Quick (and Dirty) Aggregate Queries on Low-Power WANs

Akshay Gadre
Carnegie Mellon University
agadre@andrew.cmu.edu

Fan Yi
Princeton University
fanyi@princeton.edu

Anthony Rowe
Carnegie Mellon University
agr@ece.cmu.edu

Bob Iannucci
Carnegie Mellon University
bob@sv.cmu.edu

Swarun Kumar
Carnegie Mellon University
swarun@cmu.edu

ABSTRACT

Low-Power Wide-Area Networks (LP-WANs) are seeing wide-spread deployments connecting millions of sensors, each powered by a ten-year AA battery to radio infrastructure, often miles away. By design, iteratively querying all sensors in an LP-WAN may take several hours or even days, given the stringent battery limits of client radios. This precludes obtaining even an approximate real-time view of sensed information across LP-WAN devices over a large area, say in the event of a disaster, fault or simply for diagnostics.

This paper presents QuAiL[1], a system that provides a coarse aggregate view of sensed data across LP-WAN devices over a wide-area within a time span of just one LP-WAN packet. QuAiL achieves this by coordinating multiple LP-WAN radios to transmit their information synchronously in time and frequency despite their power constraints. We design each client's transmission so that the base station can retrieve an approximate heatmap of sensed data by exploiting the spatial correlation of this data across clients. We further show how our system can be optimized for statistical and machine learning queries, all while maintaining the security and privacy of sensed data from individual clients. Our deployment over a 3 sq. km. LP-WAN deployment around CMU campus in Pittsburgh demonstrates a 4× faster information retrieval versus the state-of-the-art statistical methods to retrieve the spatial sensor heatmap at a desired resolution.

CCS CONCEPTS

• **Computer systems organization** → **Sensor networks**; • **Networks** → *Network protocol design.*

KEYWORDS

LPWAN, Aggregate Queries, sensor networks, machine learning

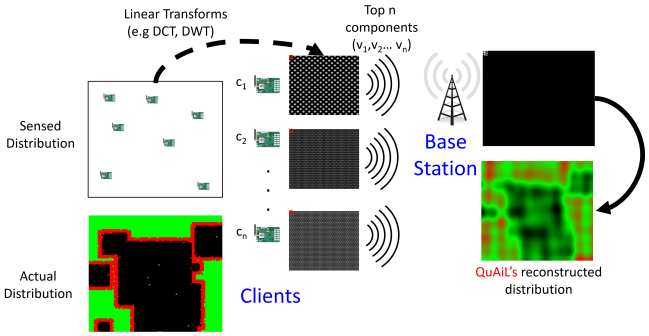


Figure 1: QuAiL achieves quick estimation of spatial distribution of sensed data across clients

1 INTRODUCTION

Dost thou love life? Then do not squander time, for that is the stuff life is made of – Benjamin Franklin

"How long does it take to query a million low-power temperature sensors during a forest fire to send out real-time evacuation alerts?". Questions like this one pose a challenge for current Low-Power Wide-Area Networks (LP-WANs), with fast expanding nationwide deployments in U.S., China and much of Europe [1, 30]. Today's LP-WAN radios are designed to transmit at low data-rate to the base station several miles away, providing up to 10 years of battery life on a AA battery. The battery constraints of LP-WAN radios necessitate an extremely slow data rate from sensors they are attached to – with each message lasting as long as several seconds [24]. This means that querying a large number of low-power devices in a city is a process that can take several hours to days. While this limitation is reasonable for many common LP-WAN applications that are latency-insensitive (e.g. monthly metering, infrastructure monitoring, environmental sensing, etc.), one might occasionally wish to query the aggregate view of large numbers of LP-WAN clients in real-time to gather diagnostics, particularly in the event of a disaster, fault or any rapidly evolving event. In this paper, we ask: "Can we at least build an approximate view of sensed data of LP-WAN radios in a city within a deadline of a few seconds – trading off resolution and accuracy in favor of latency?". As a motivating example, consider a wide-area deployment of temperature sensors that we wish to monitor in real-time during a forest fire – such as the recent California wildfires and Australia bushfires. In its simplest form, we might be interested in short statistical summaries, such

[1] Quick Aggregation in LP-WANs

as the count of number of sensors online, the mean of the sensed values such as average temperature. More generally, we may seek a spatial map of the approximate sensed value at different regions of the environment, say thermal maps. In several instances, these maps may need to be optimized for machine learning inference models that operate on the aggregate sensed view, say to declare different levels of evacuation emergency. Current state-of-the-art approaches to approximately answer such aggregate queries using statistical sampling [36] or sparse recovery [14] would still require us to query individual LP-WAN sensors, effectively taking minutes to hours, even for a modest numbers of sensors queried [13, 24].

This paper presents QuAiL, a system that responds to aggregate queries from a large number of LP-WAN nodes with minimal loss of resolution in a time-span as short as one LP-WAN packet (i.e., a few seconds at most). QuAiL enables base stations to simultaneously find approximate responses at low latency to several types of queries on aggregate sensed data such as simple statistics, spatial distributions and machine learning queries. We show how our techniques are broadly applicable to a variety of popular LP-WAN technologies – LoRa and NB-IoT. We implement and evaluate QuAiL on a 3 sq.km. pilot campus-scale deployment around CMU. Our experiments reveal a 4× faster information retrieval with 2.36× less error over sparse sampling solutions for network-scale inference.

QuAiL's approach is best understood by revisiting our forest fire example (see Fig. 1). Suppose we wish to quickly query wide-area LP-WAN enabled temperature sensors to obtain a spatial heatmap of the current impact of the forest fire. QuAiL relies on the fact that most sensor heatmaps have a high degree of spatial correlation and therefore sparse in linear domains such as the Discrete Cosine Transform – a principle used in JPEG image compression [4]. QuAiL then seeks to recover the top-n most significant non-zero terms of this linear domain quickly from distributed low-power sensors, where n is the smallest number that can be used to recover an approximate view of the sensed data at the desired resolution. We make each sensor locally compute its contribution to each of these n terms: $(v_1, \ldots, v_n)$. Each sensor then concurrently transmits n orthogonal codes at powers weighted precisely by $(v_1, \ldots, v_n)$ at the same time and frequency. QuAiL then relies on the fact that wireless signals transmitted concurrently across sensors will add up linearly at the base station. We invert the linear-transform at the base station to obtain the approximate spatial heatmap.

The rest of this paper deals with the challenges and opportunities stemming from the above in the LP-WAN context. On the challenges front, we first aim to ensure that extremely low-power signals are synchronized concurrently in time and frequency. We must do so because the radios are extremely low-power and narrowband, and therefore inherently prone to such errors. Second, we need to account for the wireless channels between the clients and the base station, which could alter the weights with which wireless signals add up at the base station. We need to do this without requiring clients to transmit individual preambles for channel equalization, which is simply too latency-intensive. Third, we must ensure that our approach can detect failures and has mechanisms to elastically improve resolution with looser latency constraints. Sec. 4 describes our approach for synchronized collaborative encoding of LP-WAN client data, as well as our choice of orthogonal codes compatible with popular LP-WAN technologies.

Next, Sec. 5 describes the varied opportunities for posing aggregate queries that our design enables by carefully choosing the linear domain in which the sensed data is expressed by individual LP-WAN nodes. We show how our system allows for simple statistics such as sum, mean and count of sensors in the network. We also study system performance with different choices of linear domains such as DCT, DWT and DFT on different classes of sensed data. Further, we show how our approach naturally fits to answering machine learning inference queries for a large class of models that have a linear initial phase such as SVM and neural networks.

Finally, we demonstrate the attractive security and privacy properties that can be achieved by randomizing the weights applied by individual sensors. We show how passive eavesdroppers cannot infer the underlying sensed data without access to these weights. We also present differential privacy guarantees that limits the extent to which even the legitimate base station can infer both the location or specific data of individual sensors by processing the received linear combination. Sec. 6 describes our approach.

Limitations: We emphasize a few important limitations of QuAiL: (1) Our approach cannot retrieve measurements from sensors in deep sleep modes. However, for those nodes that are able to transmit and receive on demand, we show significant latency improvements in aggregate queries over the state-of-the-art. (2) Our approach can be impacted by noise and interference over a wireless medium. We therefore develop fall-back mechanisms to detect failure through custom acknowledgments and checksums. We discuss and evaluate these limitations in Sec. 7 and Sec. 8.

We implement QuAiL in the ISM band on FSK and LoRa radios due to ready availability of off-the-shelf hardware. We use Semtech SX1276 clients as the NB-IoT/LoRaWAN clients. We use NI USRP N210 software radios as our base stations. We deploy 20 clients moved across an area of 3 sq. km and placed at over 30,000 unique locations around CMU. We perform real-time feasibility study on concurrent transmissions of up to 10 clients and emulate collisions of up to 10,000 client transmissions using traces across our clients measured at different time instances and locations. We then perform large scale trace-driven evaluation for forest-fires to test the efficacy of QuAiL in the motivated example. Finally, we evaluate ability of QuAiL to compute statistics and machine learning inputs on two public sensor databases – Occupancy dataset [8] and Intel-Berkeley dataset [25]. Our results show:

- A mean accuracy of 96.98% in computing the mean of sensed data under a 589 ms time constraint.
- A mean latency reduction of 4× in recovering the spatial distribution at a resolution, when compared to individual querying of sensors, sub-sampled to the same resolution.
- A mean accuracy of 3.49% for inference using neural networks with associated guarantees on privacy of data.

Contributions: Our main contributions include:

- A mechanism to provide an approximate real-time view of sensed data within one packet duration in LP-WANs by engineering specially-designed concurrent transmissions compatible with common LP-WAN technologies
- A set of system security and differential privacy guarantees.
- A deployment at CMU demonstrating limited loss of resolution amidst strict latency constraints for varied statistical, spatial distribution and machine learning inference queries.

2 RELATED WORK

Related work falls broadly in three categories:

Low-Power WANs: LP-WAN deployments in both licensed (NB-IoT [3]) and unlicensed (LoRaWAN [24]) frequencies have witnessed rapid deployment globally [1, 30]. Recent research efforts on LP-WANs have developed novel solutions for synchronization [5, 32], association [13, 22], optimizing power [6, 10], improving scalability [11, 34], and client power adaptation [42]. To our best knowledge, obtaining a low-latency aggregate view across LP-WAN clients is a problem unaddressed by prior work, and is the focus of this paper. We also highlight how QuAiL can be used complementary to parallel decoding techniques to improve it's performance further.

Data Aggregation and Compressive Sensing in Sensor Networks: There has been much work on machine learning inference based on sensed data [15, 18, 28]. Smart cities have deployed city scale sensing applications such as environmental monitoring [37], precision irrigation for parks [35], and other participatory sensing applications[17]. Past work has developed rich machine learning [35, 37] and statistical inference models [14, 36] that operate on large volumes of aggregate data across sensors. QuAiL complements these solutions by providing an approximate view of sensed data for statistical or machine learning inference that can be obtained without exhaustively and individually querying sensors. Perhaps closest to QuAiL is prior work that leverages sparse recovery techniques such as compressive sensing in traditional wireless sensor networks [7] where clients perform pre-processing to reduce communicated information. In contrast, QuAiL relies on concurrent transmissions of low-power radios to recover an approximate view of sensed information, within the stringent latency constraint of one LP-WAN packet.

Wireless Network Coding and Analog Video Coding: Analog coding in the air using wireless signals has been deeply studied by wireless researchers over multiple decades [33] including leveraging multiple-antennas [16, 23]. There has also been extensive work done on developing novel data accumulation techniques by clever encoding mechanisms[31] and optimal forwarding to minimize energy expenditure[38]. Other researchers have used various other lossy coding schemes[43] to optimize for energy by cutting redundancy. QuAiL is also related to past work [19, 27] that leverages sparsity of video data to code information in analog, albeit on point-to-point links. While QuAiL builds upon these solutions, it specifically focuses on aggregation over linear sparse domains of the underlying phenomenon to reduce the time and bandwidth required to query LP-WAN clients at scale. To achieve this, QuAiL solves LP-WAN specific challenges enabling retrieval of linear combinations over the air for massive city-scale IoT deployments.

3 QUAIL OVERVIEW

QuAiL's main task is to get a coarse estimate of spatial distribution of sensed information in an area at low latency. A key factor in achieving this is to decode clients' information at scale within the stringent time constraint of one packet. The natural approach would be to make all the clients transmit at the same time and decode as many packets as possible from the resulting collisions. Yet colliding packets suffer incredibly high packet error rates to retrieve any useful information. Instead, QuAiL aggregates information from sensors by cleverly encoding sensed information within individual packets, so that the resulting collision can still be processed to obtain the spatial distribution of sensed data.

Our approach relies on a very simple principle: at scale, received signal power from a large number of transmitting nodes would simply be a linear combination of the transmitted powers of individual sensors, subject to noise. Let p_i denote the power of the signal transmitted by sensor i and w_i denote the power of the wireless channel between the client and a base station. Then the received power from a collision of N sensors at the base station is simply:

$$y = \sum_{i=1}^{N} w_i p_i + \text{noise}$$

QuAiL's approach is to design the power of each client p_i such that it maximizes the recovery of desired spatial distribution – denoted by Y at the base station. Let us assume the client achieves this by applying setting $p_i = \Phi(x_i)$ where x_i is its sensed data and $\Phi(.)$ is a function known to all sensors and the base station. We then assume that the base station can recover its desired spatial distribution by applying a different function: $Y = \Omega(y)$. Mathematically, we write:

$$Y = \Omega(\sum_{i=1}^{N} w_i \Phi(x_i)) + \text{noise}$$

The rest of this paper describes the key challenges in making such a design practical:

(1) Synchronization: First, the above approach assumes that transmissions across low-power clients are synchronized in time and frequency. Yet, low-power clients have large frequency and timing offsets that would render the linear combination highly susceptible to errors. Second, the weights w_i due to the wireless channels between the clients and base station may not be known upfront at the base station, and vary rapidly over time. Exhaustively computing them for each client through a priori beacons would defeat the strict latency constraints of the system. Third, variation in noise can lead to incorrect measurements of spatial distribution based on various wireless impairments such as spurious transmissions, variation in wireless channels, incorrect compensation of frequency offsets. Finally, many applications might require the locally computed function Φ to result in negative values of power which do not make sense. QuAiL's solutions to the above problems along with making the system compatible with both NB-IoT and LoRa are detailed in Sec. 4.

(2) Designing Φ and Ω: Next, we need to design functions Φ and Ω to retrieve useful information from a large number of sensors. While statistical inferences require us to identify the right way to combine client information, more complex inferences over spatial distributions rely on sparsity in spatial domains. Further, many machine learning algorithms such as SVMs and neural networks operating on spatial distributions require weighted aggregation of pixels of the resultant image. Sec. 5 demonstrates a general framework for encoding nested affine functions on sensed information for the above applications and analyzes common transform domains.

(3) Security and Privacy: A key problem of encoding information in transmitted power would be security as any malicious adversary would be able to snoop over a client's data. Thus, there remains a

necessity to obfuscate client information from the malicious adversary without affecting the quality at the base station. Towards this end, QuAiL shows how by randomly spreading energy across the available bandwidth, we can provide security of client information and ensure anonymization and privacy of client's data (Sec. 6).

4 SYNCHRONOUS COLLABORATIVE ENCODING

In this section, we will describe how we can efficiently and robustly acquire linear combinations of client sensed data encoded in power in a sparse domain by tackling wireless impairments such as the wireless channel, timing and frequency offsets, noise at the clients and preserve compatibility with both commonly used LP-WAN technologies, namely NB-IoT and LoRa.

4.1 Compensating for the Wireless Channel and Hardware Impediments

At the base station, our objective is to obtain the accurate sum of the transmit powers of individual clients, as they transmit synchronously in time and frequency. Yet, in practice, signals from individual clients are weighted by the unknown wireless channel between the client and base station. Further, transmissions may not be perfectly synchronized in time and frequency. In this section, we describe our approach to mitigate both of these challenges.

Channel Compensation: A strawman approach to channel compensation is to simply ask each client to send a short preamble to the base station to calibrate for wireless channels up-front. However, doing so would entail long overall latency, particularly given that LP-WAN symbols are long and channel feedback would need to be done on a per-client basis. In contrast, QuAiL leverages a simple concept: channel reciprocity. Prior to synchronized transmissions by clients, the base station broadcasts a short beacon listened to by all clients. The clients estimate the received power of this transmission and calibrate their transmit power.

An important challenge in achieving this design is that common LP-WAN hardware may not provide the ability to manipulate transmit powers at fine-granularity. To put this in perspective, consider SX1276, a common LoRaWAN chipset that allows manipulating transmit powers only in steps of 1 dB, limiting the resolution at which sensed data can be sent. As a result, instead of encoding information in signal power, we instead manipulate the duration of the signal. This duration can be manipulated at much finer granularity enabling clients to communicate sensed information at fine-granularity. At the base station, we then measure the total power received over a specific time window to glean the total received power across clients(Fig. 2). Mathematically, let say the client needs to reduce it's transmit power by [x].{y}dB for compensation of wireless channel and T is the time of the known signal. Then the transmit power (P_{TX}) and transmit duration (T_{TX}) can be selected as follows:

$$P_{TX} = P_{TX} - x \qquad T_{TX} = T/10^{\frac{y}{10}}$$

A second key challenge is how do the clients know what energy to transmit based on the downlink channel. First, the clients normalize their sensed value between 0 and 1. They then encode their energy onto the symbol using the same mechanism as above. They, then reduce their power to such a level so that it's received power

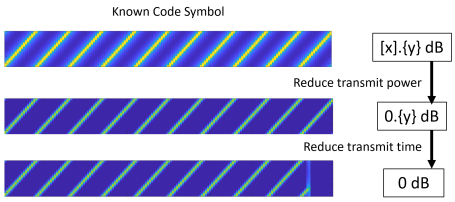


Figure 2: Channel Compensation in QuAiL

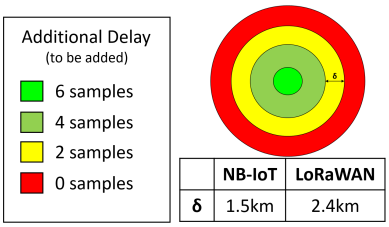


Figure 3: Timing Compensation in QuAiL

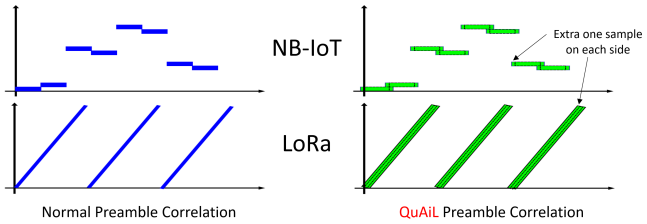


Figure 4: Thicker correlation preambles allow for small frequency and timing offsets in QuAiL

at the base station is same as the client at farthest location transmitting at maximum power. This allows all signals to be above noise as well as of equal power. Another problem is that uplink channels in cellular are different from downlink channels. Thus, QuAiL uses a discrete time Markov chain to model the uplink channel based on previous downlink and uplink channel measurements. Indeed, the uplink channel measurements will be known at only a few frequencies (via feedback sent in NB-IoT packets), while the downlink channel can be only measured from the base station's broadcast channel. Our model relies on a key assumption that holds for a narrowband channel: While the phase across subcarriers changes even across small bandwidth, the absolute value of the channel response is relatively flat across a bandwidth of 200KHz [13].

Impact of Offsets: QuAiL synchronizes clients in time by scheduling client transmissions at a fixed interval following the query beacon transmitted by the base station. Yet, this approach is susceptible to timing offsets owing to the differences in clocks and detection times between low-power clients. Further, clients may be at different distances relative to the base station, causing differences in propagation delay as well. However, QuAiL's approach to rely on average power of the received signal over a time window makes it naturally robust to timing offsets across clients. Specifically, we design QuAiL to average signal power from clients at the base station over a time window of 8.192 ms. We design this window to be significantly higher than the timing jitter of clients of popular LP-WAN technologies (see Sec. 8.1). We also compensate for the

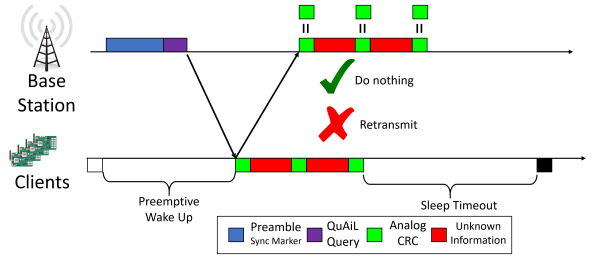


Figure 5: QuAiL's approach to achieve robustness in presence of noise

propagation delays of LP-WAN clients by delaying response from closer packets as shown in Fig. 3.

The main source of frequency offsets are the inaccurate client clocks causing a center frequency drift across time. Normally, NB-IoT and LoRa clients use offset-resilient hopping and chirp codes for base station to be able to decode them at significant distances. QuAiL's architecture requires significantly more accurate frequency synchronization to achieve addition of powers. Prior approaches have relied on one way communication using specialized client devices[9] to frequency synchronize clients over large distances. However, LP-WAN clients are not designed to correct their own frequency offsets (instead their modulation are precisely designed to avoid it!). Thus as SNR degrades, we lose some signal quality leading to errors as shown in Sec. 8.2.

4.2 Robustness to Noise

A key challenge in our approach is to ascertain whether the received linear combination at the base station is accurate, despite noise and any errors in channel or frequency offset estimation. With all the clients transmitting simultaneously, it would be impossible to ascertain the effect of a single client independently. Thus, there is a need for the base station to be able to validate the accuracy of our solution amidst noise, interference and estimation errors.

Traditional wireless protocols communicating over point-to-point links send a CRC which allows the base station to detect the presence of packet errors and require retransmission. Clearly, sending individual CRCs or checksums in our context would revert to simply querying every client and is therefore infeasible.

Instead, QuAiL designs a distributed error detection code, an analog CRC, that allows the base station to ascertain the correctness of the received information. Indeed, as the information we are trying to retrieve is the distribution of sensed value and not the individual values, our CRC should operate on the distribution. We encode a known distribution in the clients during their initialization phase to enable the CRC. To illustrate, let us suppose we want to compute $Y = \Omega(\sum_{i=1}^{N} w_i \Phi(x_i))$ where $X = \{x_1 \dots x_N\}$ are the sensed values of clients. We can then require the clients to first transmit X=A where $A = \{a_1 \dots a_N\}$ is a known matrix. This would lead to a deterministic value at the base station which can be used to verify whether the clients were able to compensate for the above offsets. For every unknown block of information (red in Fig. 5), the base station would verify the adjacent CRC blocks (green in Fig. 5). If either of the CRC blocks have an error beyond a fixed threshold, the query is retransmitted. The approach for detection and retransmission is shown in Fig. 5.

4.3 Compatibility with LoRa and NB-IoT

In this section, we design mechanisms to make QuAiL compatible with common LP-WAN protocols: LoRa and NB-IoT, without client hardware modification.

Engineering Collisions and Designing Codes: Our first challenge is that LP-WAN technologies are designed to specifically avoid collisions. While engineering collisions in LoRa is relatively simple, given that it operates on shared unlicensed spectrum where collisions are common, doing so for NB-IoT where collisions are explicitly scheduled to be avoided is much more challenging

However, there exists a set of frequencies where even NB-IoT clients can collide – the random access channel (RACH). Indeed, it is in this channel that clients vie for the spectrum resources to communicate. Further, the RACH channel allows clients to choose between specific well-defined codes (frequency hopping patterns) that are mutually orthogonal. This allows QuAiL to transmit multiple linear combinations simultaneously within one symbol. We note that QuAiL can similarly transmit multiple linear combinations in parallel in the LoRa context by choosing different frequencies of operation within the unlicensed band. The limit of number of codes that can be fit within 180kHz wide band is evaluated in Sec. 8.3.

Negative Powers: In our discussion so far, we have only allowed a mechanism to add up power linearly across clients. However, there is a key problem with this assumption: Most useful weighted linear combinations have negative weights, sometimes requiring us to measure a negative quantity. Yet, powers are positive numbers, and therefore always add up – not subtract out. Our solution to this leverages the fact that we can send independent linear combinations simultaneously through code or frequency division multiplexing. Our solution chooses to divide the available codes into two groups: one half for positive weights and one for negative weights. The base station can then subtract energies received along the negative code from the positive code to retrieve the required linear combination.

To illustrate this mathematically in the NB-IoT context, let us divide the n available codes of the RACH into two equal sets: $n/2$ positive codes – $\{c_j, j = 1, \dots, n/2\}$ and $n/2$ negative codes – $\{c_j, j = n/2 + 1, \dots, n\}$. Let us assume l_{ij} are the set of weights from clients $i = 1, \dots, m$ and up to $n/2$ target linear combinations $j = 1, \dots, n/2$. Then we can set:

$$w_{ij} = \begin{cases} l_{ij}, & l_{ij} \geq 0 \text{ and } j \leq n/2 \\ -l_{i(j-n/2)}, & l_{i(j-n/2)} < 0 \text{ and } j > n/2 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

5 ENABLING APPLICATIONS USING QUAIL

In this section, we describe how we can compute various aggregates required by many applications such as statistics, spatial distributions and weighted linear combinations for machine learning.

5.1 Statistics, Percentiles and Histograms

Traditional statistics can be typically be divided into 3 categories: single measured quantities, percentiles and distributions. Single measured quantities typically can be analyzed as map-reduce functions of sensed information with the constraint that reduce uses a sum. For example, a client can individually compute it's own average temperature, pressure and current consumption. Then, it can

communicate it to the base station where it can compute the combined average of the ensemble of sensors. Table 1 shows the client and server side functions for various commonly measured aggregates. In QuAiL's architecture, we can replace the $\ldots w_i\Phi(x_i)$ with client-side functions and the $\Omega(\sum_{i=1}^{N}\ldots$ with server side functions above to compute the statistics.

Percentiles and histograms pose an interesting challenge as the clients don't know where they lie in the group of clients being queried. A näive approach would be to keep querying clients greater than a certain measured value until you converge. However, this remains too latency intensive to fit in our architecture. Instead, we fill the available bandwidth with a large number of thin codes, more susceptible to frequency offsets yet expressing information in higher granularity. We can then ask clients to communicate in code j out of N codes as follows:

$$\Omega = I; w_i = 1; \Phi_j = \begin{cases} I, & x_i \in H_j \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where H_j is the j^{th} bin of a histogram. The above will allow an user to compute histogram as well as search for percentiles with high accuracy at low latency.

5.2 Spatial Distributions

Many inference tasks such as estimating location of a forest fire or tagging locations require spatial distribution of sensed information such as pollutant content and temperature. With large number of clients the underlying spatial information will be highly fine-grained and accurate. However, even if we assume the clients know their locations, it remains a challenging task to retrieve this information required to estimate such distributions due to scale.

A näive approach to model it in QuAiL's architecture would be to map each code c_j to one pixel and retrieve information from all clients in that pixel at that code. However, to even retrieve a 256× 256 image, this would require 65536 codes compressed into a narrow bandwidth. To reduce the number of codes required, we borrow from existing image compression literature. Most commonly used image compression algorithms like Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) are both linear[40, 41]. Further, we assume that clients know their relative location in the grid and can therefore choose appropriate weights to emulate the the above transforms. Let's say $x_{i,j}$ are the pixels of the image with x'_k a MN×1 linearized version of the image. For example, the DCT of an image M×N is simply:

$$DCT_{m,n} = \sum_{i=1}^{M}\sum_{j=1}^{N} x_{i,j} cos\Big(\frac{\pi}{M}(i+\frac{1}{2})m\Big) cos\Big(\frac{\pi}{N}(j+\frac{1}{2})n\Big)$$

$$= \sum_{i=1,j=1}^{M,N} cos\Big(\frac{\pi}{M}(i+\frac{1}{2})m\Big) cos\Big(\frac{\pi}{N}(j+\frac{1}{2})n\Big) x_{i,j} = \sum_{k=1}^{k\in M\times N} \Phi_{m,n}(x'_k)$$

This can also be represented as

$$DCT_{m,n} = DCTMAT(m,n)^{1\times MN} x'^{MN\times 1}$$

Note that clients could be provided information on grid size and their location either through calibration from the base station, or as measured from the base station. Remember that the knowledge of physical location is required due to the fact the sparse domain of our choice relies on relative location of clients. Thus, there might exist sparse domains that may work even without this knowledge.

Function	Client Side	Server Side	QuAiL eligible?
Mean	Data energy in one code	Divide energy by number of clients	Yes
Median	Histogram	Estimate median	Maybe
Mode	Histogram	Estimate mode	Maybe
Variance	Send $\|x\|^2$ and $\|x\|$ in two codes	Compute $\overline{\|x\|^2} - \overline{\|x\|}^2$	Yes
Sum	|x| in one code	Total Energy	Yes
Count	|1| in one code	Total Energy	Yes

Table 1: QuAiL's approach to calculate statistics

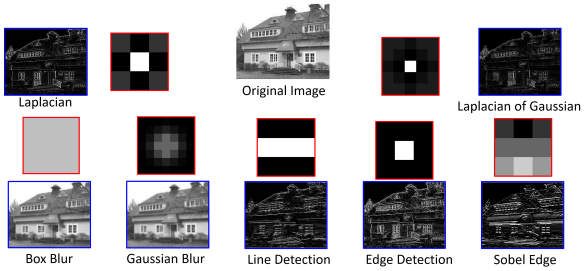


Figure 6: Convolution Filters used in image processing

Convolution Filters: To process spatial data, akin to images, one often relies on some form of filtering. Indeed, the ability of convolution filters to detect edges, prominent features and gradients across directions have seen them being widely used in image processing and an initial step for a whole class of machine learning algorithms: Convolutional Neural Networks (CNNs). Fig. 6 shows some of the common filters used for identifying various features.

A key advantage of QuAiL's architecture is that we can compute approximates of these convolutions as well in one go. There are two ways we can achieve this. First way would be to estimate the the underlying image in a sparse domain and then performing the convolution manually. Yet many filters rely on fine grained information of slopes and edges which is typically lost in sparse approximations. Instead, QuAiL takes a different approach. Observe that convolution filters are simply linear combinations of pixels across locations. This allows us to formulate the convolution with matrix F as:

$$Z = FMAT^{MN\times MN} x'^{MN\times 1}$$

Rewriting the DCT of above matrix multiplication, we get

$$DCTMAT(m,n)^{1\times MN} FMAT^{MN\times MN} x^{MN\times 1} = \sum_{k=1}^{k\in M\times N} \Phi'_{m,n}(x'_k)$$

Thus, QuAiL allows sparse retrieval of convolutions of underlying distribution of sensed data within a single query.

5.3 Machine Learning Algorithms

In this section, we describe how QuAiL can enable various machine learning algorithms on the sensed information of the clients even without relying on their data sparsity. A key aspect of QuAiL is that it can readily obtain linear combinations of the clients' sensed

information at low latency. In this regard, we make the key observation that most supervised machine learning algorithms – the first layer is a weighted linear combination of sensed information.

By modelling the first layer of machine learning algorithms, we can enable base stations to compute the required inferences. Most machine learning algorithms rely on the fact that the input vector is synchronous, or in simpler words, the underlying sparse field does not change drastically over the duration of retrieving the information from the clients. While this quasi-static assumption is true for most conventional wired and wireless sensors, a base station may require a few minutes to hours for querying a large number of sensors. QuAiL breaks this bottleneck by measuring quasi-static aggregates of large scale LP-WAN deployments within the duration of a few packets. Further, QuAiL enables client data and location privacy by a special mechanism detailed in Sec. 6.

Modelling a machine learning algorithm: The weights to clients can be sent over the air during their usual uplink communication (latency of doing this is discussed in Sec. 7). These weights can be then used during QuAiL queries to get the required linear combination. For example, let's say a neuron (η_i) in the first hidden layer, the smallest unit in a neural network, requires a weighted sum as follows: $w_i x_i + b_i$. We first identify the positive and negative weights.

$$\Phi_i^+ = \begin{cases} I, & w_i > 0 \\ 0, & \text{otherwise} \end{cases} \quad \Phi_i^- = \begin{cases} I, & w_i < 0 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Then, the base station can perform two queries to retrieve $w_i\Phi_i^+(x_i)$ and $w_i\Phi_i^-(x_i)$ which in turn can be subtracted to retrieve $w_i x_i$. The bias (b_i) can be added at the base station.

Learning the weights for machine learning: A key component of any machine learning algorithm relies on learning the weights of it's various layers on test data and changing them over time. Yet, performing such an latency-intensive task would be too power and time-consuming for clients. This raises an important question of training an machine learning algorithm catering to LP-WAN tasks.

While real-time training is a daunting task, remember that as clients last multiple years, we would likely have large traces of historical data available for training. This can allow offline training of machine learning algorithms for catering to LP-WAN clients. Indeed, latency is most often a problem only when performing machine learning inference and not during training.

Importance of regularization: Learning on large swaths of data may lead to highly variable weights leading to vastly different contributions from multiple clients. This may thus lead to high variation in power across time leading to ADC saturation. We present in Sec. 7 a mechanism that allows us to avoid it. Traditionally, machine learning algorithms are also regularized to avoid overfitting on the training data. Specifically, they do so by not allowing the weights to take unnaturally large values. This ensures that behavior of noisy input is not misclassified by the ML. This is done by an additional factor(λ) in the convex optimization function:

$$w^* = \underset{w}{arg\min} \sum_{i}^{N} ||y_i - f(x_i)||_2^2 + \lambda \sum_{i,j} w_{ij}^2$$

6 SECURITY AND PRIVACY VIA QUAIL

In this section, we will describe how we can provide security and privacy bounds for urban scenarios for client data. Security and privacy of data is required by many applications such as statistics, spatial distributions and weighted linear combinations for machine learning. For the sake of simplicity, we will restrict our discussion to retrieving weighted linear combinations of sensed information.

6.1 Security

We define **security** in QuAiL's architecture as the ability of the base station or a malicious adversary to measure a different weighted linear combination from a given weighted linear combination. We consider a passive adversary close to the base station that receives signals over the air at high fidelity. We assume that all base station queries are encrypted and the adversary has no access to private information of clients and base stations. We also assume that the number of clients far exceeds the number of codes and the fractional power received from a single client on a given code relative to the total power received on the code is negligible.

QuAiL's security properties stem from a randomized matrix $M_{n\times n}$ applied to the set of weights $W_{n\times m}$. This matrix is initialized by using a known seed at both clients and base station. Let's say QuAiL deployment has m sensors then the weight applied by the sensors is MW. Remember that a matrix with random values is almost always full-rank [12]. This randomization matrix provides two benefits: (1) Security of client information by only allowing the base station with the knowledge of M to decode the aggregates. (2) Another attractive property of this randomization is it's ability to spread energy across the codes being used to communicate. As described in Sec. 7, this improves the performance of QuAiL in presence of base stations with ADC constraints.

Recall that for any sensed input $x_{m\times 1}$, an adversary perceives the matrix MWx. Note that for all our applications, $n < m$. In this context, it is easy to see that our system is secure provided the adversary cannot infer Rx for some known rank-n matrix $R_{n\times m}$, given MWx and no prior information on M and W, provided R and MW are not equivalent. Below, we show a fundamental reason why this holds: because $m > n$, there exists an infinite number of values of x that produce the same MWx. We prove the following theorem:

Theorem 6.1. *For a fixed W, R (rank-n) and x, there is no function that uniquely maps MWs to Rx, provided R and MW are not equivalent and $MW \neq 0$.*

Proof. It suffices to prove that for any Rx, there is a $x' \neq x$ such that $Rx' \neq Rx$ but $MWx = MWx'$. Let us denote $MW = Q$. Since Q is an imbalanced $n \times m$ matrix, where $n < m$, there exists a non-zero null-space matrix $\tilde{Q}$ of rank $n - m$ and dimensions $(n - m) \times n$ such that $Q\tilde{Q} = 0$. Let z be an arbitrary $m \times 1$ vector. We define $x' = x + \tilde{Q}z$. Clearly $x' \neq x$ since Q is non-zero. It is easy to see that $Rx' \neq Rx$ since $R\tilde{Q} \neq 0$ given that R and Q are not equivalent. It is also easy to see that $Qx = Qx'$ since $Q\tilde{Q} = 0$. Given that there are infinite ways to choose z and therefore x' and Rx', it follows that it is impossible to map Qx uniquely to Rx. □

6.2 Privacy

We define privacy as the inability of the base station to infer the sensed value of a specific user with high probability, even with

Figure 7: Evaluation Testbed : Channels were collected over an 3 km^2 area. Paths show the client location and Dot represents the base station location

complete knowledge of the weights. Our definition follows the Fair Information principles [29] which emphasize minimization of data collected beyond the purpose to which it is collected. We are specifically interested in differential privacy [26], which states that a change in sensed value of one client imposes a negligible change in the distribution of powers perceived at the base station.

Mathematically, let MWx be received amidst additive Gaussian noise with standard deviation σ and written as: $R(x, \sigma) = MWx + \mathcal{N}(0, \sigma^2)^n$. The base station has full knowledge of both M and W. Let us define $\mathbf{C}$ as a set of possible values $M_{n\times n}W_{n\times m}x_{m\times 1}$. Let us assume x_1 and x_2 are two vectors of sensed inputs that differ in one entry (one client). We define (ϵ, δ)-differential privacy as [21, 26]:

$$Pr[R(x_1, \sigma) \in \mathbf{C}] \leq e^{\epsilon} Pr[R(x_2, \sigma) \in \mathbf{C}] + \delta$$

The following theorem states that QuAiL's approach is (ϵ, δ)-differentially private.

Theorem 6.2. *The function $R(x, \sigma) = MWx + \mathcal{N}(0, \sigma^2)^n$, where $\mathcal{N}(0, \sigma^2)^n$ is (ϵ, δ)-differentially private provided:* $\sigma \geq ||W||_2\sqrt{2\ln(2/\delta)}/\epsilon$

Proof. The proof follows directly from differential privacy literature – Proposition-5 in [21], assuming that the noise distribution is Gaussian with standard deviation σ and MW is a $n \times m$ matrix. □

7 DISCUSSION AND LIMITATIONS

In this section, we discuss some of the key issues one may face while deploying QuAiL with current infrastructure and some of the solutions for mitigating them.

Base station ADC saturation: The first obvious issue QuAiL may encounter is that fact that receiving codes with such wide range of powers might overwhelm the ADC. Indeed, even powerful SDRs have ADCs which span across just 16dB of SNR. Thus, it is imperative for QuAiL to normalize the powers or equalize the energy distribution for base stations to even register the received power.

QuAiL resolves this problem by using the randomization matrix M described in Sec. 6.1. Indeed, along with providing exciting security properties, the randomization matrix spreads energy across the codes to retrieve the linear combinations. This leads more comparable powers Yet, powers across codes is also governed by the clients data. With the correct skew of data, one may observe large changes in power which must be handled via smart dynamic AGCs.

Parallel Decoding for LP-WANs: Recent work has exploited the orthogonality of LP-WAN codes to decode multiple packets out of a collision of multiple packets. One such approach Choir[11] shows 6× throughput improvement over conventional LoRaWAN transmissions. QuAiL presents an interesting opportunity in leveraging these approaches to retrieve more accurate and richer aggregate estimates. QuAiL can require clients to encode their energy in frequency shifted codes that can be decoded parallely using above approaches. This will allow QuAiL to retrieve multiple aggregates within the duration of a single packet. Results in Sec.8.3 demonstrate how QuAiL can be actively combined with Choir to achieve superior performance to either scheme individually.

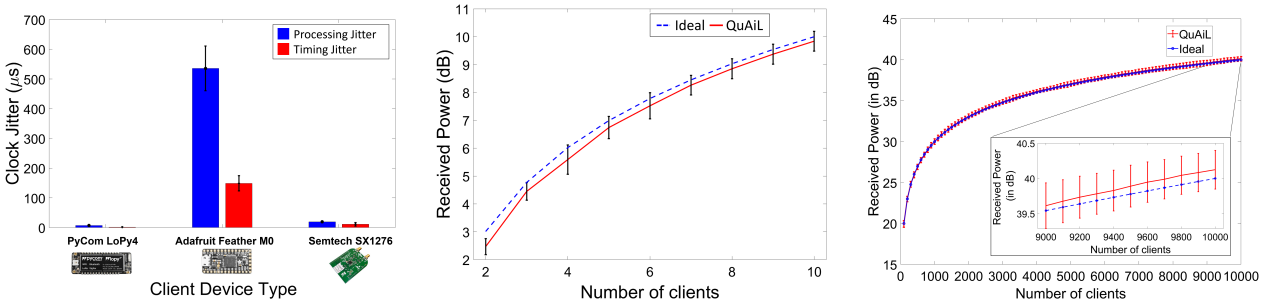
Impact on client battery life: One would think that QuAiL's approach would drain the client battery life due to every client transmitting for every query. Yet, this is not true. QuAiL's approach does not claim to sacrifice the attractive properties of long battery lives provided by LP-WANs in lieu of achieving high latency aggregation. Instead, QuAiL intends to provide an approach which can enable a complimentary ability to query massive number of clients at low-latency with high fidelity to current LP-WAN standards. As the frequency of such queries will be small, the effect on battery life of clients will be marginal.

However, one can make QuAiL more power efficient by asking the clients to reduce their transmit power equally. However, this will make QuAiL more susceptible to spurious noise providing the ability to trade-off accuracy for battery life. Results in Sec. 8.2 demonstrate the resilience of QuAiL in presence of noise.

Impact of errors: QuAiL's approach to recover spatial distributions incurs small errors in estimating individual sensed values. One may wonder how this compares to normal sparse sampling. For example, consider a flooding example. While QuAiL might be a better approach to detect areas which are flooded, one may need to query individual sensors to retrieve the amount of exact flooding in a specific area. Thus, it is recommended to use QuAiL only to recover a coarse view of the underlying sparse distribution. Yet, QuAiL's approach remains useful to get a coarse estimate of an area even with an inoperational sensor from nearby areas' information.

Limitations: While QuAiL presents a complimentary solution to current LP-WAN protocols for massive aggregate queries over large number of clients, QuAiL has the following limitations:

- **Co-existing with non-QuAiL clients, noise and interference:** While results in Sec. 8.2 demonstrate resilience to spurious noise and interference, QuAiL errors do grow with interference and noise beyond a limit. While LP-WAN transmissions are inherently resilient to collisions [11], a non-QuAiL transmission in the same time and frequency can interfere with QuAiL performance. We rely on higher-layer MAC protocols (e.g. ARQ and exponential backoff) to minimize impact of such collisions. While QuAiL may suffer from dynamic channels causing large errors due to mobility, QuAiL's analog CRC will allow QuAiL to identify if the effect on result will be significant. Same holds true even when a client goes to sleep or another client joins the base station.
- **Synchronizing queries and communicating weights:** Our approach, understandably, can only retrieve measurements



(a) (b) (c)

Figure 8: Microbenchmarks: (a) Average processing and timing jitter of various LP-WAN clients (b) Received Power (in dB) with QuAiL deployment of 10 clients (c) Received Power (in dB) for large number of clients using trace-driven emulation

from sensors that are currently not in deep sleep modes, where their RF frontends are switched off until a scheduled future time. Clients will need to be told a rough estimate of time of QuAiL queries and the weights to use for various different applications operating on them . This fundamentally limits the number of different queries that a client can respond to. Instead, we rely on the fact that the client can infer the weights of most non-ML queries by just using their location and received RSSI. This will significantly lower the problem for storing the weights yet remains a critical problem for evolving ML solutions.

8 IMPLEMENTATION AND EVALUATION

We implement QuAiL using FSK and LoRa radios on Semtech SX1276 radios to communicate with an Ettus USRP N210 emulating the base station. We collect more than 150,000 GPS-location and time stamped channel measurements across 3 km^2 area including geographical obstacles such as large buildings, hills, rivers in Pittsburgh. We then emulate collisions using the collected channels to evaluate our system at scale.

We build a light-weight NB-IoT and LoRa QuAiLstack including the NP-RACH in C++/Gnuradio on the USRP base station. Our experiments first train weights (w_{ij}) of the machine learning model under consideration based on raw data. These weights are then provided to the clients for transmitting the appropriate energy across codes. The base station receives the signals at the base stations and measures the power for all the codes to decode them. This decoded vector is used by the neural network for performing the desired inference task. We measure three quantities of interest: (1) Quality of the solution against the ground-truth (pre-labeled data); (2) Latency of communication (3) Benefit over sparse sampling approaches. Error bars in all experiments represent one standard deviation.

Real-time vs. Emulation at Scale: All of our experiments with up to 10 nodes engineer collisions in real-time. Collisions of a larger size (up to 10,000) are emulated using real-channel measurements of our 10 nodes available across 30,000 GPS locations at 5 different frequencies across time stamps[2]. Note that we store the raw I/Q samples which preserve all radio offsets of both the client and the receiver. We add measurements of channels concurrently across devices while preserving radio offsets and adding additive white Gaussian noise to the result based on observed noise distributions at the base station. We consider 20 orthogonal RACH codes and 180 kHz of bandwidth for NB-IoT and 125 kHz of bandwidth for LoRaWAN clients unless specified otherwise. To ensure repeatability, each client transmits sensed data from public datasets (see below).

[2]Data and Code is available at [2]

8.1 Micro-Evaluation

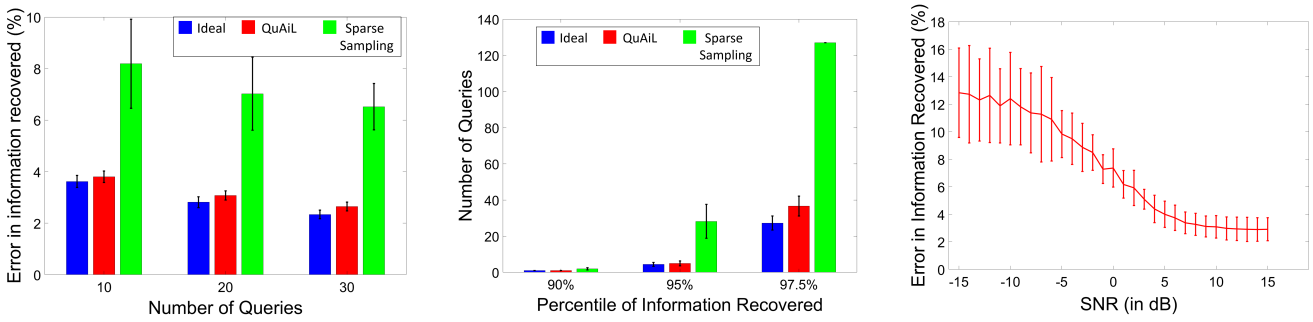
In this section, we evaluate how QuAiL operates with 10 clients simultaneously communicating with the base station.

Timing Jitter: It is a critical assumption in QuAiL that clients are able to compute time with a high enough accuracy in the presence of frequency offsets. Indeed, our channel estimation and timing offset correction abilities rely on the ability of clients to measure time correctly. We evaluate 3 of the popular LoRaWAN clients being widely used by users. Note that our evaluation of NB-IoT clients also uses the SemTech SX1276 chip in FSK mode. We use the sample Ping-Pong code in each client to evaluate the randomness in delay of transmission. We plot this as the processing jitter. We then make one of clients add a known increasing delay across packets and measure the ability of clients to measure time. We measure the average error and it's standard deviation.

Result: One sample at 125 KHz bandwidth is approximately representative of 8μs or 2.4km of distance travelled. We oversample the signal at 10 MHz to detect the variance in the delay received. As we can see in Fig 8a, for 2 of the popular chips the delay is within a limit of 1 sample. However, Adafruit Feather board had a large variance in timing of the signal. We surmise this could be either be due to faults in the specific boards or due to some intrinsic component being less capable. This shows how our correlation preambles (Fig. 4) would be able to capture all the signals.

Powers adding up: The most important result to verify the operation of QuAiL was to verify upon compensating for channels and timing offsets whether clients signals do really add up. We used 10 SX1276 clients in FSK mode sending energy in one NB-IoT subcarrier. Each client attempts to synchronize it's transmit power using approach presented in Sec. 4.1, lowering it to make client signal reach base station at power 3dB less than query signal. We then make the clients join the network one by one across queries.We run these experiments 100 times each. We also use our 150,000 collected channels from 30,000 GPS tagged locations to emulate client collisions upto 10,000 clients.

Results: As you can see in Fig. 8b, the powers of the added up signals add up linearly in received power. While the effect of powers adding up linearly is indeed being followed, we see how that effect is



(a) (b) (c)

Figure 9: Forest Fire Case Study: (a) Error in information recovered across number of queries (b) Number of queries required to recover certain percentile of information (c) Evaluation of QuAiL across various SNRs

randomized at low number of clients (law of large numbers helps). As we use the collected channel data for collisions, we see the randomized effect start to be marginalized and linear adding up of powers takes the precedent as shown in Fig. 8b.

8.2 Case study: Forest Fires

The key motivation for QuAiL was to help detect real-time disaster events such as forest fires, flooding and earthquakes. Yet, there does not exist any deployment that leverages large number of clients to sense information regarding such events (satellite images and specialized instruments are preferred). However, there has been plethora of work on predicting spread of forest fires. In fact, there is a widely accepted model that relies on fractal spreading of fire across large areas. We thus rely on this model[39] to emulate a 10km×10km area with 10,000 clients, each client occupying a cell of 100m×100m. All experiments are repeated over 1,000 times using different channels for each client from our collected channel dataset. Our aim is to retrieve the underlying distribution of burnt, burning and green areas during forest fire. We specifically assign the sensed values as follows: $0 \rightarrow Burnt; 1 \rightarrow Burning; 0.5 \rightarrow Forest$. The values were assigned, assuming a client senses temperature, burning trees will have a higher temperature while clients in far away forest will function normally. Of course, burnt clients cannot transmit. We use two baselines for evaluation: (a) **ideal** scheme where the base station knows all the DCT components; (b) **sparse sampling** scheme where the base station queries one client every query and interpolates the field. Error is measured as every cell predicted incorrectly, normalized over the 10000 cells. Note that ideal scheme also has error bars since the results were averaged across various runs of the forest fire model.

Note that this scenario is particularly designed to favor sparse sampling technique baseline where it can query clients one by one. It simply has to luck out to estimate a burning area. We even allowed it to detect burnt clients as absence of signal. Despite these benefits, a key result of our case study shows that *linear sparse aggregates obtained by QuAiL always perform significantly better than individual sparse sampling.*

Spatial Distribution Error: The first component to evaluate is the ability of approaches to estimate the underlying distribution of the area with fire spread of 3 hours. we evaluate the ability of QuAiL to estimate the underlying spatial field using the DCT domain. We compare QuAiL's performance in computing and retrieving the sparse distribution with the two baselines.

Result: We see in Fig. 9a, that across experiments, as we increase the number of queries the error in estimation of the underlying field decreases for all three approaches. However, the error is still 2.36× worse for sparse sampling over QuAiL due to the constraint of querying individual client. Indeed, upon further investigation, we see that the majority of excess error in sparse sampling is due to over or under-estimation of the fire spread.

Latency of Detection: Another key benefit of QuAiL is that of finding anomalous behaviour in a sparse field faster than sampling individual clients. However, it is critical to get the last 10% of information from the sparse field to detect these forest fires. As we saw in the previous results, it is these last shreds of information that can aid prevention services to reach the affected area. We thus evaluate the ability of the three approaches to be able to retrieve 90%, 95% and 97.5% of information of burnt areas.

Result: Fig. 9b shows that retrieving the bottom 90% of information from a sparse domain was relatively easy for each approach while remaining useless for the application at hand. As we move to retrieving more fine-grained information, we see that QuAiL can give 97.5% information upto 4x faster. Note again, this is the best case scenario for individual sparse sampling. This shows the benefit of QuAiL to rely on linear sparse domains to completely outclass sparse sampling in retrieving sparse spatial distributions.

Robustness to Noise: Another key factor that affects most wireless systems is the presence of interference and spurious noise. We add additional spurious noise to the received signal to identify the performance of QuAiL in such spurious noise scenarios. We have also highlighted some limitations regarding this in Sec. 7. Our SNR is measured such that at 0 dB SNR the farthest client transmission at highest transmit power will be just about decoded under noise. We measure the error in recovered information after 10 queries.

Result: We can observe in Fig. 9c that as the SNR worsens, much of the information for clients with frequency offsets which did not add up with the ensemble is lost. Despite these losses, we can still retrieve upto 88-90% of the underlying spatial information. Remember that the noise is so high that the base station will not be able to listen to a normal client in such a circumstance. The performance drastically improves as the noise is lowered, QuAiL reaches almost ideal performance. This shows the benefit of QuAiL's signals' powers adding up enabling even to tackle spurious noise.

(a) (b) (c)

Figure 10: Applications: (a) Effect of sparsity on QuAiL's performance (b) Comparison of QuAiL's performance with parallel decoding technique Choir[11] (c) Scatter plot of QuAiL's ability to compute approximate convolutions on [20]

8.3 Applications and Limits

Here, we will discuss the ability of QuAiL to operate in various sparsity scenarios, compute convolution filters and statistics, and limits of squeezing orthogonal codes in NB-IoT RACH.

Importance of Sparsity: QuAiL relies on the underlying sparsity of information. Yet, not all fields are sparse to extract information from just a few queries. We evaluate this by first testing the capabilities of QuAiL vs. the ideal and sparse sampling approach over a uncorrelated set of random values assigned to each sensor. Next, we evaluate a sparse domain whose underlying sparse domain may be unknown to us yet exists. Finally, we evaluate QuAiL on a domain whose sparse information is known to the algorithm. We measure the normalized error across queries.

Result: Fig. 10a shows that in a dense domain all the three approaches perform miserably with each pixel retrieving only one pixels information. This shows that in a dense domain there does not exist any solution other than querying every client. Next, we see that QuAiL is able to extract more information without necessarily the optimum information per query despite the sparse domain being unknown. Finally, with the knowledge of sparse domain, we see that QuAiL achieves almost optimum performance as achievable by oracle knowledge of sparse domains.

Comparison with Parallel Decoding Techniques: Recent work has proposed novel approaches to detect and decode multiple packets from collided packets. To demonstrate the scalability of QuAiL as well how it can be applied complementary to those approaches, we present a comparison with Choir[11] to identify the scale at which QuAiL performs better. In fact, if clients were to communicate with knowledge of these offsets we can implement QuAiL complementary to Choir as well. We measure the normalized error across queries for identifying the mean and standard deviation of uniform and normal distributed data. We allow only one query for mean and two queries for standard deviation.

Result: Fig. 10b shows that the number of clients at which QuAiL performs better than Choir depends on the distribution of the data as well as the computed aggregate. Our evaluation demonstrates that, with 99% confidence, QuAiL will outperform Choir when there exist more than 57 clients in all four cases. Further, using QuAiL complementary to Choir, outperforms both of the above approaches demonstrating the promise of QuAiL's approach.

Convolution Filters: We described in Sec. 5.2 the importance of convolution for various image processing and machine learning tasks on spatial information. We go back to the first paper[20] that used these filters to bring the accuracy of number text accuracy to 96%. When we train the convolutional neural network, we pick out the first layer of convolutions applied on the data sets. We then apply these convolutions along with DCT as described in Sec. 5.2 to evaluate the ability of QuAiL to retrieve sparse representations of the convolution. Each input is a 784 pixel image, and we enable 100 queries to estimate the convolution of the underlying image.

Result: We see that the median error across the 8 convolutions chosen is about 0.059% (Fig. 10c). Firstly, this shows that even convolutions have a underlying sparse image. Next, this highlights the ability of QuAiL to quickly get great estimates of convolution on sparse distributions without the overhead of missing out on fine-grained information. Finally, this shows the promise of QuAiL in enabling ML solutions on spatial distributions being sensed.

Statistics: Another benefit of QuAiL is the ability to estimate various statistics on the sensed information across clients in a single query. To get data for various clients, we sample client data from [25] dataset, where each of our client chooses one vector of the data. Then, compute the statistics as described in Sec. 5.1 to evaluate QuAiL's ability. We also enable the randomization matrix M to study the excess error caused due to inversion at the base station.

Result: We see in Fig. 11a that we can compute most statistics within an error bound of 5%. While this may seem excessive, remember that we are computing the exact value of a quantity. This leads to even small errors leading to high normalized error. These bounds are only for a single query. With more number of queries, we can reduce this bound significantly by repeating and randomizing queries differently.

NB-IoT code limits for QuAiL: Finally, we evaluate how finely can we divide the RACH spectrum in NB-IoT into various codes. Currently, RACH spectrum is divided into 48 parallel frequency bands and these operate in highly frequency synchronized clients. We evaluate QuAiL on estimating the average occupancy of building by sampling client data for each room from [8] dataset. We build a neural network whose input vector is 40000 sized and use various number of first hidden layer neurons as codes available. Typically, more neurons lead to higher accuracy.

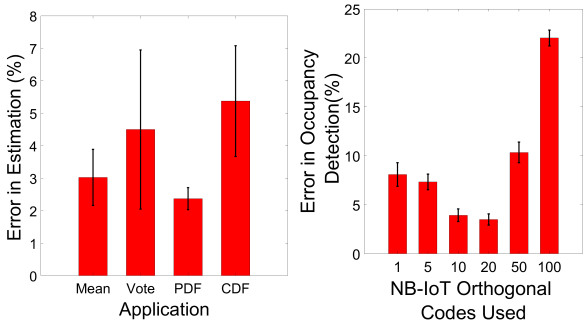


Figure 11: Limits: (a) Error in estimating statistics on [25] using QuAiL (b) Performance of QuAiL across number of orthogonal NB-IoT codes within 200KHz

Result: As described in Fig. 11b, we see that the accuracy increases as we increase the number of orthogonal codes. This is expected as the number of hidden neurons also increased. Yet as we increase the number of neurons further we reach a bottleneck as clients' frequency offsets start placing them in incorrect frequency bins. This shows the fundamental limit on orthogonal NB-IoT codes that can be squeezed in the available RACH spectrum.

9 CONCLUSION

The paper present, QuAiL, an LP-WAN solution for retrieving aggregate queries on thousands of clients within the duration of a single packet. QuAiL's approach relies on linear addition of powers of phase-asynchronous channels in the air to program these queries. QuAiL achieves this without modifying clients and is yet compatible to both major LP-WAN technologies, LoRa and NB-IoT. QuAiL achieves 4× faster aggregation of representation of forest fire maps and 2.36× lower error when the same number of queries are used over individual sparse sampling.

QuAiL's approach opens a interesting opportunity – how do you design LP-WANs where aggregate information is more useful than individual information. While individual information is important for parking and electricity meters, there is a massive untapped opportunity for LP-WANs to empower smart cities via aggregate queries detecting epicenters of earthquakes, flooding maps, forest fires faster and being able to diffuse such situations faster.

Acknowledgements: This work was supported by NSF grants 1942902, 1837607, 1646235, IoT@CyLab and Kavcic-Moura Fund. The authors would also like to thank the shepherd, IPSN reviewers, and members of WiSe and WiTech lab for feedback and support.

REFERENCES

[1] 2018. T-Mobile Launches America's First Nationwide Narrowband IoT Network. https://www.t-mobile.com/news/americas-first-narrowband-iot-network.
[2] 2020. QuAiL Code and Data. https://github.com/AkshayGadre/QuAiLIPSN2020.
[3] 3GPP. 2018. NarrowBand - Internet of Things. https://www.gsma.com/iot/narrow-band-internet-of-things-nb-iot/. Accessed: 2019-01-30.
[4] Luciano Volcan Agostini et al. 2001. Pipelined fast 2D DCT architecture for JPEG image compression. In *IEEE SBCCI*.
[5] Abdelmohsen Ali et al. 2017. On the cell search and initial synchronization for NB-IoT lte systems. *IEEE Communications Letters* (2017).
[6] Pilar Andres-Maldonado et al. 2017. Narrowband IoT Data Transmission Procedures for Massive Machine-Type Communications. *IEEE Network* (2017).
[7] Waheed Bajwa et al. 2006. Compressive wireless sensing. In *ACM IPSN*.
[8] Luis M Candanedo et al. 2016. Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models. *Elsevier Energy and Buildings* (2016).
[9] Adwait Dongare et al. 2017. Pulsar: A wireless propagation-aware clock synchronization platform. In *IEEE RTAS*.
[10] Adwait Dongare et al. 2018. Charm: exploiting geographical diversity through coherent combining in low-power wide-area networks. In *ACM/IEEE IPSN*.
[11] Rashad Eletreby et al. 2017. Empowering Low-Power Wide Area Networks in Urban Settings. In *ACM SIGCOMM*.
[12] Xinlong Feng et al. 2007. The rank of a random matrix. *Elsevier Applied Mathematics and Computation* (2007).
[13] Akshay Gadre et al. 2020. Frequency Configuration for Low-Power Wide-Area Networks in a Heartbeat. In *USENIX NSDI*.
[14] Anna Gilbert et al. 2010. Sparse recovery using sparse matrices. *Proc. IEEE* (2010).
[15] GSMA. 2018. Smart agriculture proves a natural use of NB-IoT. https://www.gsma.com/iot/news/smart-agriculture-nb-iot/. Accessed: 2019-01-30.
[16] Lajos Hanzo et al. 2010. *MIMO-OFDM for LTE, WiFi and WiMAX: Coherent versus non-coherent and cooperative turbo transceivers*. John Wiley & Sons.
[17] Huawei. 2017. Smart Shared Bicycle Lock. https://www.huawei.com/minisite/iot/en/smart-bike-sharing.html. Accessed: 2019-01-30.
[18] Intercomp. 2017. Triumph in Dubai with tests for the development of NB - IoT technologies applied to parking sensors. https://www.intercomp.it/nb-iot-technologies-applied-to-parking-sensors/?lang=en. Accessed: 2019-01-30.
[19] Szymon Jakubczak et al. 2011. Softcast: one-size-fits-all wireless video. *ACM SIGCOMM CCR* (2011).
[20] Yann LeCun et al. 1995. Convolutional networks for images, speech, and time series. *The Handbook of Brain Theory and Neural Networks* (1995).
[21] Chao Li et al. 2015. The matrix mechanism: optimizing linear counting queries under differential privacy. *Springer: The VLDB journal* (2015).
[22] Xingqin Lin et al. 2016. Random Access Preamble Design and Detection for 3GPP Narrowband IoT Systems. *IEEE Wireless Communication Letters* (2016).
[23] Hermann Lipfert. 2007. Mimo ofdm space time coding–spatial multiplexing, increasing performance and spectral efficiency in wireless systems, part i technical basis (technical report). *Institut für Rundfunktechnik* (2007).
[24] LoRa Alliance. 2016. What is the LoRaWAN Specification? https://lora-alliance.org/about-lorawan. Accessed: 2019-01-30.
[25] Samuel R Madden, Michael J Franklin, Joseph M Hellerstein, and Wei Hong. 2005. TinyDB: an acquisitional query processing system for sensor networks. *ACM Transactions on database systems (TODS)* 30, 1 (2005), 122–173.
[26] Frank McSherry et al. 2009. Differentially private recommender systems: Building privacy into the netflix prize contenders. In *ACM KDD*.
[27] Saman Naderiparizi et al. 2018. Towards battery-free HD video streaming. In *USENIX NSDI*.
[28] Sensors online. 2018. Waste-Management System Rides Narrowband IoT Network. https://goo.gl/oHCgFK. Accessed: 2019-01-30.
[29] Privacy First. 2005. The Fair Information Principles. https://www.privacyfirst.nl/acties-3/item/154-the-fair-information-principles-canada.html.
[30] R. Grech. 2018. Semtech and Comcast's machineQ Announce LoRaWAN Network Availability in 10 Cities. https://www.semtech.com/company/press/semtech-and-comcasts-machineq-announce-lorawan-network-availability-in-10-cities.
[31] R Rajagopalan et al. 2006. Data-aggregation techniques in sensor networks: a survey. *IEEE Communications Surveys & Tutorials* (2006).
[32] Ceferino Gabriel Ramirez et al. 2019. LongShoT: long-range synchronization of time. In *ACM/IEEE IPSN*.
[33] Theodore S Rappaport et al. 1996. *Wireless communications: principles and practice*. Vol. 2. prentice hall PTR New Jersey.
[34] Rapeepat Ratasuk et al. 2016. NB-IoT system for M2M communication. In *IEEE WCNC*.
[35] Luis Sanchez et al. 2014. SmartSantander: IoT experimentation over a smart city testbed. *Elsevier Computer Networks* (2014).
[36] Alfred Stein and Christien Ettema. 2003. An overview of spatial sampling procedures and experimental design of spatial studies for ecosystem comparisons. *Agriculture, Ecosystems & Environment* 94, 1 (2003), 31–47.
[37] Jesús Martín Talavera et al. 2017. Review of IoT applications in agro-industrial and environmental fields. *Elsevier Computers and Electronics in Agriculture* (2017).
[38] Hüseyin Özgür Tan et al. 2003. Power efficient data gathering and aggregation in wireless sensor networks. *ACM SIGMOD Record* (2003).
[39] Wiki. 2019. Forest Fire Model. https://en.wikipedia.org/wiki/Forest-fire_model.
[40] Wikipedia. 2019. Discrete Cosine Transform. https://en.wikipedia.org/wiki/Discrete_cosine_transform.
[41] Wikipedia. 2019. Discrete Wavelet Transform. https://en.wikipedia.org/wiki/Discrete_wavelet_transform.
[42] Changsheng Yu et al. 2017. Uplink scheduling and link adaptation for narrowband Internet of Things systems. *IEEE Access* (2017).
[43] Xinyu Zhang et al. 2009. Optimized multipath network coding in lossy wireless networks. *IEEE JSAC* (2009).