437–441.

[23] Kyle Jamieson and Hari Balakrishnan. PPR: Partial Packet Recovery for Wireless Networks. In *Proc. of ACM SIGCOMM 2007, Kyoto, Japan*.

[24] Matthew Knight and Balint Seeber. 2016. Decoding LoRa: Realizing a Modern LPWAN with SDR. In *Proceedings of the GNU Radio Conference*, Vol. 1.

[25] Mads Lauridsen, István Kovács, Preben Mogensen, Mads Sørensen, and Steffen Holst. Coverage and Capacity Analysis of LTE-M and NB-IoT in a Rural Area. In *Proc. of IEEE VTC-Fall 2016*. 1–5.

[26] Ruizhi Liao, Boris Bellalta, Miquel Oliver, and Zhisheng Niu. 2016. MU-MIMO MAC Protocols for Wireless Local Area Networks: A Survey. *IEEE Communications Surveys Tutorials* 18, 1 (Firstquarter 2016), 162–183.

[27] Kate Ching-Ju Lin, Shyamnath Gollakota, and Dina Katabi. 2011. Random access heterogeneous MIMO networks. *ACM SIGCOMM Computer Communication Review* 41, 4 (2011), 146–157.

[28] LoRA Alliance. 2016. LoRaWAN for Developers. (2016). https://www.lora-alliance.org/For-Developers/LoRaWANDevelopers.

[29] David G Luenberger. 1973. *Introduction to linear and nonlinear programming*. Vol. 28. Addison-Wesley Reading, MA.

[30] Chong Luo, Feng Wu, Jun Sun, and Chang Wen Chen. 2009. Compressive data gathering for large-scale wireless sensor networks. In *Proc. of ACM Mobicom 2009*. 145–156.

[31] SigFox Network. 2016. SigFox Technical White Paper. (2016).

[32] Chen Qian, Yunhuai Liu, Hoilun Ngan, and Lionel M Ni. 2010. ASAP: Scalable identification and counting for contactless RFID systems. In *Proc. of IEEE ICDCS 2010*. 52–61.

[33] Hariharan Rahul, Swarun Kumar, and Dina Katabi. 2012. MegaMIMO: Scaling Wireless Capacity with User Demands. In *ACM SIGCOMM 2012*. Helsinki, Finland.

[34] Rapeepat Ratasuk, Benny Vejlgaard, Nitin Mangalvedhe, and Amitava Ghosh. 2016. NB-IoT system for M2M communication. In *Proc. of IEEE WCNC 2016*. 1–5.

[35] Ettus Research. 2012. USRP N210 Datasheet. (2012). Available online at https://www.ettus.com/content/files/07495_Ettus_N200-210_DS_Flyer_HR_1.pdf.

[36] Anthony Rowe, Dhiraj Goel, and Raj Rajkumar. 2007. Firefly mosaic: A vision-enabled wireless sensor networking system. In *Proc. of IEEE RTSS 2007*. 459–468.

[37] Ramon Sanchez-Iborra and Maria-Dolores Cano. 2016. State of the Art in LP-WAN Solutions for Industrial IoT Services. *Sensors* 16, 5 (2016), 708.

[38] Wei-Liang Shen, Kate Ching-Ju Lin, Shyamnath Gollakota, and Ming-Syan Chen. 2014. Rate adaptation for 802.11 multiuser MIMO networks. *IEEE Transactions on Mobile Computing* 13, 1 (2014), 35–47.

[39] Clayton Shepard, Hang Yu, Narendra Anand, Erran Li, Thomas Marzetta, Richard Yang, and Lin Zhong. 2012. Argos: Practical Many-antenna Base Stations. In *Proc. of the ACM Mobicom 2012, Istanbul, Turkey*. ACM, 53–64.

[40] Kun Tan, He Liu, Ji Fang, Wei Wang, Jiansong Zhang, Mi Chen, and Geoffrey M Voelker. 2009. SAM: enabling practical spatial multiple access in wireless LAN. In *Proc. of ACM Mobicom 2009, Beijing, China*. ACM, 49–60.

[41] Joël Toussaint, Nancy El Rachkidy, and Alexandre Guitton. 2016. Performance analysis of the on-the-air activation in LoRaWAN. In *Proc. of IEEE IEMCON 2016*. 1–7.

[42] James E Vander Mey and Timothy J Vander Mey. 1992. Spread spectrum communications system for networks. (Feb. 18 1992). US Patent 5,090,024.

[43] Andrew J. Viterbi. 1995. *CDMA: Principles of Spread Spectrum Communication*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA.

[44] Jue Wang, Haitham Hassanieh, Dina Katabi, and Piotr Indyk. 2012. Efficient and reliable low-power backscatter networks. In *Proc. of ACM Sigcomm 2012*. 61–72.

[45] Yafeng Wu, Gang Zhou, and John A Stankovic. 2010. Acr: Active collision recovery in dense wireless sensor networks. In *Proc. of IEEE INFOCOM 2010*. IEEE, 1–9.

[46] Zhengwen Yang, Qiang Wu, Yongqiang Lu, Pei Lu, Yinghong Hou, and Manman Peng. 2012. Enhanced ALOHA algorithm for chirp spread spectrum positioning. In *Joint International Conference on Pervasive Computing and the Networked World*. Springer, 891–903.

[47] Hang Yu, Lin Zhong, Ashutosh Sabharwal, and David Kao. 2011. Beamforming on Mobile Devices: A First Study. In *Proc. of ACM Mobicom 2011, Las Vegas, Nevada, USA*. ACM, 265–276.