

TOWARD RELIABLE AND SCALABLE LONG RANGE NETWORKING FOR RURAL IOT

By

Yidong Ren

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science—Doctor of Philosophy

2025

ABSTRACT

The Internet of Things (IoT) holds great promise for transforming rural applications such as precision agriculture, infrastructure monitoring, and environmental sensing. However, enabling reliable and scalable wireless connectivity in rural areas remains a fundamental challenge due to sparse infrastructure, energy constraints, and wide-area coverage requirements. This dissertation presents a system-level exploration into building practical, low-power, and cost-effective LoRa-based networks tailored for rural IoT.

Specifically, I address four key challenges: (1) unreliable backhaul due to the absence of cellular or wired infrastructure, (2) weak LoRa signal coverage in complex and obstructed rural terrains, (3) limited scalability of existing backscatter systems for battery-free communication, and (4) inflexible physical layer encoding that fails to meet the diverse demands of rural applications. I propose and validate a series of novel techniques, including opportunistic satellite backhaul using lightweight link estimation and routing, polarization-aligned underground communication for cross-soil sensing, concurrent non-linear chirp backscatter for scalable battery-free transmission, and reconfigurable chirp encoding for adaptive coverage, throughput, and energy balancing. These techniques are evaluated through real-world deployments, hardware prototypes, and empirical experiments across rural-scale testbeds.

Together, these contributions advance the design of robust and scalable LoRa networks for rural IoT. Looking forward, this work motivates future research on integrating space–air–ground architectures, embedding joint sensing and communication capabilities, and co-designing cross-layer protocols that adapt to the dynamic and heterogeneous nature of rural environments. This dissertation lays the foundation for next-generation rural IoT systems that are not only technically efficient but also practically deployable across underserved regions.

Copyright by
YIDONG REN
2025

ACKNOWLEDGEMENTS

I am profoundly grateful for the guidance, support, and opportunities provided to me during my journey as a Ph.D. candidate.

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Zhichao Cao, for his unwavering mentorship, insightful advice, and continuous encouragement throughout my research. His vision and dedication have greatly shaped both my academic development and professional growth.

I am also thankful to my collaborators — Younsuk Dong, Prof. Mi Zhang, and Prof. Shigang Chen — for their invaluable feedback, inspiring discussions, and collaborative spirit that enriched my work.

I would like to extend my sincere appreciation to my dissertation committee members, Prof. Tianxing Li and Prof. Huacheng Zeng, for their thoughtful feedback and guidance during key stages of my Ph.D. journey.

I feel fortunate to have shared this journey with my lab mates - Huaili Zeng, Yimeng Liu, Gen Li, Maolin Gan, Jingkai Lin, Guangliang Liu, Bocheng Chen, Shichen Zhang, Fangwei Zhang, whose friendship and support made EB1100 a vibrant and motivating place—truly the best lab!

Lastly, I owe my deepest gratitude to my family. Their unwavering love, patience, and belief in me have been the foundation of all my pursuits. This work would not have been possible without their endless support.

TABLE OF CONTENTS

CHAPTER 1	INTRODUCTION	1
1.1	Motivation	1
1.2	System Challenges and Technical Limitations	4
1.3	Contributions and Proposed Techniques	6
1.4	Organization	8
CHAPTER 2	SATERIOT: HIGH-PERFORMANCE GROUND-SPACE NETWORKING FOR RURAL IOT	9
2.1	Introduction	9
2.2	Preliminary and Observation	14
2.3	System Design	21
2.4	Implementation	28
2.5	Evaluation	30
2.6	Related Work	41
2.7	Conclusion	41
CHAPTER 3	DEMETER: RELIABLE CROSS-SOIL LPWAN WITH LOW-COST SIGNAL POLARIZATION ALIGNMENT	43
3.1	Introduction	43
3.2	Preliminary and Motivation	48
3.3	System Design	52
3.4	Implementation	62
3.5	Evaluation	64
3.6	Related Work	73
3.7	Discussion and Future Work	73
3.8	Conclusion	74
CHAPTER 4	PRISM: HIGH-THROUGHPUT LORA BACKSCATTER WITH NON-LINEAR CHIRPS	76
4.1	Introduction	76
4.2	LoRa Preliminary	79
4.3	System Design	82
4.4	Implementation	88
4.5	Evaluation	89
4.6	Related Work	95
4.7	Conclusion	97
CHAPTER 5	AEROECHO: TOWARDS AGRICULTURAL LOW-POWER WIDE-AREA BACKSCATTER WITH AERIAL EXCITATION SOURCE	98
5.1	Introduction	98
5.2	Preliminary	101
5.3	System Design	104
5.4	Implementation	111

5.5	Evaluation	111
5.6	Related Work	119
5.7	Conclusion	120
CHAPTER 6	CHIRPTRANSFORMER: VERSATILE LORA ENCODING FOR LOW-POWER WIDE-AREA IOT	122
6.1	Introduction	122
6.2	Background and Motivation	126
6.3	Design of ChirpTransformer	129
6.4	Case I: Network Coverage	134
6.5	Case II: Network Throughput	143
6.6	Case III: Network Lifetime	147
6.7	Related Work	150
6.8	Discussion	151
6.9	Conclusion	154
CHAPTER 7	CONCLUSION	156
BIBLIOGRAPHY		157

CHAPTER 1

INTRODUCTION

1.1 Motivation

1.1.1 Networking for Rural IoT

Rural and remote areas have become a prominent frontier for Internet of Things (IoT) applications, including precision agriculture, environmental monitoring, and infrastructure management. In precision agriculture, for example, farms deploy IoT sensors to monitor soil moisture, weather, and crop health in real-time, enabling data-driven irrigation and fertilizer management [1, 2]. Environmental monitoring of wildlife habitats, forests, or water resources similarly relies on distributed sensor nodes in remote locations to collect critical data (e.g., climate measurements or animal tracking) over wide areas. Infrastructure management in rural regions – such as monitoring remote roads, pipelines, or power lines – also benefits from IoT devices that can report faults or conditions from far-flung locales. These rural use cases share common communication needs that are distinct from urban IoT settings. Unlike dense urban deployments, rural IoT nodes may be spread across kilometers of terrain with limited access to mains power or existing network infrastructure. Key network requirements in these scenarios include:

Long-Range Coverage: IoT sensors in farms or wilderness often reside far apart or far from any communication tower. Wireless links must reliably span distances on the order of several kilometers, far beyond the range of personal-area or local-area networks [1, 3]. Traditional short-range wireless technologies – such as Wi-Fi, Bluetooth, and Zigbee – typically cover only tens of meters and thus cannot economically service devices dispersed over large rural areas.

Energy Efficiency: Devices are usually battery-powered or rely on off-grid energy (e.g. solar), so the communication protocol must minimize energy usage. Transmit power and signaling overhead should be low enough to allow multi-year operation on batteries [4, 5]. Frequent battery replacements or recharging is impractical in remote fields or forests, so ultra-low-power communication is essential.

Affordability and Simplicity: Rural IoT deployments often operate under tight budget constraints

and in areas with sparse infrastructure. Solutions must be low-cost in terms of device hardware and operating expenses. Ideally, sensors should connect without requiring expensive base stations or licensed spectrum [1, 4, 6]. The network should be simple to deploy and maintain by non-expert users (e.g., farmers), making proprietary or complex setups less attractive.

These requirements highlight why conventional wireless options are insufficient. Wi-Fi and Bluetooth, while widely used in consumer IoT, cannot reach the long distances needed on a farm and typically consume too much power for battery-operated sensors. Cellular networks (LTE, 5G) provide wide-area coverage in populated regions, but many rural areas lack reliable cellular coverage or have coverage only near main roads. Even where available, cellular IoT connectivity entails high energy consumption and subscription costs for each device, which is prohibitive for large-scale sensor deployments. Emerging cellular-based IoT standards like NB-IoT and LTE-M improve power efficiency, but they still depend on operator infrastructure and licensed spectrum availability that may be lacking in remote locales [7]. This gap in networking options has driven interest in a new class of long-range, low-power wireless technologies.

LPWAN Technologies and LoRa: Low-Power Wide-Area Networks (LPWANs) have emerged as a compelling solution tailored to the needs of rural IoT. LPWAN technologies (such as LoRa, Sigfox, and Narrowband-IoT) are designed to send small data packets over kilometers of range while keeping devices in a low power state [1, 3]. Among these, LoRa (Long Range) has gained particular prominence as a flexible, cost-effective option for rural applications. LoRa uses a spread-spectrum modulation in unlicensed sub-GHz bands, enabling link ranges on the order of 10–15 km in rural open conditions and excellent obstacle penetration compared to higher-frequency radios [1, 7, 8]. Despite a modest data rate (typically 0.3–50 kbps), LoRa networks can support the infrequent, low-bandwidth telemetry that characterizes environmental sensors and agricultural IoT devices. Importantly, LoRa devices can operate for years on battery power because of the very low duty cycle and energy-efficient physical layer [4]. The technology’s use of unlicensed spectrum also means that farmers or communities can deploy their own LoRa networks without regulatory hurdles or recurring spectrum fees. These attributes make LoRa a promising foundation for reliable and

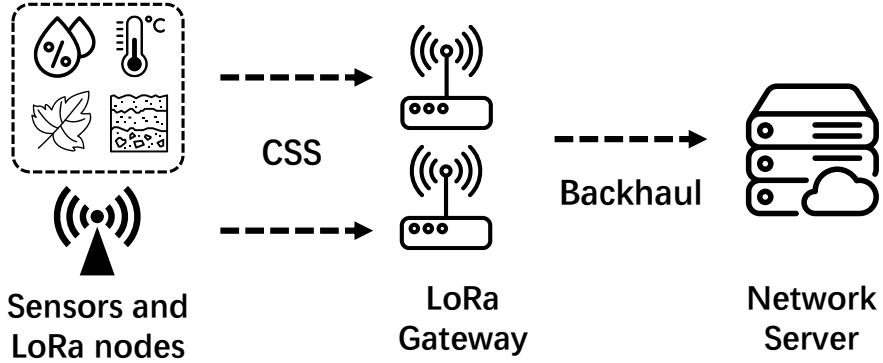


Figure 1.1 LoRa Networking Architecture.

scalable rural IoT connectivity where traditional wireless networks fall short.

1.1.2 LoRa Networking Architecture

LoRa is often used within the LoRaWAN protocol stack, which defines a complete networking architecture for connecting IoT nodes to the internet. As Figure 1.1 shows, the typical LoRaWAN architecture follows a star-of-stars topology [4, 7]: resource-constrained end devices (sensor or actuator nodes equipped with LoRa radios) transmit data over the air to intermediary devices called gateways. A gateway is a relatively simple relay that listens to LoRa channels and forward packets from any nodes in range. Crucially, LoRa gateways are bridges between the wireless LoRa domain and the internet Protocol (IP) world; they usually have a more powerful processor and an IP backhaul connection. Multiple gateways can be deployed to provide comprehensive coverage, and end devices do not need to be associated with a single gateway. The gateways then pass along received LoRa frames to a central Network Server, often located in the cloud or a central site. This front-haul (node-to-gateway) link uses LoRa radio with Chirp Spread Spectrum (CSS) [9] modulation, while the backhaul (gateway-to-server) link uses any available IP network (e.g., Ethernet, cellular, or satellite link) to route data to the server. The network server, together with an application server, performs higher-layer functions: filtering duplicate packets, managing device sessions and security keys, and forwarding the IoT data to application databases or user interfaces.

Table 1.1 Challenges, Limitations and Solutions in LoRa networking for rural IoT.

System Challenges	Technical Limitation	Proposed Solution
Rural area backhaul (Gateway-Server)	Temporal-Spatial variant satellite link	Chapter 2: Opportunistic satellite backhaul among multiple gateways [10]
LoRa wireless coverage (Node-Gateway)	Complex rural environments	Chapter 3: Polarization-aligned underground coverage [11] Chapter 6: Reconfigurable encoder-decoder co-design [5]
IoT battery reliance (Node)	Limited backscatter deployment scale	Chapter 4: Concurrent non-linear chirp backscatter [12] Chapter 5: Drone assisted scalable backscatter [13]
Diverse application (Node-Gateway-Server)	Inflexible encoding	Chapter 6: Reconfigurable encoder-decoder co-design for coverage/throughput/energy [5]

In practice, rural LoRaWAN deployments might use a single gateway to cover a local area (for instance, one gateway with an elevated antenna can cover an entire farm or village). In larger areas, multiple gateways are deployed to increase coverage and resiliency, all connected back to the same network server (forming a star topology where gateways are star points that collectively service many nodes). Gateways are often placed at high points (e.g., on top of buildings, hills, or towers) to maximize line-of-sight range. Since gateways require backhaul connectivity, they may be installed where wired internet or at least cellular backhaul is available (such as a farmhouse with DSL, or a hilltop with cellular signal). In truly isolated areas lacking any infrastructure, alternative deployment models have been demonstrated: for example, a mobile gateway on a drone or unmanned aerial vehicle (UAV) can periodically overfly remote sensors to collect their data via LoRa and then carry the data back to the internet once the drone returns to a connected zone [7]. Similarly, satellite-connected LoRa gateways (or even satellites acting as LoRa receivers) have been explored to achieve global coverage. These patterns illustrate the flexibility of LoRa networking – combining long-range wireless fronthaul with any available backhaul – to bring connectivity to devices in hard-to-reach places.

1.2 System Challenges and Technical Limitations

Table 1.1 summarizes the major system challenges faced in rural LoRa deployments, the underlying technical limitations.

1.2.1 Rural Area Backhaul (Gateway-Server)

While LoRa provides a long-range physical link between devices and gateways, the backhaul from gateways to cloud infrastructure remains a major bottleneck in rural areas. Most existing LoRa deployments rely on either wired or cellular IP connectivity, which is often unavailable or unreliable in remote terrains. Satellite communication offers global coverage but suffers from variable link quality, high latency, and cost constraints. Moreover, many rural settings experience temporal and spatial variability in link availability—e.g., satellite visibility may depend on orbital patterns, and the coverage may fluctuate due to terrain and weather. These dynamics complicate real-time data forwarding and strain energy-constrained gateway operations. An opportunistic, delay-tolerant, and energy-aware backhaul strategy is thus essential to maintain consistent service in infrastructure-sparse environments.

1.2.2 LoRa Wireless Coverage (Node-Gateway)

The theoretical range of LoRa—often advertised as 10–15 km in open environments—can be drastically reduced in practical rural deployments due to terrain-induced shadowing, vegetation density, and underground placement of sensors. In applications like precision agriculture or pipeline monitoring, sensors may be deployed underground or in heavily vegetated areas where radio signals attenuate quickly or suffer from polarization mismatch. Traditional omnidirectional antennas and fixed configuration radios are poorly suited to these non-line-of-sight or cross-medium scenarios. Enhancing fronthaul reliability thus requires both physical-layer adaptations and system-level awareness of spatial deployment constraints.

1.2.3 Battery Reliance (Node)

LoRa-based IoT devices are predominantly battery-powered, limiting device lifetime and increasing maintenance costs, especially in large-scale rural deployments. Energy-harvesting approaches, such as solar panels, are weather-dependent and expensive. By leveraging ambient source carrier modulation, backscatter systems enable battery-less communication at μW power levels, extending battery lifetime to tens of years. However, existing LoRa backscatter systems face scalability challenges due to severe signal collisions and costly sources requirement for large-scale

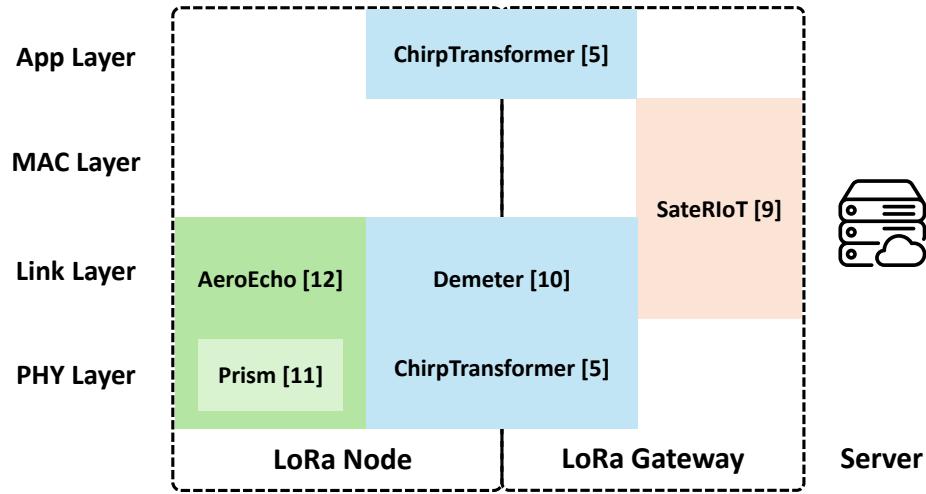


Figure 1.2 My contributions in the corresponding LoRa networking stack.

deployment.

1.2.4 Diverse Application (Node-Gateway-Server)

Different rural IoT applications have diverse performance requirements. General monitoring scenarios prioritize broad coverage across large geographic regions. Industrial sites, such as utility stations and agricultural processing, require dense sensor deployments. They need concurrent transmission for reliable network throughput. Mobile applications, such as livestock or transportation tracking, operate under highly dynamic wireless conditions. These scenarios require a fine-grained, adaptive data rate to enhance energy efficiency. However, standard LoRa uses fixed chirp-based encoding schemes and lacks flexibility in configuration. This makes it hard to adapt to the needs of diverse rural IoT scenarios.

1.3 Contributions and Proposed Techniques

To tackle these challenges and limitations, this dissertation presents a set of feasible LoRa networking solutions across the node, gateway, and server stack. These solutions span the APP layer, MAC layer, link layer, and PHY layer—some targeting specific layers, while others take a cross-layer design approach. As shown in Figure 1.2, each technique addresses critical aspects of reliability, scalability, or energy efficiency for rural IoT deployments. Besides system challenges and technical limitations, Table 1.1 also summarizes the corresponding solutions proposed in this dissertation.

Each solution chapter addresses one or more of these challenges through system-level design, implementation, and real-world evaluation.

1.3.1 Opportunistic satellite backhaul (Chapter 2)

I propose *SateRIoT* [10] as a novel network architecture with temporal link estimation and spatial link sharing that fully exploits the capability of low-cost low-earth-orbit (LEO) IoT satellites and ground LPWAN techniques. I introduce a lightweight machine learning model for bursty link estimation of satellite radio and design a multi-hop protocol with carrier sensing capability to share the data among multiple ground LPWAN radios. We deploy the system with commercial IoT satellite and LoRa radios in $3 \times 3 \text{ km}^2$, then evaluate it based on real deployment. It achieves up to $3.3 \times$ less energy consumption, $5.6 \times$ reduction in latency, and $1.9 \times$ enhancement in throughput.

1.3.2 Underground wireless coverage (Chapter 3)

Using LPWAN to collect data from buried underground sensors offers an efficient and non-intrusive solution. In such cross-soil links, signal loss arises from both propagation-induced attenuation and polarization misalignment due to varying soil conditions. However, the impact of polarization misalignment has been largely overlooked. I propose *Demeter* [11], a low-cost, low-power programmable antenna circuit design to keep reliable cross-soil communication automatically. My design utilizes a coupler to enable computational polarization adjustment on a COTS single-RF-chain LoRa radio. We evaluate it on customized circuits and commercial devices in various environmental conditions. It achieves up to 11.6 dB SNR gain, $4 \times$ horizontal communication distance, and $5.5 \times$ energy saving compared with the standard LoRa.

1.3.3 Non-linear chirp backscatter (Chapter 4)

Existing LoRa backscatter systems suffer from severe signal collisions caused by overlapping linear chirps. I propose **Prism** by using different types of non-linear chirps to modulate backscatter data based on standard LoRa carrier signals. Prism enables multiple orthogonal channels, allowing backscatter tags to transmit concurrently at the same frequency. I designed an energy-efficient method to accurately shift the frequency of the linear chirps to the non-linear chirps. It is implemented with an ultra-low power tag and standard LoRa carrier and evaluated in real environments.

The results show a $6\times$ higher transmission concurrency than state-of-the-art.

1.3.4 Drone-assisted scalable backscatter (Chapter 5)

The asymmetric distances between source–tag and tag–receiver pairs require high-cost dense source deployment. I introduce *AeroEcho* [13], a drone-assisted asynchronous decoding within the same non-linear chirp type. I co-design the excitation source and tag with a customized packet format to enable decoding for multiple tags. And then I propose two aerial routing strategies by using excitation cells to achieve optimal throughput and symbol error rate for large-scale sensor deployments. *AeroEcho* enhances the overall throughput of current backscatter transmission by $5.84\times$ and individual tag data rate by $12\times$ compared to state-of-the-art approaches.

1.3.5 Reconfigurable encoder-decoder co-design (Chapter 6)

Due to resource constraints, the encoder in existing LoRa systems remains fixed and unmodified. To expand the design space for data modulation, I introduce *ChirpTransformer* [5], a reconfigurable LoRa encoding framework that harnesses broad chirp features on commercial devices. We combine software design with hardware interrupts to dynamically modulate data using four custom features on COTS LoRa nodes without incurring additional energy or latency overhead.. The results demonstrate a $2.38\times$ increase in network coverage with a neural network decoder, a $3.14\times$ boost in throughput, and a $3.93\times$ of battery lifetime compared to standard LoRa.

1.4 Organization

The remainder of this dissertation is as follows: in Chapter 2, I discuss the opportunistic satellite backhaul; in Chapter 3, I discuss the polarization-aligned underground coverage; in Chapter 4, I discuss Concurrent non-linear chirp backscatter; in Chapter 5, I discuss the drone assisted scalable backscatter; in Chapter 6, I discuss the reconfigurable encoder-decoder co-design; in Chapter 7, I conclude this dissertation.

CHAPTER 2

SATERIOT: HIGH-PERFORMANCE GROUND-SPACE NETWORKING FOR RURAL IOT

2.1 Introduction

The Internet of Things (IoT) is revolutionizing the way we understand and interact with the physical world, enhancing precise agriculture [11, 14, 15], environment monitoring [16–18], and forest fire prevention [19, 20] among others. Low-power wide-area (LPWA) IoT techniques (e.g., LoRa [7, 21, 22], NB-IoT [23], LTE-M [24])¹ are desirable to fit the scale of rural areas. Their networking architecture consists of gateways with backhaul links allowing internet access and many sensor nodes served by those gateways. However, the lack of urban-area networking infrastructures (e.g., 5G, LTE, wired networks) in rural areas presents a cost-benefit concern regarding the huge expenses of establishing new backhaul links.

Recently, LEO satellites [25, 26] demonstrate global Internet access for ground satellite radios with space links. A satellite radio sends its packets to LEO satellites over space up-links, and then these packets are forwarded to several territorial ground stations, which are Internet-connected, through space down-links. Exploring existing LEO satellites to enable direct-to-satellite (DtS) IoT is a cheaper alternative to establishing new backhaul links on the ground. For example, SpaceX’s [27] subsidiary Swarm [28]², a commercial-off-the-shelf (COTS) DtS-IoT provider, provides global connectivity for \$5 per link per month.

SWARM uses ultra-small LEO satellites whose orbit altitude is either 462 kilometers or 510 kilometers. The connections between ground satellite radios and the satellites can be affected by many factors [29–33], e.g., weather, satellite orientation, relative antenna position between the radios and the satellites, etc. According to our measurement study (§ 2.2.3), the available connections are sparse during a SWARM satellite passing by. In 80% cases, the total connection time is less than 20% of the pass duration. Moreover, the data rate of SWARM-M138 [34] radio is as low as

¹We implement our prototype with LoRa so use LoRa to represent general LPWA IoT techniques in the rest of the paper.

²We implement our prototype with SWARM so use SWARM to represent general DtS-IoT techniques in the rest of the paper.

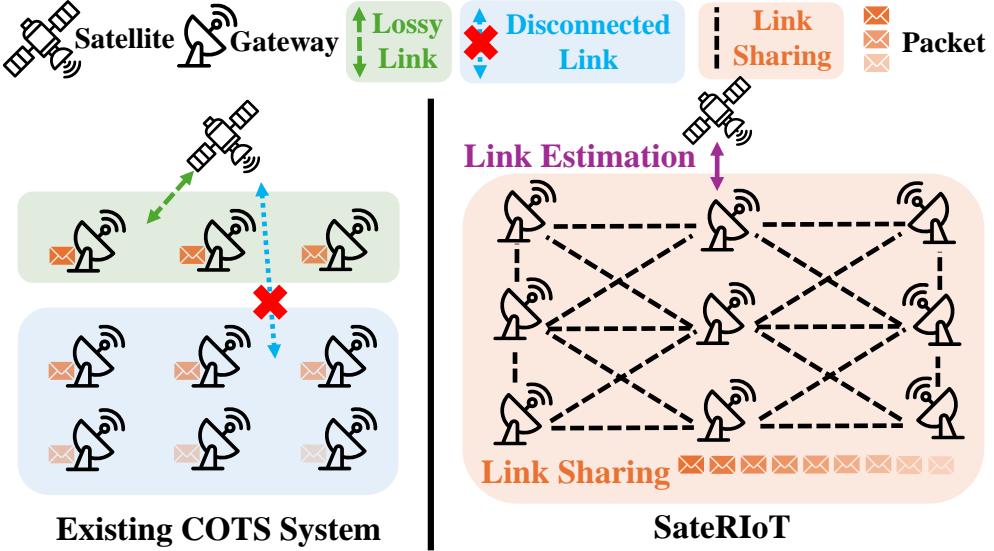


Figure 2.1 An illustration of link-aware SateRIoT with multi-hop data sharing.

1 kbps. The sparse connection time and low link capacity motivate us to rethink the ground-space IoT architecture that can fully utilize the LEO IoT satellite network to achieve optimal IoT backhaul throughput while keeping efficient energy consumption and low end-to-end delay. As shown in the left part of Figure 2.1, existing works [35–38] simply equip a satellite radio on each LoRa gateway so that the LoRa gateway forwards its collected sensory data to the Internet when a satellite passes by. However, this intuitive ground-space IoT architecture fails to achieve high performance due to the following temporal-spatial link challenges.

Challenge 1: Lossy up-links during short-window data pulling. When a gateway is connected to a satellite, the up-links could be lossy, degrading the energy efficiency of the gateway with a fixed number of packet transmissions. Specifically, deploying power infrastructure in remote areas such as farms and forests is challenging, complicating the powering of LoRa gateways [12, 39, 40]. The existing IoT gateway utilizes small solar panels for cost-effectiveness and ease of deployment. However, due to insufficient sunlight, these small panels become unsustainable in variable weather conditions such as rain, snow, cloudy days, and nighttime. Therefore, energy efficiency is important for satellite IoT gateways. The energy consumption of SWARM-M138 [34] radio reaches 7.97 mJ/bit in its Tx mode with 1 kbps, which is 408 \times larger than 0.0195 mJ/bit LoRa SF10 976 bps.

A SWARM satellite transmits data beacons to pull data from ground SWARM radios. When a

SWARM radio receives this beacon, confirming that a connection is established, it performs 6 subsequent mandatory transmission attempts (i.e., a fixed number in SWARM protocol), irrespective of how many are received. However, in such a short-term transmission window, we observe that a SWARM radio can still fail to deliver its packets to the satellite despite having received the data beacon successfully. With the temporally lossy up-link, the enforced repeated transmissions cause significant degradation of LoRa gateways' energy efficiency.

Challenge 2: Diverse connection time among spatial LoRa gateways. Due to the spatial diversity of LoRa gateways, although they all experience sparse connections, we observe the connection time is quite diverse among them. In our measurement study, when a satellite passes over four LoRa gateways, different LoRa gateways can successfully transmit their data to the satellite at different times. In most cases, only one LoRa gateway is available at a time (§ 2.2.5). If a LoRa gateway only transmits the data received by itself, the LoRa gateway with sufficient connection time and transmission capabilities will keep idle when all its own data has been forwarded. Considering the buffered data on other gateways with insufficient connection time, the overall network throughput is lowered.

To address these challenges, in this paper, we propose SateRIoT, a new network architecture with temporal link estimation and spatial link sharing that fully exploits the capability of SWARM backhaul connections, thus enabling high-performance ground-space IoT for rural areas. We estimate the temporal link behavior to determine the optimal number of data packets to save energy and allow gateways to share their idle connection time with others to maximize network throughput and lower the end-to-end delay. As shown in the right part of Figure 2.1, besides self-generated data, each LoRa gateway buffers data shared from all other gateways. When the top-middle LoRa gateway finds that k packets should be transmitted in the current time window, it will immediately transmit k packets from either itself or others. The design of SateRIoT includes three key problems to guarantee high network performance while reducing extra computation and maintenance costs.

Firstly, the quality of a SWARM link is determined by many factors (e.g., atmosphere, satellite orbit) [29–33], creating a huge feature space that exposes the challenge of balancing between accu-

racy and agility. The accuracy cannot be guaranteed if we only use simple features like weather and satellite orbit information as inputs. If more link information is counted while a SWARM satellite is passing, we have limited time for the computation of link modeling and traffic scheduling. In SateRIoT, we choose to guarantee accuracy first while optimizing its agility. Specifically, we design a bursty link model to depict the temporal link behavior. We estimate the number of successful packet transmissions in an adaptive transmission window in which the link quality is consistent. To find the optimal trigger timing that balances between sufficient feature collection and limited computation time, we propose an Acknowledgement(ACK)-triggered lightweight ensemble learning model.

Second, selecting proper model parameters is not trivial to achieve optimal performance in practice. On one hand, we need to determine a small group of specific features from many available ones as the model inputs that derive accurate link estimation. Meanwhile, we need to define an appropriate time length for each transmission window adaptively. Due to the lossy link nature, a long transmission window makes it hard to keep link consistency, degrading link estimation accuracy. However, due to link sharing among multiple LoRa gateways, short transmission windows lead to frequent model computation and network data status synchronization, causing increased energy consumption. Additionally, to ensure a transmission window can be configured in practice, it must align with the timing of the SWARM data-pulling window. To address the issues, SateRIoT proposes meticulously utilizing features from the physical layer, COTS satellite IoT protocols, and environmental information. For the optimal window length, we aim to find the longest window with consistent link quality to save energy. We adopt an empirical method to identify the optimal configuration.

Third, in link sharing, a LoRa gateway needs to collect data from others. However, when and how to collect the data is challenging. Since the connection time is sparse, after a LoRa gateway sets up a transmission window, we have no time to use energy-efficient tree-based ad-hoc routing protocols for packet pulling from others [41, 42]. Since the energy consumption of a LoRa radio is far less than that of the SWARM satellite radio, we enable link sharing with a multi-hop flooding protocol, which utilizes broadcast to achieve network-wide data consistency quickly. Specifically,

when a gateway collects new data from its sensor nodes, it initiates a flooding process, where other gateways forward the packet until all receive a copy. In this way, each LoRa gateway collects up-to-date data from others in a timely manner. In addition, carrier sense backoff is adopted to prevent packet collisions during concurrent LoRa flooding, improving network consistency and energy efficiency. On the other hand, it is possible for gateways to establish their transmission windows simultaneously. Although collisions can be mitigated by the LR-FHSS modulation [43, 44] in SWARM, duplicate packets may still occur in link sharing. To prevent duplicates, we prioritize self-generated data and shuffle relayed data. We also design beacons to lock transmitting packets and synchronize queues with the flooding service.

We implement SateRIoT on four gateways embedded with COTS SWARM M138 [34] and LoRa SX1262 [9] radio chips. We deploy these four gateways in several university farms that cover 9 km^2 areas to evaluate the performance of SateRIoT. Moreover, we have conducted extensive trace-driven emulation experiments for a large network scale with 12 gateways. Our results indicate SateRIoT delivers comparable throughput using up to $3.3\times$ less energy for each gateway. Additionally, SateRIoT reduces packet delay by up to $5.6\times$ and boosts throughput by $1.9\times$. Our contribution can be summarized as follows:

- We empirically measure the characteristics of LEO IoT satellite links with real-world deployment and observe three insights of temporal-spatial link behavior, which demonstrate the barriers of directly adopting the DtS-IoT to achieve high-performance rural IoT.
- We design SateRIoT, a new networking architecture consisting of well-designed satellite link estimation and sharing modules to enable network-wide data sharing and link-aware data transmission, enhancing network performance.
- We fully implement the proposed SateRIoT design with COTS LoRa gateways and SWARM radios. We extensively evaluate its efficiency with real-world deployment and trace-driven emulation. The results demonstrate that SateRIoT achieves equivalent throughput with $3.3\times$

less energy for individual satellite devices. Additionally, our scheduling scheme reduces packet delay by up to 5.6 \times and enhances throughput by 1.9 \times .

2.2 Preliminary and Observation

2.2.1 LEO Satellite for IoT

LEO satellites operate between 500 km and 2,000 km above Earth, offering advantages such as shorter latency and more frequent revisits compared to satellites in higher orbits. LEO satellite communication can provide high-speed broadband internet access globally, like Starlink [45] and OneWeb [46]. Additionally, LEO satellites can also facilitate low-cost IoT in rural areas with limited network infrastructure.

Satellite IoT has become an emerging field with selective commercial corporations striving to provide global IoT connectivity without resorting to expensive high-data-rate satellite communication systems. These entities have launched satellites to form extensive constellations, enabling the use of ground-based small satellite radios equipped with advanced long-range communication capabilities. For example, Sateliot [47] proposes to integrate NB-IoT for satellite communications into both current and forthcoming 5G infrastructures. Lacuna Space [48] delivers an uplink service utilizing a LoRa-inspired physical layer. SpaceX's IoT solution [27] SWARM Technologies [28] collaborates with Semtech [49] to employ long-range frequency hopping spread spectrum (LR-FHSS) modulation within the VHF bands. This operates with an uplink range of 148-150 MHz and a downlink range of 137-138MHz, aiming for exceptionally cost-effective satellite communication at a 60 USD data fee per year. In this study, we use SWARM satellite radio to evaluate the efficacy of COTS satellite IoT systems and subsequently craft our design based on these insights. SWARM with 168 satellites differs quite from traditional high-data-rate LEO satellite networks (e.g., Starlink with more than 10000 satellites) in terms of protocols, satellite size, and constellation scale. SWARM links exhibit unique characteristics such as a more sparse satellite distribution, shorter connection times, lower frequency usage, and reduced network costs [28, 32].

2.2.2 Measurement Experiments

To evaluate the network performance of the COTS satellite IoT system, we carried out outdoor measurement experiments on a rural farm. We employed 4 SWARM Eva Kits [50], each fitted with an M138 modem [34], and powered them using 18-24V DC solar panels. Over a span of 120 hours, we gathered data from 12 unique locations within a large farm, collecting approximately 3,000 packets and spanning an area of 9km^2 in rural farmland. We use infinite traffic to ensure we can capture the link status of every satellite passing by.

2.2.3 Up-link Connection Time

The propagation environment for SWARM satellites operating in the VHF bands introduces unique challenges associated with the temporal and spatial link variability [32]. For instance, atmospheric elements such as precipitation rates and cloud cover can cause time-dependent attenuation. This attenuation can vary over brief periods, making a substantial impact on link quality. Meteorological changes, especially seasonal variations, exacerbate these dynamics [31]. Due to unpredictable conditions, a fixed satellite link pattern cannot be relied upon. On the other hand, the link's spatial variability is attributed to differences in terrain and topography. These geographical features can result in shadowing and multipath effects, which differ based on the elevation angles and the terrain's contours [31, 33]. Additionally, satellite orientation is critical, especially in low Earth orbit, due to its high angle sensitivity. As a satellite progresses along its orbit, its relative positioning with respect to the ground satellite radio also changes. The elevation angle, defined as the angle between the ground's horizontal plane and the satellite's line-of-sight, influences the atmospheric path length and, subsequently, the link attenuation [25, 51].

Observation 1: Disconnected links During a SWARM satellite passes by, we divide the passing duration into several time windows with 1 minute. Given a time window, if a packet is successfully transmitted, we mark this time window as a connected time window. We define the connection time ratio as equal to the ratio between total connected time windows and total time windows in the passing duration to quantify the link disconnected property. According to the collected data traces, Figures 2.2 shows the results based on different maximum elevation angles and satellite

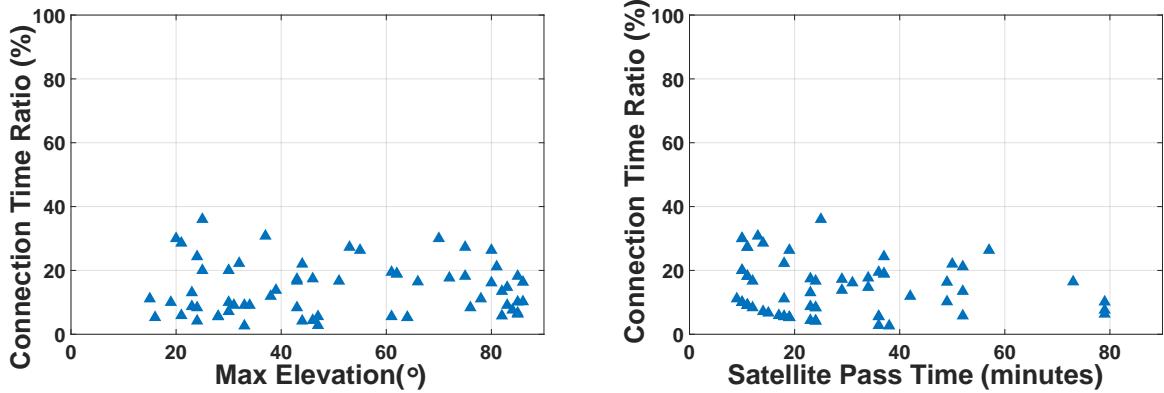


Figure 2.2 Connection time ratio.

passing duration. The results show that the connection time ratio remains consistently low. This trend holds irrespective of the maximum elevation or the duration of satellite passage. The average and median values are 14.1% and 11.9%, respectively. The highest observed ratio doesn't exceed 40%, and in 80% of the satellite passes, this ratio is below 20%. Such findings indicate the satellite links are disconnected, increasing the transmission latency.

2.2.4 SWARM Communication Protocol

Figure 2.3 depicts the communication protocol within the Swarm satellite IoT system. There are 3 basic packet types: satellite **non-data/data beacon**, **data packet**, and **ACK** (acknowledgement). Dotted edges represent the reception side and solid edges represent the transmission side. As a satellite coverage period begins, all the satellite radios awaken from their sleep mode. These satellite radios then start monitoring space, anticipating the satellite beacon. The received satellite beacon is the unsolicited message from the overhead satellite that can be classified into two types: non-data and data beacon. The data beacon informs satellite radios of the satellite's presence and its readiness to accept messages from the ground. Receiving the satellite data beacon is a prerequisite for initiating data packet transmission to the satellite. Non-data beacons are used for downlink traffic and will not trigger the satellite radio to transmit data. Multiple non-data beacons appear before data beacons. Both beacon types include information on the Received Signal Strength Indicator (RSSI), Signal-to-noise ratio (SNR), frequency, timestamp, and satellite ID.

According to the collected data traces, during times without transmissions, the non-data beacons

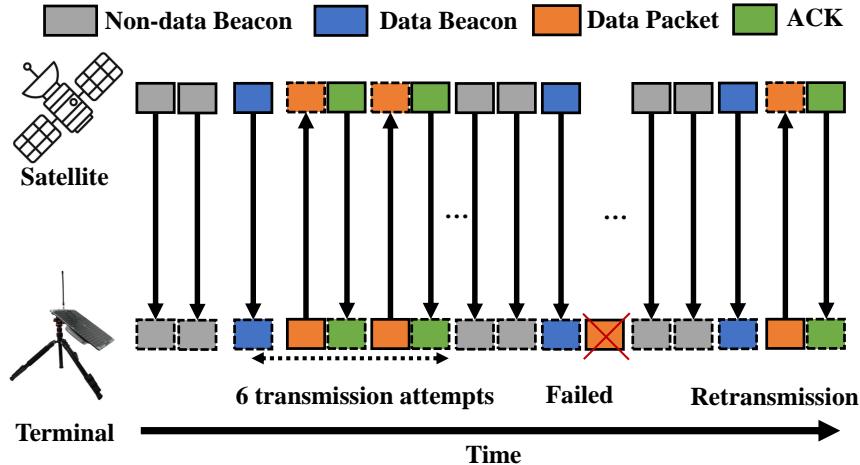


Figure 2.3 COTS SWARM IoT Communication Protocol.

were sent every 60 s. During each data transmission period, the satellite will increase the frequency of non-data beacons at the beginning and send a data beacon afterward. The time offset between two data beacons uniformly varies from 20 s to 52 s. All the data is transmitted between two data beacons. The median and average time gap between two adjacent data beacons during gateway transmission is 33 s and 35.2 s respectively.

When data packets are successfully received by the satellite, an ACK is promptly sent back to the satellite radio, as illustrated by the green block in Figure 2.3. The ACK signifies a successful data packet transmission. It includes RSSI, SNR, the sent packet ID, and frequency information. All packets delivered to the cloud correspond with acknowledgements received at satellite radio. If an ACK is not detected within a receiving window, it indicates a failed packet delivery. There are two main reasons why a packet may fail to reach the satellite: The first is when satellite radio chooses not to transmit data because of a high background noise level with RSSI exceeding -88 dBm. The second is when satellite radio transmits packets, but the satellite cannot detect them, either due to poor signal quality or interference.

Observation 2: Lossy Links. Satellite radio has a transmission buffer with a queue system called *satellite radio queue*, and messages are generally dispatched based on their entry sequence into this queue. Any unsuccessful packets are then re-queued for another attempt in the next transmission window accompanied by the following data beacon. In practice, the energy efficiency of satellite ra-

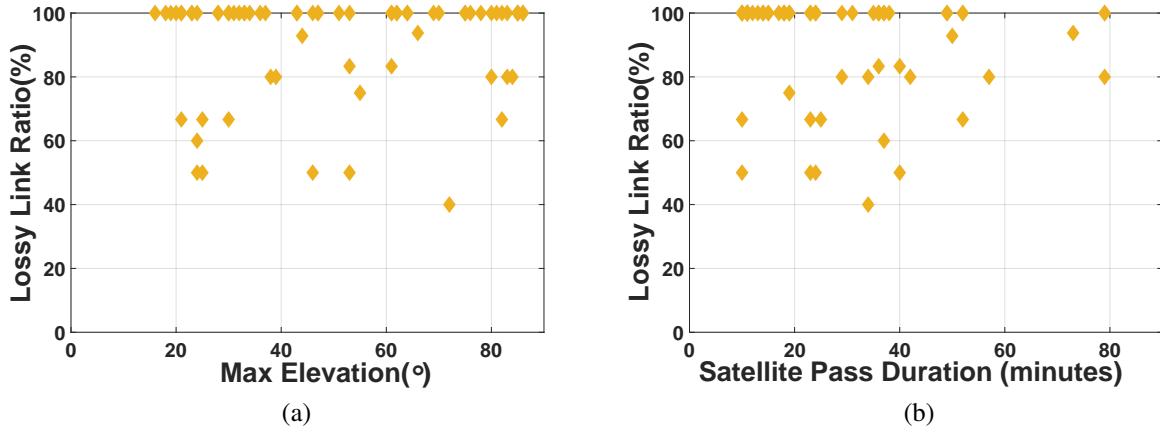


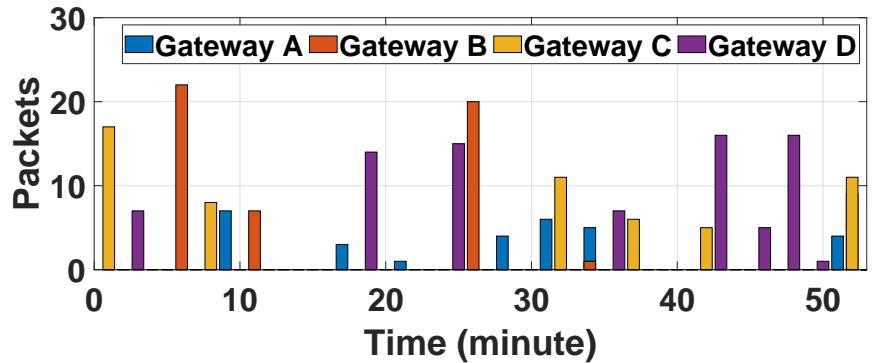
Figure 2.4 Lossy link ratio.

dio transmission performance is often degraded by failed transmissions. This inefficiency is further exacerbated by retransmissions, particularly when the communication channel deteriorates. “Lossy link” indicates the link condition when a downlink data beacon can be received while the uplink data packets may suffer loss in the following transmission windows. Using a software-defined radio, we detected 6 signal pulses on the uplink spectrum. COTS SWARM fixes the uplink transmission of 6 consecutive packets as the default setting to match its temporal dynamics. COTS radios set this fixed value because more attempts may result in high chance of transmission failures while fewer attempts might waste good transmission opportunities.

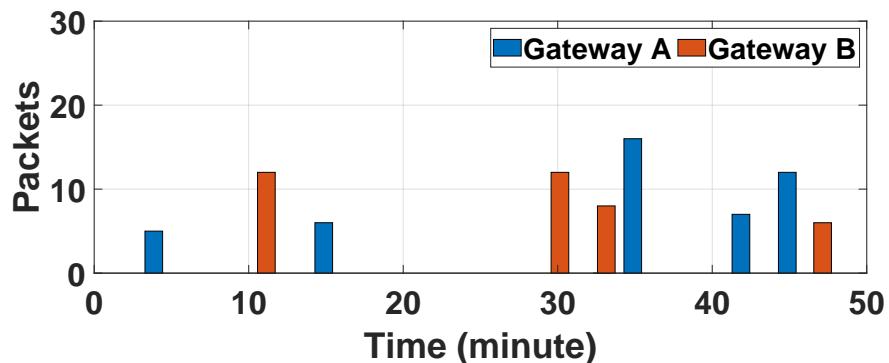
During the same satellite pass, the lossy link ratio is defined as the proportion of 1-minute transmission windows experiencing lossy links to the total number of transmission windows during the pass duration. According to our collected data trace, Figure 2.4 illustrates the lossy links ratio under different maximum elevation and pass duration settings. We can observe that 71% satellite radios experience 100% lossy links ratio, leading to significant energy wastage, especially in satellite communication radios with high energy consumption. This motivates us to predict transmission capability to reduce energy waste.

2.2.5 Geo-spatial Distributed Radios

As mentioned in Section 2.2.3, too many factors can influence the satellite communication link. The link of different geospatial locations can be variable and dynamic. To better understand the



(a) Data transmission performance of 4 gateways in each minute.



(b) Data transmission performance of 2 gateways in each minute.

Figure 2.5 Comparisons of transmitted packet number at each minute between two gateways at different distances.

link dynamics across multiple locations, we compared the performance of multiple satellite radios operating concurrently to measure the maximum transmission capability as illustrated in Figure 2.6 and Figure 2.5.

Observation 3: Dynamic spatial and temporal links. We analyzed the performance during each minute to further understand the packet transmission dynamics, as illustrated in Figure 2.5. The bars represent the number of packets successfully sent to the satellite during a 1-minute time window. In Figure 2.5a, the satellite radios of 4 gateways transmit packets alternately over time, with primarily non-overlapping periods. In most cases, only one gateway can connect to the satellite for one minute. Figure 2.5b further reveals that the two gateways' sending periods do not overlap at all, occurring at different times separately. This suggests that transmission capabilities vary by location and time.

Moreover, we analyzed the overall performance of 4 gateways arranged in a square formation,

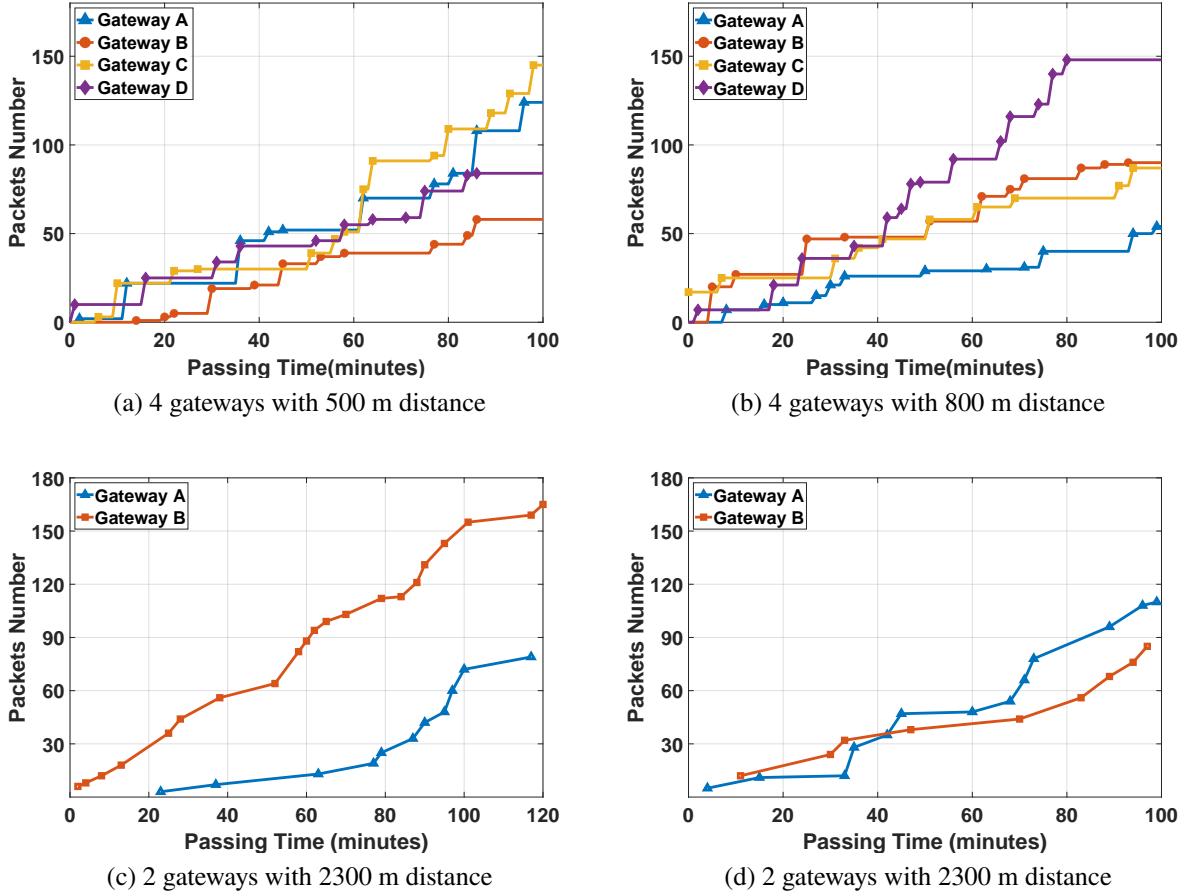


Figure 2.6 Comparisons of throughput with the same satellite pass among multiple gateways at different locations.

where each gateway was equidistant from its neighbors by either 500 m or 800 m with the same satellite pass duration. The slopes change over time for 4 gateways, demonstrating the temporal diversity. In Figure 2.6a, A, C, D exhibit varying leadership in overall transmission rates. Notably, B and D experience minimal data transmission for about 15 minutes, and C surges ahead after a 62-minute delay, maintaining its lead thereafter. As the observation period progresses, the performance gap among A, D, and B significantly increases, with A and B achieving $1.9\times$ and $1.5\times$ the packets of D. Conversely, Figure 2.6b shows D's progressive increase, with B and C alternatively leading to a similar trend, ultimately achieving comparable throughput. A maintains low data rates, managing only 54 packets over 100 minutes, $2.8\times$ less than D, which significantly falls short of transmission requirements.

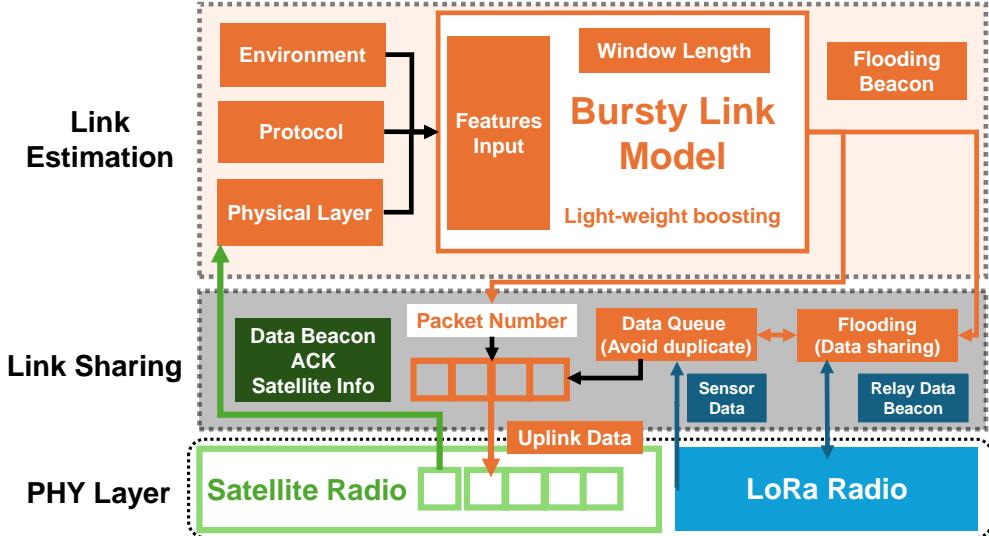


Figure 2.7 The network architecture in SateRIoT.

We also compare the performance of two terminals with 2300 m distance. Figure 2.6c illustrates that A begins its data transmission 20 minutes after B. And B progressively extends its lead over time. As a result, by the end of a 120-minute period, B has achieved approximately 2 \times the throughput of A. Conversely, Figure 2.6d presents a scenario where the performance disparity between the gateways narrows over the same distance. Although B maintains its lead initially, A catches up by the 43-minute, matching B's cumulative data transmission. Subsequently, A overtakes B and sustains this advantage up to the 100-minute, achieving 1.3 \times throughput compared to B. Collectively, these illustrations underscore the dynamic performances of satellite radio under varying temporal and spatial scenarios.

Temporal and spatial link dynamic creates varying transmission capabilities among satellite radios. Some satellite radio may struggle to send packets, while others have surplus capacity after completing the delivery of packets within their own covered ground area. Thus, the network throughput is degraded.

2.3 System Design

2.3.1 System In A Nutshell

In SateRIoT, a LoRa gateway is equipped with a SWARM radio to access the Internet through SWARM satellites. On the other hand, it can use its LoRa radio to communicate with other LoRa

gateways and collect sensory data from sensor nodes. All LoRa gateways operate in a distributed manner. Figure 2.7 illustrates the network architecture design of SateRIoT, which consists of three layers: link estimation, link sharing, and physical layer communication radio.

In the link estimation layer, the core design is the **bursty link model** (§ 2.3.2). First, we build a *lightweight model structure* (§ 2.3.2.1). The temporal burstiness of a link determines the short stable duration of the temporal links for data packet transmission. Based on our understanding of SWARM protocols, we design an ACK-triggered scheme to estimate the link, balancing model accuracy and agility to make it practical for real-time link estimation. Second, *features* (§ 2.3.2.2) are selected to be the most informative for accurate temporal link estimation. These features include information from the physical layer, environmental characteristics, and COTS satellite protocols. Third, we empirically select the longest *window length* (§ 2.3.2.3) to maintain consistent link quality while keeping the computation and system overhead low.

To combine the capabilities of multi-gateways, we design **link sharing** module to improve the overall network performance by exploring link spatial diversity. *the multi-hop flooding protocol* (§ 2.3.3.1) uses the LoRa radio to enable efficient network-wide data sharing of sensory data or flooding beacon. *Priority data queue* (§ 2.3.3.2) avoid duplicate transmission by setting priority order among data packets. The data packets in the priority data queue are managed by both the link model and the flooding protocol.

2.3.2 Bursty Link Modeling and Estimation

According to the observation of temporal lossy link, it suggests the up-link is short-term bursty, which means the link behavior (e.g., success or failure) is only stable in a varied short-time window. We design a bursty link model that estimates how many packets can be successfully transmitted in a stable transmission window. To guarantee the current transmission window is bursty for successful transmission and collect sufficient information to estimate the number of successful transmissions, instead of using satellite beacons to trigger a link estimation, we trigger the link estimation after an ACK is received, indicating a packet has been received at the satellite side. Moreover, we can collect additional critical features from the packet ACK.

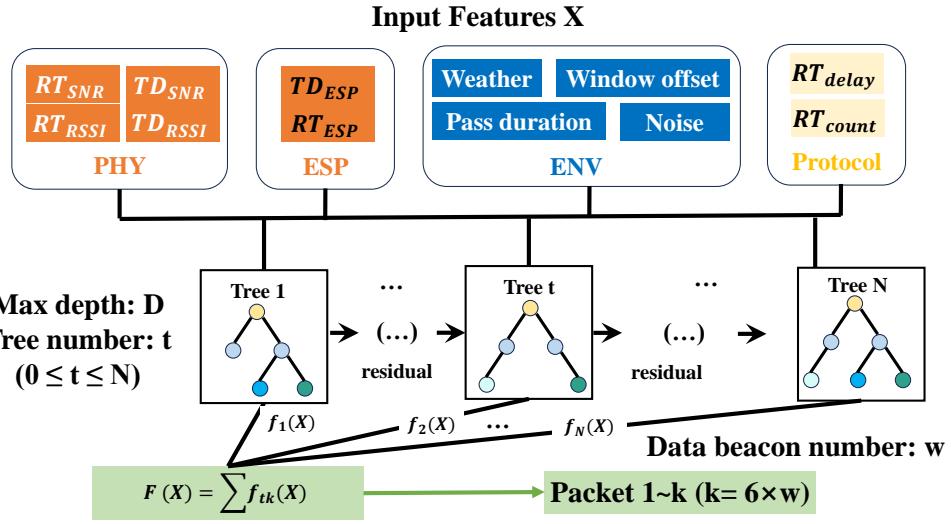


Figure 2.8 Boosting based link estimation model.

Specifically, based on our observation in §2.2.4, a gateway attempts data packet transmission once it has received a data beacon from a passing-by satellite. To estimate the current link quality, the gateway only adds a probe packet to its satellite radio queue. Once the gateway receives the acknowledgement of the probe packet from a satellite, we initiate a link estimation process to determine how many data packets should be set as pending status in a transmission window, whose time length is predetermined as T_{tx} , including multiple data beacons. From our measurement study, both the PDR and SNR remain stable across these consecutive data beacons. We leverage the short-term stability to perform link estimation for a transmission window to lower the link estimation overhead. When all pending data packets are transmitted once or the transmission window expires, regardless of whether all ACKs are received, we dequeue the left data packets from the satellite radio queue and insert a probe packet into the satellite radio queue again to trigger the link estimation of the next transmission window.

2.3.2.1 Light-weight model structure

To predict the versatile link within a given transmission window, we use boosting model architecture to build an effective, lightweight machine-learning algorithm for link estimation and scheduled packet prediction. Figure 2.8 illustrates our link estimation model architecture and transmission capability prediction. We utilize the four kinds of features as the input X and try to predict

1 to k implement classifications where $k = 6 \times w$, w represents data beacon amount given the duration of the current transmission window T_{tx} . The model $F_0(x)$ is initialized to the logarithm of class priors, i.e., for all x , $F_0(x) = [\log p_1, \log p_2, \dots, \log p_N]^T$, where p_k is the proportion of class k in the training set. The tree index in our boosting model is t from 1 to N :

1. For each class k , compute the pseudo-residuals:

$$r_{ik}^t = - \left[\frac{\partial L(y_i, F(x_i))}{\partial F_k(x_i)} \right]_{F(x)=F_{t-1}(x)}$$

L is the multi-class logarithmic loss. For a dataset of N samples, the total multi-class logarithmic loss is:

$$L = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K y_{ik} \log(p_{ik})$$

y_{ik} indicates whether sample i is in class k (1 if true, 0 otherwise). p_{ik} is the model's predicted probability that sample i belongs to class k .

2. For each class k , fit a new tree $f_{tk}(x)$ with r_{ik}^t .

$$f_{tk}(x) = \arg \min_f \sum_{i=1}^M L(r_{ik}^t, f(x_i))$$

3. Update the model: For each class k , find the coefficient γ_{tk} that minimizes the overall loss, and update

$$\gamma_{tk} = \arg \min_\gamma \sum_{i=1}^N L(y_i, F_{t-1}(x_i) + \gamma f_t(x_i))$$

$$F_N(x) = F_0(x) + \sum_{t=1}^N \sum_{k=1}^{6w} \gamma_{tk} f_{tk}(x)$$

During the training process, we adjust data or feature sample ratios to randomly selectively use data and features for each tree rather than employing the entire set. This random feature and data diversity makes the final model more robust, enabling it to better adapt to various data distributions. We can effectively reduce the risk of overfitting the training data, thereby enhancing the model's generalization ability. Our model balances uneven real data sets by giving more weight to under-represented categories during training. We also use random sampling to oversample/desample the data with different labels and balance the class distribution.

2.3.2.2 Link features

The input of the link estimation model.

Physical layer features: The successful transmission of the probe packet indicates a satellite data beacon and an acknowledgement have been received by the gateway. RT_{rss} and RT_{snr} indicate the RSSI and SNR of the received satellite data beacon. TD_{rss} and TD_{snr} indicate RSSI and SNR of the acknowledgement of the probe packet. The RSSI and SNR values represent the physical propagation property of the current down-link from the satellite to the gateway.

Expected Signal Power (ESP) feature: RSSI and SNR can be distorted by environmental noises. To focus on satellite signals, we combine RSSI and SNR to generate an ESP [8, 52] value, which indicates the signal attenuation along the propagation path. The ESP feature is calculated as follows:

$$ESP = RSSI + SNR - 10 \log_{10}(1 + 10^{0.1 \cdot SNR}) \quad (2.1)$$

Then we can get RT_{esp} and TD_{esp} with physical layer features.

Environmental features: We include several environmental features as follows: *Noise*: RSSI of background noise. This is measured by SWARM radio when no satellite appears. *Elevation*: The maximum elevation during a satellite pass can be found on the SWARM website [28] given the location of the gateway. *Passing duration*: The passing duration of a satellite is announced on SWARM website [28] according to the location of the gateway. *Transmission Window Offset*: The time offset between the current estimated transmission window and the time that the satellite starts to pass the area. The relative antenna position between the gateway and the satellite is different at different time offsets. *Weather*: Based on the weather released by local weather station, we use four-levels quantization to define the weather values from sunny to drizzle.

Protocol features: RT_{delay} : The time delay between the received data beacon and acknowledgement of the probe packet. RT_{count} : The number of satellite non-data beacons during a 30-seconds period before the acknowledgement of the probe packet.

2.3.2.3 Link window length

The predicted window length of our link estimation model is significant. Predicting long windows can lead to low link estimation accuracy since only 2% of the minutes have data transmission modes lasting more than 2 minutes. Consequently, selecting overly long observation windows can significantly degrade link estimation accuracy. Conversely, shorter windows necessitate more frequent transmission of LoRa beacons, which increases both energy consumption and computational overhead. Therefore, optimizing the window length is critical to achieving a balance between estimation accuracy and system efficiency. To maintain consistency and continuity with the transmission patterns, the window length must align with the natural data transmission cycles. Higher link quality and more reliable data transmission are indicated by consecutive data beacons. Inappropriate window lengths can disrupt these continuous transmission periods, resulting in a marked decrease in estimation accuracy. Thus, careful selection and tuning of window lengths are essential for improving link quality assessments and ensuring efficient network operation.

2.3.3 Link Sharing

2.3.3.1 Multi-hop Packet Flooding Protocol

The multi-hop flooding protocol provides a primitive networking method to enable network-wide packet sharing. The basic idea is that a LoRa gateway initiates a flooding process once it has a packet to share. There are two types of packets: data packets and flooding beacons, with the latter used to manage the data queue. A LoRa gateway immediately forwards the packet once it receives one from another LoRa gateway. A flooding packet starts from the original gateway. In the first round, the packet will be delivered to the closest gateways. Next, these gateways will keep relaying this packet to their next-hop adjacent gateways. Eventually, the packet can be delivered to all gateways [53, 54]. For IoT data collection systems in rural areas, flooding protocol is easy to deploy and implement in the low-cost IoT gateways without coordination overhead, avoiding complex network traffic patterns.

Carrier-sense based Collision Avoidance. It is possible that several LoRa gateways initialize multiple flooding processes in a short period. Additionally, several LoRa gateways may receive the

same packet and start to forward simultaneously. Without noticing others' packet transmission, the potential packet collision could degrade the reliability of the data sharing. It will be worse in dense network deployment with larger flooding scale. To solve this, We adopt channel activity detector (CAD) on COTS LoRa radio chips [55] to enable low-cost carrier-sense-based collision avoidance. Before a LoRa gateway forwards a received packet, it will wait for a random initial backoff, then repeat carrier sense until the channel is clean.

Flooding Beacons. The link model of a gateway triggers two types of beacons to synchronize the data packets among all gateways with the flooding protocol. Firstly, when the gateway determines which packets will be transmitted in a transmission window, it sends out a beacon including information on these packets. Secondly, when the transmission window ends, the gateway sends a beacon indicating which packets have been successfully transmitted.

Network Consistency. In case a LoRa gateway misses a packet from other gateways due to LoRa link dynamics [8]. Each LoRa gateway maintains the status of its local buffered packets and broadcasts the status with the schedule of a Trickle timer [56]. If a gateway receives others' status and finds an inconsistency with its local status, it will request other gateways to send the missing packet. The trickle timer will be reset when a gateway receives a request. This strategy offers both a systematic approach to maintaining information consistency and a proactive method to resolve potential mismatches in a distributed environment. In this way, all LoRa gateways consistently buffer all packets and receive the flooding beacons from others for later global packet transmission scheduling.

2.3.3.2 Priority Data Queue Management

Priority and data enqueue: The priority data queue structure includes a self-generated data queue and a relay data queue. The self-generated data queue has higher priority than the relay data queue. Since self-generated data packets are unique across different gateways, duplicate packets will be prohibited when multiple gateways simultaneously send data packets to a satellite with non-collided frequency hopping. We do order shuffling for the relay data queue. Namely, when a gateways received a relay data packet, it will insert the data packet to a random position in the relay

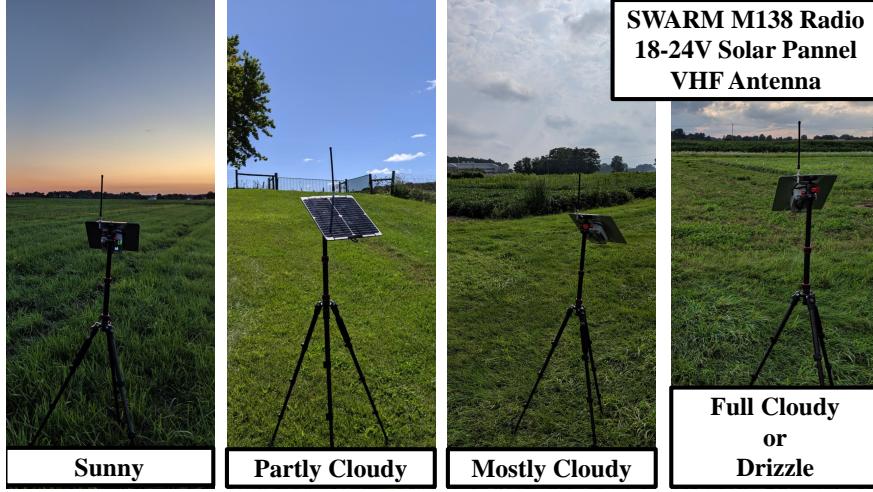


Figure 2.9 Outdoor deployment of satellite IoT radio with different weather.

data queue. In this way, when multiple gateways concurrently transmit data packet, the number of duplicate packets will be further reduced since they will forward the relayed data in different order.

Packet holding and releasing When a gateway receives a flooding beacon from others indicating the data packets they will transmit in the coming transmission window, the gateway will check its queue. For those identical data packets, the gateway will hold them for a holding period that equals two transmission windows $2T_{tx}$. The holding packets have no chance scheduled if a transmission window is coming. In the holding period, if the gateway receives another flooding beacon (§2.3.3.1) indicating the holding packets are successfully transmitted, it will dequeue them directly. When the holding period expires, the gateway will release them. In this way, we actively reduce potential duplicate transmissions. For each gateway, when a transmission window ends, it will dequeue packets whose ACKs have been received.

2.4 Implementation

Our methods are fully compatible with LoRaWAN and COTS satellite devices without additional hardware or centralized coordination. The data for the SWARMIoT only costs 5 USD each month [28], much less than the cost required for network infrastructure construction in rural areas.

Ourdoor SWARM satellite radio deployment: We employed 4 Swarm Eva Kits [50], each fitted with an M138 Modem [34], and powered them using 18-24V DC solar panels. Figure 2.9 shows the outdoor Swarm ground gateway deployment scenarios to collect data in four different weather

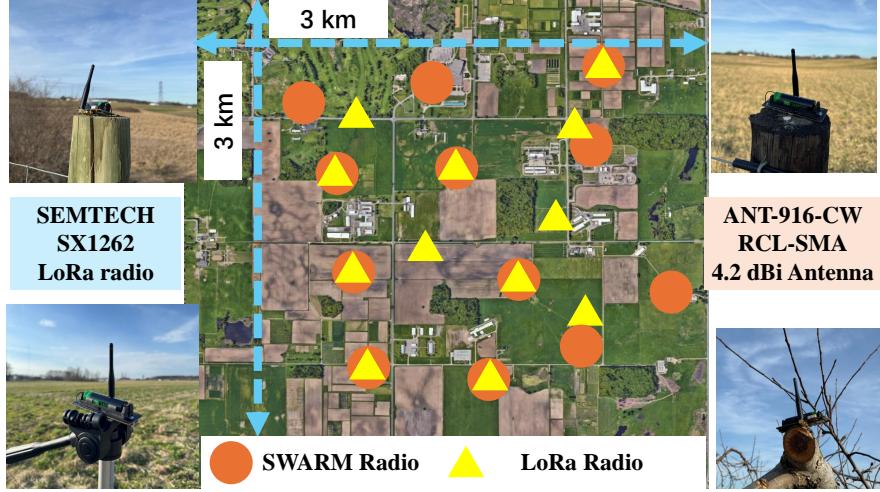


Figure 2.10 Outdoor deployment of flooding experiments and satellite radio locations.

conditions (e.g., sunny, partly cloudy, mostly cloudy, fully cloudy or drizzle, rain). Two similar weather conditions can coexist during the same time period in a close area. The locations of SWARM devices are shown as orange circles in Figure 2.10 around 9 km^2 rural area farmland. For the satellite radio, the length of a packet is 192 bytes that can contain multiple sensory data packets.

Outdoor LoRa radio deployment: Figure 2.10 depicts the outdoor deployment of LoRa radios in 9 km^2 farm zones. The yellow triangles represent locations. We deploy LoRa devices in elevated positions with open space, such as high fences, small trees, or on tripods at heights ranging from 1.5 m to 2.5 m. We use 4.2 dBi gain LPWA antennas [57] and 12 SX1262 [9] radios controlled by ESP32 MCUs [58] operating on US915 ISM bands with SF12 and 125kHz bandwidth. We use LMAC-1 [55] to enable CSMA for LoRa transmission.

Link estimation model: The predicted transmission window is set to 120 s when $w = 4$, which is the maximum duration observed with continuous data beacons with consistent link quality (§ 2.3.2.3). The model generates 330 trees, each with a maximum depth of 5. We use the objective function 'multi:softprob' to show each class's probability distribution and the Mean Squared Error (MSE) as the loss function. We design our model based on XGBoost [59] and SMOTE [60]. The inference time per sample is 0.373 milliseconds on a Raspberry Pi 4 Model B [61]. Given the lightweight nature of the model, it is particularly well-suited for deployment on resource-constrained devices. We can anticipate exceptionally rapid inference times when the model is deployed at LoRa gateways.

2.5 Evaluation

Performance metrics: To evaluate the overall performance for uplink transmission, we focus on the cumulative number of delivered packets as **Throughput** and the time from packet generation to successfully delivered for each packet as **Latency**. For the link estimation model’s performance, we rely on **Accuracy** and **Mean Absolute Error (MAE)** as metrics. We measure the performance of individual gateway using **Energy Efficiency**, represented as actual transmission attempts for each packet. For multi-hop flooding protocol performance, we employ **Latency** as the key indicator to present the packet delay among multiple gateways.

Baseline method: **1. COTS:** We use the existing COTS protocol and network architecture of Direct-to-satellite IoT devices as our baseline. Each gateway directly sends all the packets from their satellite communication radio’s transmission buffer without any link estimation or link sharing. This is evaluated and compared in energy efficiency evaluation (§2.5.2.2) and uplink data transmission (§2.5.1). **2. LDB** link estimation model: For link prediction, we only use the latest data beacon (LDB) to guarantee agility at first. The input features include RSSI, SNR, ESP of data beacon, noise, weather, satellite elevation, pass duration, and transmission window offset. **3. ENV** link estimation model: We build a linear regression model to predict the total delivered data amount during one satellite pass duration with input from weather, satellite elevation, pass duration and location index. We use the ENV model to compare energy efficiency performance (§2.5.2.2). **4. SateRIoT-w:** We use different link window lengths as our baselines. We select the w (as mentioned in Section 2.3.2.2) values of 1,2,3,5 data beacon numbers to compare the different SateRIoT scheme.

Default settings: The default packet generation frequency of each gateway is 1 packet per minute. The packet includes multiple sensory data frames from its covered area. The default weather value is 2, partly cloudy.

2.5.1 Overall Performance

In this section, we evaluate the overall performance in throughput and latency with SateRIoT and COTS protocol. The delay and possible duplicate caused by link sharing are considered in

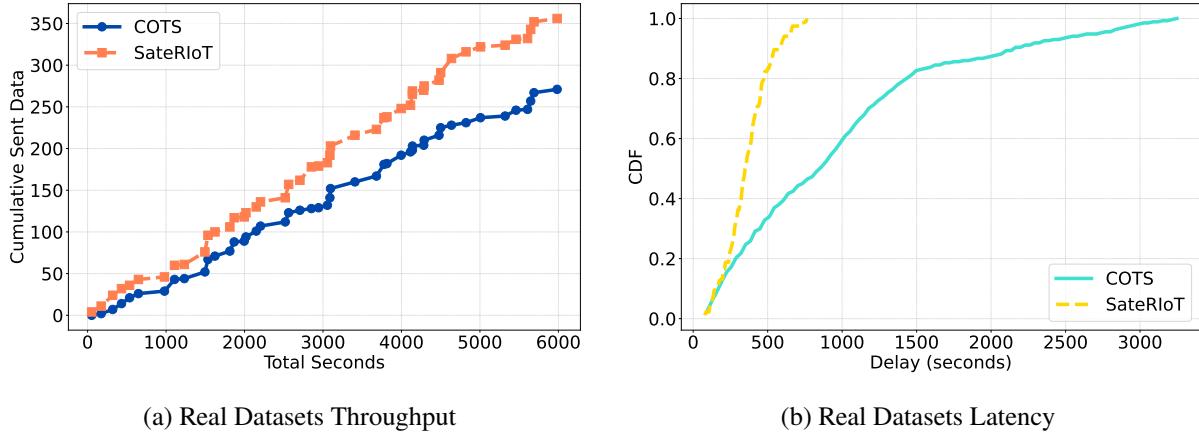


Figure 2.11 Performance in real data of 4 gateways

calculating the latency and throughput according to the results from §2.5.3. Experiment A uses real data collected simultaneously from 4 gateways in different outdoor locations. Experiment B employs trace-driven datasets from 12 gateways to conduct emulation, involving 4 gateways transmitting concurrently during 3 different time periods with similar satellite orbits and weather conditions. Experiment C simulates 12 gateways, with ground-truth data generated by our accurate link estimation model.

A. Real datasets experiments We use data collected outdoors over 100 minutes from 4 gateways in §2.2.5 as ground-truths. We use our link estimation model and real collected information as input to predict the link with real and emulate the throughput and latency. Figure 2.11a depicts the throughput variations over time for all four gateways. As time progresses, the throughput disparity between SateRIoT and the COTS method widens. After 100 minutes (6000 seconds), SateRIoT can achieve $1.31\times$ cumulative sent data packet compared to COTS protocol. Additionally, Figure 2.11a details the latency from packet generated time to delivery time. It is obvious that SateRIoT offers substantially shorter latency compared to the COTS approach. Specifically, around 80% of the packets can be transmitted in less than 8 minutes, compared to the COTS method, which requires up to 24 minutes to achieve the same level of packet transmission. This results in SateRIoT being $3\times$ faster. All the packet in SateRIoT reaches a delay of less than 750 s while COTS extends to 3250 s. This indicates that the maximal latency of COTS can be up to $4.3\times$ longer than SateRIoT.

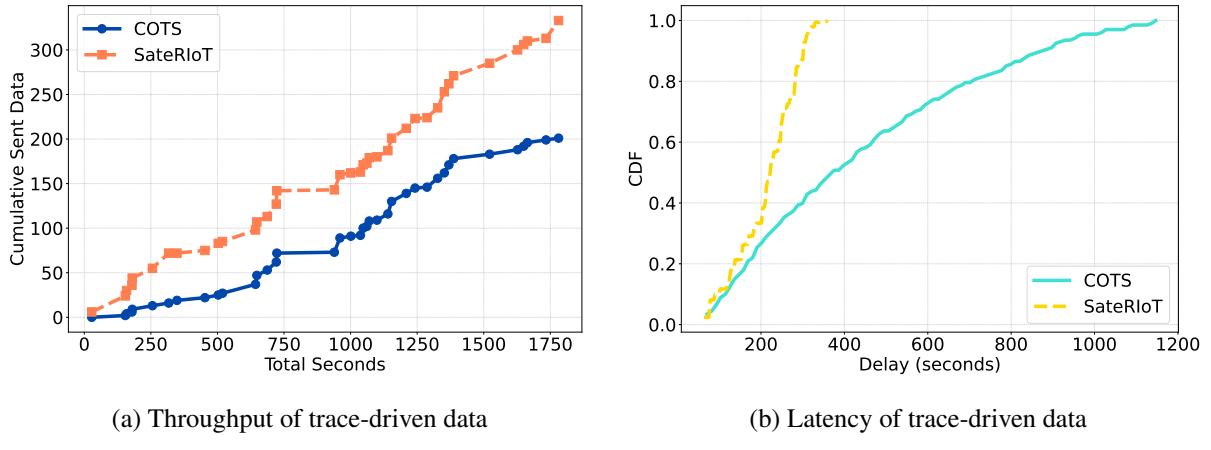


Figure 2.12 Performance of trace-driven 12 gateways.

These findings affirm the superiority of SateRIoT in terms of throughput and latency in real-world scenarios. The overall performance can be further improved when more gateways join the uplink transmission.

B. Trace-driven experimental settings: To evaluate more gateways performance with SateRIoT, We enlarge the gateways number to 12. We use 4 satellite radio devices as a group. Each deployed 3 times to emulate scenarios where gateways in 12 locations attempt to schedule data transmissions during the same satellite pass. We select 3 satellite traces that were closely matched in terms of elevation and pass duration. The maximum elevations and pass duration are 71° with 31 minutes, 67° with 29 minutes, and 69° with 30 minutes, respectively. In each satellite trace, we deploy 4 gateways to collect real data sets from 4 distinct locations. The locations of the gateways vary across the 3 satellite passes, ensuring coverage of 12 unique locations in total. This allows us to effectively emulate a scenario where 12 gateways concurrently connect to the same satellite.

Results: Figure 2.12a shows that SateRIoT transmits 332 packets, which is $1.65\times$ more data than the COTS baseline. Over time the data transmission gap between them widens because the baseline focuses on individual transmissions, missing the opportunity to use the connection time fully. SateRIoT efficiently schedules data and controls traffic, maximizing the use of the transmission window across all gateways with a reliable connection. Figure 2.12b reveals that SateRIoT delivers all packets to the satellite within 357s, while the COTS protocol takes up to 1147s. Therefore,

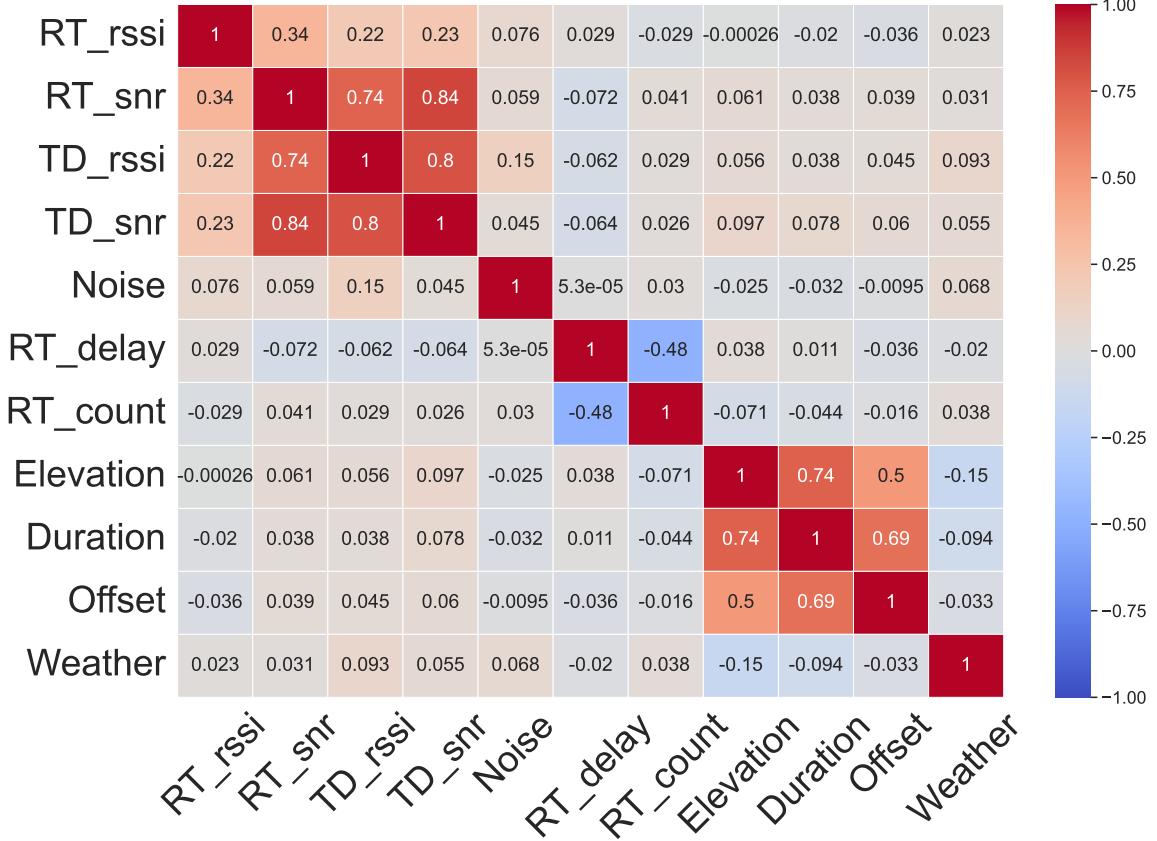
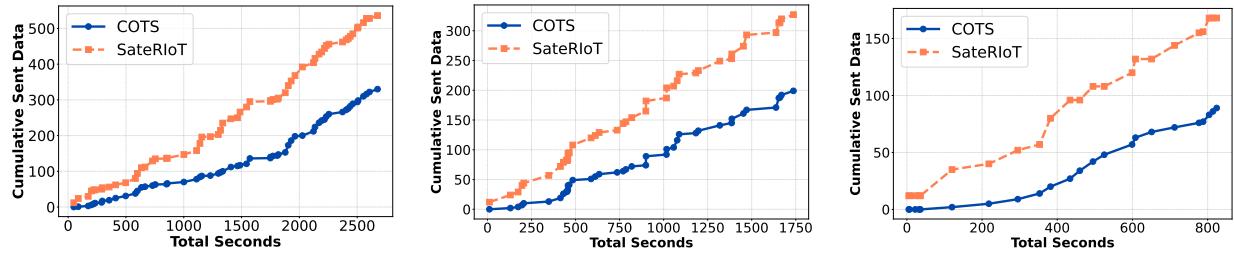


Figure 2.13 Correlation heatmap of input features.

SateRIoT is up to 3.4× faster, sending 80% of its packets in 280s compared to 700s for the COTS.

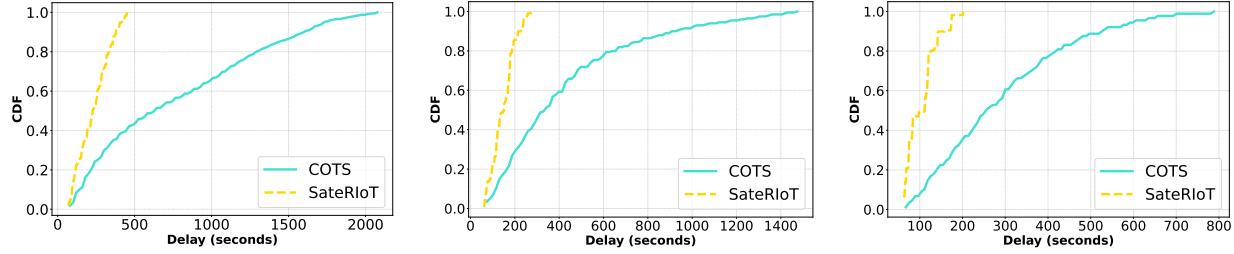
C. Generative datasets: Data availability for situations where elevation, pass duration, and weather conditions are extremely similar is severely restricted. In order to broaden the scope of our emulation experiments to encompass a wider range of satellite orbits and weather conditions, we deployed 2 or 4 satellite radios at 12 different locations, each tracking multiple satellite passes. This allows us to generate datasets for simulating uplink data transmission among multiple satellite radios during the same satellite pass. Figure 2.13 provides a clear overview of the interrelationships within the original features directly recorded from collected real datasets. We can easily observe that RT_{rss} , RT_{snr} , TD_{rss} , TD_{snr} exhibit strong positive correlations, while RT_{delay} and RT_{count} display notable negative correlations. Additionally, pass duration, transmission window offset, and weather also demonstrate noteworthy positive correlations with each other.

Based on the above observations, we establish a process to determine feature values using their



(a) Throughput of 12 gateways with max elevation 81° in 45 min. (b) Throughput of 12 gateways with max elevation 66° in 30 min. (c) Throughput of 12 gateways with max elevation 28° in 14 min.

Figure 2.14 The throughput performance of generative data sets.



(a) Latency of 12 gateways with max elevation 81° in 45 min. (b) Latency of 12 gateways with max elevation 66° in 30 min. (c) Latency of 12 gateways with max elevation 28° in 14 min.

Figure 2.15 The latency performance of generative data sets.

inherent physical significance and a correlation heatmap. Initially, we randomly pick the maximum elevation from real datasets and then choose a pass duration that aligns with this elevation. We randomly generate varying frequencies of connection time windows and set start times from collected real datasets, considering the same maximum elevation and pass duration. Second, We randomly select a value for RT_{snr} from the real datasets. Then we randomly choose a TD_{snr} from the subset of real datasets that match the chosen RT_{snr} . Following this rule, based on determined RT_{snr} and TD_{snr} , we randomly select RT_{rss} and TD_{rss} values. Third, we randomly chose RT_{delay} from collected datasets, followed by a random RT_{count} based on consistent RT_{delay} . In addition, we randomly set background noise RSSI from -106 dB to -87 dB ranging as real noise level.

Generative datasets experiments: We conduct further experiments to validate performance in various satellite scenarios with the generative datasets. We set the maximum satellite elevations to 81°, 66° and 28° with corresponding pass durations of 45, 30, and 14 minutes at 12 different locations. Figure 2.14a, Figure 2.14b, and Figure 2.14c demonstrate that SateRiot can achieve

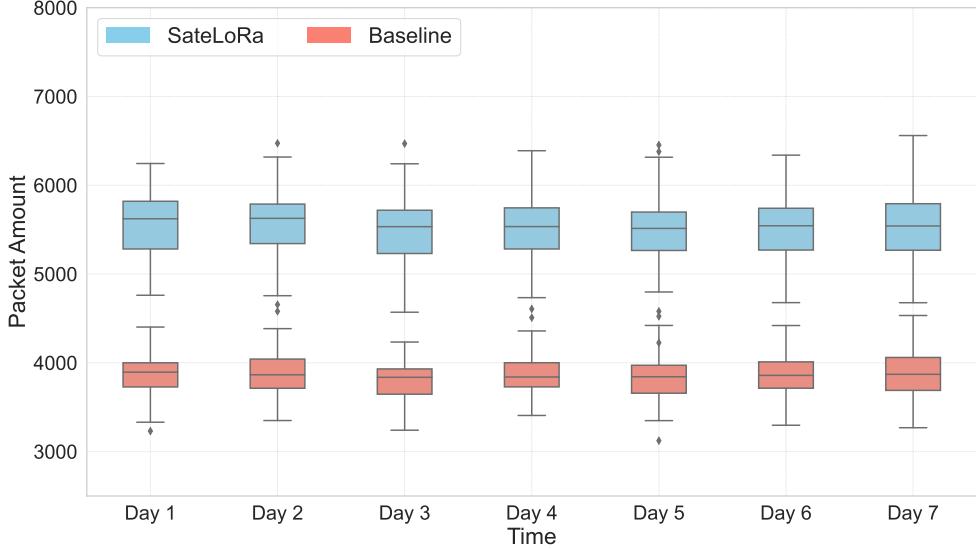


Figure 2.16 Throughput of large-scale simulation.

1.6×, 1.7× and 1.9× the data amount of baseline during one satellite pass. the data volume of the baseline within a single satellite pass, respectively. Meanwhile, Figure 2.15a, Figure 2.15b, and Figure 2.15c highlight that SateRIoT can achieve packet delivery latency that is 4.6×, 5.6× and 3.93× lower than the COTS protocol at most, respectively.

Large-scale simulation: We further simulate long-term application in 7 days. We select elevation and pass duration from the real satellite orbit and emulate the performance of multiple satellite passes using the generative datasets model. We repeat the 7-day experiment 100 times to obtain large-scale simulation results to discover the performance gain of SateRIoT further. The boxplot figures as shown in Figure 2.16. The accumulative packet volume of SateRIoT consistently exceeds the COTS baseline over the 7 days, maintaining a steady range approximately from 1.5 to 2.0 ×.

2.5.2 Link Estimation Model

In this section, we compare the performance among different link estimation trigger schemes and different link estimation transmission window lengths. The ground truths for link estimation during transmission windows are derived from local logs of real outdoor experiments. Then, we use the link estimation model and input feature derived from the logs to predict transmission link capability by comparing it with the real link. MAE is derived by comparing predicted values from the link model to ground truths. We use $w(windows)$ 1,2,3,4,5 to represent the data beacon amount

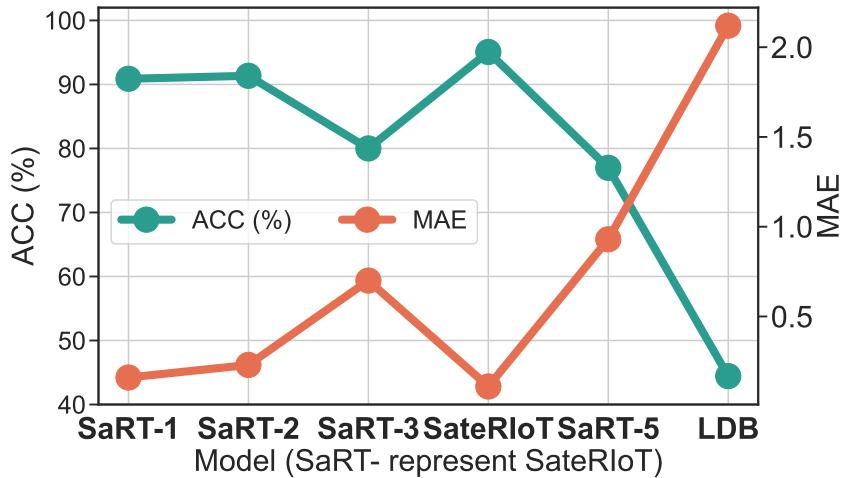


Figure 2.17 ACC and MAE of different link models.

during one transmission window.

2.5.2.1 Accuracy

We split our datasets randomly with 20% of the total as test datasets. The predicted accuracy of window 4 can be up to 95.07% ACC with only 0.11 MAE. The feature importance of TD ESP and RT ESP features are 0.16 and 0.13, respectively. This affirms the significance of our feature engineering. The results affirm the reliability of our link estimation model, which can enhance the performance of individual gateways and optimize overall data transmission.

Comparison among different models: The comparison results are shown in Figure 2.17. the accuracy is only 44.44% and 2.12 MAE for the LDB model. This suggests that the link information from the data packet acknowledgement is necessary to guarantee accuracy. The ACCs of window 1,2,3,5 are 90.88%, 91.35%, 80.00%, 76.98% respectively. The MAEs of window 1,2,3,5 are 0.16, 0.23, 0.70 and 0.93 respectively. The performance for SateRiot-1 and SateRiot-2 are similar to SateRiot-4, whereas SateRiot-3 performs poorly as it may interrupt continuous data pattern that should have been captured in window 4, leading to inaccurate predictions of the last data beacon with 18 classifications. When the window length exceeds 4, performance declines because only 2% of transmission periods last longer than 120 seconds. This highlights the importance of selecting an optimal window length to balance accuracy and efficiency. Based on empirical testing, we have selected w=4 as our optimal window length.

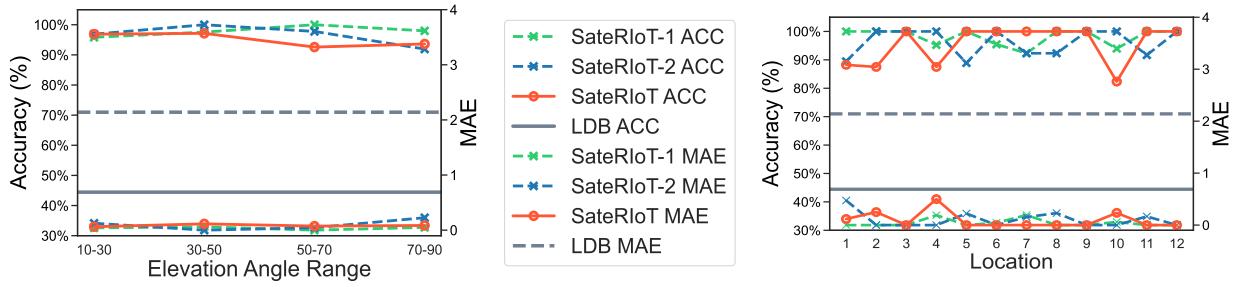


Figure 2.18 ACC and MAE of different models with various elevation and locations

Performance with different maximum elevation and locations: To verify the performance in different scenarios, we evaluated our model’s performance in 12 locations and multiple satellite elevation values. Figure 2.18a displays the distribution of ACC and MAE for various groups of maximum elevation angles, categorized into "10-30", "30-50", "50-70", and "70-90" degrees. All groups achieve more than 91% accuracy and a very low MAE of less than 0.3. This suggests that our link estimation model is versatile and functions effectively across different elevation angles and for satellites with varying orbits. Figure 2.18b presents the distribution of accuracy and MAE for link estimation performance across 12 locations within a farm. Notably, our model achieves nearly 100% accuracy and zero MAE at 8 locations. Three locations surpass 87% accuracy, and one location achieves 82%, with all MAEs remaining below 0.7. This deviation is within an acceptable range and has minimal impact on the estimations and gateway performance. These results highlight the robustness and reliability of our bursty link model across different elevations and geographic locations.

Remark: Overall, the versatility of our model across varying elevation angles, combined with its robustness in different geographic locations, demonstrates its efficacy in diverse operational conditions. Balancing window length and triggering post-ACK receipt are key strategies to maximize link estimation accuracy and system efficiency in LoRa networks. To improve adaptability and scalability, it can be further trained in more complex environments.

2.5.2.2 Energy Efficiency

Energy experimental settings: We conducted experiments using real datasets mentioned in §2.2.5 as test datasets. As outlined in Section 2.2.4, the COTS method attempts to transmit 6 times after

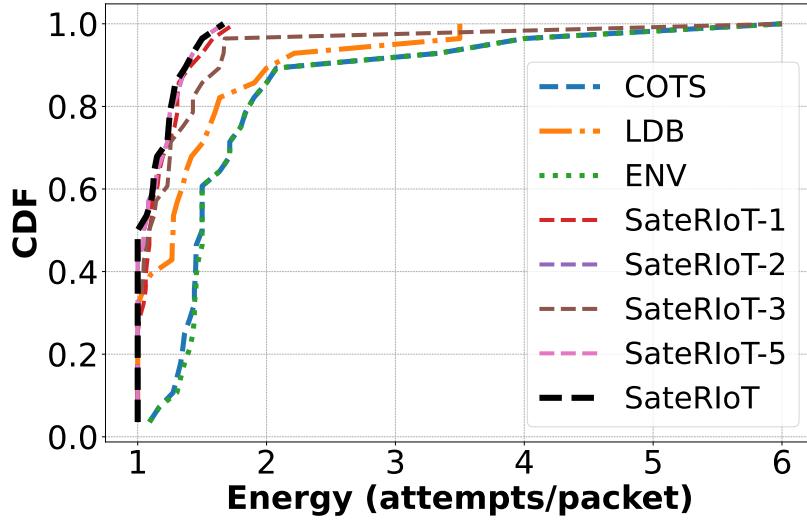


Figure 2.19 CDF comparison of energy efficiency and link waste.

the gateway receives each satellite data beacon. It continues attempting transmissions whenever a satellite data beacon arrives. **ENV** stops the transmission if the cumulative packet amount reaches its predicted value during the satellite pass duration. **LDB** predicts the data packet transmission capability after one data beacon arrives and schedules predicted attempts. SateRIoT and SateRIoT-n stop after successfully transmitting a predicted number of packets. Such an approach helps reduce energy consumption by avoiding unreliable links. We predict and calculate the average attempts for each packet as an energy efficiency metric during one satellite pass duration. Then, we count the energy efficiency distributions from multiple satellite passes to compare the performance.

Results: Figure 2.19 compares the energy efficiency distributions between SateRIoT and the baseline methods. SateRIoT consistently requires fewer than 1.65 attempts per packet across all scenarios, in contrast to LDB, ENV, and COTS, which require up to 3.5, 6, and 6 attempts per packet, respectively. For 80% of the cases, SateRIoT's model prediction keeps energy efficiency under 1.25, while LDB and ENV, along with COTS, reach 1.6 and 1.85, respectively. Over 50% of the cases with SateRIoT transmit packets without extra energy waste, whereas LDB achieves this in 32%. For each packet at one gateway, averagely, SateRIoT attains up to 3.3× or 28.15 J lower energy usage than the COTS protocol during a single satellite pass. We can also observe that the model with different window lengths exhibits a similar trend in energy efficiency, but SateRIoT performs

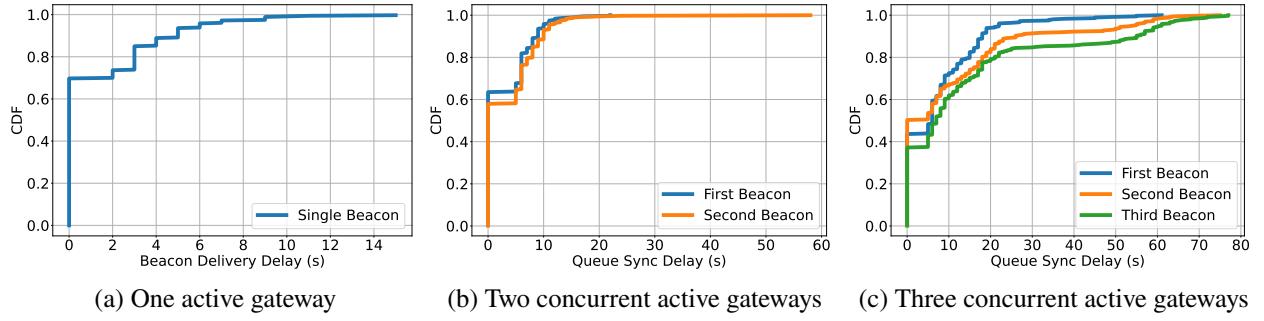


Figure 2.20 The latency distribution of flooding beacons in multi-hop protocol.

slightly better than other SateRIoT-n solutions thanks to the higher accuracy. This demonstrates that SateRIoT, unlike the more generalized link prediction methods like LDB or the COTS protocol, utilizes precise link estimation for energy-efficient transmissions. The overall energy saving will increase as the number of gateways grows.

2.5.3 Multi-hop Flooding Protocol

Beacon flooding experiments: We conducted experiments to evaluate the performance of flooding protocol in terms of reliability and latency. Initially, we designated one gateway in active mode and initiated its beacon broadcast according to its unique transmission plan to all other gateway using our CSMA-enabled multi-hop protocol. Then, we recorded the received time for all packets. This process was repeated 10 times for each gateway, designating each one as the active gateway in turn. For a setup of 12 gateways, we obtained latency data from a total of 30 experimental runs.

One active gateway results: Figure 2.20a depicts the overall latency distribution for all the LoRa radios. The PDR is 100%. Approximately 70% of the standby gateways can receive the flooding beacon from the active gateway within a latency of less than 2 s. Furthermore, around 94% of the standby gateways can capture the flooding beacon from the active gateway within a 5-second window. This indicates that in 94% of cases, beacon dissemination from two active gateways with a time offset of larger than 5 s can prevent packet duplication. In our real data sets measured in § 2.2.5, instances where a concurrent time offset of less than 5 s occur in less than 2% of all effective transmission slots. This latency level is sufficiently low to facilitate timely data transmission.

Concurrent flooding beacons experiments: We seek to understand the performance when two

gateways broadcast their beacons concurrently with a random time offset ranging from 0 s to 5 s. We documented both the arrival time of the first beacon and the arrival times of both beacons at each gateway. Moreover, to examine the extreme scenarios where three gateways become active in close succession, we orchestrated an experiment where three beacons were broadcasted in rapid succession with a random time offset ranging from 0 to 5 seconds, either between the first and second beacon or between the second and third beacon. We record the arrival times of three beacons for each gateway. Such instances accounted for less than 0.1% of our emulation data. This three-beacon experiment was also repeated 120 times.

Latency Results: Figure 2.20b illustrates that approximately 90% of the gateways can receive both beacons under 9 s, which may result in a maximum of 4 duplicate packets. Additionally, around 60% of the gateways can receive both beacons within 5 s. Figure 2.20c indicates that around 60% beacons can be received with a delay of up to 20 s. around 60% beacons can be received with a delay of up to 9 s. When comparing three figures in Figure 2.20, it's evident that the time required to reach approximately 80% of gateways approximately triples with the addition of each beacon.

Remark: The effective performance of the CSMA-enabled multi-hop flooding protocol enables SateRIoT to operate efficiently. Its impressive time tolerance within 5 s ensures that concurrent cases are infrequent and hardly impact overall data transmission or create duplicate issues.

Overall Energy Analysis: SateRIoT achieves a comparable throughput for each individual gateway while maintaining approximately half the energy consumption of the baseline method. The propagation path between sensor nodes and satellites is usually hundreds of kilometers long, dramatically increasing the energy consumption to transmit the same amount of data. For example, the power consumption of SWARM-M138 [34] modem is 12.24 J in Tx mode, while Semtech SX1262 LoRa radio [9] only consumes from 0.002 J to 0.045 J [62] with a 192-byte packet from SF7 to SF12, which is 6120 \times to 272 \times less. In the busiest situations with the most power consumption mode with SF12, one satellite transmits one packet; this requires data sharing with LoRa radio transmission at most 12 times. The energy consumption of the LoRa packet is much lower than the power consumption of satellite radio.

2.6 Related Work

Satellite Networking: L2D2 [25] presents a distributed scheduling system to reduce latency for downlink data transmission with hybrid ground stations. Umbra [63] proposes a withholding scheduling scheme for backhaul from satellites to large ground stations to overcome the uneven queueing effect. Li et al. [64] evaluate that the orbit of COTS satellites is unpredictable because of collision avoidance for Starlink [45]. However, SateRIoT does not heavily rely on orbit parameters but instead on coarse elevation and duration information. STARRYNET [65] builds an open-source experimental framework to simulate complicated network behaviors. SpaceCore [26] designs a stateless architecture to enable 5G deployment at satellites. Serval [66] enables near-real-time insights for latency-sensitive imagery applications by emerging computational capabilities on the satellites and ground stations. Compared to these works, SateRIoT focuses on the low-cost satellite IoT in rural areas.

Satellite-based IoT: The research work on satellite IoT mainly focuses on network modeling and physical layer design. Fraire et al. [67] present a sparse direct-to-satellite constellation design combined with LoRa. Zhang et al. [68] utilize Bernoulli–Rician message to enable channel estimation and user activity detection. Qian et al. [69] proposed a symmetry chirp spread modulation. Spectrumize [70] utilizes satellite movement-induced Doppler shift as a unique identifier for detection and decoding. In contrast, SateRIoT designs and implements a network backhaul for rural area IoT by space link.

2.7 Conclusion

To conclude, we introduce SateRIoT, a novel IoT backhaul architecture that merges LPWA technology on the ground with cost-effective IoT LEO satellites in space to support efficient rural area networking. First, we design a bursty link model to predict packet transmission capacity, reducing unnecessary data transmission. Next, we refine the model by selecting key features and optimizing the window length. To achieve link sharing, we develop a multi-hop flooding protocol to maintain data sharing among all gateways and use a priority-based queue structure to avoid duplicate transmissions. We implement SateRIoT with COTS satellite IoT and LoRa radios and evaluate the

performance on real deployment and real-world collected traces. The results show that SateRIoT can achieve $3.3\times$ less energy consumption for an individual gateway. For overall performance, SateRIoT reduces latency for packet delivery up to $5.6\times$ and improves overall throughput by $1.9\times$.

CHAPTER 3

DEMETER: RELIABLE CROSS-SOIL LPWAN WITH LOW-COST SIGNAL POLARIZATION ALIGNMENT

3.1 Introduction

Internet-of-Things (IoT) plays a critical role in precision agriculture [71–77], in which soil monitoring is an essential part. For example, understanding soil moisture levels in the crop root-zone area can benefit irrigation water efficiency [78]; in-soil nitrate monitoring leads to a deeper understanding of fertilization efficiency [75, 77, 79, 80]. Currently, commercial soil monitoring systems [81–83] deploy underground soil sensors while leaving communication modules on the ground. However, it is desirable to bury communication modules under the ground as well so that they will not interfere with other agricultural activities (e.g., mowing, harvesting, fertilization, irrigation) [84, 85]. Such *cross-soil communication* has become a key feature of agricultural IoT [86, 87] and has attracted significant interests in recent years [72, 88–92].

Considering the large-range and long-term deployment in rural farms, rather than ad-hoc wireless sensor networks (WSN) [85, 91, 93, 94], Low-power Wide Area Networks (LPWAN) naturally fit to enable agricultural IoT by embracing the wireless technologies featured by low energy consumption and long communication distance [95]. LoRaWAN [21] is a popular LPWAN technology operating on the unlicensed band. It supports long-range data collection from LoRa nodes (typically with sensors) to a gateway. Compared to other cellular LPWAN techniques (e.g., NB-IoT [6], LTE-M [96]), LoRaWAN provides a low-cost way to flexibly deploy new infrastructure in rural farm areas where the cellular infrastructure is not available. Some measurement studies [88, 97–99] have shown the feasibility of adopting LoRaWAN to achieve cross-soil communication. However, achieving a reliable cross-soil LoRaWAN in practice is not trivial. In our in-field measurement (§ 3.2.1), we observe that the soil condition changes (e.g., moisture, temperature) can lead to a significant signal-to-noise ratio (SNR) variation up to 19.2 dB over a cross-soil LoRa link. Moreover, according to our empirical study of the SNR variation in cross-soil LoRa communication (§ 3.2.2), compared to the variation of signal attenuation, polarization misalignment between the transmit-

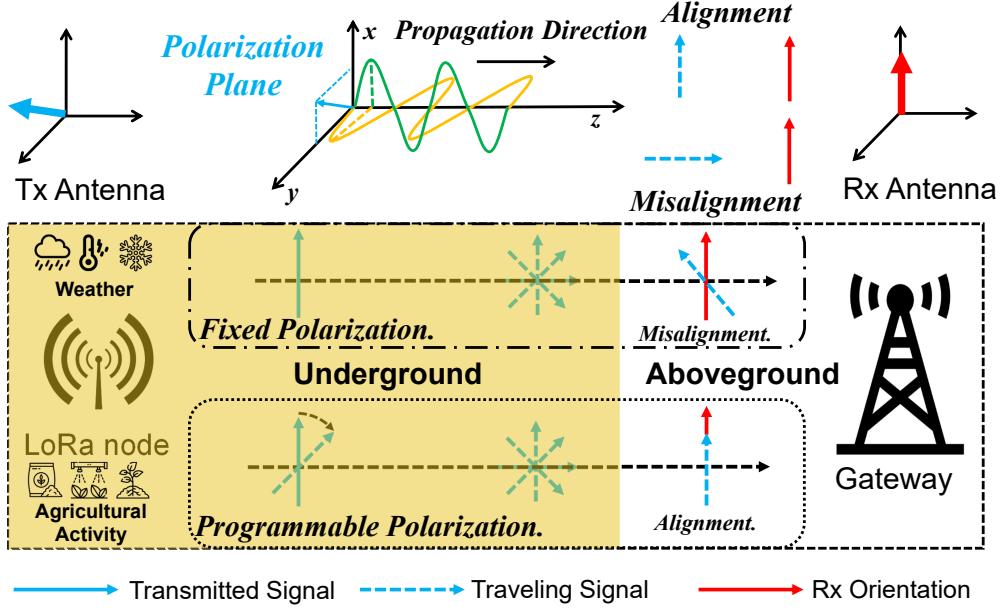


Figure 3.1 The illustration of polarization misalignment problem (top) and the idea of cross-soil communication and programmable polarization in Demeter (bottom).

ter and receiver antennas contributes more, leading to 13.2 dB SNR variation. For the available configurations on commercial off-the-shelf (COTS) LoRa nodes [100], such an SNR variation will shorten the communication distance by 4× (§ 3.5.2) and reduce energy consumption per day by up to 82% (§ 3.5.3).

The reason for cross-soil polarization misalignment is illustrated in Figure 3.1. LoRa nodes and gateways adopt low-cost linearly polarized antennas. The polarization (i.e., blue arrows) of an electromagnetic wave emitted from an antenna specifies the geometrical orientation of its oscillations. The received signal power can be maximized if the polarization of the received signal is aligned with the receiver's antenna orientation (i.e., red arrows). However, the soil's microscopic structure and components keep changing over time, subject to different environmental conditions, such as weather and agricultural activity. Consequently, as shown at the bottom of Figure 3.1, when an electromagnetic wave travels through the soil with complicated scattering, refraction, and reflection [101–105], its polarization changes over time. Even if the LoRa node's antenna and the gateway's antenna are aligned in advance, the cross-soil transmission can cause polarization misalignment between the received signal and the LoRa gateway's antenna, degrading its power

Table 3.1 Qualitative comparison between Demeter and alternative methods regarding COTS compatibility (COTS.), cross-soil reliability, polarization self-alignment capability, and protocol overhead in soil communication. “Pol.” and “Dir-Ante.” stand for “polarization” and “directional antenna”, respectively.

Challenge Method \	Cross- COTS. soil Reli- ability	Pol. S-alig. Capa- ble	Protocol Over- head	
Challenge Method	Cross- COTS. soil Reli- ability	Pol. S-alig. Capa- ble	Protocol Over- head	
Dual Pol. [107]	No	High	Yes	Low
Circularly Pol. [107]	Yes	Low	No	Low
Dir-Ante. [108, 109]	Yes	Low	No	Low
Scheduling [110, 111]	Yes	High	No	High
Demeter (Ours)	Yes	High	Yes	Low

significantly.

There is no prior work on cross-soil LoRa communications for the buried LoRa nodes to send their sensor data to a LoRa gateway over a long distance. We want a solution that is low-cost, interoperable with existing LoRa devices, reliable, and self-adjusting for polarization alignment, with low overhead, such that the LoRa nodes buried underground can operate uninterruptedly for a long time. For low cost and interoperability, we prefer a COTS-compatible solution. Specifically, the LoRa technology is commercialized by Semtech [106] only, with its proprietary radio chips, which all LoRa nodes and gateways in actual deployment use. We say a LoRa node (or gateway) is COTS-compatible if it can be made from the Semtech LoRa radio chips with other (possibly custom-designed) circuitry and computing/storage/signal processing components. A survey of relevant technologies from the literature shows in Table 3.1 that none of them is satisfactory in meeting all the challenges faced by cross-soil LoRa communications: COTS compatibility, reliability, polarization self-alignment, and low overhead, which we elaborate below.

Challenge 1: COTS Compatibility. There is prior work using a dual-polarized antenna (consisting of two orthogonal polarization antennas) and two RF chains to fully collect the energy of arbitrary polarization signals [107]. Although the work was not directly on LoRa, it is conceivable that its method could in principle be applied on a LoRa gateway to receive polarization-misaligned signals from an underground LoRa node. However, this method is not COTS-compatible because all existing COTS LoRa gateways [112–114] only have one RF chain in their radio chips, not two

RF chains required in [107]. Hence, not only does this method carry a higher cost, but it is not applicable to the existing deployments using COTS gateways.

Challenge 2: Cross-soil Reliability. In aboveground and open-space communications, LoRa nodes can utilize circularly polarized antennas, which generate circular polarization signals, to mitigate the polarization misalignment problem [107, 115]. However, the signal reflection, refraction, and scattering of under-ground communications are much more complex, leading to dramatic axial ratio distortion of circular polarization [116–119]. This distortion means the wave is no longer circular polarization and can result in a severe signal loss at LoRa gateways, degrading communication reliability [118–123].

Challenge 3: Polarization Self-alignment Capability. Low-cost electronically switchable directional (ESD) antennas [108, 109, 124, 125], providing antenna gain in a certain direction, can be equipped on LoRa nodes to enhance SNR. However, these directional antennas carry a higher cost in the LoRa band, and they are not capable of mitigating polarization misalignment. Moreover, the antenna gain from low-cost ESD antennas in the low-frequency band is usually too limited to tolerate the observed 13.2 dB SNR variation completely [126–130].

Challenge 4: Protocol Overhead. LoRa nodes can estimate cross-soil links and schedule transmissions under favorable conditions for good polarization [110, 111]. However, considering the continuously changing soil conditions, maintaining an accurate estimation and scheduling the transmissions may bring considerable computation and energy overhead, particularly for low-cost LoRa nodes.

To address the above challenges, in this paper, we propose Demeter, a low-cost and low-power polarization-programmable antenna system on COTS-compatible LoRa nodes to enable reliable cross-soil LoRa communication. It is inter-operatable with the LoRa gateways in existing deployment. Our key idea is to design a single RF-chain compatible circuit to dynamically adjust the initial polarization of LoRa node transmission. As shown at the bottom of Figure 3.1, considering the polarization changes during underground signal propagation, the LoRa node adjusts the polarization of its transmitted signal dynamically to ensure the polarization of the received signal is aligned with

LoRa gateway's antenna.

The design of Demeter involves three key problems.

First, due to the single RF chain resource on existing LoRa nodes, it is not trivial to design a circuit that enables flexible polarization adjustment. To address this challenge, Demeter uses an RF splitter to split the raw signal from the single RF chain into two signals. In addition, Demeter adopts a dual-polarized antenna, which takes the two signals as the inputs of its two orthogonal polarization dipole antennas to generate a linearly polarized transmission signal. The polarization of the transmission signal can be adjusted by configuring the amplitude ratio of the two signals.

Second, considering the energy constraints on COTS LoRa nodes, energy-exhausting hardware components (e.g., voltage attenuator, amplifier) should not be involved in signal amplitude control. How to control the amplitudes of the two split signals with low-power hardware components is another challenge. To address this challenge, Demeter converts an amplitude ratio to a configurable phase offset between the two signals by utilizing a low-power digital phase shifter and a passive hybrid coupler. The phase shifter adds a pre-configured offset to the phase of one signal. Moreover, Demeter uses the hybrid coupler to mix the two signals to generate two new signals with the corresponding amplitude ratio for the two orthogonal dipole antennas and further customize the hybrid coupler for the LoRa frequency band.

Third, due to the occasional change of the soil condition, the polarization of cross-soil signals is changing over time. To lower link maintenance overhead while keeping agile to the polarization change, when to trigger and how to calibrate the end-to-end polarization alignment between LoRa nodes and gateways are not trivial. To address these challenges, Demeter triggers a calibration process adaptively according to potential soil condition changes in a day. A LoRa gateway initializes a polarization calibration process and asks the targeted LoRa node to transmit multiple beacons with different polarizations. Then we develop a heuristic algorithm to search the best polarization, balancing the searching accuracy and energy overhead on LoRa nodes.

We implement a Demeter prototype with a COTS digital phase shifter [131], a COTS RF splitter [132], a COTS dual-polarized antenna [133], and a customized hybrid coupler on a COTS LoRa

node with Semtech SX1276 radio [134]. We have conducted extensive experiments in various soil environments. Our results show that compared with standard LoRa, Demeter can achieve up to 9.94 dB SNR gain outdoors, 4 \times horizontal communication distance, and at least 20 cm underground buried depth improvement.

In summary, our contributions are summarized as follows:

- To the best of our knowledge, we are the first to develop a reliable cross-soil LPWAN, which is an essential part of agricultural IoT and in-situ soil sensing for realizing precise agriculture in scale.
- We propose a low-cost and low-power polarization-programmable antenna system on COTS single RF chain LoRa nodes. Moreover, we design practical methods to monitor and calibrate polarization alignment over a cross-soil LoRa link with low maintenance costs.
- We implement the prototype of Demeter and evaluate it in real-world deployments. The results show an SNR gain of up to 9.94 dB in real environments, 4 \times communication distance, and at least 20 cm deeper deployment depth improvement on average. Moreover, Demeter decreases energy consumption per day by up to 82%.

3.2 Preliminary and Motivation

LoRa is a wireless communication technology that uses chirp spread spectrum (CSS) modulation to overcome the effects of narrowband interference and multipath fading [135]. This allows LoRa devices to communicate over long distances with low power consumption [95, 136]. Figure 3.2 illustrates a typical cross-soil LoRa communication system, where LoRa nodes are buried in underground soil and LoRa gateways are deployed in aboveground air. The sensors on the LoRa nodes generate hourly/daily sensory data that is sent to the gateways over a cross-soil link. Many factors (e.g., rainfall, snowfall, temperature, animal activity, plant growth) incur significant soil condition changes over time, leading to unstable cross-soil LoRa links and resulting in packet loss and energy waste.

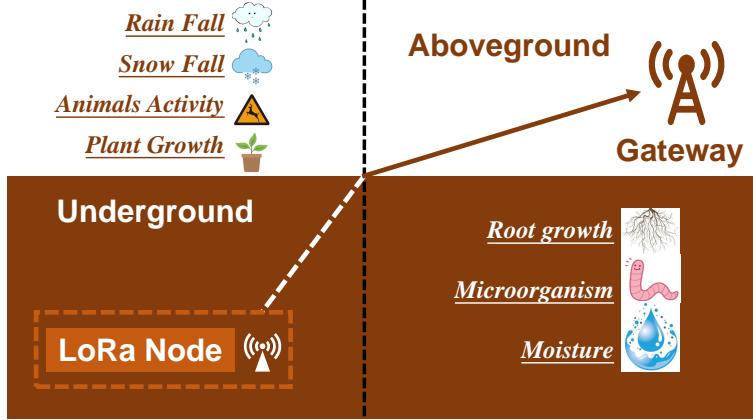


Figure 3.2 The illustration of cross-soil LoRa communication system.

3.2.1 In-field Cross-soil Link Measurement

Agriculture soils are categorized into more than ten types, as determined by the ratio of three mineral components (i.e., sand, silt, and clay) according to USDA soil texture document [137,138]. For example, clay soil consists of more than 50% clay while loam has about 40% sand, 40% silt, and 20% clay. In agriculture, loam soil holds a significant position due to its ideal balance of sand, silt, and clay particles. This balance affords loam soil excellent properties for plant growth, making it highly desirable for farming and gardening.

We conducted a 9-day in-field experiment to explore the quality of cross-soil LoRa communication in a wheat farm. The soil is Cvraab-Conover loam. Its mineral composition is about 40% sand, 40% silt, and 20% clay, respectively [137–139]. Rainfall comes on Day 2 and Day 9. A moisture sensor [140] connected to a COTS LoRa node is buried at 30cm depth underground, and we deploy a COTS LoRa gateway aboveground. The horizontal distance between the LoRa node and gateway is 10m. The LoRa node sends a soil moisture reading every 30 minutes. The gateway extracts RSSI (Received Signal Strength Indicator), SNR, and VWC (Volumetric Water Content) that measure the ratio of the volume of water to the unit volume of soil from each received packet.

Figure 3.3 shows the results. From VWC curve, we can observe two rainfalls (i.e., VWC shifting up) on Day 2 and Day 9. Correspondingly, the SNR and RSSI decrease sharply on Day 2 and fluctuate on Day 9. Moreover, we observe another RSSI and SNR drop and fluctuation on Day 6 and Day 4, respectively. Based on the historical weather record, abrupt temperature changes appear

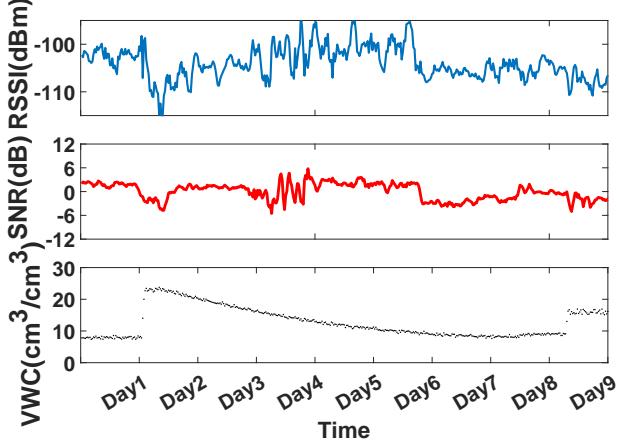


Figure 3.3 9-days performance of in-field cross-soil communication.

these days. Overall, the SNR and RSSI are extremely unstable across 9 days. The RSSI difference reaches to 21.32 dBm. The SNR ranges from 10.4 dB to -8.8 dB. The total 19.2 dB SNR variation verifies that soil condition (e.g., moisture, temperature) changes can degrade communication reliability in real-world deployments.

3.2.2 Cross-soil Signal Polarization Study

For the cross-soil LoRa link in our in-field measurement, the observed dramatic SNR variation consists of two parts. One part is from the change of signal attenuation determined by soil's dielectric properties [94, 141, 142] that keep changing along with the changes of soil condition (e.g., moisture, temperature). The other part is from signal polarization misalignment [115]. COTS LoRa nodes and gateways adopt linearly polarized antennas. The polarization of an electromagnetic wave describes the orientation of the oscillating electric field. The electromagnetic wave of LoRa signals is transverse and travels in a specific direction. A magnetic field and an electric field are perpendicular to each other and the direction of propagation. The linear polarization indicates the orientation of the electric field, which oscillates in a fixed plane determined by the magnetic and electric fields and can be purely vertical, horizontal, or any angle in between. When the polarization of the received signals is well aligned with a LoRa gateway's antenna orientation, the energy of the electric fields can be fully captured by the receiving antenna. Consequently, the SNR of the received signals can be maximized. However, in cross-soil communication, signal scattering, reflection, and refraction are inevitable during signal propagation in soils, making the polarization variation at the

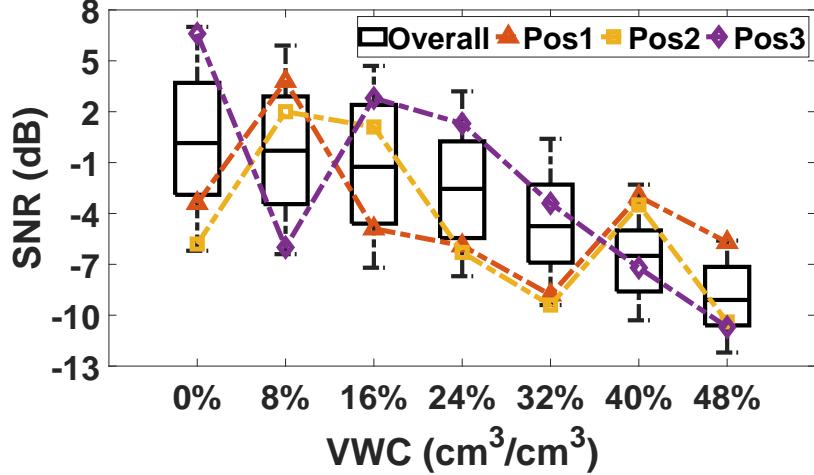


Figure 3.4 SNR variation analysis across multiple VWC settings.

receiving antenna [104, 143–146], resulting in inevitable polarization misalignment.

We have conducted controlled experiments to investigate the SNR variation incurred by polarization misalignment in cross-soil communication. In an indoor environment, we bury a COTS LoRa node in a plastic container of sandy soil (i.e., > 90% sand) and utilize a COTS LoRa gateway to receive packets. To imitate the soil moisture changes in our in-field measurement, we change VWC from 0% to 48%. When VWC is 48%, the soil tends to be water-saturated, which only happens when flooding appears in the real world. Thus, we do not set a higher VWC. To verify the influence of polarization misalignment, for each VWC, we manually vary the polarization of the received signals by uniformly rotating the antenna of the LoRa gateway to 16 different orientations in a plane perpendicular to the direction of electromagnetic wave propagation. For each antenna orientation, the LoRa node will transmit 10 packets. Then, the LoRa gateway collects the packet SNR.

In Figure 3.4, the five-value boxes show the observed SNR distribution at different VWC configurations. We can see that the SNR variation between the maximum and the minimum SNR under a VWC configuration is 13.2 dB on average. Such a huge SNR variance is purely caused by polarization misalignment. Moreover, for each VWC configuration, the maximum SNR is achieved when the polarization is well aligned. Thus, the changes in the maximum SNR reflect the changes in signal attenuation. We can see the maximum SNR monotonously decreases from 7.0 dB to -

5.6 dB, namely the signal attenuation increases 12.6 dB, when VWC increases from 0% to 48%. In our in-field measurement, the VWC is in the range of [7.8%, 23.1%]. The SNR variation caused by signal attenuation is about 5.8 dB which is $2.3 \times$ less than the 13.2 dB SNR variation brought by polarization misalignment. By adding both variations, the total SNR variation is 19.0 dB, which is relatively consistent with the observed 19.2 dB.

Furthermore, we verify that the signal polarization is indeed changed by soil condition change (i.e., VWC increase). The red, yellow, and purple dashed lines represent the SNR of three specific antenna orientations. If the cross-soil signal polarization is constant under different VWC, the trend of the three dashed lines should be similar to the monotonous decrease of the maximum SNR. However, the patterns of the three dashed lines are non-monotonic and totally different. Specifically, the standard deviations of the SNR difference between each dashed line and the maximum SNR are 4.66 dB, 4.2 dB, and 3.78 dB. The dynamic pattern and high deviation verify that the soil condition changes dramatically influence the quality of received signals.

3.2.3 Motivation

The observed reliability vulnerability motivates us to compensate for the polarization misalignment in cross-soil LoRa communication. On the other hand, a LoRa gateway may cover hundreds or thousands of LoRa nodes. Different LoRa links have diverse polarization misalignments. If we compensate for the polarization misalignment at the gateway side (e.g., rotating the gateway's antenna), the gateway has to assign different LoRa nodes' transmissions at different times. The calibration process, transmission scheduling, and network synchronization will consume considerable energy on LoRa nodes, compromising the energy benefits of the achieved reliable data transmission and incurring new scalability concerns. This motivates us to compensate for polarization misalignment *at the LoRa node side* compatible with the standard LoRaWAN media access control (MAC).

3.3 System Design

Figure 3.5 provides an overview of Demeter. Demeter consists of three parts: 1) the hardware design of a programmable antenna that allows the LoRa node to configure its signal polarization; 2) a polarization alignment calibration algorithm that generates the optimal LoRa node's setting to

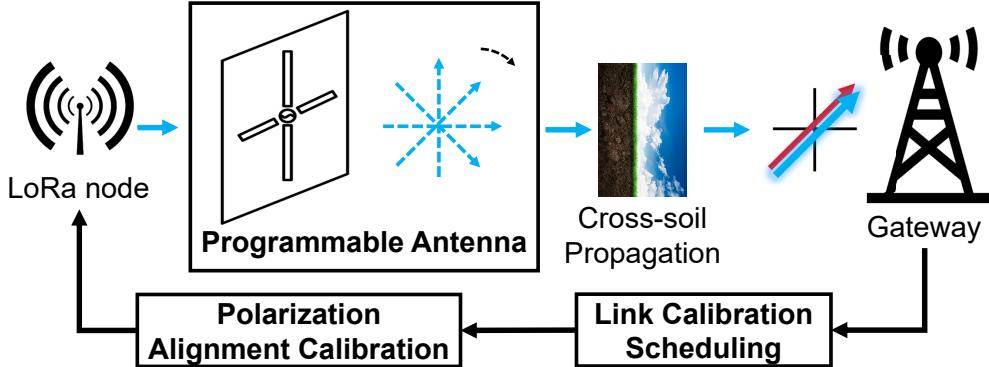


Figure 3.5 System Overview of Demeter.

achieve polarization alignment at the LoRa gateway; and 3) a link calibration scheduling method that optimizes system energy efficiency while keeping communication reliability.

3.3.1 Hardware Architecture

Figure 3.6 illustrates Demeter’s hardware architecture, which consists of four components as follows:

Passive RF Splitter: To achieve an adjustable polarization degree with only one RF chain and dual-polarized antenna, the first step is to split the single-channel signal into two channels. We use a 3 dB wide-band 2-way passive splitter to divide a signal into two identical signals, each with half the amplitude of the raw signal.

Phase Shifter: An adjustable phase shifter adds a phase offset to the input signal. Demeter adopts a programmable digital phase shifter [131] which can be embedded into COTS LoRa nodes with a 3.3 V voltage supply.

Hybrid bridge Coupler: The hybrid coupler is implemented as a four-ports (i.e., two inputs, two outputs) 3 dB 90° bridge with center frequency at the US915 band for LoRa. A hybrid coupler divides the signal of each input port into two signals with the same amplitude and 90° phase difference, then crossly combines two signals from two different input ports to generate the signals at two output ports.

Dual-polarized Antenna: A dual-polarized antenna consists of two dipole antennas with orthogonal linear polarization orientations and an overlapping geometrical center. The simple structure makes it easy to manufacture and has been widely used in wireless communication.

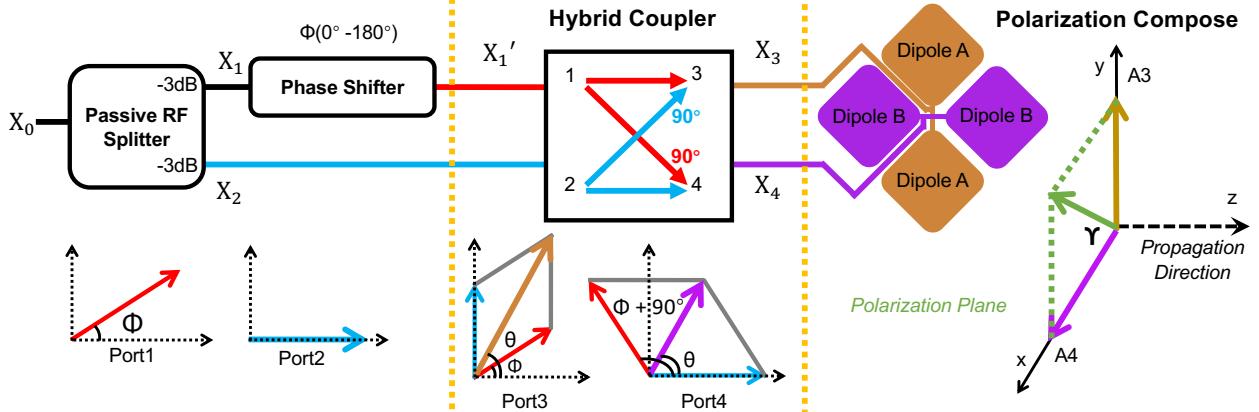


Figure 3.6 Demeter’s hardware architecture and adjustable polarization principle.

3.3.2 Adjustable Polarization Principle

As shown in Figure 3.6, the raw signal transmitted by a LoRa node is X_0 with amplitude A and phase α , which can be expressed as $X = f(A, \alpha, t)$, where t is the time. First, we use the passive RF splitter to divide the raw LoRa signal into two identical signals with half amplitude: $X_1 = f(\frac{A}{2}, \alpha, t)$ and $X_2 = f(\frac{A}{2}, \alpha, t)$. For the first way signal X_1 , the digital phase shifter is used to add a phase offset ϕ . We assume the phase shifter is ideal without extra loss, the amplitude of X_1 is still $\frac{A}{2}$. Then we have $X'_1 = f(\frac{A}{2}, \alpha + \phi, t)$. We can see that the phase difference between X'_1 and X_2 is ϕ , too.

The port1 and port2 of the hybrid coupler take X'_1 and X_2 as input, respectively. The output signals at port3 and port4 are X_3 and X_4 . To calculate the value of X_3 and X_4 , we use X_{ji} to represent the partial output signal at port i from the input signal at port j . As shown in Figure 3.6, X_3 consists of two parts: X_{32} (i.e., the blue arrow at port3) and X_{31} (i.e., the red arrow at port3). The hybrid coupler adds a 90° phase offset on X_2 to generate X_{32} (i.e., the blue crossing line). Then, X_3 can be calculated as follows:

$$\vec{X}_{32} = f\left(\frac{A}{2}, \alpha + \frac{\pi}{2}, t\right) \quad (3.1)$$

$$\vec{X}_{31} = f\left(\frac{A}{2}, \alpha + \phi, t\right) \quad (3.2)$$

$$\vec{X}_3 = \vec{X}_{32} + \vec{X}_{31} = f\left(\frac{A}{2}, \alpha + \frac{\pi}{2}, t\right) + f\left(\frac{A}{2}, \alpha + \phi, t\right) \quad (3.3)$$

Port3 vector \vec{X}_3 (i.e., the brown arrow in Figure 3.6) is vectors sum of \vec{X}_{32} and \vec{X}_{31} . $\vec{X}_{32}, \vec{X}_{31}$ compose a rhombus with same amplitude. Therefore, the phase value of \vec{X}_3 should be the mean

phase of \vec{X}_{32} and \vec{X}_{31} . Let the amplitude of \vec{X}_3 be A3. We can simplify \vec{X}_3 as:

$$\vec{X}_3 = f(A3, \alpha + \frac{\phi}{2} + \frac{\pi}{4}, t) \quad (3.4)$$

Similarly, X_4 is the combination of X_{42} (i.e., the blue arrow at port 4) and X_{41} (i.e., the red arrow at port 4). Based on X'_1 , the hybrid coupler adds a 90° phase offset to generate X_{41} (i.e., the red crossing line). We have X_4 as follows:

$$\vec{X}_{41} = f(\frac{A}{2}, \alpha + \phi + \frac{\pi}{2}, t) \quad (3.5)$$

$$\vec{X}_{42} = f(\frac{A}{2}, \alpha, t) \quad (3.6)$$

$$\vec{X}_4 = X_{41} + X_{42} = f(\frac{A}{2}, \alpha + \phi + \frac{\pi}{2}, t) + f(\frac{A}{2}, \alpha, t) \quad (3.7)$$

Let us indicate the amplitude of \vec{X}_4 as A4,

$$\vec{X}_4 = f(A4, \alpha + \frac{\phi}{2} + \frac{\pi}{4}, t) \quad (3.8)$$

We can see that X_3 and X_4 have the same phase while their amplitudes A3 and A4 are different.

We further feed X_3 and X_4 to the two feeding ports of the two dipole antennas of the dual-polarized antenna. Then the two dipole antennas produce two orthogonal electric fields. Generally, an electromagnetic plane wave that moves in the z -axis direction has two components in x -axis and y -axis [147].

$$E_x = E_{xm} \cos(wt - kz + \beta_x) \quad (3.9)$$

$$E_y = E_{ym} \cos(wt - kz + \beta_y) \quad (3.10)$$

$\vec{E} = \vec{e}_x E_x + \vec{e}_y E_y$ represents for electric field and m means magnitude/amplitude. At the two dipole antennas in the dual-polarized antenna of Demeter, we have two orthogonal electric fields in the x - y plane. Both the wave phases in x and y are equal to $\beta = \alpha + \frac{\phi}{2} + \frac{\pi}{4}$. We set the same z to 0 here.

$$||\vec{E}|| = \sqrt{E_x^2 + E_y^2} = \sqrt{E_{xm}^2 + E_{ym}^2} \cos(wt + \beta) \quad (3.11)$$

$$\gamma = \arctan(\frac{E_y}{E_x}) = \arctan(\frac{A3}{A4}) \quad (3.12)$$

According to the principle of vector composition, we can change the amplitude ratio $\frac{A_3}{A_4}$ between X_3 and X_4 by adjusting the phase offset ϕ to determine the degree of linear polarization γ (i.e., the green arrow at polarization compose). The propagation direction (i.e., black dashed arrow) of electromagnetic waves is the normal of the polarization plane. Finally, we can generate signals with different linear polarization degrees.

3.3.3 Coupler Architecture

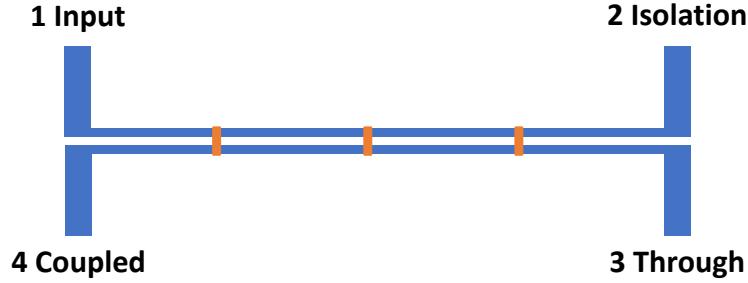
Some hardware components of our system, such as the phase shifter, the passive RF splitter, and the dual-polarized antenna are COTS devices that can be obtained easily. The hybrid coupler for a LoRa node should be low-cost with a small size without any power supply. However, the sub-1 GHz bridge coupler which meets our demand is limited on the market. Thus, we design the bridge coupler by ourselves.

The purpose of the coupler is to produce 90° phase difference between the signals of two output ports when the input signal is only from one input port and the other one is isolated. Three kinds of RF bridge couplers can achieve the target: codirectional, contradirectional, and transdirectional with different isolated port positions relative to input ports. Compared to the other two solutions, as illustrated in Figure 3.7a, a transdirectional coupler can avoid crossover and direct current bias problems because the isolated port and input port are on the same side, making the coupler smaller and scalable for IoT devices [148].

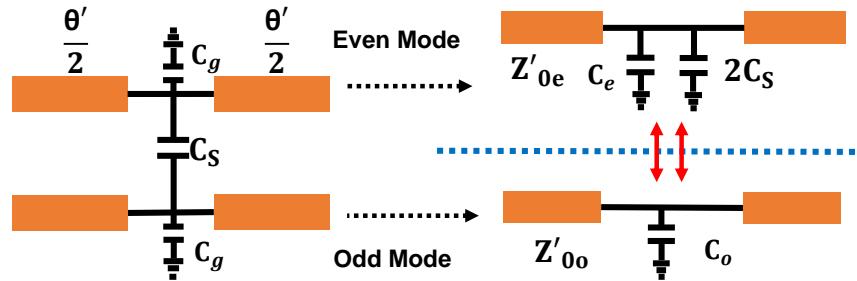
Transdirectional Coupled Line. We adopt an even-odd mode analysis [148] at four ports of the hybrid coupler in Figure 3.7. Port1 is input and port2 is isolated while port3 is the through and port4 is coupled. V_i ($i=1,2,3,4$) represents the voltage at port*i* and V_0 is the generator voltage. Let Z_{in}^e be the input impedance at port1 for the even mode and Z_{in}^o be the input impedance of the odd mode. Z_0 is the load impedance of the transmission line in this coupler.

$$Z_{in}^{(e,o)} = Z_{0(e,o)} \frac{Z_0 + jZ_{0(e,o)}\tan\theta}{Z_{0(e,o)} + jZ_0\tan\theta} \quad (3.13)$$

$$V_1^{(e,o)} = V_0 \frac{Z_{in}^{(e,o)}}{Z_{in}^{(e,o)} + Z_0} \quad (3.14)$$



(a) Hybrid coupler geometric structure



(b) Equivalent circuit for each unit cell

Figure 3.7 The illustration of the hybrid coupler design.

According to the symmetrical structure of the transdirectional coupler, the voltage at port 4 is

$$V_4 = V_4^e + V_4^o = V_1^e - V_1^o = V_0 \left[\frac{Z_{in}^e}{Z_{in}^e + Z_0} - \frac{Z_{in}^o}{Z_{in}^o + Z_0} \right] \quad (3.15)$$

Based on the definition of Z_{in}

$$Z_{in} = \frac{V_1}{I_1} = \frac{V_1^e + V_1^o}{I_1^e + I_1^o} \quad (3.16)$$

$$Z_0 = \sqrt{Z_{0e} Z_{0o}} = \sqrt{Z_{in}^e Z_{in}^o} = Z_{in} \quad (3.17)$$

Combined with Equation 3.15. N is the coupling coefficient.

$$V_4 = V_0 \frac{j(Z_{0e} - Z_{0o}) \tan \theta}{2Z_0 + j(Z_{0e} + Z_{0o}) \tan \theta} \quad (3.18)$$

$$N = \frac{Z_{0e} - Z_{0o}}{Z_{0e} + Z_{0o}} \quad (3.19)$$

$$V_4 = V_0 \frac{jN}{\sqrt{1 - N^2} \cos \theta + j \sin \theta} \quad (3.20)$$

Similarly, we calculate V_3 for port 3 as follows:

$$V_3 = V_3^e + V_3^o = V_0 \frac{\sqrt{1 - N^2}}{\sqrt{1 - N^2} \cos \theta + j \sin \theta} \quad (3.21)$$

We can notice that there is always 90° phase difference between V_3 and V_4 due to the same denominator and orthogonal numerator. The scattering parameter S can be computed with voltage ratios ($S_{ij} = \frac{V_j}{V_i}$).

$$S = \begin{bmatrix} 0 & 0 & -\sqrt{1-N^2}j & N \\ 0 & 0 & N & -\sqrt{1-N^2}j \\ -\sqrt{1-N^2}j & N & 0 & 0 \\ N & -\sqrt{1-N^2}j & 0 & 0 \end{bmatrix} \quad (3.22)$$

Then, to keep the absolute value of S_{31} , S_{32} , S_{41} , and S_{42} identical for generating the designed X_3 and X_4 signals, we set N as $\frac{\sqrt{2}}{2}$. The S parameter of our coupler becomes:

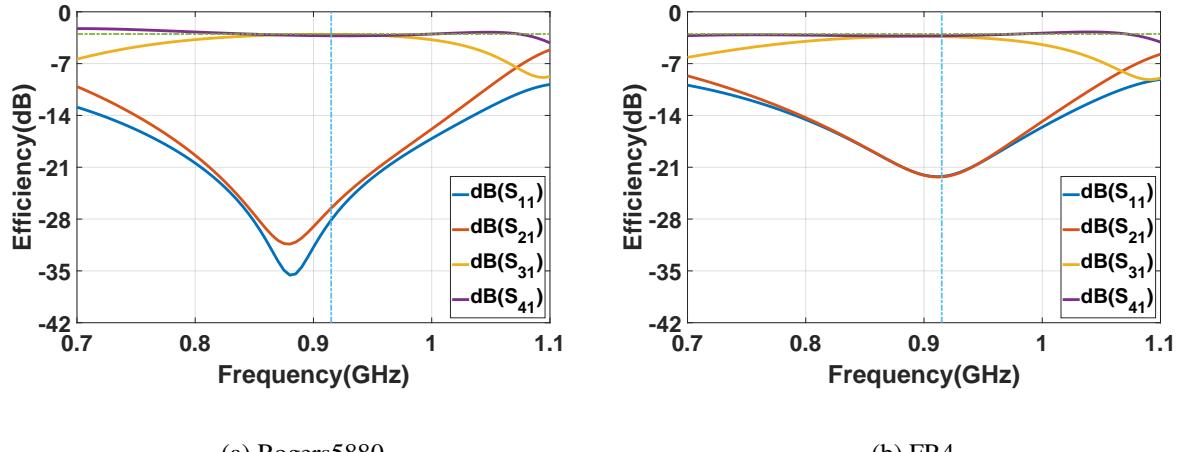
$$S = \begin{bmatrix} 0 & 0 & -j & 1 \\ 0 & 0 & 1 & -j \\ -j & 1 & 0 & 0 \\ 1 & -j & 0 & 0 \end{bmatrix} \quad (3.23)$$

Periodically Loaded Design. For a conventional coupled line, we need to satisfy Equation 3.17 under the necessary constraint of the transdirectional operation:

$$\theta_e - \theta_o = (2n + 1)\pi \quad (3.24)$$

However, it is challenging to adjust θ_e , θ_o , Z_{0e} and Z_{0o} simultaneously because the spacing between the coupled lines would impact both the impedances and phase velocities. Inspired by transdirectional coupled-line couplers working on 3.6 GHz [149], we adopt periodically loaded coupled lines to overcome this challenge.

We design multiple unit cells with capacitors between coupled lines. The unit cell, consisting of several capacitors and transmission lines, is shown in Figure 3.7a. The orange rectangles are patch capacitors. Here N is the number of unit cells. Let θ' be the electrical length of each unit cell and Z'_0 the characteristic impedance of the transmission line as the equivalent circuit that Figure 3.7b shows. C is the capacitance between the coupler transmission lines. C_g is the capacitance between the coupler and the ground. The shunt susceptances of periodic load on even and odd modes are l_e



(a) Rogers5880 (b) FR4

Figure 3.8 Coupler efficiency with different substrates.

and l_o . According to the design in [149], we have:

$$\cos \frac{\theta}{N} = \cos\theta' - \frac{b}{2} \sin\theta' \quad (3.25)$$

$$Z_0 = Z'_0 \sqrt{\frac{2\sin\theta' + b\cos\theta' \pm l}{2\sin\theta' + b\cos\theta' \mp l}} \quad (3.26)$$

$$l_{(o,e)} = \Omega C_{o,e} Z'_{0(o,e)} \quad (3.27)$$

$$C_s = \frac{C_o - C_e}{2} \quad (3.28)$$

According to Equations 3.25, 3.26, 3.27, and 3.28, the electrical length for each unit cell can be assigned with a preconfigured value. Then we can calculate the capacitance value of C_s accordingly for the LoRa frequency band.

HFSS Simulation: We use the High-Frequency Structure Simulator (HFSS) to perform electromagnetic simulation on our system. Initially, we use Roger5880 0.5mm thickness substrate with low signal energy loss to ensure high efficiency and verify the system. Figure 3.8a shows the efficiency of the coupler across the four ports in the US915 band. S_{31} and S_{41} can achieve -3.04 dB and -3.24 dB, which are consistent with half amplitude split of X_{31} and X_{41} . And the reflection loss in S_{11} and S_{21} are ultra-low as -28.13 dB and -26.52 dB, achieving high energy efficiency. However, Rogers5880 is expensive hindering us from deploying Demeter on a large scale. In addition, Rogers5880 is too soft to be distorted easily, causing unexpected circuit changes and high loss.

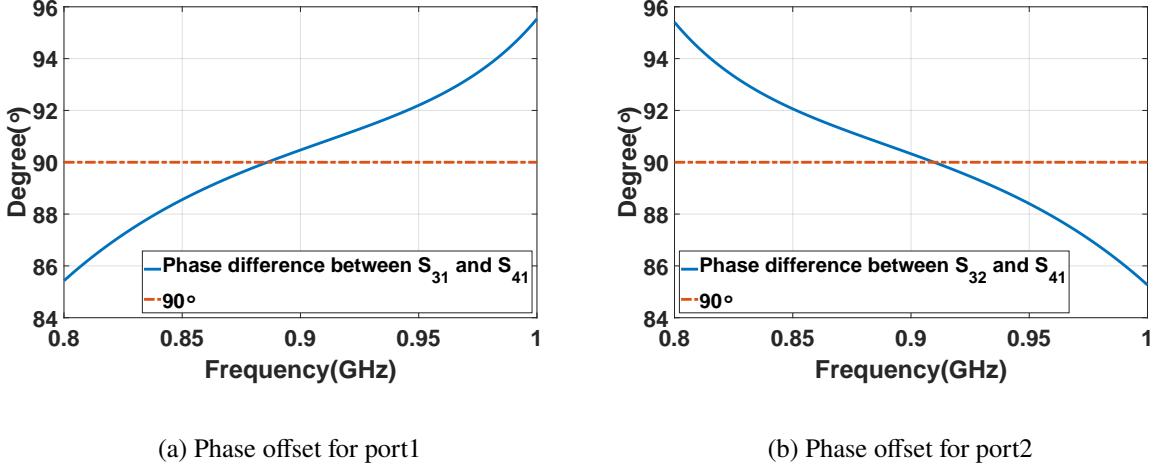


Figure 3.9 Cross phase shift in Demeter’s coupler.

To benefit IoT deployment, we redesign our coupler with substrate FR4 at 1.6mm thickness, which is one of the cheap, hard, and easily processed substrate materials. In HFSS, by adjusting the geometrical shape and parameters like electrical length, bridge width, and capacitors, we successfully achieve a 90° phase offset with low loss as shown in Figure 3.9. The phase difference between S_{32} and S_{42} range from to 89.85° to 90.22° in the US915 band and S_{32} and S_{42} range from to 90.6° to 90.92°. Figure 3.8b shows that S_{31} and S_{41} are -3.35 dB and -3.24 dB. And the reflection loss in S_{11} and S_{21} are -22.23 dB and -22.29 dB, still very low levels. The performance is comparable with Roger5880.

3.3.4 Polarization Alignment Calibration

The calibration problem is to find the optimal phase offset configuration of the phase shifter that makes the signal polarization aligned between a Demeter node and a LoRa gateway, namely the observed SNR is the highest at the LoRa gateway. Instead of traversing all possible phase offset configurations, we design a heuristic SNR peak searching algorithm to minimize the control overhead.

SNR Peak Searching Problem: The digital phase shifter can provide n different discrete phase offsets, which generate different polarization degrees of the output signal. We use the term “phase index” (i.e., from 1 to n) to represent these phase offsets in ascending order. We aim to find out the

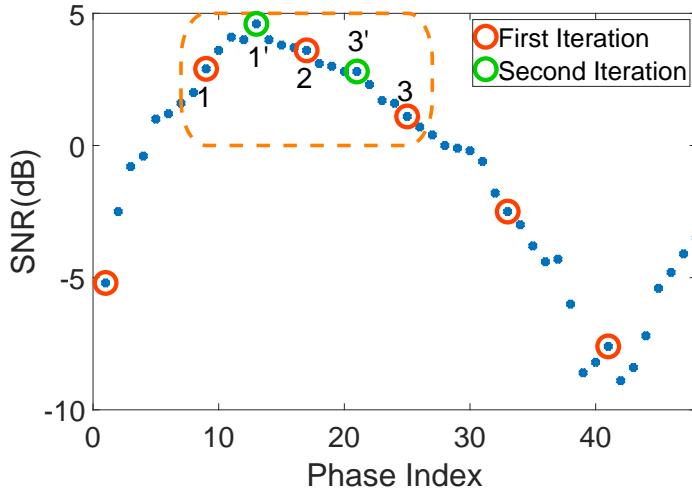


Figure 3.10 The illustration of peak search algorithm of Demeter for polarization alignment calibration.

phase index that enables polarization alignment, namely obtaining the highest SNR at the gateway side. Intuitively, a LoRa node can send n probing beacons that cover all available phase indexes to a LoRa gateway. Then, the gateway feedbacks the one with the highest SNR to the node. For example, in Demeter prototype, we have 45 phase indexes. Figure 3.10 shows the SNR changes (e.g., blue dots) for a cross-soil link deployed in the real world. We can see phase index 13 with the highest SNR 4.6 dB should be selected.

Heuristic Calibration Algorithm: Demeter transmits beacons with varied polarizations in a short duration. The cross-soil communication links are quite stable during a short period [150, 151]. Therefore, in Figure 3.10, the SNR changes comply with a sine wave approximately, where only one valley and one peak exist. By noticing this property, we can reduce the transmission number of probing beacons of current SF in a heuristic way. The algorithm contains two parts. First, we use six uniformly distributed phase indexes to determine the optimal phase index's range. Then, we use binary search to find the optimal phase index in this range iteratively. For example, Figure 3.10 shows that the LoRa node first transmits six probing beacons using the phase indexes marked by the red circles. The gateway feedbacks that the optimal phase index is in the range of either [1,2] or [2,3]. Then, the node sends another two probing beacons with phase indexes 1' (i.e., the middle of [1,2]) and 3' (i.e., the middle of [2,3]). The gateway feedbacks that the new search ranges [1',2]

and [2, 3']. The node and gateway repeat this process to find the optimal phase index. In practice, we control the number of iterations to balance the polarization accuracy and the energy cost at the node side.

3.3.5 Link Calibration Scheduling

In our empirical study in Section 3.2.1, we observe that the SNR of a cross-soil LoRa link is influenced by the changes in soil conditions (e.g., moisture, temperature) significantly. When soil conditions change, the SNR is also degrading or fluctuating. With this observation, we need to trigger link calibration more frequently if soil conditions change fast. In Demeter, LoRa nodes and gateways periodically trigger the polarization alignment calibration Θ times in a day. If the soil condition (e.g., rainfall, snowfall, fertilization, irrigation) changes frequently, we set a large Θ . Otherwise, we keep Θ small to reduce the energy costs. Therefore, based on the weather report, agriculture activity plan, and soil sensory data, we empirically configure Θ every day.

3.4 Implementation

We implement a prototype with three COTS hardware modules and a customized hybrid coupler operating on the US915 band. The COTS hardware modules include a 3 dB XRDS-RF 698-2700MHz passive RF splitter [132], an ANT627-NF-PANEL-MIMO-OD [133] dual-polarized antenna, and a digital phase shifter PE44820 [131] with 45 available phase offset indexes. Considering the loss of the phase shifter, we apply another identical one with no phase offset to the other signal, balancing the strength of the two split signals.

We implement the hybrid coupler on two kinds of substrate: FR4 with 1.6 mm thickness and Rogers5880 with 0.5 mm thickness. The 0402 patch capacitors with values of 4.3pF and 3.3pF are used in Demeter on the Rogers5880 and FR4 coupler, respectively. As shown in Figure 3.11b, the length and width are 63 mm and 25 mm, which is a relatively compact size for the IoT device working in sub-1G frequency. We use a vector network analyzer to measure the S parameters of our hybrid couplers. The phase difference between S_{31} and S_{41} is 89.6° and the phase difference between S_{32} and S_{42} is 90.7° . The signal loss of Demeter is 1.5 dB. The extra transmission power consumption of Demeter is $600 \mu\text{W}$, which is neglectable compared with COTS LoRa nodes' tens

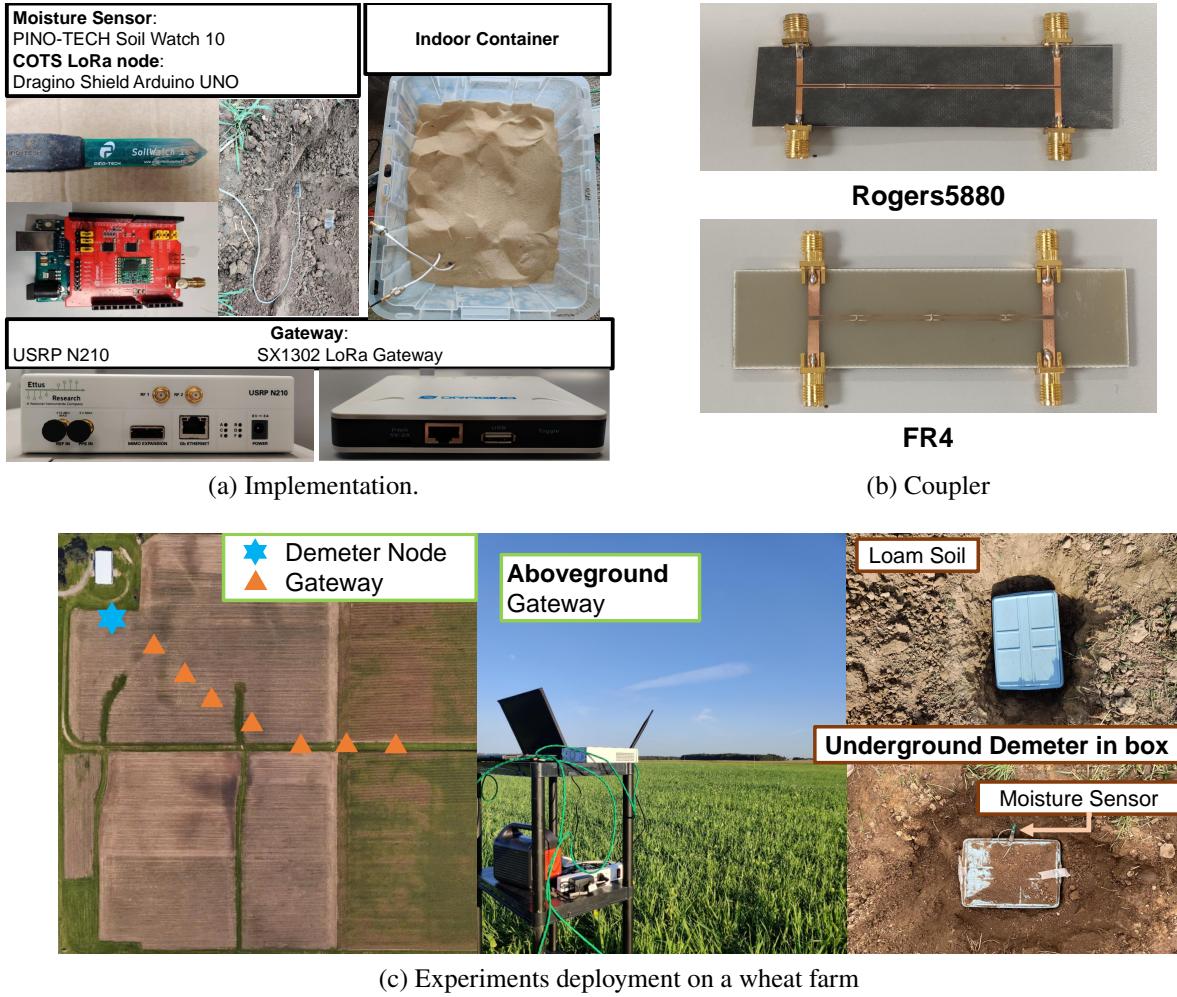


Figure 3.11 The implementation of Demeter prototype and experiment environments. The box containing a battery, a LoRa node, and our prototype of Demeter RF components is buried in loam soil at the top right corner of subfigure (c). Moisture experiments with an additional moisture sensor are shown in the bottom right corner of (c).

of mW transmission power. Moreover, since we use a wide-band dual-polarized antenna and phase shifter, the total cost of the Demeter prototype is 150 US dollars. We can lower the cost to about 20 US dollars by integrating all these modules into a unified circuit with a US915 band RF-front. The cost can be further lowered with volume production.

As Figure 3.11a shown, we implement a Demeter node by connecting Demeter to a COTS LoRa node with Semtech SX1276 radio chip [134]. For LoRa gateways, we use UHD and GNUradio to control a USRP N210 [152] with an SBX 400-4400 RF daughterboard [153]. We also use a COTS LoRa gateway with Semtech SX1302 radio chip [112] and SoilWatch 10 moisture sensor [140]. We

implement the polarization alignment calibration algorithm using the standard LoRaWAN Class A MAC, which is the default and most energy-efficient one. We run a heuristic algorithm script at the gateway to evaluate the SNR from the LoRa node in soils. We set RECEIVE_DELAY as 10 s to give enough response time for the gateway to generate feedback. We adopt 2 iterations and send 7 probing beacons in total in each calibration process. At the receiving window, the LoRa node turns off the phase shift function but keeps receiving signals through the phase shifter, which does not influence the signal reception.

3.5 Evaluation

We evaluate the communication reliability of Demeter in both indoor and in-field environments. In addition, we demonstrate the energy efficiency of Demeter regarding different calibration trigger frequencies.

Performance Metrics: We use two metrics to evaluate the performance of Demeter. The first is **SNR** of the received packets, which is a widely used metric in wireless communication to indicate communication reliability. The higher the observed SNR is, the more reliable the communication is [154]. All the SNR gain values are net gain. The second is **Energy Consumption per Day**. We use Joule per day (J/Day) to present energy consumption under various settings.

Baseline Method: We compare Demeter with the default linear polarized antenna without polarization alignment. In our implementation, we only use one dipole antenna of the dual-polarized antenna to generate the baseline signal with the same power as Demeter.

3.5.1 Indoor Experiment

Experiment Settings: We conduct indoor experiments to verify the SNR gain of Demeter in polarization alignment for cross-soil communication. We bury a Demeter node in a plastic container of sandy soil and utilize the COTS LoRa gateway to receive packets as Figure 3.11a shown. The distance between the antenna and the gateway is 20 m and the in-soil distance between the antenna and the edge of the container is 0.6 m. Then we gradually add water and stir the soil evenly to change VWC from 0% to 48%. For Demeter, we calibrate the polarization, and the antenna orientation is randomly placed, then we calculate the average SNR of 10 packets. For the baseline method, we

Table 3.2 Comparison between COTS LoRa devices and Demeter performance in sandy soil with different moisture levels. The metric is SNR with unit dB.

VWC (cm^3/cm^3)	0%	8%	16%	24%	32%	40%	48%
Max SNR	5.9	5.2	5.7	3.2	0.2	-2.6	-6.1
Min SNR	-5.9	-6.5	-7.2	-7.7	-9.6	-10.6	-12.7
Demeter SNR	6.1	5.1	6.0	3.3	0.5	-2.1	-5.9

collect and estimate the SNR of 10 packets from the LoRa node for each of the 16 different orientations of the manually rotated antenna of the LoRa gateway. For each VWC level, we keep the same wireless link between the two methods by switching the transmitter connected to the antenna ports without changing soil conditions. The spreading factor (SF) of LoRa transmission is 10.

Results: Table 3.2 shows the maximum (i.e., Max SNR) and minimum (i.e., Min SNR) SNR values of the baseline method at different antenna orientations and the average SNR value of Demeter at different VWC levels. We can see that the Demeter’s SNR is 0.21 dB higher than Max SNR on average. Moreover, Demeter SNR can achieve 11.6 dB SNR improvement compared to Min SNR. The results indicate that Demeter can achieve good polarization alignment to maintain high communication quality compared to manual polarization alignment adjustment.

3.5.2 In-field Experiment

We evaluate the performance of Demeter under five different configurations in terms of environment and deployment outdoors. We conduct experiments on a wheat farm as shown in Figure 3.11c. In the experiments, we dig a hole and bury a battery, a LoRa node, and our prototype of Demeter RF components (as mentioned in Section 3.3) in a plastic box to protect devices. Considering the potential ultra-low SNR outdoors, we utilize the USRP as the gateway to collect raw signals and compute average SNR among multiple symbols. After the polarization alignment calibration, we collect 12 packets to measure the average SNR value for Demeter. For the baseline, we also collect 12 packets to measure the average SNR. We set a 45° inclined antenna (Figure 3.14) of the gateway towards to underground node in baseline and Demeter. The default moisture is 11%, and the default soil type is loam soil.

1) Horizontal Distance Experiment Settings: As shown in Figure 3.11c, we conduct experiments

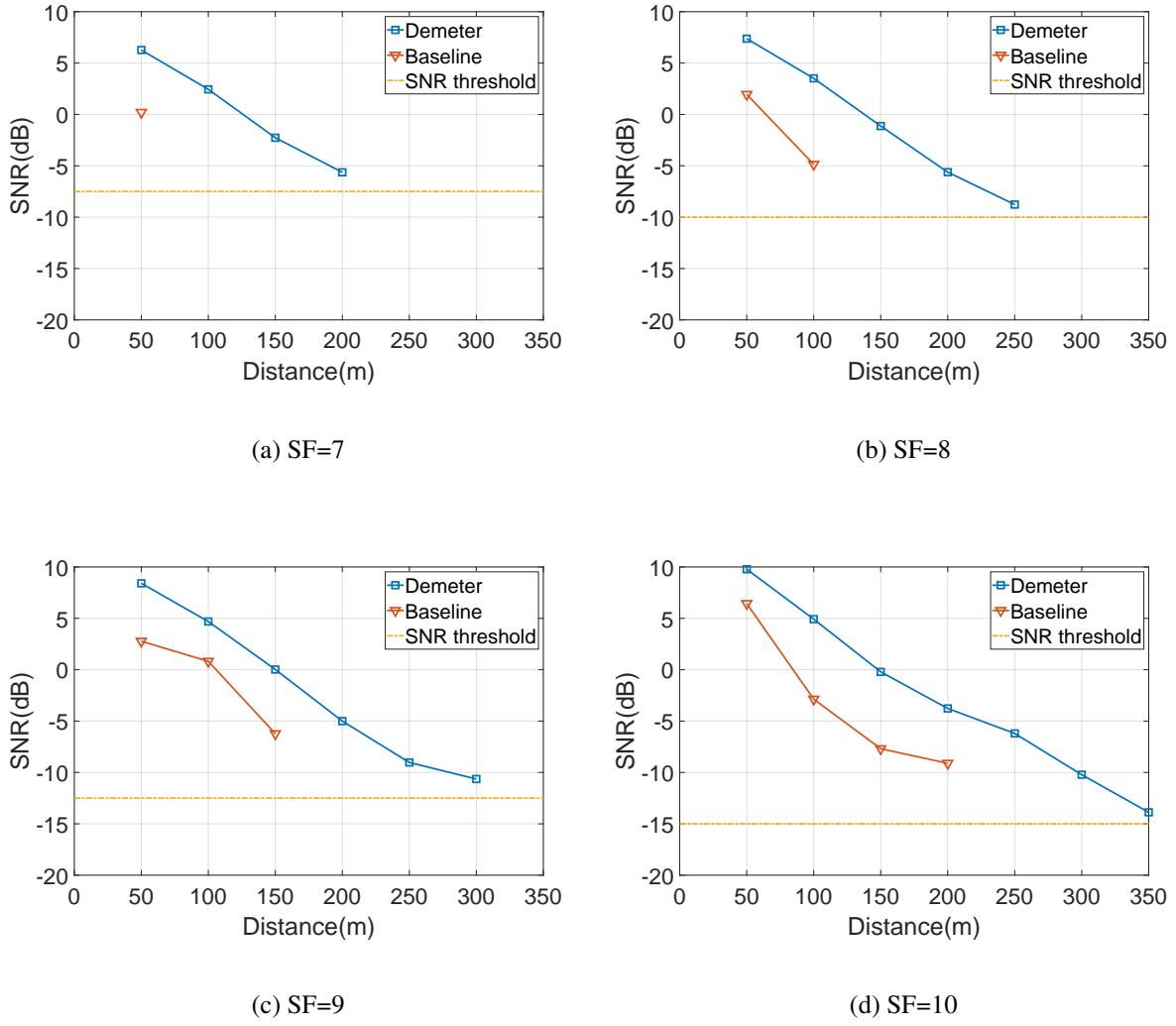
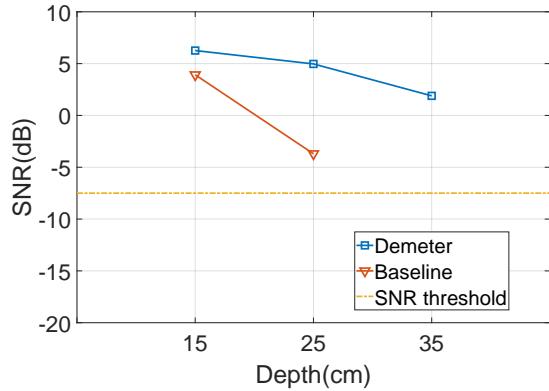


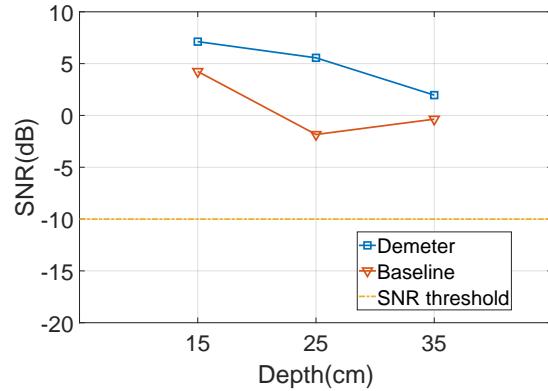
Figure 3.12 Impact of horizontal distance to Demeter.

in multiple horizontal distances from 50 m to 350 m. Orange triangles represent the locations of the gateway at various distances. The blue star is the location of the buried node. The buried depth is 35 cm. We set different SFs from 7 to 10 in the experiments.

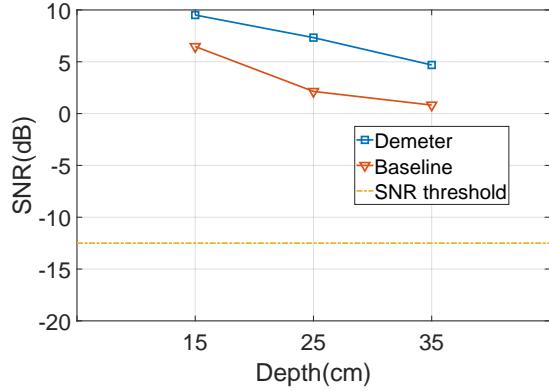
Results: As Figure 3.12b and Figure 3.12a shown, compared to the baseline, the SNR gain of Demeter is at least 8.87 dB and 9.94 dB with SF-8 at 150 m and SF-7 at 100 m respectively. The average SNR gain of SF9 and SF10 are 6.16dB and 6.55dB, respectively. The average SNR gain of Demeter achieves 6.86 dB across different configurations. We can observe that the SNR difference between Demeter and the baseline varies because the polarization of the baseline is random when



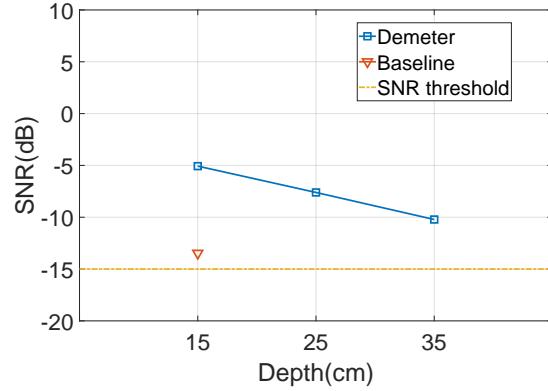
(a) SF=7, Distance=100m



(b) SF=8, Distance=100m



(c) SF=9, Distance=100m



(d) SF=10, Distance=300m

Figure 3.13 Impact of depth to Demeter.

Demeter achieves polarization alignment. Due to terrible polarization, the baseline signals may drop lower than the required SNR for successful decoding. Standard LoRa may increase SF to adapt. Instead, Demeter can recover wireless communication without varying SF. In Figure 3.12, if the distance is beyond 250 m, the baseline suffers from misalignment polarization, and Demeter trigger polarization alignment calibration to improve the signal SNR from under threshold to above threshold. Every time SF is added by one, the distance of Demeter can be enlarged by 50 m. For SF7, SF8 and SF9, distance promotion is 4×, 2.5×, 2× and 1.75× respectively in the Figure 3.12a, Figure 3.12b , Figure 3.12c and Figure 3.12d. Based on the results, Demeter can achieve 2.5×



Figure 3.14 The three different antenna orientations.

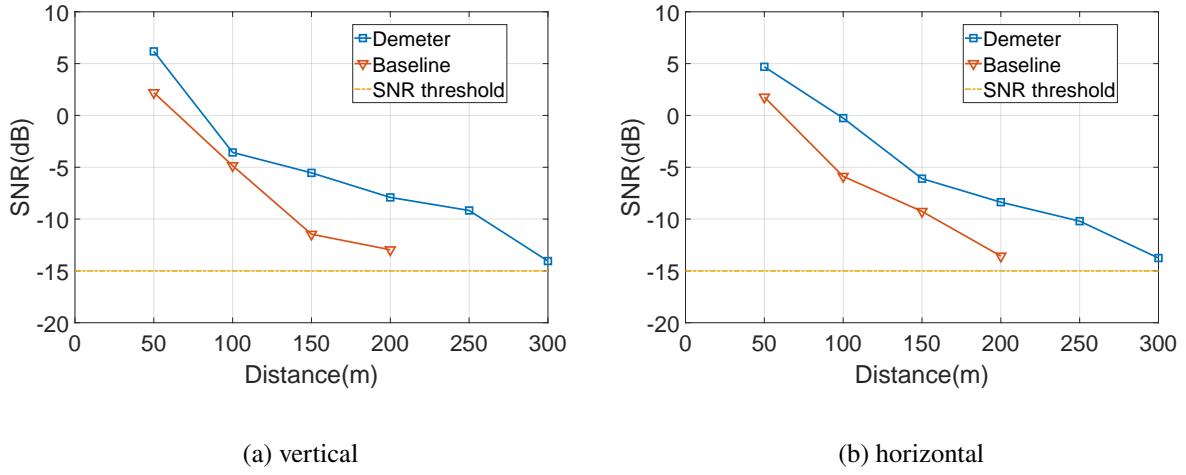
communication range on average compared with COTS LoRa. It is equivalent to a $6.56\times$ coverage area gain.

2) Depth Experiment Settings: To verify the performance of Demeter in different underground depths, we placed the Demeter node into three depth levels underground, 15 cm, 25 cm, and 35 cm, as SF is set from 7 to 10. We evaluate Demeter under the horizontal distance as 100 m and 300 m.

Results: Figure 3.13a, Figure 3.13b, and Figure 3.13c show the results. In all settings, we can see the SNR gain of Demeter compared to the baseline. Under 300 m horizontal distance and SF-10, the baseline signal can be detected only when the depth is 15 cm. Demeter make the LoRa signal appear again when depth is 25 cm or 35 cm as shown in Figure 3.13d. The depth of Demeter can be extended to $2.33\times$ compared with the baseline. In Figure 3.13b, the SNR gain is only 2.2 dB at 25 cm depth while the SNR gain is 8.67 dB with 35 cm depth. This indicates the wireless link is unstable when depth changes. The average SNR gain of Demeter achieves 5.30 dB across different configurations. Besides horizontal distance, Demeter can also improve maximal soil depth by polarization alignment calibration for cross-soil wireless communication.

3) Antenna Orientations Experiment Settings: The different antenna orientations of the gateway can influence the polarization alignment directly. In this experiment, the gateway antennas with vertical and horizontal orientations are selected to explore the impact of antenna orientation for Demeter. SF is set as 10, and the depth is 35 cm. Figure 3.14 shows the three kinds of antenna orientation.

Results: The measurement results are illustrated in Figure 3.15. We can observe that compared to the baseline, the average SNR gain of Demeter in the vertical antenna (Figure 3.15a) is 6.41 dB,



(a) vertical

(b) horizontal

Figure 3.15 Impact of antenna orientations to Demeter.

which is close to the 6.34 dB SNR gain of the horizontal antenna (Figure 3.15b), and 6.55 dB of 45° inclined antenna in Figure 3.12d. This verifies that even if the gateway antenna is put in different orientations suffering from beam mismatching and polarization plane changes, Demeter can work perfectly to provide reliable communication.

4) Moisture Experiment Settings: We explore the impact of different moisture levels (11%, 16%, 25%, 36%, 47%). We use a moisture sensor to measure the VWC before we bury Demeter node underground. The measurement depth of VWC sensor and Demeter node is 35 cm. SF is set as 10.

Results: As illustrated in Figure 3.16, the average SNR gains with 16% VWC and 25% VWC are 6.41 dB and 6.24 dB respectively compared to the baseline. Both values are close to 6.55 dB gain in Figure 3.12d with VWC 11%. The SNR gains decrease slightly due to VWC growing up as the attenuation also increases. The SNR gains in Figure 3.16b range from 3.54 dB to 8.64 dB while SNR gains range from 4.25 dB to 8.49 dB in Figure 3.16a. With higher moisture levels in Figure 3.16c and Figure 3.16d, the overall SNR decreases because of the moisture-caused attenuation. But Demeter can still compensate 5.29 dB SNR on average when VWC=36% and enlarge 4× horizontal distance in flooding status with VWC=47%. This indicates Demeter can accommodate different soil environments with various moisture conditions to achieve polarization alignment.

5) Soil Type Experiment Settings: To explore the influence of soil type, we evaluate our system

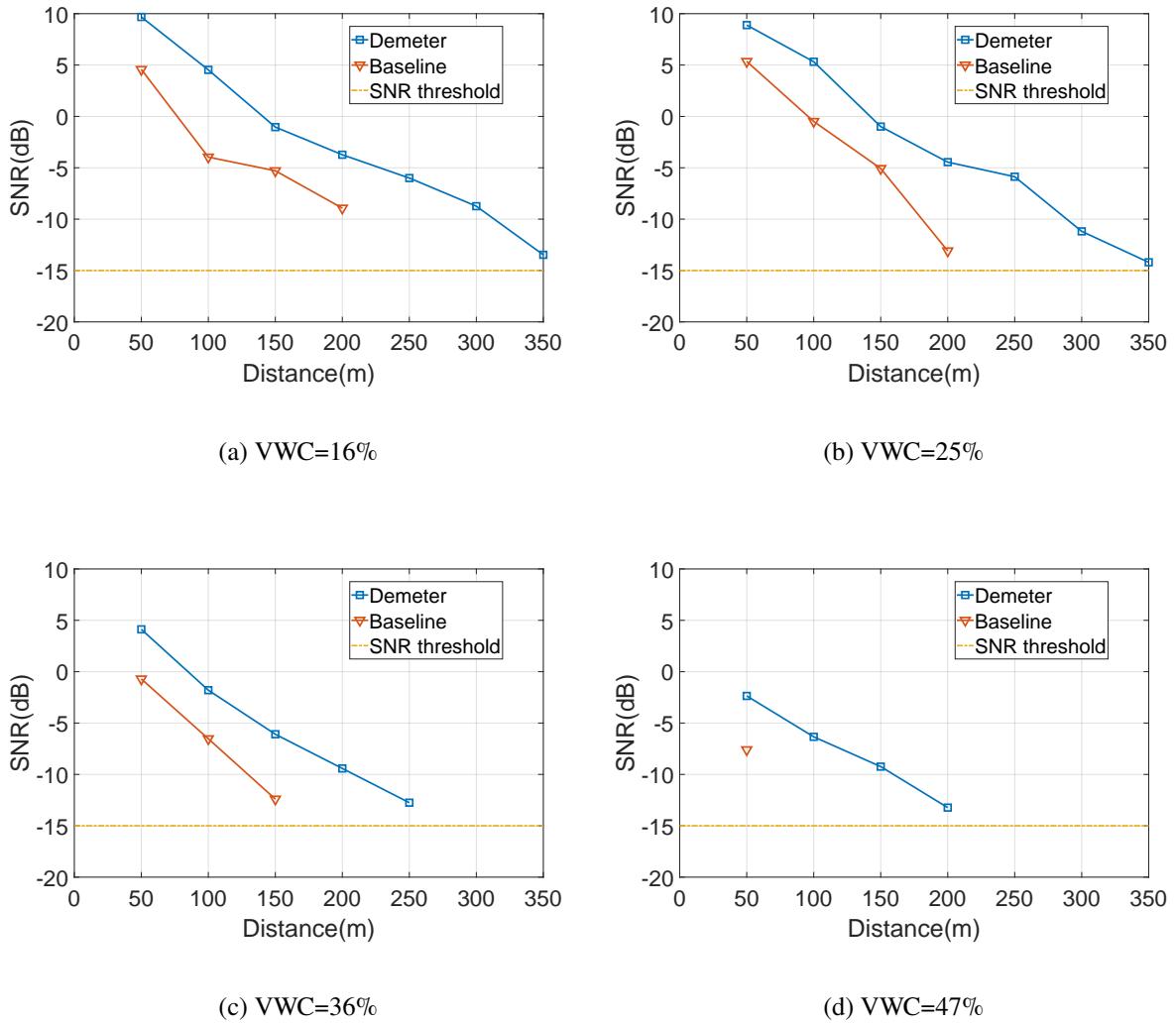


Figure 3.16 Impact of moisture to Demeter.

under three agricultural soil types: loam soil, clay soil, and sandy loam soil. The percentages of silt, sand, and clay can categorize soil. Loam has balanced components and good fertility. It holds moisture and nutrients well, drains excess water easily, and allows air to circulate around plant roots, which is ideal for gardening and agriculture. Clay takes the most percentage of clay soil, which has smaller and fewer pore spaces than loam soil, which means that water drains slowly and can cause waterlogging and compaction. Sandy loam soil has more sand compared to loam. It has a coarse texture and a loose structure that allows water and air to flow through easily. In the experiments, the depth is 35 cm and SF is set as 8.

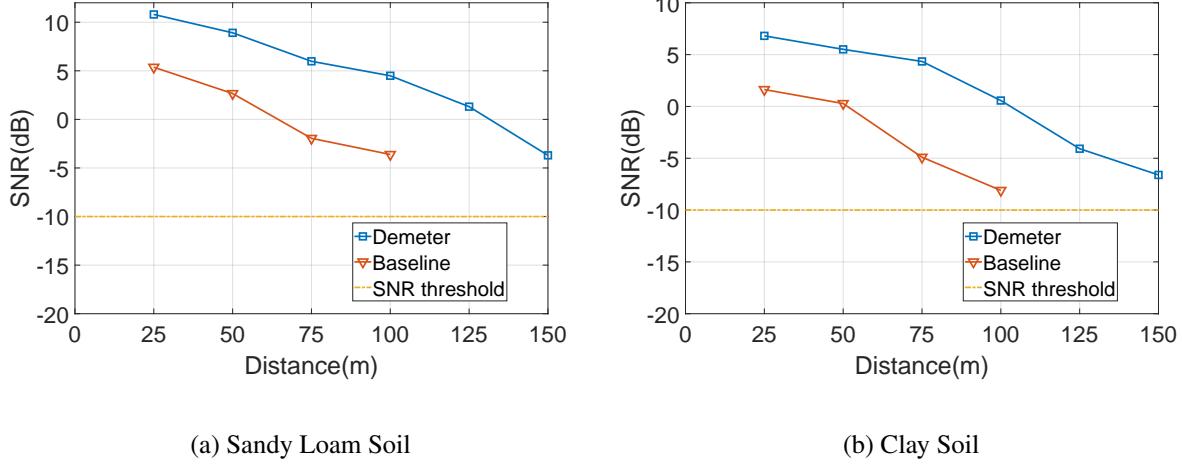


Figure 3.17 Impact of soil type to Demeter.

Results: The results for loam soil are shown in Figure 3.12b. Figure 3.17 shows the results for sandy loam and clay soils. Demeter achieves higher SNR in sandy loam of Figure 3.17a than loam with the same horizontal distance while Demeter achieves lower in clay soil of Figure 3.17b than in loam. Baseline signals in sandy loam yield the best performance, while clay soil ones are the worst. The SNR difference between these two types of soil is 3.39 dB. The SNR gain is 6.9 dB, 6.93 dB, and 7.08 dB in loam, sandy loam, and clay soils, respectively. This indicates Demeter can work in multiple types of soils and achieve good communication quality and similar SNR gains.

3.5.3 Energy Consumption Analysis

Experiment Settings: By leveraging the SNR gain of Demeter, we can increase the data rate with a lower SF to save energy compared to the baseline under the same link budget. Previous experiment results indicate Demeter can achieve at least 5 dB SNR gain, which means we can reduce SF by 2 grades with a lower SNR threshold compared to the standard LoRa [40, 100, 134, 155]. For example, if the standard LoRa has to use SF12, we can use SF10 for packet transmission. Given different link budgets and SF settings, the energy saving for each packet ranges from 0.0569 J to 0.5786 J with 3.3 V voltage.

First, we make a LoRa node sending six packets per hour to evaluate the energy efficiency of Demeter compared to the standard LoRa under different polarization calibration frequencies.

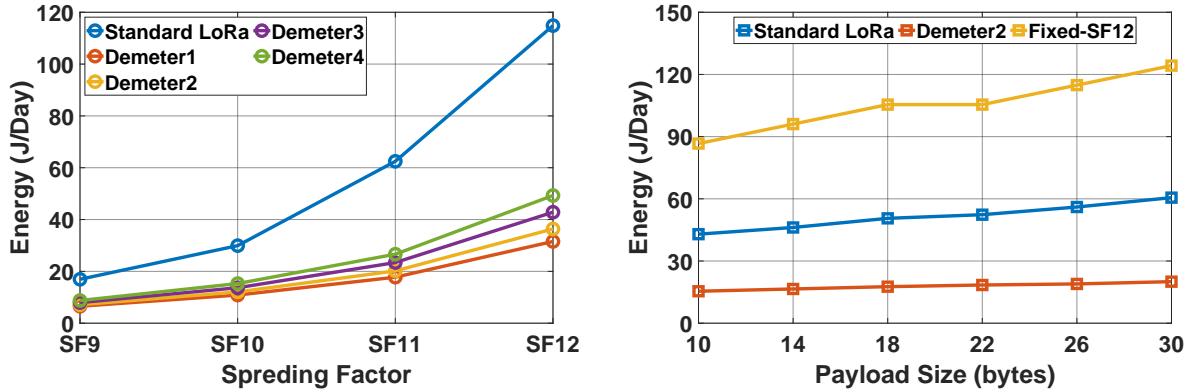


Figure 3.18 The analysis of energy consumption per day under various settings.

Based on the fact that significant fluctuations in SNR occur only 5 times over a 9-day period in Figure 3.3, we set Θ from 0.5 to 6. Specifically, Demeter 1 triggers the calibration once every two days. Demeter 2, 3, and 4 trigger the calibration 2, 4, and 6 times per day. By default, the payload size is set to 26 bytes. We calculate the energy consumption per day under four different link budgets from SF9 to SF12.

To further verify the performance of various payload lengths, we adopt six payload sizes from 10 bytes to 30 bytes. For each payload size, we calculate the average energy consumption per day of 144 packet transmissions under link budgets from SF9 to SF12 (i.e., 36 packets for each link budget). We compare Demeter 2 with the standard LoRa and Fixed-SF12, which always applies SF12 to transmit packets no matter the current link budget.

Results: In Figure 3.18a, Demeter 1–4 outperforms the standard LoRa under different link budgets. For example, compared to the standard LoRa, Demeter 2 reduces energy consumption per day by 68.3% and 57.9% for SF12 and SF9, respectively. Moreover, we can see that energy consumption per day increases when calibration trigger frequencies increase. Demeter 1’s average energy consumption per day under the four SF settings is 34.1% of the standard LoRa while Demeter 4’s increases to 48.5%. Figure 3.18b illustrates the energy consumption under different payload sizes. Demeter 2 reduces energy consumption per day by 82.0% compared to Fixed-SF12 and 63.4% compared to the standard LoRa. For a LoRa node with a 4,400 mAh energy store (i.e., two mid-grade Alkaline AA batteries), this would extend the system lifetime roughly 5.5 \times . The results indicate the

energy benefits of polarization alignment are far more than the energy cost incurred by alignment calibration in Demeter.

3.6 Related Work

Cross-soil IoT Systems: Ad-hoc Wireless Underground Sensor Networks (WUSN) [85, 91, 93, 156, 157] is well studied for cross-soil IoT. For example, Wang et al. [156] propose a soil measurement system based on WUSN. Dong et al. [158] and Silva et al. [85] present autonomous precision irrigation systems with WUSN. Compared to ad-hoc WUSN, we focus on cross-soil LPWAN, a parallel IoT paradigm. Moreover, some works [88, 89, 159] utilize COTS LoRa and NB-IoT to demonstrate the feasibility of cross-soil LPWAN. In comparison, Demeter further enhances the reliability, communication range and depth, and energy efficiency of LoRa cross-soil communication.

In-soil Communication Method: Some works [90, 91, 160, 161] focus on extending the depth a sensor can be buried at with electromagnetic waves [91, 161], magnetic coils [90], and acoustic signals [160]. For example, Sun et al. [90] adopt magnetic induction to reduce signal path loss in soils. The communication system [160] can achieve up to 50 m buried depth with a low-cost acoustic transceiver. Although magnetic and acoustic signals are able to tolerate the signal loss brought by soils, they cannot enable long-range communication over the air, which hampers the farm-scale agricultural data collection needs. In contrast, Demeter is able to achieve much longer cross-soil communication distance for general-purpose agricultural IoT systems.

Polarization-aware Communication for IoT: LLAMA [115] implemented a reconfigurable meta-surface to rotate the polarization of signal between transmitter and receiver. Polartracker [110] and Polarscheduler [111] propose a polarization-aware link model to communicate in polarization-aligned periods. In comparison, Demeter enables adaptive polarization alignment by a customized RF front-end for cross-soil LPWAN with low protocol overhead.

3.7 Discussion and Future Work

VWC range in our field study. Our in-field experiments were conducted just after a wheat harvest. Therefore, no irrigation was applied, leading to lower VWC values than those reported in the growing season [162–165] in our field study. Although the controlled experiments (e.g.,

Table 3.2, Figure 3.16) demonstrate a consistent performance gain when the VWC increases to 48%, we plan to conduct a large-scale deployment in the next growing season to evaluate Demeter performance in a more realistic VWC.

Hobbyist soil moisture sensor. The soil moisture sensor [140] we used to measure the groundtruth VWC is a hobbyist-level sensor. Although it can correctly report the trend of soil moisture changes, the accuracy of soil moisture measurement is not guaranteed. Since the specific characteristics of the soil greatly influence the accuracy of soil moisture sensors, a more accurate sensor with a soil-specific calibration would be used. In our future work, we will employ a calibratable and accurate professional-level soil moisture sensor for exploring soil-specific polarization

Adaptive polarization calibration for large-scale deployment One of our future works is to go beyond the physical layer design in this paper to build a reliable and energy-efficient network stack for achieving large-scale deployment. Considering environment and soil diversity in large-scale deployment, different LoRa nodes may need to be calibrated with different schedules. Instead of our periodical polarization calibration, on-demand polarization calibration will play a significant role in large-scale deployment as it allows a LoRa node to calibrate its polarization adaptively. A machine learning based method may be feasible to determine the trigger for a polarization calibration by taking multi-dimensional environmental and soil property changes as inputs. However, it is challenging to balance inference accuracy and computation cost on energy-limited LoRa nodes. Moreover, we will integrate RF front-end circuit design of Demeter to facilitate large-scale deployment and quantify the performance and cost benefits compared to other cross-soil communication solutions.

3.8 Conclusion

In this work, we present Demeter, a novel system that enables reliable cross-soil LPWAN communication. The main contribution of Demeter is a low-cost antenna system that aligns the polarization between LoRa node and gateway antennas and works with COST single RF-chain LPWAN radios. We also develop a low-cost hybrid coupler to encode and change polarization degrees with phase information. Furthermore, we propose a heuristic algorithm to calibrate the polarization

alignment automatically. We implement Demeter with customized PCB circuits and COTS devices to evaluate its performance in various soil conditions and scenarios. Demeter can work with various soil types and different environmental conditions. The results show that Demeter can achieve up to 9.94 dB SNR gain outdoors and 11.6 dB indoors, 4 \times horizontal communication distance, at least 20cm underground depth improvement, and up to 82% energy reduction compared with COTS LoRa node.

CHAPTER 4

PRISM: HIGH-THROUGHPUT LORA BACKSCATTER WITH NON-LINEAR CHIRPS

4.1 Introduction

LoRa [1, 7] is known for its low-power and long-range communication enabling low-power wide-area network (LPWAN) for massive IoT, which has been seen as an important part of the next-generation networks [8]. LoRa uses Chirp Spread Spectrum (CSS) modulation, in which a LoRa symbol is represented by a linear chirp. During the demodulation (called “*dechirp*”), the energy of the whole chirp will be accumulated at a signal frequency in the spectrum to enable reliable decoding even when the single strength is below the noise floor. By leveraging the advantage of LoRa’s noise tolerance, some works [166–169] show the designs which adopt LoRa’s chirps in backscatter systems to achieve long communication distance (e.g., 1.1 km to 2.8 km) benefiting the deployment of the backscatter systems in practice [170]. On the other hand, backscatter tags usually transmit data without a time backoff. Concurrent LoRa backscatter transmissions in a large-scale system increase the collision probability. As a result, the system reliability might be significantly degraded [169]. How to resolve the signal collision is a critical issue in applying LoRa backscatter in general.

Existing works [168, 169] addressed this problem by creating more orthogonal logical channels in the frequency domain, allowing multiple backscatter tags to transmit concurrently by selecting different logical channels. For example, NetScatter [169] can decode parallel packets with distributed chirp spread spectrum coding and a specially designed excitation signal. P²LoRa [168] achieves parallel decoding by shifting the ambient LoRa chirps with different frequencies. The evaluation results show that 256 and 101 concurrent transmissions can be realized by NetScatter and P²LoRa, respectively. However, these methods cannot step further to support over 500 or more concurrent transmissions efficiently due to the LoRa’s narrow-band nature.

Recently, CurvingLoRa [171] has shown the possibility of exploiting non-linear chirps to enable high-throughput LoRa concurrent transmission. First, different types of non-linear chirps can be used to transmit data concurrently [172]. Second, the transmissions with the same type of non-

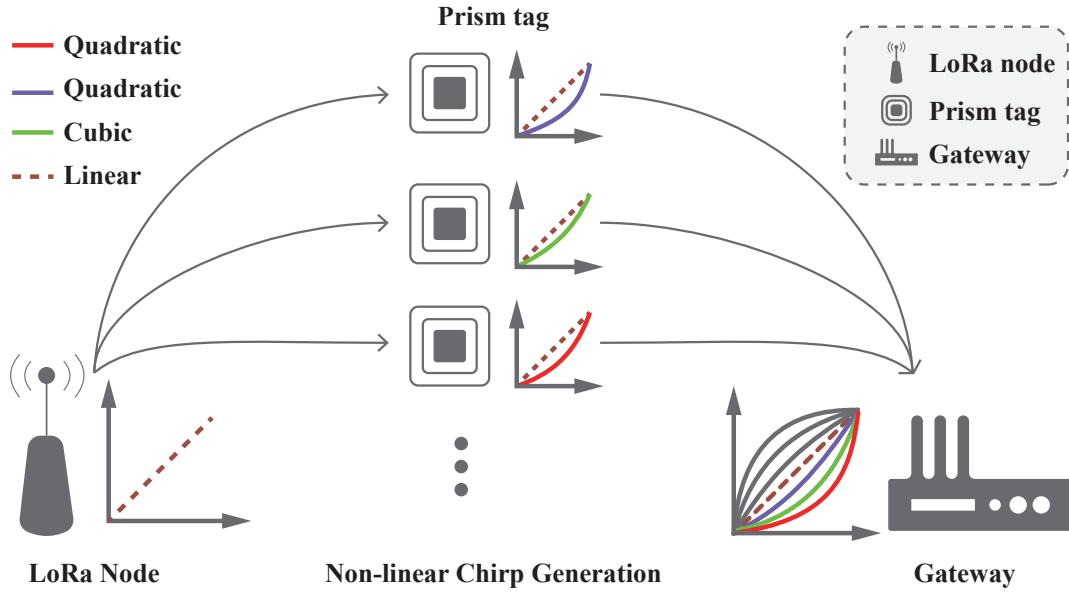


Figure 4.1 The illustration of Prism. The tags take linear chirps as excitation sources and backscatter different types of non-linear chirps for concurrent transmission.

linear chirp can be concurrently decoded when the timing of these transmissions is not exactly aligned [171]. This motivates us to rethink the design of the LoRa backscatter system with non-linear chirps.

In this paper, we propose Prism, a high-throughput LoRa backscatter method by embracing the non-linear chirps to enable concurrent transmissions. As shown in Figure 4.1, a commercial-off-the-shelf (COTS) LoRa node generates linear chirps (e.g., the dashed brown chirp) as excitation signals. When multiple (e.g., three) Prism tags detect the excitation signals, they transform the linear chirps to different non-linear chirps (e.g., the solid blue, green, and red chirps) to modulate their information. When a gateway receives these signals with the different chirp types, we can easily distinguish the excitation and backscatter chirps to decode the information of different tags [171].

However, converting a linear chirp to its non-linear counterpart with limited resources on a backscatter tag is not trivial. To enable an efficient chirp-type transform, we must address the following challenges:

First, frequency shifts between an excitation linear chirp and a backscatter non-linear chirp change over time. It is challenging to generate the time-variant frequency shifts to transform the

chirp types in a lightweight manner. In Prism, we use a timer to trigger different frequency shifts at different times. When the timer interrupt is triggered, a Micro Controller Unit (MCU) generates an integer input indicating the targeted frequency shift at the current time to a Digital Analog Converter (DAC). Then the DAC outputs a specific voltage, which is the input of a Voltage Controlled Oscillator (VCO). At last, the VCO outputs a square wave signal with the targeted frequency, which controls a radio frequency switch to change the antenna impedance/density to add the specific frequency shift. In this way, we keep the energy consumption of the chirp-type transform low.

Second, it is not trivial to calculate the frequency shifts accurately. In Prism, we use a frequency-shift function to calculate the frequency shifts over time. Specifically, all chirps have their math functions, which can be linear, quadratic, cubic, or higher-order, to shape different chirp types. We can accurately calculate the frequency shifts between two different types of chirps by using their math function in the time-frequency domain. Hence, we use the same way to generate the frequency-shift function between the linear function and the corresponding non-linear function.

Third, due to various noises, how to reliably demodulate backscatter non-linear chirp symbols is the last challenge. In Prism, we observe that for a backscatter non-linear chirp symbol, the whole chirp energy will be accumulated at two peaks on the spectrum after a non-linear dechirp process. We develop a phase search method to combine the two peaks coherently to enhance the received signal strength. Moreover, a misaligned non-linear chirp suffers from severe frequency leakage. We utilize pilot linear chirps in an excitation LoRa packet to generate pilot non-linear chirps to calibrate such frequency offsets. In these ways, we can demodulate backscatter chirp symbols reliably.

We implement Prism with a COTS LoRa device, a customized PCB circuit, and a USRP. We evaluate its performance in outdoor and indoor scenarios. The results show that Prism achieves the highest throughput of 68.36 Kbps when 700 tags concurrently transmit, indicating a 6 \times higher transmission concurrency than state-of-the-art. In addition, in the same channel, the bit error rate of the seven concurrent transmissions is less than 1% by using seven different types of non-linear chirps. The contributions of this paper are summarized as follows:

- We first introduce non-linear chirps to LoRa backscatter systems. The system scalability is

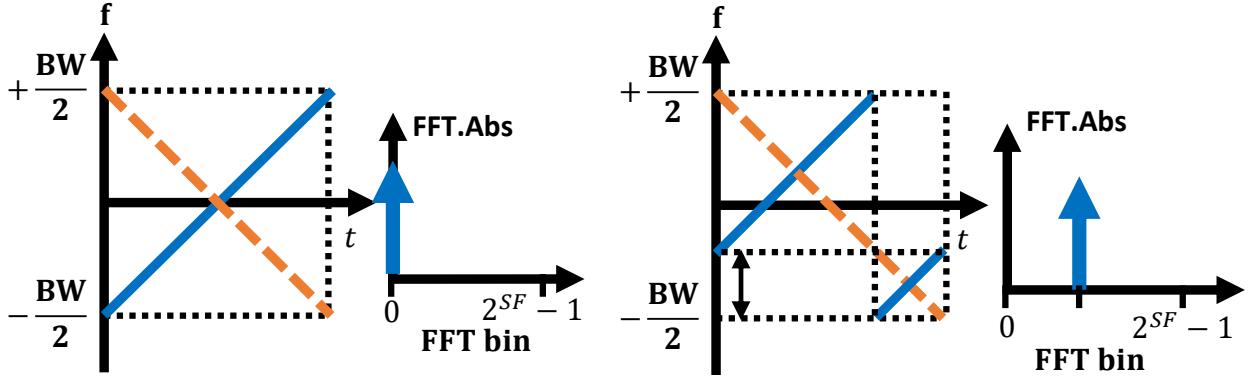


Figure 4.2 The illustration of LoRa CSS modulation and demodulation with the initial frequency shifted up-chirp and the base down-chirp.

significantly improved with our enhanced network throughput.

- On resource-limited backscatter tags, we propose a lightweight and low-power method to convert a linear excitation chirp to non-linear ones, which can be reliably decoded at a gateway.
- We prototype Prism and evaluate its performance in real environments. The results show the highest throughput of 68.36 Kbps with 700 tags. It is 6× higher than state-of-the-art in terms of transmission concurrency.

4.2 LoRa Preliminary

In this section, we briefly introduce LoRa CSS modulation and explain the background knowledge of the LoRa concurrent transmission enabled by non-linear chirps.

4.2.1 LoRa CSS Modulation

A kind of CSS mechanism is adopted by LoRa. Given a bandwidth BW , as shown in Figure 4.2, the basic communication unit of LoRa CSS modulation is a linear base up-chirp (e.g., the solid blue line in the left sub-figure) whose frequency increases linearly from $-\frac{BW}{2}$ to $\frac{BW}{2}$ over time, which can be denoted as $C(t) = e^{j2\pi(-\frac{BW}{2} + \frac{k}{2})t}$. The key character of the CSS modulation is that a time delay in a chirp can be transformed into a frequency shift. Therefore, encoded data bits can be modulated by the initial frequency offset or the cyclic time shift in the base up-chirp (e.g., the solid blue line in

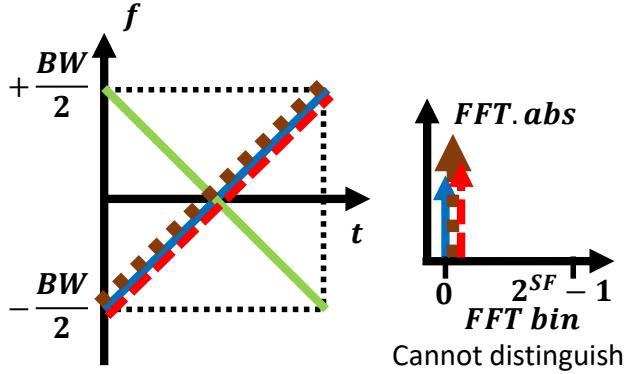


Figure 4.3 The illustration of the decoding failure due to the collision between the linear chirp symbols from different backscatter tags.

the right sub-figure). A chirp symbol can carry multiple bits of information, which is determined by the spreading factor (SF). By combining symbols with CSS modulation, a LoRa packet consists of many chirp symbols, encoding multiple pieces of information.

The process of demodulation is called “dechirp”. During the dechirp, it will multiply a received chirp symbol with a base down-chirp, which is the conjugate of the base up-chirp (e.g., the dashed orange line in Figure 4.2). After FFT (Fast Fourier Transform), a peak will appear at an FFT frequency bin, showing the initial frequency of the received chirp symbol. Therefore, if a LoRa configuration defines N different initial frequency offsets, its SF equals to $\log_2 N$.

4.2.2 Concurrent Transmission in LoRa Backscatter

Collision happens when multiple concurrent LoRa backscatter packets arrive at a gateway in a close time. The time difference comes from the different propagation delays of the multiple tags. Usually, after detecting the identical excitation LoRa signals, the arrival time difference of the concurrent backscatter chirp symbols at a gateway is less than a microsecond [168]. As shown in Figure 4.3, assume three tags use linear chirps to modulate data in the same channel. When we apply the dechirp on the overlapped linear chirp symbols, we will observe three energy peaks at close FFT bins on the spectrum. For a tag (e.g., blue linear chirp), we cannot reliably detect its energy peak if it is overwhelmed by the interference energy peaks on the spectrum. This situation will happen when the linear chirp symbol may experience a larger attenuation than the interference chirp symbols due to the near-far issue [171]. As a result, we inevitably have decoding ambiguity with two overlapped

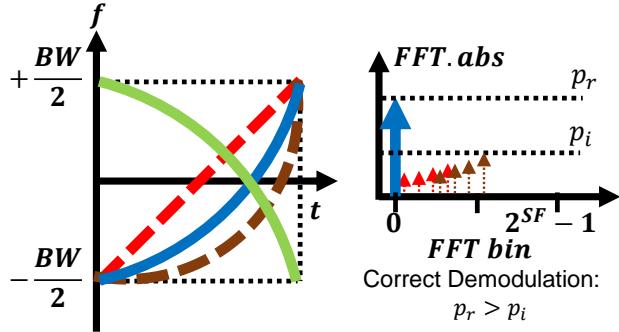


Figure 4.4 The illustration of the collision resolving between different types of linear and non-linear chirps from different backscatter tags.

packets in concurrent LoRa transmissions. Only the stronger packets are possibly decoded, but they will suffer more from the decoding ambiguity with massive concurrency.

To resolve the LoRa collision problem, CurvingLoRa [171] defines a non-linear base up-chirp whose frequency increases non-linearly from $-\frac{BW}{2}$ to $\frac{BW}{2}$ over time. There are multiple types of non-linear chirps, like quadratic, quartic, or trigonometric. As shown in Figure 4.4, we have three backscatter tags that use linear (red), quadratic (blue), and quartic (brown) chirps to modulate data, respectively. Their backscatter chirps overlap with each other. For demodulation, if the initial frequency offset of the interference chirp symbol is f_0 , applying the dechirp on the corresponding chirp symbol can be described as follows:

$$e^{j2\pi(f_0+f(t))t} \times e^{-j2\pi f(t)t} = e^{j2\pi F(t)t} \quad (4.1)$$

where $f(t)$ is a time-frequency function to depict the shape of a base up-chirp, and $-f(t)$ indicates the conjugate base down-chirp. $F(t)$ is a dechirp frequency function to depict the resulting signals. In Figure 4.4, the f_0 is 0, and the time-frequency functions of linear, quadratic, and quartic chirps can be expressed as $f_{\text{linear}}(t) = a_1 t$, $f_{\text{quadratic}}(t) = b_1 t^2$, $f_{\text{quartic}}(t) = c_1 t^4$, respectively.

Taking the demodulation of the quadratic non-linear chirps as an example. The quadratic base down-chirp is the green curve in Figure 4.4. When we apply the quadratic dechirp on the quadratic chirp, we obtain $F_{\text{quadratic}}(t)$ as follows:

$$F_{\text{quadratic}}(t) = 0 \quad (4.2)$$

which will be an energy peak at FFT bin 0, as we expected. However, when we apply the quadratic dechirp on the linear and quartic chirps, we obtain $F_{linear}(t)$ and $F_{quartic}(t)$ as follows:

$$F_{linear}(t) = a_1 t^2 - b_1 t^3 \quad (4.3)$$

$$F_{quartic}(t) = c_1 t^5 - b_1 t^3 \quad (4.4)$$

We can see that instead of the constant frequency in $F_{quadratic}(t)$, the frequencies of $F_{linear}(t)$ and $F_{quartic}(t)$ are time-variant with a non-zero t , leading to the frequency changes over time. Therefore, the energy of the linear and non-linear interference chirp symbols is no longer accumulated to high energy peaks on the spectrum. As shown in Figure 4.4, the energy of linear and quartic interference chirps (e.g., the small red and brown dashed energy noises) spreads over multiple, clustered FFT bins on the spectrum. This scattering effect makes the blue symbol can be demodulated successfully with an accumulated energy peak (e.g., the blue solid energy peak in Figure 4.4) on the spectrum. Then, we can utilize specific types of dechirp windows to demodulate LoRa backscatter packets with specific chirp types in concurrent transmissions of the multiple tags. Thus, the energy scattering effect among different types of non-linear chirps enables multiple quasi-orthogonal logical channels. The collision tolerance ability of non-linear chirps motivates us to develop a LoRa backscatter involving non-linear chirps with higher network throughput.

4.3 System Design

4.3.1 System Overview

The system overview is shown in Figure 4.5. We adopt the same tag hardware framework with P²LoRa [168]. A Prism tag includes a preamble detection circuit, MCU, DAC, and VCO. The ultimate goal of our design is to enable N resource-limited backscatter tags in the same channel to capture an excitation linear chirp and transform it into N orthogonal non-linear chirps, respectively, which can be concurrently decoded at a gateway. To achieve the goal, the end-to-end system consists of three parts: backscatter modulation, non-linear signal demodulation, and non-linear chirp decoding. First, for a Prism tag, the preamble detection circuit wakes up its MCU once a linear LoRa excitation packet is detected. Then the MCU will control the DAC and VCO to shift the

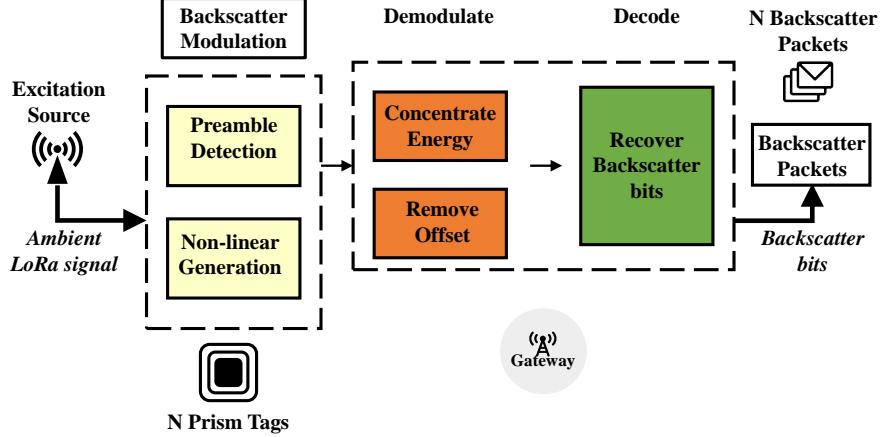


Figure 4.5 The overview of Prism system.

frequency of the excitation linear chirps to generate its non-linear chirps with a frequency-shift function. Second, at the gateway, we calibrate various frequency offsets in the received non-linear chirps and concurrently demodulate them with the dechirp by utilizing the energy scattering effect. Finally, after demodulating the chirp symbols from different tags with different types of non-linear chirps, we decode the bits embedded in these backscatter non-linear chirps from each tag.

4.3.2 Preamble Detection

Prism adopts a low-power preamble detection circuit [168, 173] to identify the arrival of linear LoRa excitation packets. The circuit includes three parts: impedance matching, envelop shaping, and preamble detection. The first two parts can capture the repetitive pattern of the continuous base up-chirps in the preamble of a LoRa packet. The last part compares the captured pattern with a template to determine whether linear LoRa excitation chirps are coming.

4.3.3 Non-linear Backscatter Chirp Generation

The first task of generating a non-linear backscatter chirp is to add frequency shifts on a linear excitation chirp. As shown in Figure 4.6, after the linear excitation chirps are detected, we use the MCU to control the output voltage of the DAC, which determines the frequency of square wave signals generated by the VCO. The square wave signals are added on linear excitation chirps to generate the non-linear backscatter chirps.

Specifically, the MCU uses 16-bit serial data to control the DAC. The upper 4 bits are control

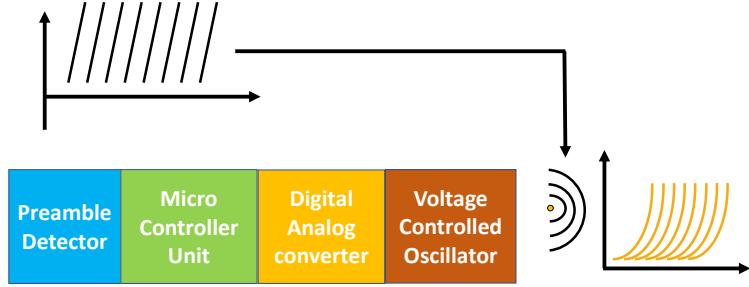


Figure 4.6 The design of a lightweight frequency shift generation on a Prism tag.

bits, and the lower 12 bits give the input data. The output voltage V_{OUT} of the DAC is given with the following equation:

$$V_{OUT} = \frac{V_{REF} \times N}{4096} \times gain \quad (4.5)$$

where N is the numeric value of the DAC's binary input code, V_{REF} is a constant value as the reference voltage, and the value of *gain* is 2, decided by the peripheral circuit. The output voltage V_{OUT} of the DAC is the input of the VCO, generating a square wave signal, whose frequency f_{OUT} can be calculated with the following equation:

$$f_{OUT} = \frac{1MHz \times 50k}{N_{DIV} \times R_{VCO}} \times \left(1 + \frac{R_{VCO}}{R_{SET}} - \frac{V_{OUT}}{V_{SET}}\right) \quad (4.6)$$

where N_{DIV} is 2, V_{SET} is the voltage on the SET pin, nominally set as 1.0V, and the R_{VCO} and R_{SET} are two 100k resistors. With Equation 4.5 and 4.6, we can set a 16-bit integer value ranging from 0 – 65535 to N to generate a square wave signal with the specific frequency f_{OUT} . The square wave signal further controls a radio frequency switch to add f_{OUT} frequency shift on the linear excitation chirps.

When a tag can shift the frequency over linear excitation chirps, the next step is to determine the frequency shift needed at a specific time. In Figure 4.7, the left sub-figure illustrates the frequency difference between a red linear base up-chirp and an orange non-linear base up-chirp (e.g., a quadratic shape), which are time aligned. We subtract the frequency of the linear base up-chirp from that of the non-linear base up-chirp, defining the phase shifts needed over time, as shown in the right sub-figure. For different linear excitation chirps, we use the same sequence of frequency shifts to generate their non-linear counterparts for simplicity. In addition, the MCU uses a timer to trigger different frequency shifts over time.

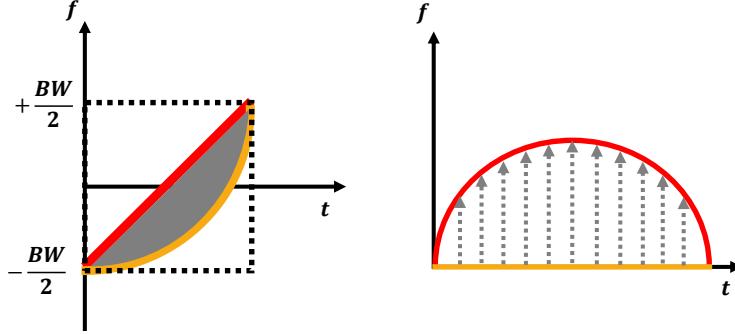


Figure 4.7 The time-frequency transform from a linear excitation to a non-linear chirp.

Intuitively, we can use a look-up table to store the frequency shifts at different time points. However, in order to achieve an accurate non-linear chirp generation, we must store fine-grained frequency shifts, which are hard to be supported with the limited storage space on the low-power MCU. Instead of the look-up table, we compute the frequency shifts with an accurate time-frequency function. We adopt the way proposed by CurvingLoRa [171] to define a non-linear base up-chirp as a monotonic curve growing from $(0, -\frac{BW}{2})$ to $(\frac{2^{SF}}{BW}, -\frac{BW}{2})$. The coordinates x and y indicate time and frequency, respectively. The monotonic curve function $f_c(t)$ of a linear or non-linear chirp can be represented by the sum of a series of polynomial functions in the time-frequency domain as follows:

$$f_c(t) = \sum_{i=0}^n k_i t^i, t \in [0, \frac{2^{SF}}{BW}], f_c(t) \in [-\frac{BW}{2}, \frac{BW}{2}] \quad (4.7)$$

Hence, for a linear base up-chirp, $k_0 = -\frac{BW}{2}$ and $k_1 = \frac{BW^2}{2^{SF}}$. Its curve function is the following:

$$f_{linear}(t) = -\frac{BW}{2} + \frac{BW^2}{2^{SF}} t \quad (4.8)$$

In addition, to depict a quadratic non-linear base up-chirp, $k_0 = -\frac{BW}{2}$ and $k_2 = \frac{BW^3}{2^{2SF}}$. Its curve function is the following:

$$f_{quadratic}(t) = -\frac{BW}{2} + \frac{BW^3}{2^{2SF}} t^2 \quad (4.9)$$

By subtracting Equation 4.8 from Equation 4.9, we can obtain a frequency-shift function $f_\Delta(t)$ as follows:

$$f_\Delta(t) = \frac{BW^2}{2^{SF}} t - \frac{BW^3}{2^{2SF}} t^2 \quad (4.10)$$

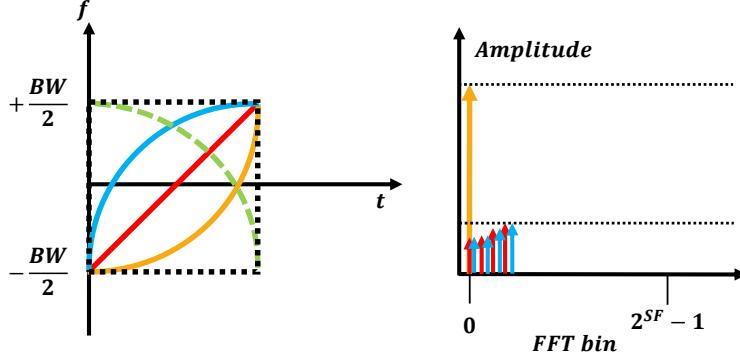


Figure 4.8 The illustration of the backscatter non-linear chirp demodulation with a linear base up-chirp as excitation.

Given the values of SF and BW, the MCU can accurately compute the frequency shift at any time. For different non-linear types at different tags, we change the coefficients k_i in Equation 4.7 to calculate the corresponding frequency-shift function $f_\Delta(t)$.

4.3.4 Non-linear Chirp Energy Concentration

As we use square wave signals to add a frequency shift on a linear excitation chirp, the backscatter signals will contain two non-linear chirps that are symmetric about the linear excitation chirp. Take a linear excitation base up-chirp as an example. As shown in Figure 4.8, a gateway will receive three signals, including a red linear excitation chirp, an orange targetted non-linear chirp and a blue mirror non-linear chirp. In addition, a green chirp is the base non-linear down-chirp used for demodulating the targeted non-linear chirp. With a non-linear dechirp, as shown in the right sub-figure, the energy of the mirror non-linear and linear excitation chirps will be spread over multiple FFT bins due to the energy scattering effect of non-linear chirps. Thus, the targeted non-linear chirp can be demodulated successfully even when the interference of the linear excitation chirp is stronger.

However, when the linear excitation chirp is no longer a base up-chirp, the energy of the targeted non-linear chirp will spread to two peaks on the spectrum after the non-linear dechirp, as shown in Figure 4.9. The reason is the frequency offset between the non-linear base down-chirp and the targeted non-linear chirp, degrading the interference tolerance ability. To achieve the same interference tolerance ability by taking a linear base up-chirp as the excitation signal, we manually

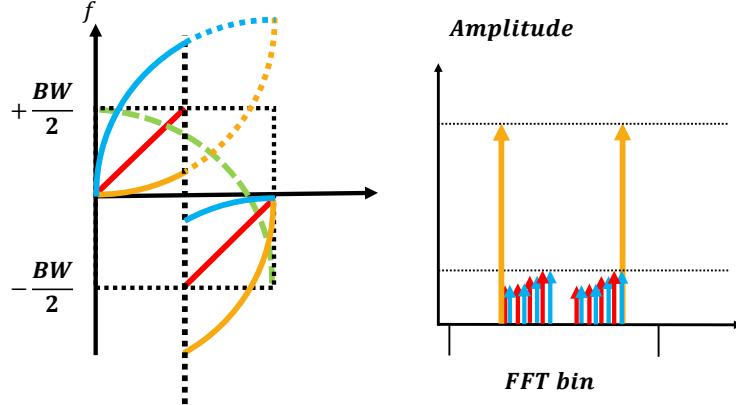


Figure 4.9 The illustration of the two energy peaks on the spectrum after a non-linear dechirp with a normal linear excitation chirp.

combine the two energy peaks on the spectrum to ensure that the energy of the whole targeted non-linear chirp is concentrated. To ensure the two peaks can be superposed coherently, we search for all possible phase differences between the two peaks and select the phase difference that can maximize the energy of the superposed peaks.

4.3.5 Chirp Offset Removal

In practice, compared to the linear excitation chirps, non-linear backscatter chirps from a Prism tag experience a propagation delay as its ToF (time of flight), leading to an unexpected time offset (TO) between the non-linear chirps and their base down-chirps during demodulation. In addition, another misalignment may also be caused by carrier frequency offset (CFO). In the non-linear dechirp, the CFO and TO will spread the spectrum power of non-linear chirps into multiple frequency bins. Thus the backscatter symbols cannot be demodulated successfully.

The Start-Frame-Delimiter (SFD) of a linear excitation LoRa packet contains two linear base down-chirps. After a Prism tag detects the preamble of the linear excitation LoRa packet, based on the linear base down-chirps in the SFD, it will generate two non-linear base down-chirps, which are to calibrate the TO and CFO. Similar to multiplying a base down-chirp to up-chirps and conducting FFT in the dechirp, we define a reverse-dechirp as multiplying a base up-chirp to down-chirps and conducting FFT. The basic observation is that when we apply the reverse-dechirp to the two non-linear base down-chirps, the CFO makes the energy peaks far away from FFT bin 0 and the TO



Figure 4.10 The implementation of an excitation LoRa node, Prism tags, and a USRP gateway.

makes the energy of the base down-chirps scatter in multiple FFT bins after the reverse-dechirp. First, we search the TO to achieve the two maximum repetitive energy peaks with the reverse-dechirp. Then, according to the FFT bins where the energy peaks appear, we calculate and calibrate the CFO. By doing so, we successfully remove the chirp offsets, including the TO and CFO, and the non-linear reverse-dechirp ensures the process is inter-tag interference resilient.

4.3.6 Backscatter Data Recovery

After the chirp offsets are calibrated, we use the non-linear dechirp to demodulate the backscatter non-linear chirps with a sliding window. A Prism tag uses On-Off-Keying (OOK) modulation to embed ‘0’ or ‘1’ into a non-linear chirp symbol. Specifically, if we detect an energy peak using the non-linear dechirp in a time window, it represents bit ‘1’. Otherwise, the bit is ‘0’. Beyond OOK, we can be easily extended to n -Frequency Shift Keying (FSK) by adding a n -level frequency shift on the backscatter non-linear chirps to represent different bits. The different frequency offsets are equivalent to different time offsets during the dechirp. Considering the energy scattering effect of non-linear chirps with different shapes or time offsets, the inter-tag interference of n -FSK is negligible.

4.4 Implementation

We implement Prism tags using the same hardware framework with P²LoRa [168]. The preamble detection module includes a three-stage voltage-doubling amplifier HSMS-285C [174] and a low-power voltage comparator LPV7215MG [175]. The backscatter module consists of an MCU

STM32L011D3P6 [176], a low-power DAC MAX5530 [177], a low-power VCO LTC6990IS6 [178], and a reflective radio frequency switch ADG902 [179]. The excitation source is a COTS LoRa node with Semtech SX1276 [134] radio chip. For a gateway, we use the UHD+GNU-radio to control a USRP N210 [152] with an SBX 400-4400 RF daughterboard [153] to record raw signals in the air, which are further processed in MATLAB. We use a 3 dBi gain omnidirectional antenna for Prism tags and the USRP. The total power consumption of a Prism tag is $695\mu W$ during working. Figure 4.10 shows the implementation of the Prism tag, the LoRa node, and the USRP.

4.5 Evaluation

In this section, we evaluate the performance of Prism to answer the following questions:

- **Q1 (§V-A):** In outdoor environments, does Prism perform as reliably as existing LoRa backscatter with linear chirps for long-distance backscatter communication?
- **Q2 (§V-B):** In indoor environments, does Prism perform as reliably as existing LoRa backscatter does with linear chirps for long excitation distance?
- **Q3 (§V-C):** In general, can Prism reliably scale to the scenario of massive concurrent transmissions?

Basic Experimental Settings: The default SF, BW, coding rate (CR), and transmission power of the excitation LoRa node are set as 10, 125 kHz, $\frac{4}{5}$, and 20 dBm. The SF setting is consistent among the excitation LoRa node, backscatter tags, and gateway. For the indoor and outdoor deployments, we define the distance between the excitation Lora node and a tag as source-to-tag distance (d_{ST}). Similarly, the distance between a tag and the gateway is defined as tag-to-receiver (d_{TR}). At each location/configuration, the excitation LoRa node transmits 300 LoRa packets, and a Prism tag modulates 28 bits of data based on an excitation LoRa packet to calculate the bit error rate.

Performance Metrics: We use two performance metrics as follows to indicate the reliability and scalability of a LoRa backscatter system.

- *Bit Error Rate (BER)* is a metric to evaluate the performance of communication reliability over a channel that has been altered due to noise, interference, distortion, or bit synchronization.



Figure 4.11 The deployment of our outdoor experiments.

tion errors. BER is widely used to demonstrate the channel noise resilience of a communication system. The lower the BER is, the higher the reliability of the system is. Similar to P²LoRa, we set BER as 10^{-4} if all data bits are successfully decoded.

- *Network Throughput* refers to the total data delivered through a network system in a unit of time. Network throughput is critical to show the scalability of a network system. The higher the throughput is, the better the scalability is. In our experiments, we use bits per second (bps) or Kbps as the unit to quantify the network throughput.

Baseline Method: We compare Prism with state-of-the-art concurrent LoRa backscatter P²LoRa [168].

4.5.1 Outdoor Experiments

Experiment Setup: Figure 4.11 shows the deployment of our outdoor experiments. We deploy a Prism tag at different locations (e.g., blue locations) on our campus. It uses a quadratic type of non-linear chirps defined as $f(t) = t^2$. The excitation LoRa node is put at the locations (e.g., green locations) close to the tag, and d_{ST} is 5 m. We put the USRP gateway (e.g., orange location) at a fixed location. Then, we move the pair of the excitation LoRa node and the tag. Thus, d_{TR} varies from 50 m to 600 m. We repeat the same experiments by setting SF as 9.

Results: As shown in Figure 4.13, we can see that the BER is almost the same between P²LoRa and Prism at all locations. If d_{TR} is not greater than 400 m, both Prism and P²LoRa can decode all data bits successfully. When d_{TR} comes to 500 m or larger, the BER gets larger. For example, the

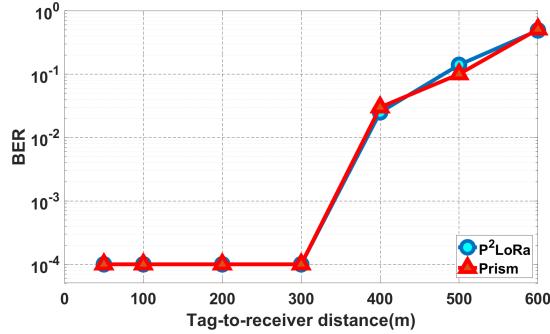


Figure 4.12 Outdoor BER comparison (SF=9).

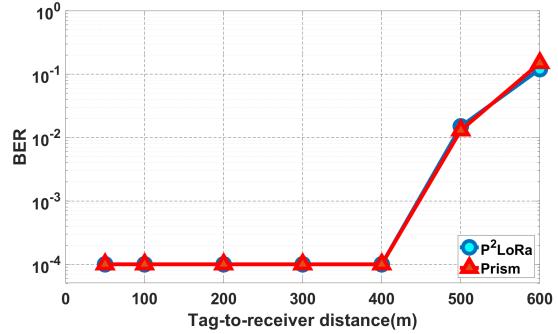


Figure 4.13 Outdoor BER comparison (SF=10).

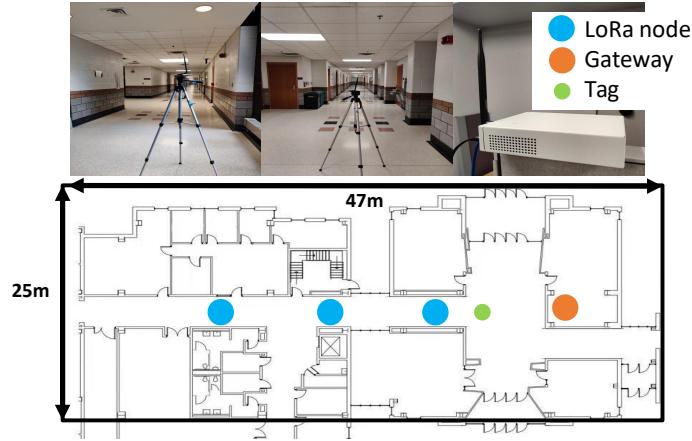


Figure 4.14 The deployment of our indoor experiments.

BER is 1.5% when d_{TR} is 500 m. When d_{TR} increases to 600 m, the BER of Prism is 14.9%, which is 1.5% higher than P²LoRa. Such a difference can be mitigated by upper-layer bit error correction methods [180].

In Figure 4.12, a similar trend can be observed when we set SF as 9. Since SF-9 chirps are less noise resilient than SF-10 chirps, when d_{TR} is equal to or less than 300 m, both systems are able to decode all the transmitted bits reliably. As the distance increases to 400m, the BER of both P²LoRa and Prism increases to 2.5%. In addition, as the distance increases to 600m, the BER level increases to 50%.

Remark: Prism realizes a similar backscatter distance to P²LoRa. Therefore, non-linear chirps can be adopted to enable long-range LoRa backscatter.

Experiment Setup: Figure 4.14 illustrates the deployment of our indoor experiments. We deploy Prism in our office building. The gateway is put in our office, and the excitation LoRa node and

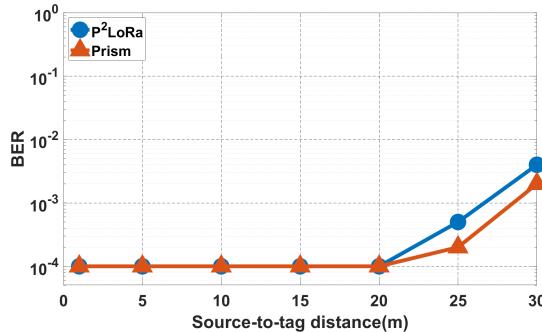


Figure 4.15 Indoor BER comparison (SF=10).

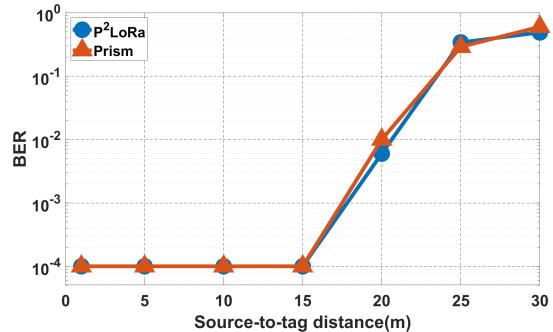


Figure 4.16 Indoor BER comparison (SF=9).

tag are deployed in a corridor. A wall separates the line of sight path between the gateway and the backscatter tag. We use the same quadratic non-linear chirp as we do in the outdoor experiments. The d_{TR} is fixed to 15 m, and the d_{ST} varies from 1 m to 30 m. We repeat the same experiments by setting SF as 9.

Results: As shown in Figure 4.16, as P²LoRa does, Prism can reliably decode the data bits when the d_{ST} is not greater than 20 m. When the d_{ST} is larger than 20 m, the BER of Prism and P²LoRa increases since the excitation linear chirps become weaker with the large d_{ST} . In addition, Prism achieves a bit lower BER than P²LoRa. However, the achieved BER is less than 1%, indicating similar communication reliability.

When the SF is set to 9, we observe a similar trend in Figure 4.15. We can see that the d_{ST} is reduced to 15 m for reliable communication since SF-9 chirps are more vulnerable to noises. The communication reliability decreases quickly as the d_{ST} is larger than 20 m. The BER of Prism can reach 60.3% when the d_{ST} is 30 m which is much higher than that using SF-10 chirps.

4.5.2 Indoor Experiments

Remark: Prism experiences a similar influence brought by enlarging the distance between the tag and the excitation LoRa node to P²LoRa. Thus, non-linear chirps are a perfect alternative for linear chirps in LoRa backscatter systems.

4.5.3 Overall Throughput under Collision

Experiment Setup: We conduct a trace-driven emulation to evaluate the network throughput of Prism in various backscatter collisions. We collect different types of backscatter non-linear and

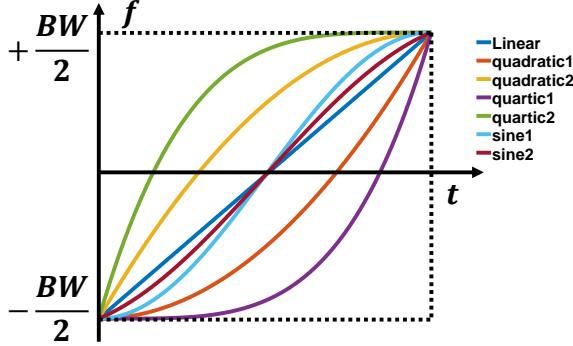


Figure 4.17 Prism uses seven types of non-linear chirps.

linear chirps in our indoor deployment. Then, we superpose these collected backscatter chirp symbols to create different backscatter collisions. A total of seven different types of linear or non-linear chirps are used in this experiment. These chirp types and the corresponding time-frequency functions are listed as follows:

- *linear* : $f(t) = t$
- *quadratic1* : $f(t) = t^2$
- *quadratic2* : $f(t) = -t^2 + 2t$
- *quartic1* : $f(t) = t^4$
- *quartic2* : $f(t) = -t^4 + 4t^3 - 6t^2 + 4t$
- *sine1* : $f(t) = \sin(t)$, $t \in [-\pi/2, \pi/2]$
- *sine2* : $f(t) = \sin(t)$, $t \in [-3\pi/8, 3\pi/8]$

The shapes of the base linear and non-linear up-chirp are shown in Figure 4.17.

First, we compare Prism with P²LoRa when the collisions happen in the same channel. Specifically, Prism uses the seven different types of chirps to modulate data bits for 7 concurrent tags, but the 7 tags only use linear chirp to modulate data in P²LoRa. We randomly select n ($n \leq 7$) tags to transmit data concurrently. Second, we emulate the collision scenarios across multiple channels. We adopt the method of P²LoRa to assign different tags to 100 different channels. The number

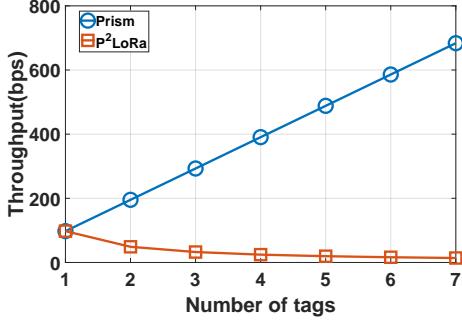


Figure 4.18 Throughput comparison between P²LoRa and Prism in the same channel.

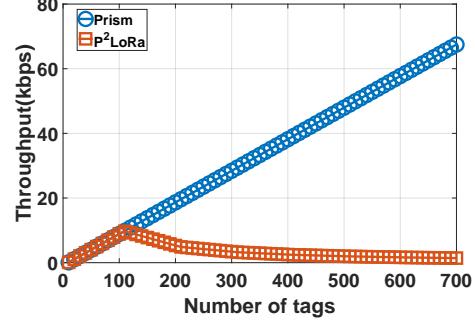


Figure 4.19 Throughput comparison between P²LoRa and Prism across multiple channels.

of concurrent tags increases from 1 to 700. When the number of concurrent tags is less than 100, we assign each tag with a different channel using linear chirps as P²LoRa does. When the number continuously increases, we uniformly assign a newly coming tag to a random channel, then select a type of non-linear chirp that is not used by other tags in the channel. In comparison, P²LoRa assigns channels uniformly and randomly to new tags which use linear chirps only. In addition, to mimic the near-far effect in the collisions, we randomly select a Signal-Interference-Ratio (SIR) between any two collided non-linear chirp symbols in the range between -10 dB and 0 dB [171, 172]. Given the number of collisions, we generate 40 collision scenarios with diverse types of non-linear chirps, channel assignments, and SIR to calculate the average throughput as our results. If the data bits of a tag are successfully decoded, the throughput of the tag equals $\frac{BW}{2^{SF}} \times CR = 97.7$ bps. Then, the throughput of a collision scenario is calculated as the number of successfully decoded tags multiplies the throughput of a single tag.

Results: With only one channel, as shown in Figure 4.18, the throughput of Prism linearly increases as the number of backscatter tags increases. However, as the number of tags increases, the throughput of P²LoRa decreases and approaches zero eventually. The results verify that the seven types of linear and non-linear chirps are orthogonal enough to enable successful concurrent transmissions among seven tags. Compared to P²LoRa, the backscatter concurrency is improved by 6×.

When we use 100 channels and 700 tags, as shown in Figure 4.19, the throughput of Prism increases linearly as the number of concurrent tags increases. This verifies that Prism successfully supports 700 tags to transmit concurrently. Prism achieves the maximum throughput 68.36 Kbps

with 700 tags. When the number of concurrent tags is not greater than 100, P²LoRa exhibits a similar throughput increase as Prism. However, as the number of tags increases beyond 100, the throughput of P²LoRa drops to zero gradually. When the number of tags reaches 700, the awful throughput of P²LoRa can be expected due to the severe throughput loss in each signal channel.

Remark: Prism utilizes the energy scattering effect of different types of non-linear chirps to increase the concurrency in each channel, thus improving the throughput significantly. In addition, Prism is parallel to P²LoRa and can be easily extended to scenarios with multiple channels. Moreover, we only demonstrate seven types of non-linear chirps. Hopefully, we can further improve the throughput of Prism by involving new types of non-linear chirps. How to achieve the maximum concurrency with non-linear chirps is still an open question.

4.6 Related Work

Generally, a backscatter system uses a radio frequency switch to either reflect or absorb excitation signals. The batteryless property makes it attractive in many IoT applications. For example, Liu et al. [181] propose ambient backscatter that uses some ubiquitous wireless signals (e.g., TV and cellular) in surrounding environments to obtain energy for backscatter communication. Zhao et al. [182] propose OFDMA-enabled Wi-Fi backscatter that enables tags to produce backscatter signals at the frequencies of orthogonal subcarriers, which are subsequently synthesized into an OFDM symbol by a receiver. Prism focuses on LoRa backscatter and achieves a scalable backscatter system with interference-tolerant concurrent transmissions enabled by different types of non-linear chirps. We summarize the related work from the following aspects.

LoRa Backscatter: Talla et al. [167] propose LoRa backscatter that uses a single-tone excitation signal to generate a LoRa packet by shifting the frequency with a radio frequency switch. With a specific excitation signal, Netscatter [169] further proposes to combine CSS and OOK modulation, assigning different cyclic shifts of a linear chirp to different concurrent tags, to support parallel decoding of concurrent LoRa packets. By taking COTS LoRa linear chirps as excitation signals, PLoRa [166] adopts an FPGA to shift LoRa linear chirps by two frequency offsets to modulate data. Moreover, P²LoRa [168] develops a modulation method to support parallel decoding with ambient

LoRa linear chirps as excitation signals. However, the scalability of P²LoRa relates to the available bandwidth resources, which are usually limited in LoRa, a narrow-band communication technology. Additionally, P²LoRa suffers from inter-tag inference, which may comprise its performance in practice. In comparison, Prism is parallel with existing methods to enhance the scalability of LoRa backscatter systems by exploring the quasi-orthogonal logical channels created by different types of non-linear chirps in the same physical channel.

Non-linear Chirps: Non-linear frequency modulation (NLFM) [183] has been widely used in radar systems, which is a cost-effective pulse compression method [184] with high resolution, improved SNR, and effective interference suppression. NLFM modulation function has an inherent spectrum weighting that results in lower sidelobe levels compared with linear frequency modulation(LFM). In LoRa, CurvingLoRa [171] adopts non-linear chirps to improve the scalability of LoRa networks by enabling a LoRa gateway to successfully decode concurrent transmissions with the same type of non-linear chirps. Moreover, CurveALOHA [172] proposes a media access control (MAC) protocol with multiple types of non-linear chirps to increase the network throughput. In contrast, Prism is a scalable LoRa backscatter system, which converts ambient linear LoRa chirps emitted from COTS LoRa radios to multiple non-linear counterparts, which can be reliably decoded at a gateway simultaneously.

LoRa Collision Resolving: LoRaWAN [1] adopts the least restrictive MAC protocol ALOHA [185], which enables nodes to transmit data upon waking up. When multiple LoRa packets from different nodes are transmitted concurrently, collisions inevitably appear. Existing works focus on extracting distinguishable features of the collided packets in the time or frequency domain. For example, Choir [186] matches data bits to a LoRa node with its frequency change due to its oscillator deficiency. FTrack [187] divides collisions by exploring the distinct tracks on the spectrum and symbol edges on the time domain. CIC [188] combines the spectrum from different parts of a chirp symbol, proposing a sub-symbol method to cancel interference signals under ultra-low SNR environments. mLoRa [189] utilizes the consecutive interference cancellation to iteratively demodulate the data bits with no interference. CoLoRa [190] utilizes the spectral peak ratio of a misaligned chirp as a

characteristic to differentiate between overlapped LoRa packets. Pyramid [191] applies a sliding demodulation window to track the changes in energy peaks to resolve collisions. NScale [192] improves noise resistance in the iterative peak recovery by varying peak scaling factors within two consecutive windows. PCube [193] classifies chirp symbols from concurrent packets by utilizing unique phases of the air channel. However, since backscatter tags usually have no back-off mechanism, the time offset among the collided chirp symbols is almost zero, which is hard to be handled by the existing methods. In addition, the existing methods suffer from the near-far problem, indicating a situation that a strong signal from a near node significantly reduces the SNR of a weak signal from a far node. Moreover, LMAC [194] and p-CARMA [195] operate at the MAC layer and try to use Channel Activity Detection (CAD) and CSMA (carrier-sense multiple access) to detect a potential collision and avoid it with a backoff. However, the CAD and CSMA are energy-heavy for a LoRa backscatter system. In contrast, Prism adopts non-linear chirps, creating new logic channels to avoid these issues for practical LoRa backscatter concurrent transmissions.

4.7 Conclusion

To conclude, we develop a high-throughput LoRa backscatter system Prism by using the different types of non-linear chirps to enable concurrent transmissions of multiple backscatter tags in the same channel. Specifically, Prism takes ambient LoRa linear chirps from COTS LoRa nodes as the excitation source. Then, a Prism tag uses a timer to trigger a low-power frequency shifting process, which transforms the linear excitation chirps to the non-linear backscatter chirps according to an accurate frequency-shift function. At the gateway side, we concentrate the energy of non-linear chirps during the dechirp and decode the data bits encoded by backscatter tags. We prototype Prism tags and gateways with customized PCB circuits and USRP, respectively. We evaluate Prism’s performance in indoor and outdoor environments. Prism achieves the highest throughput of 68.36 Kbps and supports 700 tags to transmit concurrently with a bit error rate that is less than 1%. With more non-linear chirp types, Prism is expected to perform better.

CHAPTER 5

AEROECHO: TOWARDS AGRICULTURAL LOW-POWER WIDE-AREA BACKSCATTER WITH AERIAL EXCITATION SOURCE

5.1 Introduction

Recently, the Internet of Things (IoT) has revolutionized agricultural methods, giving rise to smart agriculture [196]. This breakthrough incorporates interconnected agricultural sensors, profoundly boosting efficiency, productivity, and sustainability in managing both crops and livestock [2, 15, 18, 197–199]. Crucial to this transformation is the demand for cost-effective hardware with broad communication range capabilities, vital for spanning vast agricultural landscapes [11, 200, 201]. With the deployment of numerous sensors, low power requirements are essential, consequently reducing the frequency of battery replacements and maintenance [5, 196, 202].

Backscatter communication, a cutting-edge IoT technology, utilizes existing carrier waves to modulate information bits, eliminating the need for amplifiers and carrier modulation found in active radio systems [203]. Recent research on Long Range (LoRa) backscatter [166–169] aims to integrate LoRa communication with backscatter radios, achieving low energy consumption (e.g., ≤ 1 mW) while sustaining long communication distances between tags and receivers (e.g., 1.1km to 2.8km), potentially meeting the demands of agricultural IoT: long-distance, low-power, and cost-effectiveness [7].

However, several factors currently make long-range backscatter communication impractical, as summarized in Table 5.1. The first issue is the high deployment cost. The effective communication range between the excitation source and the backscatter tag in existing solutions is roughly 20× shorter than the distance between the tag and a LoRa gateway, spanning less than 15 meters in a single tag static backscatter system [1, 166, 167, 204]. This restriction forces the deployed position of tags to be very close to the excitation source compared to the distance to the receivers, leading to the dense deployment of excitation sources and increased costs. Additionally, the dense tag deployment causes collisions and reduces scalability [1]. Some existing works do not support multi-tag backscattering [166, 167, 204], while others require high spectrum occupation to support multi-tag

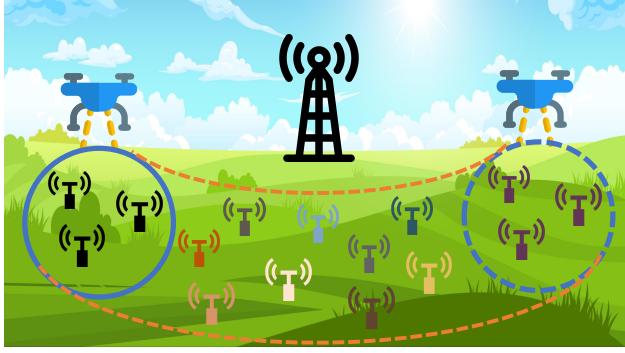


Figure 5.1 Illustration of AeroEcho.

networking [168, 169, 205]. This limitation prevents the scalability of tags in agricultural settings, restricting the potential for high throughput. Moreover, long excitation systems [206–208] focus solely on shortening the distance between the excitation source and the tag and reducing costs, but they overlook the importance of scalability.

This paper introduces AeroEcho, a novel low-power wide-area data collection system that utilizes backscatter with a high-speed aerial excitation source and fixed gateways. The concept is illustrated in Figure 5.1. Inspired by the wide range of usage of drones as infrastructure in modern smart farms [209], AeroEcho shifts the excitation source from stationary locations to unmanned aerial vehicles (UAV), reducing the cost of dense excitation source deployment for tags widely distributed in the farmland. The excitation source and tag are co-designed for multi-tag decoding in the same frequency channel to boost scalability. We deploy receivers with fixed gateways because of the complexity of full-duplex transceivers and limited computation and storage resources on the drones. However, achieving AeroEcho design entails overcoming three significant challenges:

The initial challenge is that when one excitation source signal is transmitted, all the backscatter tags within the communication range can be woken up and start backscatter modulation due to the standard synchronized symbols in the commercial-off-the-shelf (COTS) excitation device. This inevitably leads to signal collision and requires excitation signal cancellation at the receiver side. To address this challenge, we co-design our excitation source and tag with a customized packet format to avoid unnecessary synchronization and collision for backscatter signals. Furthermore, we propose asynchronous decoding methods to distinguish and decode the signals from different

Table 5.1 Existing long-range wide-area backscatter comparison. (Tag data rate is measured in SF12 and 125kHz bandwidth)

	Cost	Scalability	Max Throughput	Tag data rate
Single tag [166, 167, 204]	High	Low	13.6kbps	24.5bps
Parallel decoding [168, 169, 205]	High	Low	250kbps	30.5bps
Long excitation [206–208]	Low	Low	19.6kbps	-
AeroEcho	Low	High	1.46Mbps	366bps

tags without the complex carrier cancellation process at gateways.

Next, the random and dense distribution of tags on the land poses a challenge. More tags with asynchronous decoding can decrease communication performance due to a higher symbol error rate (SER). Achieving an optimal balance between the number of active tags and SER to maximize throughput becomes imperative. To tackle this issue, we introduce an excitation cell methodology to facilitate optimal multi-tags decoding performance. Specifically, the UAV selectively activates only those tags within a predefined circular area designated as the excitation cell. The radius of this cell is calibrated to encompass the maximum permissible distance from the source to any given tag.

Lastly, accounting for UAV range constraints, tag energy efficiency, and unpredictable tag placement, designing a flight path that ensures comprehensive coverage of multiple radio excitation cells within a specific area presents a formidable challenge. This challenge is formalized as an optimization problem, emphasizing either the energy efficiency of backscatter tags or the range efficiency of UAVs while considering constraints related to communication and coverage reliability. To tackle this issue, we have devised two distinct route planning strategies: Rectangular Displacement and Annular Trajectory, tailored to optimize coverage and energy efficiency.

We implement AeroEcho utilizing customized PCB circuits and software-defined radios operating on TV white space spectrum, thereby circumventing the 0.4-second on-air time limitations inherent in industrial, scientific, and medical (ISM) bands and extending communication distances [210]. Our evaluation encompasses real-world farm scenarios and trace-driven large-scale emulation. The results demonstrate that signals from 71 tags can be decoded with less than 1% SER

on the same frequency channel, showcasing a 10 \times increase in concurrency compared to existing methods. Furthermore, the overall throughput can be expanded to 1.46 Mbps across multiple frequency channels. Our routing scheme yields up to 1.5 \times lower energy consumption for tags and up to 9 \times shorter range for UAVs. In summary, the contributions of this paper are delineated as follows:

- We propose a comprehensive LoRa backscatter system with an aerial excitation source to enable scalable agricultural IoT.
- We develop practical methodology and efficient algorithms to optimize backscatter concurrency, network throughput, and energy efficiency, simultaneously.
- We prototype AeroEcho and evaluate its performance in real environments, emulation, and simulation. The results show that the maximum concurrency of AeroEcho is 10 \times of the state-of-the-art with the same frequency resources. Moreover, AeroEcho improves the overall throughput by 5.84 \times and individual tag data rate by 12 \times .

5.2 Preliminary

5.2.1 Backscatter in Agricultural IoT

IoT devices predominantly utilize battery-powered sensors for data transmission [11]. The transmission process involves generating baseband signals, modulating these signals with carrier signals, and amplifying the signals. The consumption of substantial energy for carrier modulation and amplification presents an issue for devices requiring frequent agricultural sensor data reporting. In contrast to active radios, backscatter radios use signals from excitation sources as carrier signals and modulate their data on top of carrier signals, significantly reducing power consumption [166, 203].

Typically, a backscatter system includes an excitation source, a tag, and a receiver. We can shift power consumption from the sensor node (tag) to a shared ambient source, where the energy use for wireless communication is negligible. However, rural farmlands lack infrastructure. Although low-power wide-area networks (LPWANs) are available, their gateways are sparsely distributed

kilometers apart [7, 10], unable to support long excitation distances or many concurrent tag transmissions. Instead, drones, widely used in fertilization and irrigation on smart farms, are a good choice for bringing the source close to the tag [2, 209].

5.2.2 LoRa Backscatter

Recent years have seen significant advancements in the design of LoRa-based backscatter radios, combining the low-power backscatter technology with long-range techniques to improve performance [166, 167]. LoRa is designed for IoT up to 10 km with Chirp Spread Spectrum (CSS) modulation [4, 154]. The basic unit of LoRa modulation is a linear up-chirp whose frequency increases linearly with time across the whole bandwidth [40, 211]. The key to LoRa modulation is that a time delay in a chirp can be transformed into a cyclic frequency shift. The initial frequency can modulate encoded data bits as cyclic time shifts. The demodulation process is ‘dechirp’ as defined in Equation 5.1. Where f_0 represents the initial frequency and f_c is the up-chirp. It multiplies a received chirp symbol with a base down-chirp (the conjugate of the base up-chirp $-f_c(t)$) whose frequency decreases linearly over time. After the Fast Fourier Transform (FFT), a peak appears at an FFT frequency bin, corresponding to the initial frequency of the received chirp symbol. LoRa defines N different initial frequency offsets to encode $\log_2 N$ bits.

$$\left(e^{j2\pi(f_0+f_c(t))t} \right) * \left(e^{-j2\pi f_c(t)t} \right) = e^{j2\pi f_0 t} \quad (5.1)$$

5.2.3 Asynchronous Decoding Rationale

When multiple tags are excited by the same COTS excitation source, traditional linear chirp-based backscatter signals will suffer severe collision issues. We can observe multiple energy peaks on the spectrum and cannot distinguish signals reliably [1, 205]. Netscatter and P²LoRa [168, 169] solve the problem by taking multiple frequency channels.

The nature of a non-linear chirp [171] makes it easy to solve collision issues. It still utilizes CSS modulation, replaces the linear chirp signals $f_c(t)$ with a non-linear chirp time-frequency function, and the dechirp of non-linear chirps can be expressed in Equation 5.1.

Orthogonality among different non-linear chirp types. The different types of non-linear chirps [205]

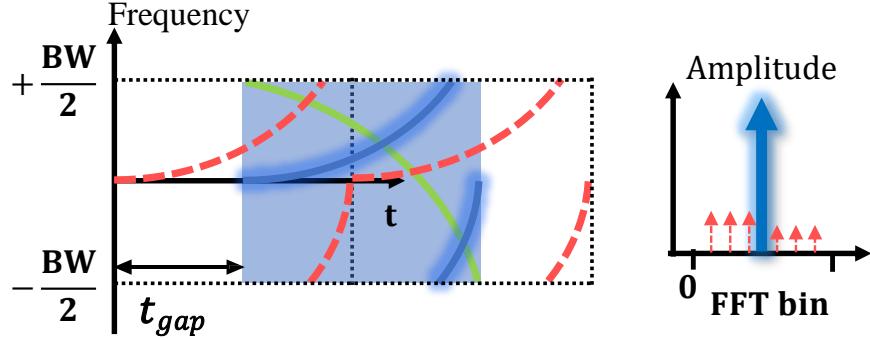


Figure 5.2 Asynchronous decoding rationale.

define the math function of non-linear quadratic chirp as $f_{non1}(t) = k_1 t^2 + k_2 t + k_0$ and quartic function as $f_{non2}(t) = z_1 t^4 + z_2 t^3 + z_3 t^2 + z_4 t + z_0$. After demodulation with non-linear chirp, the output signals are shown as follows respectively:

$$\begin{cases} F_{non1}(t) &= f_0, \\ F_{linear}(t) &= f_0 + \sum_{i=0}^1 x_i t^i - \sum_{j=0}^2 k_j t^j, \\ F_{non2}(t) &= f_0 + \sum_{m=0}^4 z_{4-m} t^m - \sum_{n=0}^2 k_n t^n \end{cases} \quad (5.2)$$

Only $F_{non1}(t)$ is one constant. A corresponding peak can be detected on the spectrum, while the energy of mismatched type symbols spreads over the spectrum. However, the orthogonal non-linear chirp types are limited and complicated to design and implement. There are only 6 given non-linear chirp formulas [171, 205]. It is hard to meet the requirements of hundreds of sensors deployed in agriculture.

Asynchronous decoding among the same non-linear chirp type. Further, an intrinsic property of non-linear chirps necessitates precise temporal alignment. This requirement provokes a reconsideration of our approach to non-linear chirp backscatter. The frequency function of non-linear is time-variant, leading to the spectrum energy distribution changing over time. As shown in Figure 5.2, the energy of red interference symbols spreads over multiple, clustered FFT bins due to the misalignment with the dechirp window. If we demodulate a quadratic non-linear chirp with a time offset t_{gap} , the mathematical function of dechirp can be expressed as follows:

$$F_{non1-offset}(t) = f_0 + k_2 t_{gap}^2 + 2k_1 t_{gap} \times t + k_2 t_{gap} \quad (5.3)$$

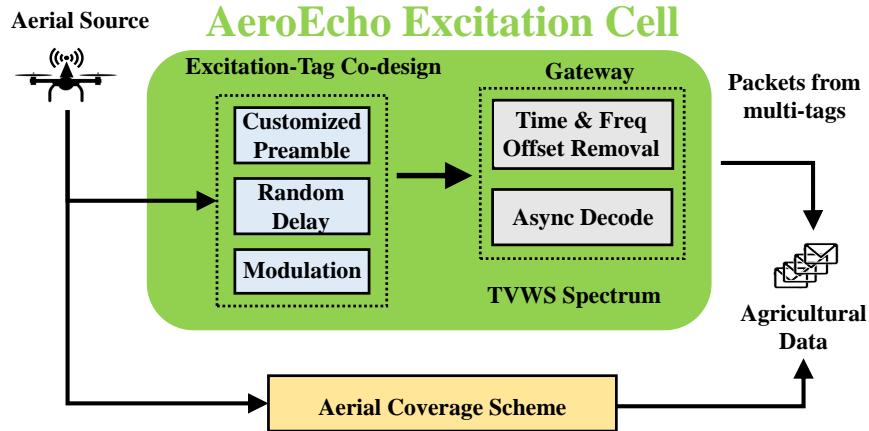


Figure 5.3 System overview.

The output frequency is a quadratic function of time instead of a constant. The scattering effects make it easy for the target symbol (blue solid line) to be demodulated, and we can see a strong energy peak on the spectrum. As a result, the energy of the interference symbol can be regarded as noise. We can do asynchronous decoding by sliding the dechirp window. The same type of non-linear chirps builds multiple quasi-orthogonal logical channels by time offsets among different chirps. However, creating time offsets cannot be applied to backscatter systems with standard LoRa excitation sources. This motivates us to co-design excitation-tag.

5.3 System Design

As shown in Figure 5.3, AeroEcho consists of a mobile excitation source on a UAV, backscatter tags, and a gateway. The basic unit of AeroEcho is the excitation cell. The single excitation cell takes the UAV excitation signals transmission location as center and the maximal excitation-source-to-tag distance as radius. There are multiple tags distributed in a single cell. Tags can be woken up (5.3.1) and modulate their own data on top of the carrier signals with a random time delay (5.3.2). At gateways, AeroEcho removes offsets and demodulates symbols (5.3.3). RF source on UAV transmits excitation signals with a prescribed coverage scheme (5.3.4) across multiple excitation cells. Finally, we can recover data bits from various tags.

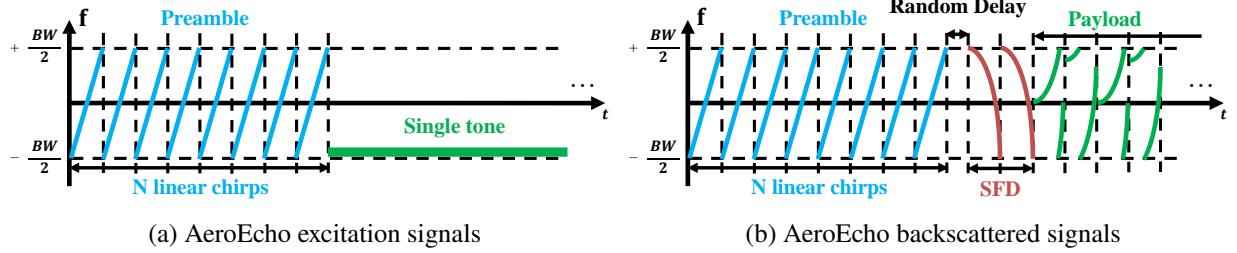


Figure 5.4 Illustration of AeroEcho excitation source signals and backscattered signals.

5.3.1 Excitation Source and Tag Co-design

When a UAV reaches the excitation point, it transmits preamble signals to activate AeroEcho backscatter tags. Initially, we employ eight repeated linear chirps as preamble signals, a design choice that aligns with the standard LoRa. We devised a passive preamble detection circuit to identify the arrival of these excitation signals. This circuit comprises impedance matching, an envelope detector, and a comparator. The impedance matching and the envelope detector are designed to recognize the repetitive pattern of the repeated linear chirps in the preamble. The comparator, the third component, evaluates if our tag can be activated by juxtaposing the reference voltage with the output voltage from the first two components. The excitation signals format is illustrated in Figure 5.4a. We adjust the amount and amplitude of linear chirps in the preamble with different transmission power to change the sensitivity of the preamble detection circuit. This enables AeroEcho tags to achieve different excitation source-to-tag distances in different excitation cells under various environmental factors (e.g. weather, agricultural activities) without additional hardware or software modification. Adjusting a UAV instead of individually reconfiguring multiple tags makes the system adaptive and scalable. Then, we utilize single-tone signals as carrier signals to allow flexible modulation for non-linear chirps.

As Section 5.2.3 mentions, time offsets among different chirp symbols can lead to scattering effects to distinguish signals from other tags. When a UAV hovers at a specific location and emits excitation signals, it can cover the backscatter tags within the circular area defined by its excitation cell. However, in agricultural settings, tags equipped with sensors are often distributed irregularly [2]. On the other hand, using such circular coverage patterns cannot seamlessly cover a farm

without overlaps. If fixed offsets are assigned, it becomes challenging to synchronize sensors at random locations with varying backscatter modulation times across multiple excitation cells. Every time we adjust the deployment of the farm or add new sensors, we need to reschedule the delay design among hundreds of tags. The maintenance cost is high. To enable concurrent transmission, AeroEcho tags give different random offsets ranging from 0 to 1 symbol time as waiting time after the preamble, as shown in Figure 5.4b. Different random time offsets among multiple tags can create orthogonal logical channels to support concurrent transmission.

5.3.2 AeroEcho Packet Format

After preamble detection, AeroEcho tag can wake up and assign a specific frequency shift, converting single-tone signals into non-linear chirps. We use a microcontroller to control the voltage output of the digital-analog converter. The voltage is the input of a voltage-controlled oscillator. After that, the RF switch and antenna adjust the impedance and radiate the signals, adding the frequency shift on the carrier signals. We formulate the voltage function accordingly to create a non-linear chirp generation that matches the signals' desired final time-frequency shape. In the time-frequency domain, non-linear chirp functions can be expressed as polynomial functions.

$$f_c(t) = \sum_{i=0}^n k_i t^i, t \in [0, \frac{2^{SF}}{BW}], f_c(t) \in [-\frac{BW}{2}, \frac{BW}{2}] \quad (5.4)$$

For a non-linear base up chirp of quadratic function, $k_0 = -\frac{BW}{2}$, $k_2 = \frac{BW^3}{2^{2SF}}$ and for encoded chirps, $k_0 = -\frac{BW}{2}$, $k_1 = -\frac{BW^2}{2^{SF-1}}$, $k_2 = -\frac{BW^3}{2^{2SF}}$ (mentioned in Section 5.2.3). After excitation signals identification and random time delay, we modulate single tone signals to generate two non-linear base down-chirps SFD (red curves) as shown in Figure 5.4b to synchronize initial time and frequency. This process is discussed in Section 5.3.3. For SFD chirps, $k_{SFD0} = \frac{BW}{2}$, $k_{SFD2} = -\frac{BW^3}{2^{2SF}}$.

Like linear chirp modulation, we transfer the cyclic time offset to the initial frequency offset for each symbol. We can generate different time-variant voltage curves with different cyclic time offsets to modulate multiple initial frequency symbols. Then, we can implement non-linear CSS modulation on backscatter tags, which is a considerable advantage compared to existing work with OOK. The spreading factor can vary from 1 to 12, which means AeroEcho encodes 1 bit to 12

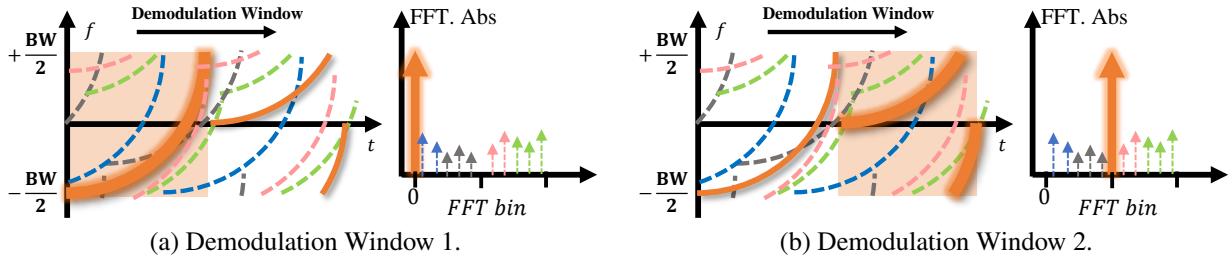


Figure 5.5 Illustration of the sliding window for non-linear chirps demodulation at the gateway.

bits for each symbol. The multiple spread factor and flexible initial frequency offset choices meet the multiple data rate requirements. We use the Heaviside step function [212] to express the time-frequency function of non-linear chirp with the impact of cyclic time offset (denoted as t_o) on the t can be expressed as follows:

$$f_{non1}(t) = -\frac{BW}{2} + \frac{BW^3}{2^{2SF}} [t^2 + (2\text{Heaviside}(t - t_o) - 1)t_o^2] \quad (5.5)$$

5.3.3 Asynchronous Decoding

Thanks to the single-tone excitation signals, AeroEcho do not need to operate complicated excitation carrier signals cancellation. The demodulation necessitates precise time synchronization owing to the sensitivity of non-linear chirps to time offsets, as outlined in Section 5.2.3. Random set delays and sampling time offset induced time offsets (TO) can distribute the spectral power across multiple frequency bins, and hardware-induced carrier frequency offset (CFO) shifts the energy peak. According to Equation 5.3 in Section 5.2.3, the initial frequency offset caused by TO and CFO for non1 can be expressed as: $\Delta f = k_1 TO^2 + CFO$.

To remove offsets, after the preamble and random time delay, we generate two non-linear base down-chirps as the start of frame delimiter (SFD) shown in Figure 5.4b. We multiply base up-chirps with two SFD chirps and identify the TO corresponding to the spectrum's two most substantial repetitive energy peaks. Then, we calculate the CFO according to the location of the shifted energy peaks. This effectively mitigates the influence of TO and CFO. Moreover, the non-linear SFD provides resilience against interference among tags.

After offset removal, we can receive signals like Figure 5.5 shown. For example, Orange, pink,

grey, green, and blue symbols come from four tags. We use sliding windows of one symbol length to demodulate signals from each tag. The sliding window aligns with the first orange chirp of initial frequency $-\frac{BW}{2}$, and a peak appears at the initial FFT bin. The receiver then takes other color symbols not aligned with the demodulation window as noise. When the sliding window aligns with the second orange symbol with an initial frequency of 0, a peak appears in the middle of the spectrum. Similarly, we can also decode all data from different tags.

5.3.4 Aerial Coverage Scheme

The RF source at UAVs transmits excitation signals at the centers of circles with a radius of \mathbf{r} equal to maximal D_{st} , creating basic excitation cells. Tags are distributed in a specific area with random locations. We aim to plan multiple excitation cells collecting data from tags with a specific density in a given area and keep the low SER. Meanwhile, we need to minimize the total energy consumption of all the backscatter tags and UAV flight ranges. The problem can be defined as follows:

$$\begin{aligned} & \underset{r,s,N}{\text{minimize}} \quad E, F = \frac{R}{s} \\ & \text{subject to} \quad \text{SER}_N \leq \text{SER}_{\text{threshold}} \quad N \geq \rho \cdot s \end{aligned} \tag{5.6}$$

R represents the flight range to cover the area s . E is the total energy consumption of backscatter tags and $F = \frac{R}{s}$ is the flight range efficiency of UAV, representing range per unit area. With concurrent backscatter transmission amount N for a single excitation cell, we must consider the symbol error rate SER_N , which is required to be less than or equal to a maximum acceptable threshold $\text{SER}_{\text{threshold}}$. Within this area, backscatter tags are distributed randomly but with a uniform deployment density symbolized as ρ . The concurrency N should also be larger than the tag amount (the product of ρ and s).

We propose two methods: Rectangular displacement and Annular trajectory scheme. The rectangular method achieves seamless coverage but wastes energy and exhibits data collection unfairness due to repetitive tag wake-ups. The annular method is energy-efficient but requires multiple rounds to cover all tags and is not suitable for time-sensitive applications.

Rectangular Displacement Coverage: As illustrated in Figure 5.6, the gateway is at the center of

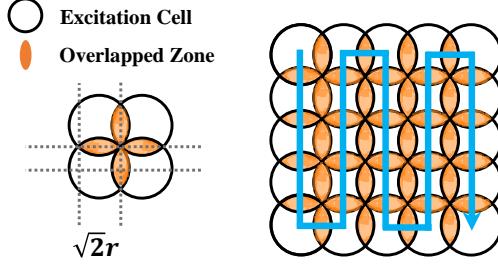


Figure 5.6 Rectangular Coverage Scheme

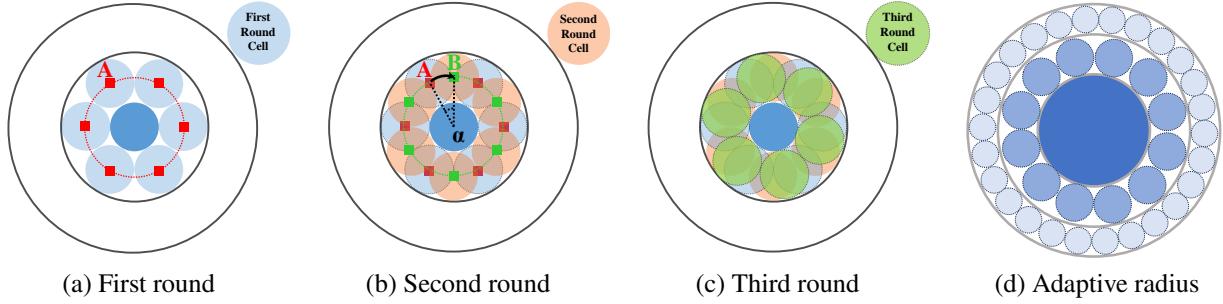


Figure 5.7 Illustration of annular coverage schemes.

the figure. We have m^2 excitation cells, and the column and row are \mathbf{m} for the square deployment. Four cells have only one intersection point and four identical overlapped orange zones (orange shape). Then the UAV can follow the blue trajectory to travel all the excitation points and seamlessly collect all the sensor data. The width of the orange shape is $\sqrt{2}r$. We can calculate the overlapped area with a geometric method as follows:

$$\left\{ \begin{array}{l} S_{overlap} = m(m-1)(\pi - 2)r^2 \\ E_s = e\rho(S_{overlap} + S) \quad R_s = (i^2 - 1)r \end{array} \right. \quad (5.7)$$

However, the tags in the overlapped orange areas are triggered to perform backscatter modulation multiple times in the same collection round. This may lead to energy waste and unfairness in IoT sensor data collection. This creates complicated maintenance problems and is not acceptable for some energy-sensitive applications.

Annular Trajectory Coverage: As depicted in Figure 5.7a, the dark blue cell is the initial excitation annulus. UAV initiates the transmission of the first excitation signals from the central point of the dark blue cell, which also serves as the gateway location. The drone then moves to the next adjacent

annulus, traversing the centers of the light blue excitation cells from the starting point A. When the drone reaches all the red points and transmits the excitation signals, it is termed one round. To cover all the tags in this annulus, AeroEcho utilizes multiple rounds at each annulus. Figure 5.7b illustrates the second round. The shallow orange circles are excitation cells in the second round with beginning point B. After two rounds, the angle offset on flight trajectory between A and B is α_1 . Above 95% tags' data can be collected. The drone flies and transmits signals along the green center points for each round. Likewise, in the latter rounds, the drone starts with a predefined angle offset α from the start point of the last round. One of the optimal settings of initial α_1 is $\frac{\pi}{6}$. For the latter two rounds, we take two quartiles in $[0, \alpha_x]$ as α_{x+1} and α_{x+2} . Figure 5.7c illustrates the third round coverage scheme with the green shallow circles ($\alpha = \frac{\pi}{12}$), which eventually cover over 98.7% area for the annulus.

As Equation 5.8 shows, C_i symbolizes the ratio of the annular area to the average coverage area of each round. This implies that ensuring complete coverage of all the tags within one annulus requires a flight distance C_i times the single annular trajectory's distance. $1 \leq C_i \leq 1.35$ can guarantee the data collection of more than 75% sensors for each round. Every tag can only be triggered $\frac{1}{C_i}$ in each round on average. The experimental results of coverage rate are discussed in Section 5.5. The flight range of UAV covering all the tags of the current annulus can be denoted as R_c on average.

$$d = \sum_j r_j \quad C_i = 4dr_j \cdot \arcsin \frac{r_j}{d} \pi r^2 \quad R_c = 2\pi \sum_i C_i \cdot d \quad (5.8)$$

Given the limited battery capacity, we aim for the UAV to travel as short as possible to ensure minimal energy consumption by backscatter tags. A larger excitation radius can expand the single-cell area, resulting in a shorter displacement of the drone. However, a larger radius also implies a reduced signal strength from the tag to the receiver and a shorter range. To cover with higher efficiency, we propose an adaptive radius scheme. As depicted in Figure 5.7d, we use three excitation cells with different radii. The same excitation cell is arranged in the same annulus. From the inside out, we select the maximum radius at varying distances while ensuring the tag-to-receiver



Figure 5.8 Implementation of the excitation source, tags, and gateway in the farm.

distance does not exceed the current distance. This strategy provides the UAV's range efficiency and coverage.

5.4 Implementation

Figure 5.8 shows the devices we use for our experiments. The wake-up module consists of a three-stage voltage-doubling amplifier HSMS-285C [174] and a low-power voltage comparator LPV7215MG [175]. The modulation module includes STM32L011 [176] MCU, a low-power voltage-controlled oscillator LTC6990IS6 [178] and a reflective RF switch ADG902 [179]. The excitaion source is HackRF One [213] on DJI Spark at TVWS spectrum. We use GNU-radio to control a USRP N210 [152] as a gateway, and then we do signal processing in MATLAB. The total energy consumption AeroEcho tag is $538\mu W$. The cost is less than 10 US dollars.

5.5 Evaluation

In this section, we conduct experiments to verify the performance of single tag, concurrent transmission, throughput, data rates and UAV routing scheme. The default SF=12, bandwidth (BW) is 250kHz, the frequency band is 470MHz at TVWS spectrum, the coding rate is $\frac{4}{5}$, and transmission power is $14dBm$. The default non-linear chirp type is quadratic $1-f(t) = t^2$ as mentioned in 5.3.2. AeroEcho tag modulates 28 symbols of information on each packet. We conduct experiments in different scenarios with multiple source-to-tag distance(D_{st}), source-to-receiver distance(D_{tr}). The SER is set to 10^{-4} if all the symbols are decoded successfully.

Metrics: Symbol Error Rate (SER): SER is the ratio of data symbols being incorrectly decoded due to noise or other impairments. A lower SER indicates a more robust and efficient transmission

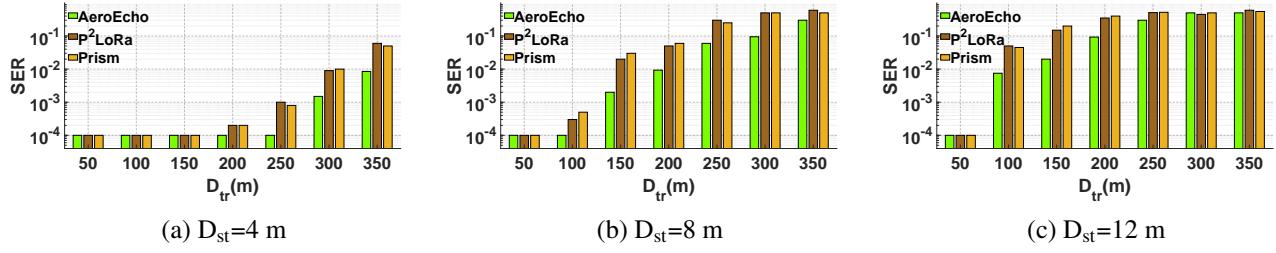


Figure 5.9 Impact of excitation cell radius to BER of AeroEcho with different D_{tr} .

system. **Concurrency**: The maximal tags amount to a backscatter to support simultaneously with the corresponding SER threshold. **Throughput**: Actual speed at which data is successfully transferred for the backscatter networks with all the working tags. **Data rates**: The maximum number of data bits transmitted for each tag per second. **Energy Consumption**: The total trigger times of backscatter tags per unit area. **Range Efficiency**: The average flight range of the UAV for covering each tag.

Baseline: Netscatter [169], P²LoRa [168] and Prism [205].

5.5.1 Excitation-tag co-design performance

Experiment Settings: In outdoor experiments, we adopt a single tag to verify the non-linear chirp backscatter signal performance with TVWS. We put the USRP receiver at a fixed location and moved the tag from 50m to 350m while keeping the excitation source on the drone at 3m height with multiple $D_{st} = 4\text{m}$, 8m , or 12m in horizontal distance to AeroEcho tag. Prism and P²LoRa with SF12 and 250kHz at 915 ISM bands are baseline methods.

Results: As shown in Figure 5.9a with Logarithm scale, we can discover that AeroEcho, Prism and P²LoRa all can decode all the bits successfully if D_{tr} is equal to or less than 150m. As D_{tr} grows from 200m, the SER of the two baseline methods increases more than AeroEcho. When the $D_{tr} = 300\text{m}$, the SER of the two baseline methods reduces to about 1% while AeroEcho only is 0.15%. Finally, the SER declines and achieves 0.009, 0.06, and 0.05 for AeroEcho, P²LoRa and Prism, respectively. In Figure 5.9b, we can also see that the error demodulation occurs when $D_{tr} = 100\text{m}$ for P²LoRa and Prism. When the $D_{tr} = 200\text{m}$, the SER of AeroEcho reaches about 1% and the SER of Prism and P²LoRa is 6 \times and 5 \times of the SER of AeroEcho. Finally, the SER for two

Table 5.2 SIR and max D_{tr} with different excitation cell radius

Radius (m)	4	6	8	10	12	16
Max D_{tr} (m) (SER=1%)	350	260	200	150	120	50
SIR (dB)	4.5	7.2	9.8	11.5	13.3	16.1

baseline methods becomes larger than 10% when $D_{tr} = 250m$. As shown in Figure 5.9c, we can also observe that all symbols can be decoded successfully when $D_{tr} = 50m$. When the $D_{tr} = 100m$, the SER of AeroEcho reduces to 0.008, greatly lower than that of Prism and P²LoRa, which are 0.05 and 0.045, respectively. Finally, the SER of AeroEcho and two baseline methods rises to more than 10% when $D_{tr} \geq 200m$. Based on the results, we also measure and find the maximal D_{tr} of SER=1% with 6m, 10m, 16m D_{tr} , as shown in Table 5.2. This helps the experiments for the UAV routing scheme.

Remark: In conclusion, the performance regarding SER under different communication distances of a single backscatter tag of AeroEcho at TVWS bands is greater than that of two baseline backscatter techniques operating at ISM bands.

5.5.2 Excitation Cell

We conduct trace-driven experiments to explore the impact of maximal concurrency on excitation cell radius (maximal D_{st}) under massive concurrent collision. Different excitation cell radii lead to different SIR (Signal-to-interference-ratio) ranges. SIR range refers to the ratio of the strongest and weakest signals. We collect non-linear chirps in real environments and do large-scale collision emulation with different SIR to determine maximal concurrency under different SER thresholds.

SIR Experiments: We use fixed D_{tr} and then we move UAV to different relative distance to tag with fixed location, making different excitation cell radius with maximal horizontal D_{st} from 0 to 4m, 6m, 8m, 10m, 12m or 16m. The experimental deployment is shown in Figure 5.10. Gateway is located 100m away from tag. In the beginning, the UAV is right above the tag. Then UAV moves right between the tag and gateway. Afterward, UAV moves circularly with step $\frac{\pi}{2}$ to three other locations. We measure the SIR(maximal SNR variation) in different configurations. The results

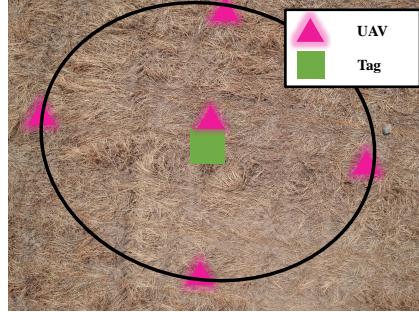


Figure 5.10 The deployment of different excitation cell traces collection (drone's view).

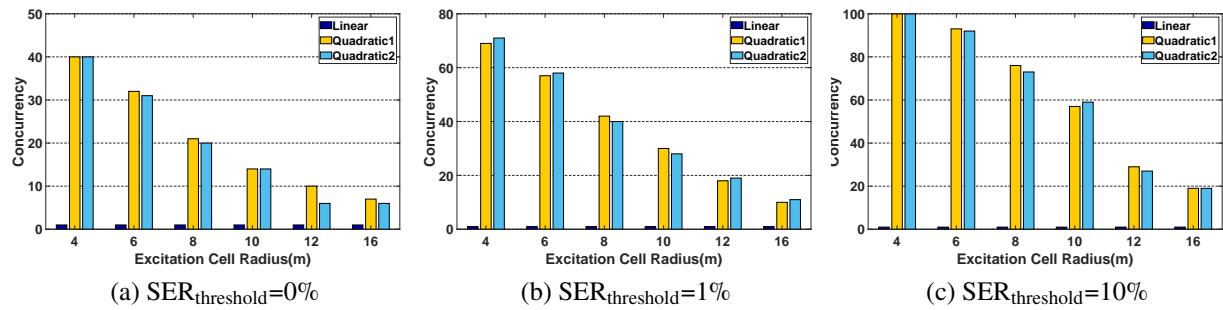


Figure 5.11 Impact of excitation cell radius to concurrency of AeroEcho and linear chirp system.

are shown in Table 5.2. We can observe that SIR increases with D_{st} increasing.

Concurrency Experiment Settings: We collect accurate signals from different locations with diverse channels to conduct large-scale emulation of massive collisions. We improve the link diversity by varying SIR randomly in six ranges as listed in Table 5.2. We also adopt a random time delay among symbols ranging in [0,1] symbol time. We add multiple symbols from 2 to 100 and then decode each symbol using a non-linear down-chirp template with sliding windows. Two kinds of non-linear quadratic chirp and linear chirp are used. The math abstract formula of two non-linear chirps are non1- $f(t) = -\frac{BW}{2} + \frac{BW^3}{2^{SF}}t^2$ and non2- $f(t) = -\frac{BW}{2} + \frac{BW^2}{2^{SF-1}}t - \frac{BW^3}{2^{2SF}}t^2$.

Results:

Figure 5.11 shows the concurrency capacity with different SER thresholds. In Figure 5.11, it is evident that linear chirps suffer from severe collisions. Linear chirp-based backscatter cannot support concurrent transmission. When the excitation cell radius is 4m, quadratic1, and quadratic2 can successfully decode all the data bits when concurrency is 40, as Figure 5.11a shows. The two can achieve 7 and 6 tags when the excitation cell radius is 16m. Figure 5.11b illustrates the overall

Table 5.3 Throughput comparison between different LoRa backscatter techniques and combinations

	Netscatter	P ² LoRa	Prism + P ² LoRa	AeroEcho (1%)	AeroEcho (10%)	AeroEcho (10%) + Prism	AeroEcho (1%) + P ² LoRa	AeroEcho (1%) + P ² LoRa (SF=10)
Throughput	1.95kbps	2.82kbps	9.77kbps	20.5kbps	33.0kbps	132kbps	1.03Mbps	1.46Mbps
Tag data rates	30.5bps	30.5bps	30.5bps	366bps	330bps	330bps	366bps	1220bps
Concurrency	64	100	400	70	125	500	3500	1500

concurrency with SER=1% is larger than the concurrency without errors at each radius. Quadratic1 and quadratic2 support (69,71), (42 40), (10 11) concurrent transmission with radius=4m, 8m, and 16m, respectively. We can also observe that they can support more than 100 concurrency when radius=4m and SER=10%. Even with the largest radius of 16m, they can achieve 19 concurrency. Prism [205] can only support 7 tags with SER=1% due to the availability of only 7 chirp types. AeroEcho is 10× the concurrency capacity of Prism and 70× of linear chirp-based backscatter system.

Remark: When the excitation cell radius grows, the SIRs among different backscattered signals will also increase, causing less concurrency. A larger radius also means faster coverage and a shorter UAV range. We carefully select the radius to balance coverage density and UAV range.

5.5.3 Throughput and data rates

Experiments Settings: To evaluate the current transmission throughput performance of existing backscatter techniques, we conduct a series of experiments at TVWS bands. As shown in Table 5.3, we emulate the overall throughput and single tag data rates under various concurrency values with their theoretical maximum concurrency. Our experiments involve multiple setups. We use P²LoRa and Netscatter for basic throughput measurement with concurrency 64 and 100 respectively. We integrate Prism with P²LoRa, expanding to 100 frequency channels and incorporating 4 non-linear chirp types. We also measure the performance of AeroEcho with two different SER thresholds (1% 10%) with 70 and 125 concurrency respectively. Moreover, AeroEcho (SER=10%), combined with Prism, creates orthogonal logical channels by assigning 125 random offsets across 4 non-linear chirp types. We combine AeroEcho (SER=1%) with P²LoRa and implement 70 random time offsets with a single non-linear chirp type over 50 frequency channels. These experiments

aim to thoroughly evaluate the performance of backscatter communication systems under different configurations and channel conditions. $Sf=12$ and $BW=125\text{kHz}$.

Results: As Table 5.3 shown, maximum concurrency of Netscatter is 64 with 125KHz. P²LoRa takes 100 frequency channels to achieve only 2.82kbps. Prism + P²LoRa extends the orthogonal channels by non-linear chirp type but is still restricted by the frequency channels. The first three methods have the same data rate for individual tags: 30.5bps. When combining AeroEcho with multiple non-linear chirp types or orthogonal frequency channels, they can all achieve the same 366bps single tag data rate, 12× of the conventional methods. The overall throughput of AeroEcho (1%) + P²LoRa can be up to 1.03Mbps. When the SF=10, we emulate experiments with 50 non-linear chirps with random offsets from 50 tags in 30 orthogonal frequency channels, and the overall throughput can be up to 1.46Mbps, which is 5.84× the max throughput of the previous backscatter system (250bkps of Netscatter [169]). In addition, we also verify the single tag rate can be up to 1.46kbps when bandwidth is 500kHz and SF is 12. This indicates that the flexible non-linear CSS modulation enables AeroEcho to encode more data for each symbol or tag, which supports higher overall throughput than previous methods. In addition, the data communication performance of AeroEcho can be easily extended when combined with orthogonal frequency methods or multiple non-linear chirp types methods without extra loss.

5.5.4 Aerial Routing Scheme

In this section, we compare the energy consumption of tags between the rectangular scheme and the annular scheme. We also compare the range efficiency of UAV between fixed radius and adaptive radius of annular AeroEcho.

Energy Experimental Settings: According to the findings detailed in Section 5.5.1, we determined the maximum transmission distance (D_{tr}) across varying radii of excitation cells. We adopted a 1% SER as the reliability benchmark for the data collection system. In the rectangular scheme, the count of columns and rows varies as integers from 1 up to K, with K signifying the point at which the furthest excitation cells attain their maximum D_{tr} for a given radius. Similarly, we define the number of concentric circles as integers from 1 to N for the annular scheme, where N indicates when

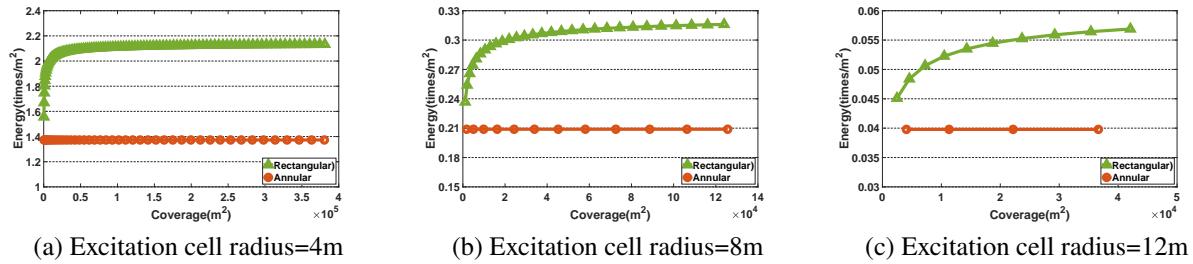


Figure 5.12 Comparison of backscatter tags energy consumption with different excitation cell radii.

the outermost excitation cells reach their peak D_{tr} . Both schemes maintain an identical density of tag distribution. We calculate the total coverage area using each method's maximum D_{tr} . This enables us to simulate the overall trigger frequency of backscatter tags per square meter as the energy consumption metric.

Results: Figure 5.12 illustrates the energy consumption differences between the annular and rectangular schemes. At a 4m radius, the annular scheme maintains low energy consumption at approximately 1.37, while the rectangular scheme's energy consumption spikes to 2 at a coverage area of $7300m^2$, remaining around 2.1 times/ m^2 for larger areas—about 1.5 times higher than the annular scheme. At an 8m radius, the annular scheme's energy consumption is roughly 0.21, with the rectangular scheme's trigger times per square meter reaching 0.316, indicating slower energy growth. Coverage is constrained for a 12m radius due to a significantly shorter maximum D_{tr} , with the annular scheme at 0.04 times/ m^2 and the rectangular scheme increasing from 0.045 to 0.057. The energy disparity between the two schemes grows as the coverage area expands.

Remark: The study reveals that while the rectangular scheme leads to more incredible energy waste for backscatter tags than the annular scheme, it enhances UAV range efficiency. Thus, the annular scheme is preferred for energy-sensitive backscatter tags, and the rectangular scheme is better for optimizing the UAV range.

Coverage Rate Experiments Settings: To verify the performance of coverage reliability of the annular coverage scheme, we simulate the coverage rate for each annulus from inside to outside and compare it with the rectangular coverage scheme.

Results: Figure 5.13 illustrates the coverage rate of two rounds for annulus 2 to 44 from inside to

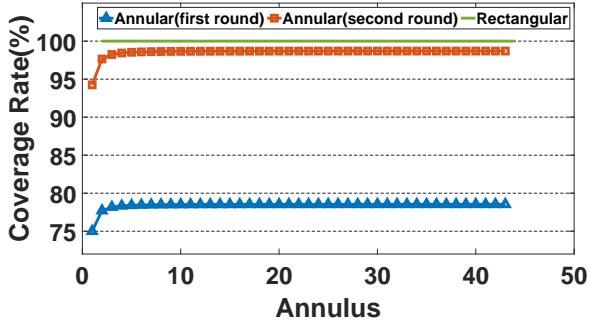


Figure 5.13 Coverage rate at the 1st and 2nd rounds.

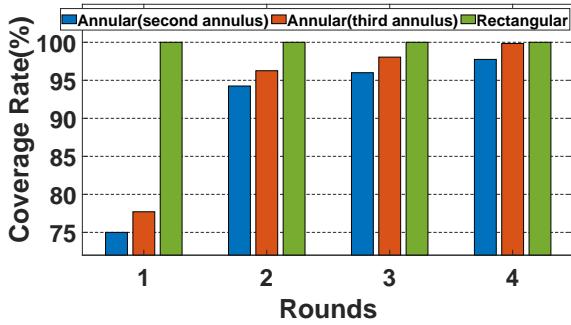


Figure 5.14 Coverage rate at 4 rounds.

outside. We can observe that the rectangular scheme achieves full coverage. In the first round, the coverage rate of annular mostly reaches 75% to 80%. In the second round, the most annulus can achieve more than 97% coverage rate except the second annulus. Figure 5.14 shows the coverage rate for the second and third annulus from 1–4 rounds. The second annulus covers 97.75% tags, and the third annulus covers 99.84% tags in the fourth round, respectively. Both can achieve 96% in the third round.

Range Efficiency Experimental Settings: To satisfy different sensor coverage densities, we need to select an adaptive cell radius to achieve the maximal range efficiency for UAVs. We compare adaptive radius with fixed radius to conduct experiments with different density levels and coverage radii. Density is 1 tag/m². The radius of the coverage area is 350m.

Results: As Figure 5.15 illustrates, the range efficiency of the two methods has the largest difference of 0.04m/tag when the coverage radius is 350m. With coverage increasing, the available excitation cell radius reduces, and the difference between the two methods decreases. The range efficiency is

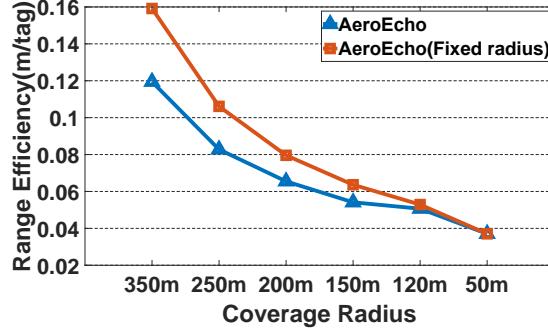


Figure 5.15 Range efficiency of fixed radius and adaptive radius.

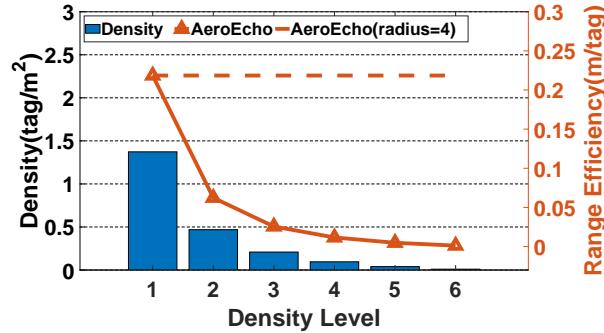


Figure 5.16 UAV range efficiency with radius adaption and tags density.

0.037 when the coverage radius is 50m, which means only an excitation cell with a 16m radius can be used. This implies that AeroEcho performs better in range efficiency with more excitation cells with different radii and further coverage area. Figure 5.16 illustrates maximal density for excitation cells with different radii. From density levels one to six, we have 1 to 6 available excitation cell radii, respectively. The range efficiency at density level 1 and density level 6 are 0.2185 and 0.0012, respectively. It is obvious that range efficiency decreases from density level 1 to 6, and it decreases more and more slowly.

Remark: Adaptive excitation cell radius can achieve better range efficiency than inflexible settings.

5.6 Related Work

Long Range Backscatter: Talla et al. [167] employ a tone signal to create a linear LoRa packet by shifting frequency. PLoRa [166] manipulates signals with two distinct frequency shifts. Utilizing special excitation signals, Netscatter [169] decodes numerous LoRa packets simultaneously by merging CSS modulation with OOK. PACT [214] enables concurrent transmission but relies on

expensive hardware each priced between 50-290 USD. P²LoRa [168] uses the existing LoRa signals for parallel decoding but requires substantial bandwidth and frequency resources. Prism [205] achieves concurrency with different non-linear chirps to create a limited orthogonal coding space. Contrastingly, AeroEcho proposes the using same type of non-linear chirps and mobile excitation to improve concurrency and scalability further with low overhead.

LoRa Collision Resolving: LoRaWAN, utilizing the ALOHA protocol [185], experiences collision when multiple nodes transmit simultaneously. Various solutions, such as Choir [186], FTrack [187], and CIC [188], resolve these collisions by extracting unique features from overlapping packets in the time or frequency domain. Others such as mLoRa [189], CoLoRa [215], Pyramid [191], NScale [192], and PCube [216] use interference cancellation, spectral peak ratios, energy peak tracking, peak scaling factor variation, and unique phase utilization, respectively. However, they lack a collision-resolving mechanism and suffer from the near-far problem [1]. CH-MAC [217] uses coding and hopping, and LMAC [194] attempts to avoid collisions using CAD and CSMA at the MAC layer, but they are energy-intensive for LoRa backscatter systems. AeroEcho employs non-linear chirps with backoff to create new logical channels, offering a practical solution for concurrency.

UAV Routing for Backscatter system: Yang et al. [218,219] use the UAV as receiver and provide a multiple access solution for time division. Han et al. [220] optimize the trajectory by detecting the presence of parasite devices. Previous studies offered general UAV-aided backscatter solutions, which lacked specificity for agricultural contexts to solve the high cost and scalability issue. AeroEcho introduces an energy-efficient backscatter system for reliable data collection in agricultural IoT environments.

5.7 Conclusion

To conclude, we develop a TVWS long-range backscatter system AeroEcho for agricultural IoT scenarios by using non-linear chirps to enable concurrent transmissions and a UAV as a mobile excitation source. We design the excitation signals on UAVs and modulation methods on backscatter tags to improve the throughput of concurrent transmission by setting different time offsets among con-

current backscatter tags. We also adopt a non-linear SFD to synchronize backscatter signals from multiple tags at the gateway side. The routing scheme achieves the balance of energy/coverage efficiency and UAV flight range. We implement AeroEcho with customized low-cost hardware and software-defined radio. We evaluate its performance with real environment signals. The results show that 71 tags can transmit concurrently with less than 1% bit error rate by using the same non-linear chirp in the same channel, resulting in a 10 \times higher transmission concurrency than state-of-the-art. Moreover, AeroEcho improves the overall throughput of current backscatter transmission by 5.84 \times and individual tag data rate by 12 \times compared to the state-of-the-art.

CHAPTER 6

CHIRPTRANSFORMER: VERSATILE LORA ENCODING FOR LOW-POWER WIDE-AREA IOT

6.1 Introduction

Long Range (LoRa) technology has emerged as a highly promising solution for seamlessly connecting unattended Internet-of-Things (IoT) devices on a large scale [7, 22]. As of March 2023, the global presence of LoRa networks has expanded significantly, with 181 public operators worldwide facilitating connectivity for over 300 million IoT devices [221]. This wide-area IoT technology has found application in a multitude of scenarios. For instance, Amazon Sidewalk [222] employs LoRa to bridge connections with smart IoT devices located beyond the range of conventional home Wi-Fi networks. In the agricultural sphere, Microsoft FarmBeats [223, 224] utilizes LoRa to efficiently gather data from remote sensors deployed across vast farmlands. Additionally, LoSee [95, 225, 226] leverages LoRa to track shared bicycles within urban environments.

Figure 6.1 shows the network architecture underlying LoRa systems for IoT data collection [22]. LoRa nodes are responsible for encoding sensory data and transmitting data packets to LoRa gateways. These gateways serve as crucial intermediaries; upon receiving the packets, they decode the data and subsequently relay it to network and application servers through backhaul networks. It is important to note that LoRa nodes, equipped with low-cost commercial-off-the-shelf (COTS) Semtech LoRa radios [134, 227], operate under constrained resources. In stark contrast, LoRa gateways and servers utilize potent COTS Semtech radio [112, 113], some even employing Software Defined Radio (SDR) [152], to manage energy-intensive and computationally demanding tasks. Furthermore, distinct LoRa applications exhibit diverse performance considerations. For example, agricultural and urban deployments prioritize ubiquitous coverage [11, 92, 95, 136, 224], whereas industrial dense deployments contend with potential packet collisions necessitating scalable throughput [171, 228, 229]; In mobile and semi-outdoor applications (e.g., transportation, elder-care), the dynamic environmental conditions can swiftly deplete the energy of LoRa nodes, underscoring the need for sustainable adaptation strategies [230–232]. Consequently, existing

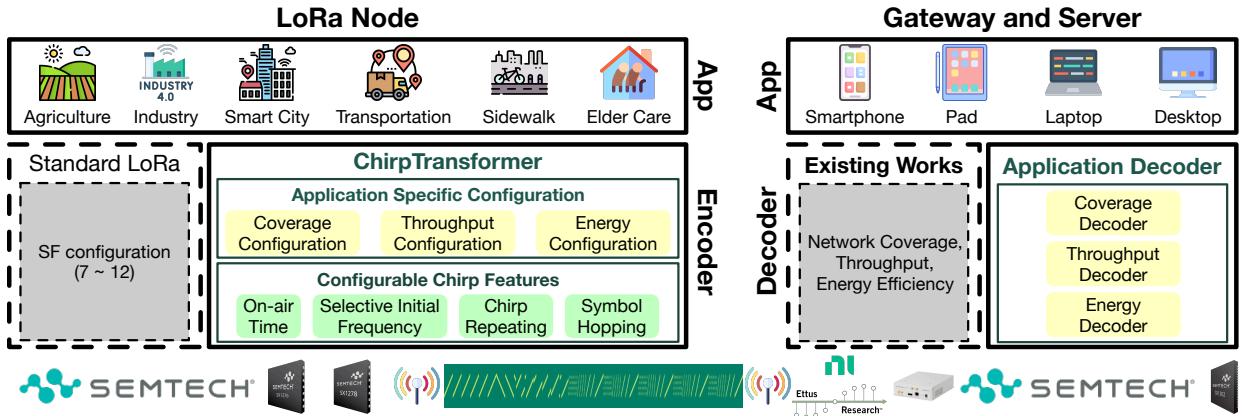


Figure 6.1 An illustration of the ChirpTransformer encoding framework in LoRa systems.

works predominantly focuses on optimizing network coverage [40, 100, 155, 232–237], throughput [189, 191, 229, 238–242], and energy efficiency [230, 243–246] at the gateway and server side, often leaving LoRa nodes untouched due to their resource limitations.

However, the efficacy of decoding designs in these systems heavily relies on how LoRa nodes encode data. LoRa employs Chirp Spread Spectrum (CSS) modulation, where data is encoded by linear chirps. The standard LoRa encoder uses a configurable parameter known as the spreading factor (SF) to determine the data rate of a chirp symbol. Notably, all chirp symbols within a LoRa packet adhere to the same SF setting. With COTS LoRa nodes offering only six available SFs ranging from 7 to 12 for configuring LoRa encoding, the standard LoRa encoder's reliance solely on packet-level SF configuration presents limitations in optimizing end-to-end performance across diverse application scenarios concerning network coverage, throughput, and energy efficiency.

In this work, we present an innovative approach to LoRa encoder design aimed at enhancing the versatility and overall performance of LoRa networks. Our proposed solution, ChirpTransformer, reimagines the conventional LoRa encoding framework by introducing a set of four configurable chirp features. These features enable a more extensive range of encoding options at LoRa nodes, as depicted in Figure 6.1. Through the utilization of these configurable parameters, a diverse array of encoding techniques can be devised and tailored to meet the specific coverage, throughput, and energy efficiency requirements of diverse applications. Furthermore, our approach extends beyond node-side enhancements; we have developed corresponding decoding methodologies for gateways

and servers. These decoding methods leverage the pre-configured chirp features implemented by the encoders, aiming to optimize the end-to-end performance of the LoRa network comprehensively.

Challenge #1: Chirp Feature Design. Our objective is to construct a set of specific chirp features that collectively create an informative feature space. However, the pursuit of achieving this goal does not involve indiscriminately adding numerous chirp features, as this would unnecessarily complicate the LoRa network stack. To tackle this challenge, ChirpTransformer goes beyond solely incorporating time and frequency domain data; it strategically designs four distinct chirp features aimed at enhancing the informativeness of the feature space. First, ChirpTransformer controls the on-air time of a symbol, which encompasses one or more chirps. Second, ChirpTransformer selects a collection of initial frequencies that a chirp can employ to encode data. Third, ChirpTransformer involves a novel intra-symbol chirp pattern. Departing from the conventional approach of comprising a symbol with just one chirp, a ChirpTransformer symbol may consist of repeated chirps sharing the same initial frequency offset. Fourth, ChirpTransformer incorporates an innovative inter-symbol chirp pattern. In contrast to using identical types of chirps, ChirpTransformer employs diverse chirp patterns across different symbols within a LoRa packet.

Challenge #2: Implementation on COTS LoRa nodes. Numerous chirp features, such as non-linearity [12, 171, 247] and interleaving [248], pose implementation challenges on COTS LoRa nodes, necessitating costly node replacements, especially in large-scale deployments. To circumvent this issue, the generation of these designed chirp features must not require hardware modification or additional energy consumption. To address this challenge, we have devised a lightweight symbol converter that transforms the chirp features – such as on-air time, selective initial frequency, and chirp repeating – into corresponding chirps configured for SF. Furthermore, we leverage an existing hardware interrupt specifically designed for frequency hopping to incorporate our symbol-hopping chirp feature without incurring extra overhead.

Challenge #3: Application-specific Encoder-Decoder Co-design. With our suite of four chirp features, we unlock the potential for multiple encoder configurations to enhance application-specific performance. To maximize these performance gains, our focus lies in optimizing performance

through an end-to-end approach, encompassing encoder configuration and decoder co-design. To address this challenge, we have undertaken three case studies targeting network coverage, throughput, and energy efficiency:

- 1) *Network Coverage*: Addressing weak signals is crucial [95, 100]. The standard LoRa encoder extends on-air time to ensure adequate energy reception, countering noise interference. As the SF increases, the required signal-to-noise-ratio (SNR) for successful decoding decreases. ChirpTransformer takes a different approach, enhancing noise tolerance by widening the feature distance between adjacent symbols and leveraging a neural-enhanced decoder to maximize SNR gain (§6.4).
- 2) *Network Throughput*: Enabling concurrent transmission [171, 229, 241] is a primary concern. While LoRa permits simultaneous transmissions with different SFs, the varying on-air times across these SFs result in imbalanced SNR tolerance among concurrent transmitters. ChirpTransformer utilizes intra-symbol chirp repeating to create novel orthogonal encoding configurations and employs template-based decoding to resolve collisions (§6.5).
- 3) *Network Energy Efficiency*: Fine-tuning encoding configurations for optimized data rates under various noise levels is crucial [230, 243]. The standard LoRa encoder supports only six data rates, whereas ChirpTransformer achieves 23 configurations by adjusting the number of available initial frequencies using a selective peak searching algorithm for decoding. (§6.6).

We have implemented end-to-end ChirpTransformer systems for each case study, utilizing COTS LoRa nodes and a USRP N210 SDR. Using our expansive campus-scale testbed spanning 2800 m × 1700 m, we have conducted extensive experiments to evaluate ChirpTransformer’s performance against both standard LoRa and a state-of-the-art benchmark. The results confirm ChirpTransformer’s superiority across all three case studies.

In summary, our contributions are listed as follows:

- We propose ChirpTransformer, an innovative LoRa encoding framework incorporating four chirp features. This framework significantly enhances encoding methods, enabling efficient adaptation to diverse applications.
- ChirpTransformer seamlessly integrates with COTS LoRa nodes without introducing additional

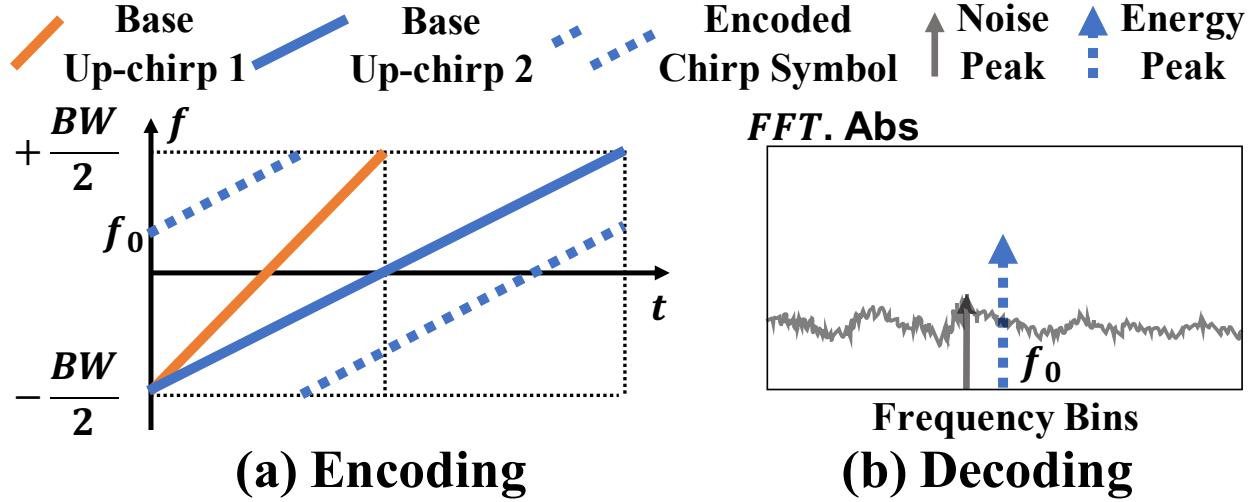


Figure 6.2 The illustration of (a) shifted initial frequency encoding and (b) energy peak detection decoding.

overhead. Through three comprehensive case studies, we showcase its ability to optimize end-to-end network performance by adopting an encoder-decoder co-design.

- We implement ChirpTransformer on COTS LoRa nodes and conduct extensive experiments using a campus-scale testbed. The results demonstrate ChirpTransformer’s exceptional performance, achieving a $2.38 \times$ increase in network coverage, a $3.14 \times$ improvement in network throughput, and a $3.93 \times$ battery lifetime compared to the standard LoRa.

6.2 Background and Motivation

6.2.1 Standard Encoding and Decoding

Figure 6.2 illustrates the encoding and decoding method of the standard LoRa. As shown in Figure 6.2(a), the standard LoRa encoder uses bandwidth (BW) to configure a base up-chirp (e.g., the orange chirp), whose frequency increases linearly from $-\frac{BW}{2}$ to $\frac{BW}{2}$ over time. Notably, the on-air time of a base up-chirp is adjustable. For example, the blue base up-chirp 2 spends more time swiping the whole BW than the orange base up-chirp 1. LoRa uses SF configuration, ranging from 7 to 12, to control the on-air time. Given the pre-configured BW and SF , the on-air time is $\frac{2^{SF}}{BW}$. Thus, the on-air time is doubled when SF increases by 1.

Given the base up-chirp, data bits are encoded by shifting the initial frequency of a base up-chirp to f_0 . For example, in Figure 6.2(a), we shift the initial frequency of the blue base up-chirp

2 to f_0 . When the chirp frequency meets $BW/2$, the rest of the chirp will be shifted down and restarts from $-BW/2$. As a result, the blue dashed chirp forms a typical chirp symbol. Moreover, The range of f_0 is $[0, BW)$. The number of data bits encoded by a chirp symbol is also determined by the frequency chip, indicating the distance between two adjacent initial frequency offsets. On COTS LoRa nodes, the frequency chip is $\frac{BW}{2^SF}$. Thus, in $[0, BW)$, we have 2^SF initial frequencies to represent data. Given an SF configuration, a chirp symbol encodes SF data bits.

At the decoder side, the dechirp [100, 241] is the standard decoding process in LoRa. First, a received chirp symbol is multiplied with a base down-chirp (i.e., the conjugate of a base up-chirp). The Fast Fourier Transform (FFT) is then used to focus on the energy of the chirp symbol at a single tone, which corresponds to the initial frequency offset of the chirp symbol, on the spectrum [238, 241]. For example, in Figure 6.2(b), the blue dashed arrow indicates the energy peak by applying FFT on the encoded chirp symbol shown in Figure 6.2(a). Then, we find the frequency bin where the highest spectral energy peak appears to determine the initial frequency offset f_0 . Once we obtain the initial frequency offset, data bits can be decoded. On the other hand, to decode a received chirp symbol successfully, the energy peak of the received chirp symbol should be higher than the highest energy peak derived from noises on the spectrum. For example, in Figure 6.2(b), the blue dashed arrow is higher than the grey arrow, indicating the noise peak, to guarantee f_0 can be found correctly.

6.2.2 SF-configured LoRa Performance

We can see that the standard LoRa encoder is purely controlled by SF configuration. We have six SF configurations, ranging from 7 to 12, on COTS LoRa nodes [134, 227]. In a LoRa packet, all chirp symbols employ the same SF configuration for simplicity. As such, LoRa's performance highly depends on SF configuration. We analyze the influence of SF configurations for SNR tolerance and data rate as examples.

SNR Tolerance: SNR tolerance indicates the SNR threshold, above which a chirp symbol can be successfully decoded. SNR tolerance determines the capability of network coverage. As the on-air time of a chirp symbol increases, more energy can be combined to form a higher energy peak on

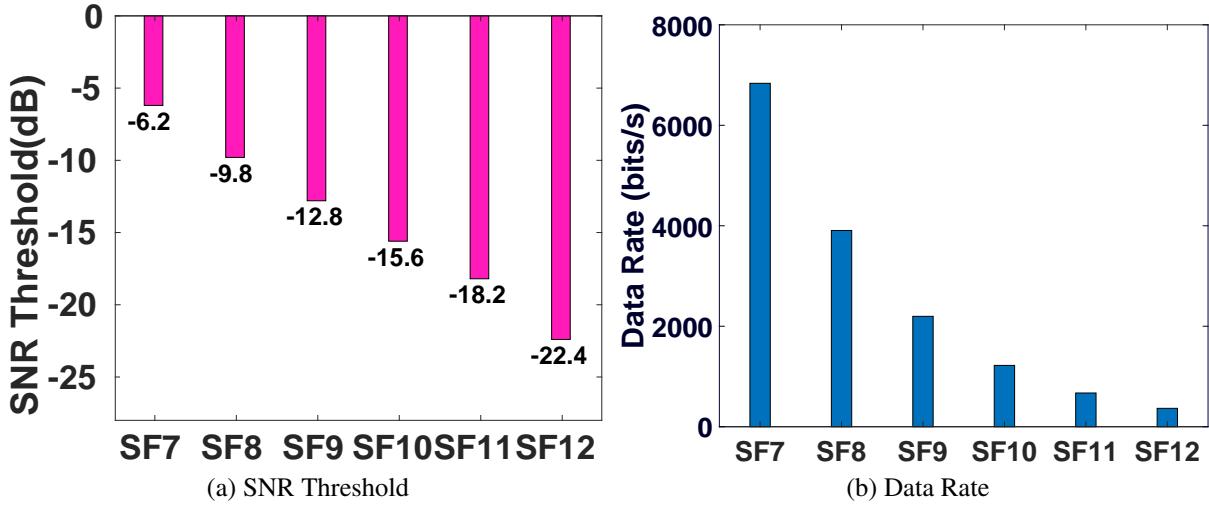


Figure 6.3 The influence of SF configurations on (a) SNR tolerance and (b) data rate.

the spectrum, making it easy to detect the energy peak under strong noises. Thus, a chirp symbol with larger SFs can achieve better SNR tolerance. We set BW as 125 kHz and empirically measure the required SNR of the dechirp under different SFs with a synthesis dataset [100, 154]. Figure 6.3a shows the SNR threshold of different SF configurations, taking 1% symbol error rate (SER) as the criteria of successful decoding. The minimum SNR threshold reaches -22.4 dB.

Data Rate: Data rate indicates how many data bits can be successfully transmitted per second without collision. Given a certain SNR level, a higher data rate leads to higher energy efficiency. Given the settings of SF and BW , a chirp symbol encodes SF data bits, and its on-air time is $\frac{2^{SF}}{BW}$. Thus, the data rate is $\frac{SF \cdot BW}{2^{SF}}$. A chirp symbol with a larger SF has a longer on-air time, lowering its data rate. When BW is set as 125 kHz, the data rate under different SFs is shown in Figure 6.3b. We can see that the data rate is reduced by 42.9%-45.7% with an SF increment. As a tradeoff, Figure 6.3a shows that increasing SF by one can achieve a 2.6-4.2 dB gain of the SNR threshold.

6.2.3 Motivation

The purely SF-controlled LoRa encoder is simple, providing SNR tolerance as low as -22.4 dB and six data rate options under six SNR thresholds. However, with SF-12 configuration, the LoSee measurement study [95] has shown that although the longest communication range can reach 3.2 km - 3.5 km, a gateway can only cover about an irregular 11 km^2 - 12 km^2 area in an urban environment,

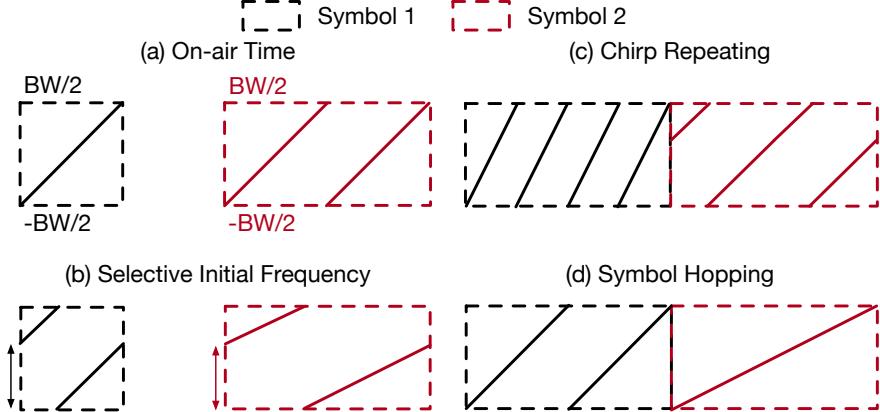


Figure 6.4 The illustration of the four chirp features in ChirpTransformer.

which is far from needed to achieve ubiquitous wide-area coverage. On the other hand, in those environments with dynamic link budgets [230, 231], only six SF options are too coarse-grained to achieve sustainable energy efficiency. This motivates us to rethink the LoRa encoder design, particularly on COTS LoRa nodes, for inherently supporting various LoRaWAN applications.

6.3 Design of ChirpTransformer

6.3.1 Chirp Feature Design

Figure 6.4 illustrates the four chirp features, representing the information from four domains, that are used to define a symbol in ChirpTransformer.

First, we define a time-domain feature, on-air time, which defines the propagation time of a symbol. A symbol contains one or multiple chirps, and the total propagation time is its on-air time. As shown in Figure 6.4(a), Symbol 1 contains one base up-chirp, and Symbol 2 has longer on-air time with two base up-chirps.

Second, we select a set of frequencies in the range $[0, BW]$ to define the available initial frequency offsets. In Figure 6.4(b), an initial frequency offset indicates the shape of a chirp compared to the base up-chirp. Therefore, the selective initial frequencies are frequency-domain features to determine the shapes of those chirps used to encode data.

Third, given the on-air time of a symbol, we design an intra-symbol chirp pattern, called *Chirp Repeating*, to depict the repeated identical chirps in a symbol. As shown in Figure 6.4(c), Symbol 1 consists of four repetitive base up-chirps, and Symbol 2 contains two repetitive chirps with the

same initial frequency offsets. The chirp repetition is a time-domain pattern, while the same initial frequency offset of those chirps reflects the frequency characteristic.

Lastly, given a fixed on-air time for all symbols in a packet, we design an inter-symbol chirp operation, called *Symbol Hopping*, to create a new pattern domain for encoding. The basic idea is to hop different chirp repeating patterns among the symbols in a packet instead of using the same configuration. As shown in Figure 6.4(d), Assume Symbol 1 and Symbol 2 are in the same packet. Symbol 1 has two repetitive base up-chirps, but Symbol 2 revises the chirp repeating pattern with only one base up-chirp. Based on the four chirp features, we abstract four parameters to configure encoders versatiley: 1) on-air time (OT), 2) available initial frequency offsets (IFO), 3) chirp repeating times (CRT) in a symbol, and 4) the number of available chirp patterns in symbol hopping (SH). ChirpTransformer adjusts the four-configuration-tuple (OT, IFO, CRT, SH) to cope with a specific performance demand.

6.3.2 Chirp Feature Configuration

On COTS LoRa nodes, we have six types of SF-configured chirps. Based on these chirps, we illustrate the supported configurations of our four chirp features.

OT indicates the on-air time of a symbol. On COTS LoRa nodes, the on-air time of a chirp is determined by six SFs from 7 to 12. Its on-air time is $\frac{2^{SF}}{BW}$, where BW is the bandwidth. The minimum on-air time is with SF-7. Similar to the way of using SF to determine on-air time, our OT is in the range of $[7, +\infty)$. When OT is k , the on-air time of a symbol is $\frac{2^k}{BW}$. In ChirpTransformer, the on-air time of a symbol could be extended flexibly. Since the maximum SF is 12, a symbol has to contain multiple chirps when OT is larger than 12.

IFO indicates the available initial frequency offsets for encoding. In SF-configured encoding, an SF- k chirp is used to encode k data bits. In ChirpTransformer, given OT- k chirps, the value of IFO can be 2^i , where i is in the range of $[1, k]$. Thus, the configuration (OT- k , IFO- i) means that we use OT- k chirps to encode i data bits by shrinking the available initial frequency offsets from 2^k to 2^i . For example, the configuration (OT-7, IFO-6) indicates using OT-7 chirps to encode 6 data bits. Half of the initial frequency offsets in OT-7 chirps will no longer be used for encoding. When

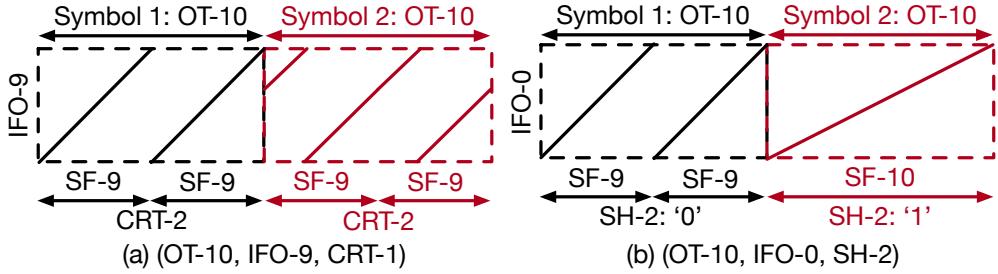


Figure 6.5 Two examples to illustrate the concepts of CRT and SH. (a) parameter tuple (OT-10, IFO-9, CRT-1); (b) parameter tuple (OT-10, IFO-0, SH-2).

OT is larger than 12, the maximum IFO can be 2^{12} on COTS LoRa nodes. In such cases, the IFO values are from IFO-1 to IFO-12.

CRT indicates the number of repetitive identical chirps in a symbol. We know that the on-air time of an OT-10 symbol equals eight SF-7 chirps, four SF-8 chirps, two SF-9 chirps, and one SF-10 chirp. Similarly, when the on-air time of a symbol is OT- k , we use CRT- j repetitive SF- $(k - j)$ chirps with the same initial frequency to fill the symbol, where j is in the range of $[\max\{k - 12, 0\}, k - 7]$. For example, as shown in Figure 6.5(a), (OT-10, IFO-9, CRT-1) means the on-air time of each symbol is OT-10. Each symbol consists of two (i.e., 2^1) identical SF-9 (i.e., $10 - 1$) chirps, which have 2^9 available initial frequency offsets for encoding.

SH represents the available chirp repeating patterns of symbol hopping. SH has three values: 0, 2, and 4. If SH is 0, the symbols in a packet follow the same pattern without symbol hopping. The data bits encoded by a symbol purely rely on the IFO settings. If SH is 2 or 4, we have 2 or 4 different patterns. We can use the different patterns to encode 1 data bit or 2 data bits. When SH is not 0, CRT will be invalid. Given the on-air time of a symbol OT- k ($k \leq 12$) and the available chirp patterns SH- t , the t chirp repeating patterns of a symbol can be represented by one SF- k base up-chirp, or two SF- $(k - 1)$ repetitive base up-chirps, ..., or 2^t SF- $(k - t)$ repetitive base up-chirps. We must keep $k - t \geq 7$. And when k is larger than 12, the SH- t chirp repeating patterns include 2^{k-12} SF-12 base up-chirps, 2^{k-11} SF-12 base up-chirps, ..., or 2^{k-13+t} SF- $(13 - t)$ base up-chirps. For example, as shown in Figure 6.5(b), (OT-10, IFO-0, SH-2) means the on-air time of a symbol is OT-10. IFO-0 indicates we do not use initial frequency offsets to encode data. In this case, with SH-2, we have two symbols. One is two repetitive SF-9 base up-chirps, indicating bit ‘0’. The other

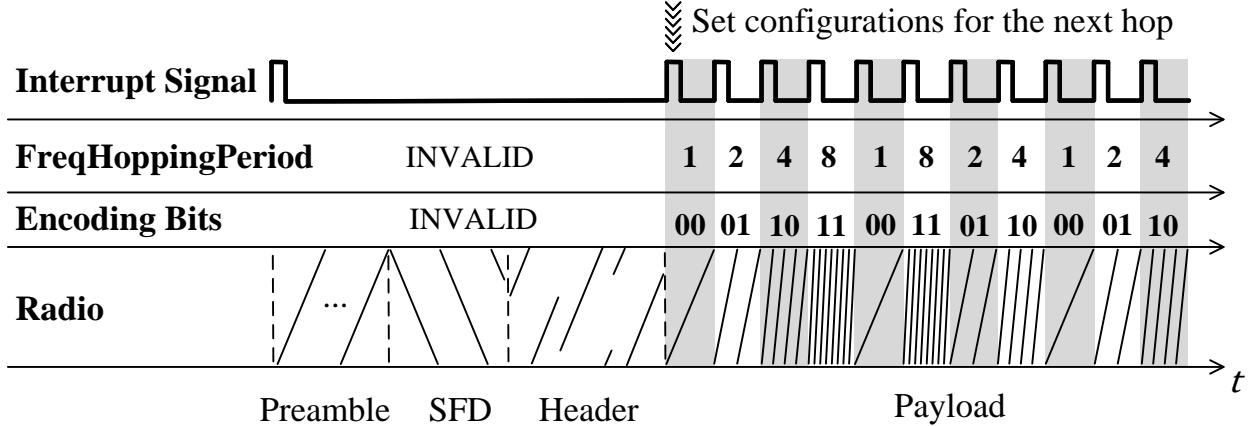


Figure 6.6 Timeline of the symbol hopping feature, leveraging a hardware interrupt.

is one SF-10 base up-chirp, indicating bit ‘1’.

6.3.3 COTS Feature Implementation

A ChirpTransformer packet consists of four parts: preamble, start frame delimiter (SFD), header, and payload. The format of the preamble, SFD, and header are the same as the standard LoRa. We put the values of (OT, IFO, CRT, SH) in the header. The symbols in the payload are encoded by the encoding method determined by (OT, IFO, CRT, SH). By default, COTS LoRa nodes support SF-configured packets in which all chirp symbols follow the same SF configuration. However, if symbol hopping is enabled, a ChirpTransformer packet consists of symbols with different SF chirps. Thus, without symbol hopping, we design a symbol converter to translate the payload of a ChirpTransformer packet to SF-configured chirp symbols. On the other hand, we leverage a hardware interrupt to implement symbol hopping.

Given OT- k , IFO- i , and CRT- j configurations, the symbol converter includes three steps. In the first step, it determines the corresponding SF configuration. If only one chirp exists in a symbol, the SF configuration corresponds to OT- k . Otherwise, the symbol contains a chirp repeating pattern CRT- t . Then, the SF configuration is $k - t$ corresponding to the SF of each repetitive chirp. In step two, we calculate the number of SF chirps to form a symbol. If CRT is not applied, only one SF chirp exists. For CRT- j , the number corresponds to 2^j . In the last step, we set the initial frequency offset for each SF chirp. Since IFO- i must be a subset of IFO-SF, we directly assign the initial frequency offset of each chirp in a ChirpTransformer symbol to the corresponding chirp generated

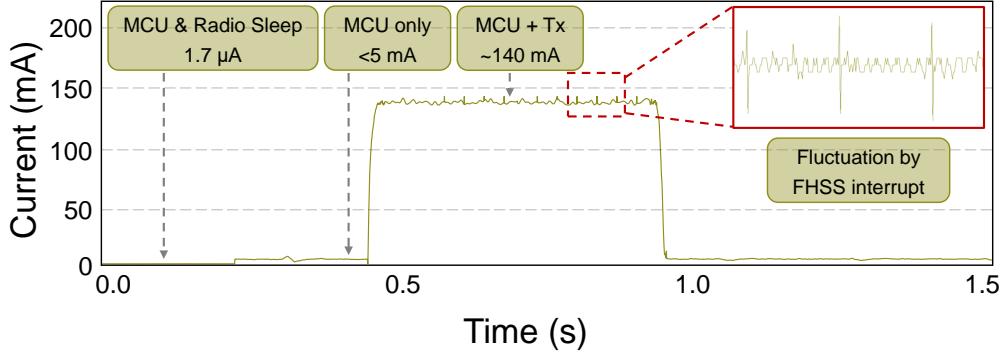


Figure 6.7 Current profile for a ChirpTransformer transmission with symbol hopping.

in step two. In this way, a ChirpTransformer symbol can be translated to a series of SF-configured chirp symbols.

We implement the symbol hopping feature based on the frequency hopping capability of COTS LoRa nodes [227]. The LoRa standard requires LoRa nodes to support frequency hopping, enabling long-duration packet transmission without violating the maximum permissible channel dwell time. The key principle behind the frequency hopping scheme is hardware interrupt, named ChangeChannelFhss, that enables LoRa nodes to select and switch to a new channel during packet transmission. After a predetermined hopping period, the transmitter and receiver change to the next channel in a predefined list of hopping frequencies to continue transmission and reception of the next portion of the packet. Our key observation is that a LoRa node can modify not just the channel but all configurations every time it triggers the ChangeChannelFhss interrupt. Thus, we can implement symbol hopping on a COTS LoRa node without adding extra hardware by making the node periodically trigger the ChangeChannelFhss interrupt and change its SF configuration during packet transmission. Moreover, we keep the transmission on a single channel by setting the list of hopping frequencies as identical frequencies.

For example, we use (OT, IFO-0, SH-4) to encode 2-bit data with four different chirp repeating patterns. Figure 6.6 shows the timeline of the encoder for transmitting a packet. The transmission starts with the preamble, SFD, and header. At the beginning of each transmission, an interrupt signal ChangeChannelFhss is generated, where the interrupt handler programs the frequency, SF, and hopping period for the first hop of the payload. The interrupt signal is cleared after all config-

urations have been settled. Then, during payload transmission, the transmitter periodically triggers the ChangeChannelFhss interrupt to modify the SF configuration for ChirpTransformer data modulation. We use our symbol converter to determine the SF for each chirp repeating pattern. The time that each hop of transmission will dwell is determined by FreqHoppingPeriod, which is an integer multiple of symbol periods. As illustrated in data modulation, the periods of all symbols should be identical to the OT configuration. Thus, we determine the FreqHoppingPeriod based on the SF configuration of chirps in the corresponding hop, i.e., $FreqHoppingPeriod = 2^{OT-SF}$. The new configurations are programmed within the current hopping period to ensure it has been set when the next hop begins. The interrupt computation is much shorter than the symbol on-air time, causing no extra latency.

6.3.4 Energy Profiling

To demonstrate the energy overhead brought by the hardware interrupt processing, we estimate the energy profile of a ChirpTransformer transmission with symbol hopping using the Monsoon HV Power Monitor [249]. The node is powered by 3.6 V. Figure 6.7 shows the current profile of the transmission, including the instant current for all radio access phases. The LoRa node stays in sleep mode when it does not transmit data. The radio transmission consumes the highest amount of energy by a large margin. The power consumption of the MCU is much less than that of the radio circuit. Thus, the ChangeChannelFhss interrupt processed by the MCU only introduces a small current fluctuation during the packet transmission, which has a neglectable impact on the whole energy consumption of the LoRa node.

6.4 Case I: Network Coverage

In the first case study, we focus on reliable weak signal decoding to extend LoRaWAN coverage [100]. We are seeking a design to tolerate the SNR lower than the -22.4 dB SNR threshold of the standard LoRa (§ 6.2.2). With ChirpTransformer, the key idea is to maximize the feature distance among different symbols during encoding beyond SF-12 with SH configurations.

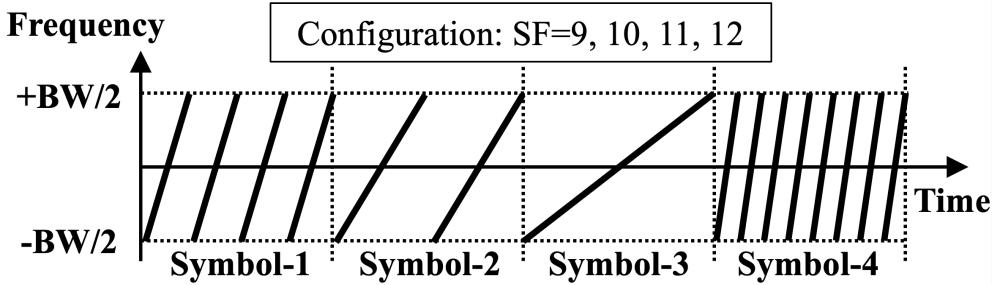


Figure 6.8 The illustration of symbol hopping encoding, using four chirp repeating patterns (e.g., SF-9 – SF-12) to represent 2-bit data.

6.4.1 SH Encoder-Decoder Co-design

Encoding Method: Our design utilizes the chirp pattern domain created by symbol hopping and prohibits using initial frequency offsets (i.e., IFO-0) for data encoding. We use four chirp patterns to represent each 2-bit data (i.e., SH-4). To obtain four chirp repeating patterns, the minimum OT configuration is OT-10 so that we have the SF-7, SF-8, SF-9, and SF-10 chirps to construct four different patterns. When OT increases, the SNR tolerance will be enhanced while the data rate reduces. Figure 6.8 shows an example of (OT-12, IFO-0, SH-4), which uses four chirp repeating patterns of SF-9 – SF-12 to define 4 symbols. We use SH-[9-12], SH-[8-11], and SH-[7-10] to represent (OT-12, IFO-0, SH-4), (OT-11, IFO-0, SH-4), and (OT-10, IFO-0, SH-4), separately.

Neural-enhanced Decoding Method: Our decoding problem is to extract a symbol’s chirp patterns (i.e., 4-class classification). Inspired by the neural-enhanced decoder in NELoRa [100], Chirp-Transformer also aims to decode the symbols in a neural-enhanced manner to maximize the SNR gain from both the encoder and decoder side. Like NELoRa, our neural-enhanced decoder converts symbols into time-frequency spectrograms as input feature maps. We simplify NELoRa’s architecture by using a lightweight first Conv2d module with fewer filters and replacing the LSTM layer with a bidirectional GRU layer, which is more computation-efficient for temporal feature extraction. Figure 6.9 shows the concise network structure, which consists of seven modules in total. Specifically, the first four modules aim to generate a filter mask to be multiplied with the input spectrogram. Then, we feed the masked spectrogram into a three-module classifier for chirp pattern recognition.

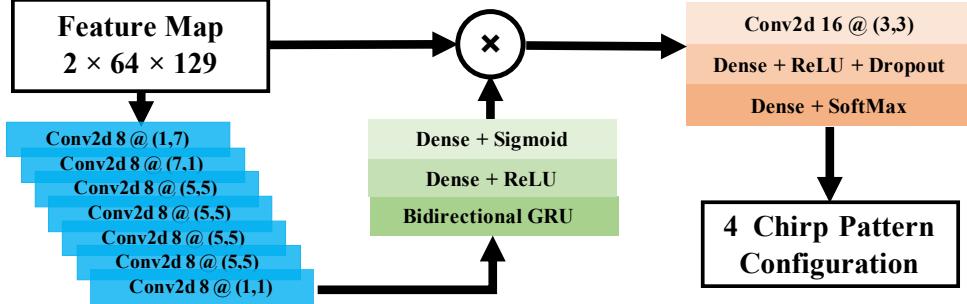


Figure 6.9 The structure of the neural-enhanced decoder.

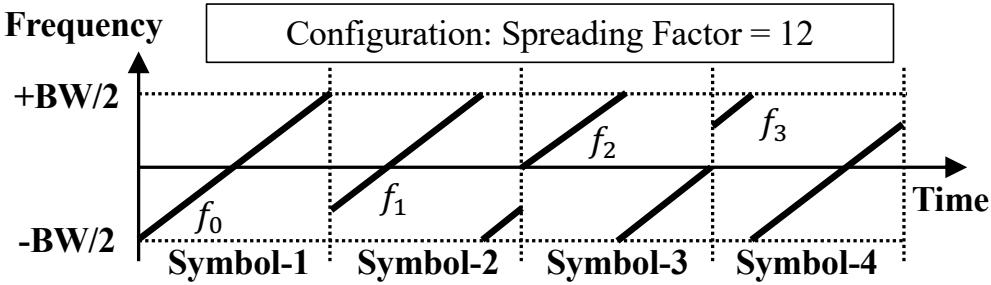


Figure 6.10 The illustration of the IFO-2 based LoRa encoder, using four initial frequency offsets (e.g., f_0, f_1, f_2, f_3) to represent 2-bit data.

6.4.2 IFO-2 based Encoder-Decoder Co-design

Encoding Method: With the similar design principle of our symbol hopping encoder, we can set the available initial frequency offsets as 4 (i.e., IFO-2) to encode 2-bit data. Specifically, as shown in Figure 6.10, we select four SF-12 chirps with deterministic initial frequency offsets (e.g., f_0, f_1, f_2, f_3) to encode 2-bit data. We set the initial frequency offsets as $-\frac{BW}{2}, -\frac{BW}{4}, 0$ and $\frac{BW}{4}$ to distinguish these symbols as much as possible in the feature space. To keep a similar data rate to the symbol hopping encoding, IFO based encoding has three configurations, namely IFO-2-10, IFO-2-11, and IFO-2-12, which use SF-10, SF-11, and SF-12 chirps to form symbols matching SH-[7-10], SH-[8-11], and SH-[9-12] symbols.

Neural-enhanced Decoding Method: The decoding problem becomes classifying the initial frequency offsets, which is still a four-class classification. We adopt the same deep neural network (DNN) structure in Figure 6.9 to classify IFO-2 based symbols for decoding.

6.4.3 Weak Signal Packet Detection

After a LoRa node transmits a packet through the air, the packet should be reliably detected at the decoder side. Then, its payload is divided into multiple symbols and fed into the DNN-based decoder for chirp pattern recognition. Since the preamble part of a ChirpTransformer packet remains the same as the standard LoRa encoder, we need a reliable way to detect the LoRa preamble under ultra-low SNR. Given a period of received signals, we identify whether they contain a LoRa preamble, which consists of a series of base up-chirps, as an indicator of whether a packet is coming. We divide the received signals into N symbol-length signal segments, where N is the number of base up-chirp symbols in a LoRa preamble. Then, we combine the N segments as a superposed signal segment. All the base up-chirp symbols are constructively superposed if a preamble is in the received signals [100]. Considering the random initial phase of each base up-chirp symbol due to carrier frequency offset (CFO) and sampling frequency offsets (SFO), along with the quadratic distribution of the phase shift related to frequency bias and chirp symbol index, we heuristically determine the optimal phase compensation [100, 238]. After phase compensation, we coherently sum up the N symbol-length signal segments of the received signals. Then, we use a standard base up-chirp symbol to calculate its cross-correlation with the superposed signal segment. We treat the detection of a significant correlation peak as a sign of a successful preamble detection. The threshold for peak detection is based on channel estimation, which is six standard deviations of the mean noise correlation. The index of the correlation peak also indicates the boundary of the received LoRa preamble symbols. Therefore, we can align the timing of the demodulation window and extract the aligned symbols from the payload part using the detected preamble.

6.4.4 Implementation

We have implemented ChirpTransformer on COTS LoRa nodes and use an SDR as a gateway, shown at the bottom of Figure 6.11. Specifically, the USRP N210 SDR platform captures over-the-air LoRa signals by operating on a UBX daughter board with a sampling rate of 1 MS/s for both ChirpTransformer and standard LoRa, which is a widely used sampling rate setting [40, 100, 171, 238]. Since the maximum BW is 500 kHz in LoRa, COTS LoRa gateways should support a



Figure 6.11 USRP N210 based gateway and COTS LoRa nodes deployed in a campus environment.

sampling rate of at least 1 MS/s according to the Nyquist–Shannon sampling theorem. The captured signal samples are then delivered to a back-end host, preprocessed, and demodulated by the decoder algorithms. The COTS SX1278 [227] based LoRa nodes transmit ChirpTransformer packets with random payloads. They are deployed in a campus environment, including both indoor and outdoor scenarios, as shown at the top of Figure 6.11. ChirpTransformer implements the DNN model on a Raspberry PI 4 [250] with an average inference time of 0.26 s over 100 runs and a memory requirement of 17.19 MB. In comparison, the compressed NELoRa model under the SF-10 setting reaches 152.9 MB in memory and 0.97 s for inference. This makes NELoRa 3.7× slower and 8.9× larger in memory than ChirpTransformer. This can further help reduce the overhead of the DNN model deployment at the gateway.

6.4.5 Baseline Methods and Metrics

Besides the standard LoRa, which uses SF-12 chirps for encoding and the dechirp for decoding, we choose Ostinato [251], a state-of-the-art encoder-decoder co-design for weak signal decoding beyond SF-12, as one of our baselines.

Ostinato: repetitive SF-12 chirps based encoder + chirp coherent-combining based decoder. Ostinato [251] can be regarded as a special case of ChirpTransformer’s chirp repeating encoding. Specifically, Ostinato uses repeated SF-12 chirps, having the identical initial frequency offset, to encode the same data as a single SF-12 chirp does. Then, the decoder coherently combines the

multiple SF-12 chirps to obtain a more obvious energy peak during dechirp than any individual SF-12 chirp. We implement the encoder by using our chirp repeating feature to generate the repeated SF-12 chirps and adopt the phase calibration method to fine-tune the chirp coherent combining for reliable decoding.

We use two metrics to indicate the performance of weak signal decoding.

- *Symbol Error Rate (SER)* is widely used to demonstrate the channel noise resilience of a physical layer design given different SNR [100, 238, 239]. A low SER is desirable.
- *SNR Threshold* is the lowest SNR that a physical layer design can keep the SER under a predetermined value. By default, we set the predetermined SER as 1%.

6.4.6 Communication Reliability Evaluation

Setup: We evaluate the SNR threshold of ChirpTransformer under SH-[7-10], SH-[8-11], SH-[9-12], IFO-2-10, IFO-2-11, and IFO-2-12 configurations. We further evaluate the SNR threshold of **Ostinato** with three comparable configurations, using single, two, and four repetitive SF-12 chirps to compose a symbol, indicated as **Ostinato-1**, **Ostinato-2**, and **Ostinato-4**. In the experiments, we collect high-SNR symbols on our campus testbed. Then, we inject random noises to generate weak signals with arbitrary SNR levels using the same method of NELoRa [100]. We use the synthesis symbols to calculate the SNR threshold, indicating communication reliability. We generate the same number of synthesis symbols to train the DNN decoder for our symbol hopping and IFO-2-based encoders.

Results: The results are shown in Figure 6.12.

General SNR Gain: The SNR thresholds of symbol hopping encoder are -24.5 dB, -27.4 dB and -28.8 dB using SH-[7-10], SH-[8-11], and SH-[9-12] encoding settings, separately. Compared with -22.4 dB SNR threshold of the standard LoRa under SF-12, we achieve the maximum 6.4 dB SNR gain, which is much larger than the approximate 2 dB SNR gain of NELoRa [100]. This indicates that ChirpTransformer significantly improves the communication reliability of LoRa. LoSee [95], an urban LoRa measurement study, introduces a link model to predict the packet delivery ratio (PDR) within a square coverage area based on link SNR. The coverage area is defined where the

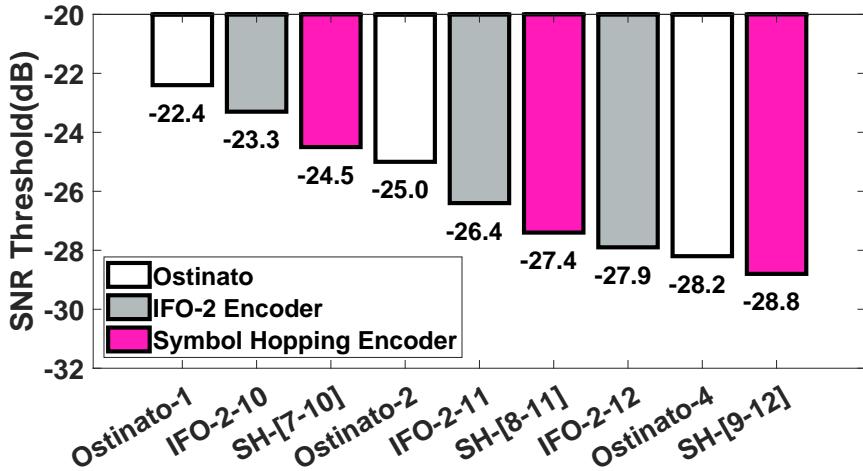


Figure 6.12 The comparison of SNR threshold between ChirpTransformer and **Ostinato**.

overall PDR value exceeds 70%. With the link model of LoSee and 6.4 dB SNR gain, ChirpTransformer can achieve approximately 2.38 \times coverage area compared with the standard LoRa in the urban environment.

SNR Gain from symbol hopping: the IFO-2 based encoder adopts the same DNN-based decoder with the symbol hopping encoder but encodes data with the same type of chirps. We can see that the SNR thresholds of the IFO-2 based encoder are 1.2 dB, 1.0 dB, and 0.9 dB higher than the symbol hopping encoder with the identical on-air time settings of OT-10, OT-11, and OT-12. On average, the symbol hopping encoder can tolerate 1.03 dB lower SNR than the IFO-2 based encoder. This verifies our symbol hopping feature provides a larger feature space than IFO-2 feature, which makes the different symbols can be easily distinguished with a DNN model. The SNR thresholds of SH-[7,10], SH-[8,11], and SH-[9-12] are 2.1 dB, 2.4 dB, and 0.6 dB lower than **Ostinato-1**, **Ostinato-2**, and **Ostinato-4**, separately. On average, ChirpTransformer achieves 1.7 dB SNR gain compared to **Ostinato**. This indicates our symbol hopping feature outperforms the chirp repeating pattern in **Ostinato** with complicated phase calibration for chirp coherent-combining.

6.4.7 Campus-scale Evaluation

Setup: SH is adopted by those LoRa nodes experiencing weak links to improve network coverage. The SH encoder extends its range and reduces the SNR threshold by lowering the data rate and neural-enhanced decoder. We conduct campus-scale experiments to verify the gain of SH encoder-

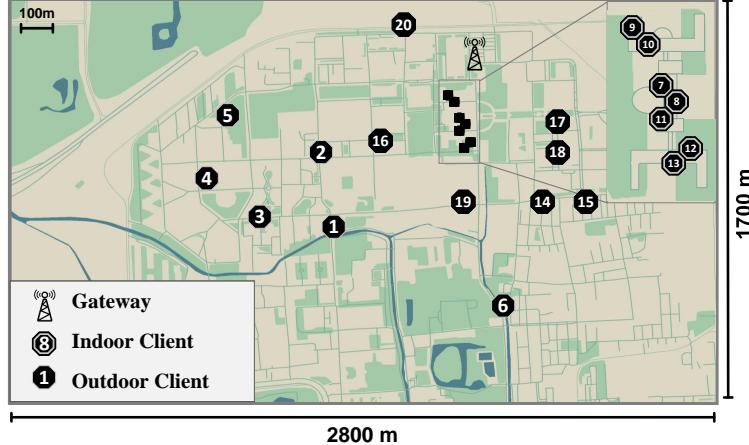


Figure 6.13 The campus-scale testbed with 20 LoRa nodes at 20 indoor and outdoor locations.

decoder co-design with extremely low SNR conditions. Figure 6.13 illustrates the deployment of our outdoor testbed at a campus ($2800\text{m} \times 1700\text{m}$). We randomly deploy 20 COTS LoRa nodes at 20 different NLOS positions covering indoor and outdoor scenarios. For indoor nodes, concrete walls are the main obstacles. Buildings and trees are the obstacles in the outdoor environment. For each LoRa node at a position, we collect tens of packets, and each contains 40 payload symbols. We repeat the data collection with three encoding methods of the standard LoRa, ChirpTransformer using SH-[9,12] configuration, and Ostinato using **Ostinato-4** configuration, respectively. We first evaluate the preamble detection accuracy of our design compared to the standard LoRa. In addition, for those detected packets, we compare the decoding SER between ChirpTransformer and Ostinato. For our SH-[9,12], we apply the DNN-based decoder pre-trained with our synthesis dataset collected in § 6.4.6 on the weak signal packets.

Results for packet detection: To understand whether a ChirpTransformer packet can be successfully detected at extremely low SNR conditions, we evaluate ChirpTransformer’s performance on packet detection for all 20 positions. Figure 6.14a shows the comparison of the packet loss rate between ChirpTransformer and the standard LoRa. Figure 6.14b shows the CDFs of the packet loss rate for ChirpTransformer and the standard LoRa. We can see that the standard LoRa suffers a high packet loss rate at all positions, where more than 90% of packets are undetected at 10 out of 20 positions. In comparison, ChirpTransformer achieves a lower than 6% packet loss rate at all positions. This is because the standard LoRa detects LoRa pREAMbles by searching for continu-

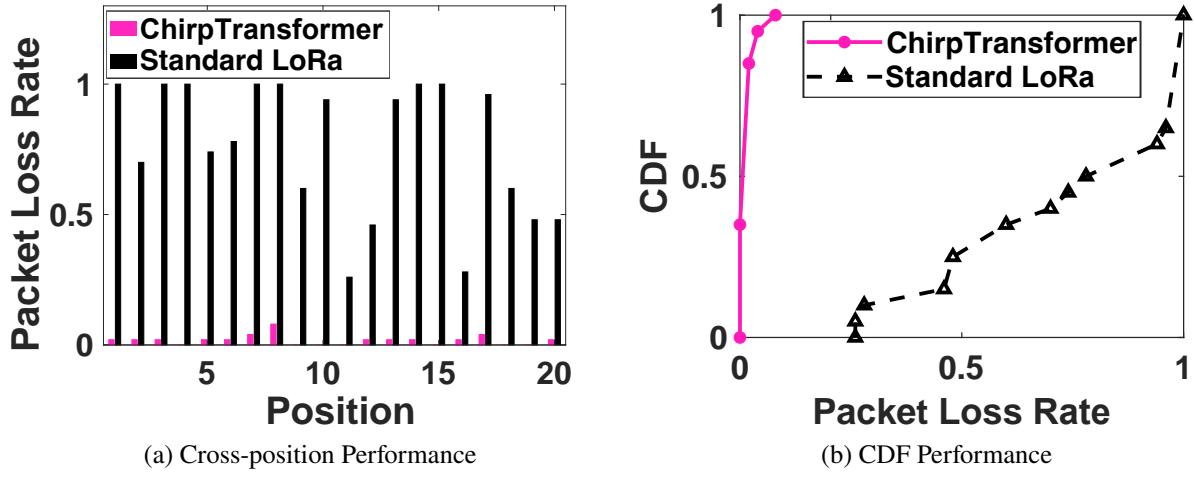


Figure 6.14 The packet detection accuracy on our campus-scale testbed with extremely low SNR.

ous identical frequency-domain energy peaks by applying the dechirp to the continuously received signals. Thus, it requires the energy peak of each chirp symbol in the preamble to be detectable. However, the real SNR at most of the positions is much lower than the SF-12 SNR threshold (e.g., -22.4 dB), leading to undetectable energy peaks. After mitigating the CFO and SFO in a preamble, ChirpTransformer concentrates the energy of all preamble chirp symbols to detect a LoRa packet. The results verify that ChirpTransformer can reliably detect any LoRa packets at the extremely low SNR.

Results for decoding SER: We utilize our preamble detection design to detect Ostinato packets. Then, we compare ChirpTransformer with Ostinato by computing the decoding SER for the detected symbols at all 20 positions. As shown in Figure 6.15b, we can see the median SER of Ostinato is about 79.25%. In comparison, ChirpTransformer can achieve a much lower median SER of 40.54%, indicating the boundary of the communication range will be enlarged greatly. The reason is that Ostinato suffers from severe noises in the wild environment, leading to the signals not being coherently combined. At the same time, our neural-enhanced decoder can tolerate it by exploring multi-dimensional features. The specific SER across 20 positions is shown in Figure 6.15a. Our SER is lower than Ostinato at all 20 positions. In addition, we observe that ChirpTransformer has a higher SER in some positions, such as positions 4, 7, 8, 14, and 15, than others. The reason is that some coexisting wireless interference also brings new noise patterns that the pre-trained DNN

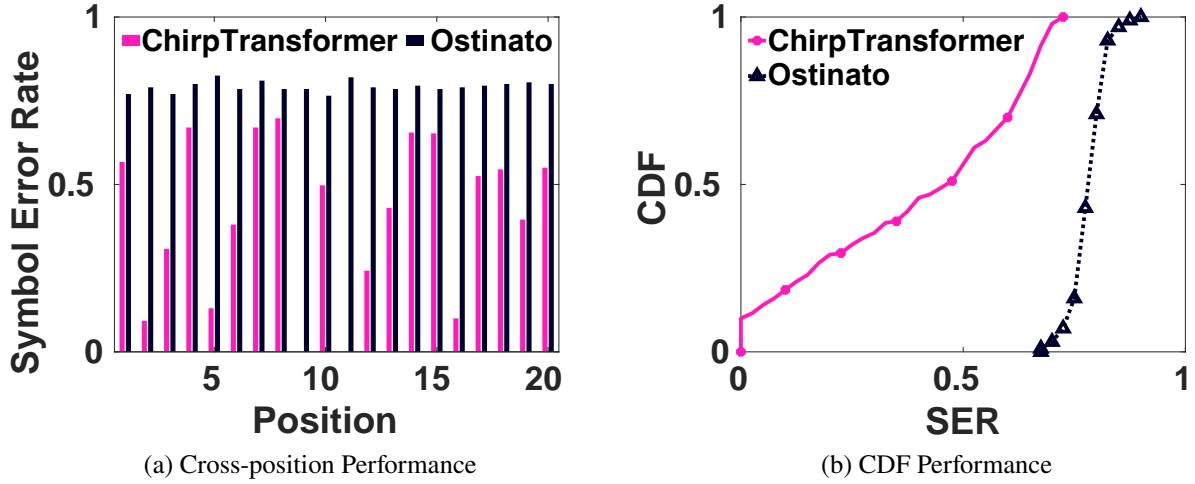


Figure 6.15 The comparison of communication reliability on our campus-scale testbed.

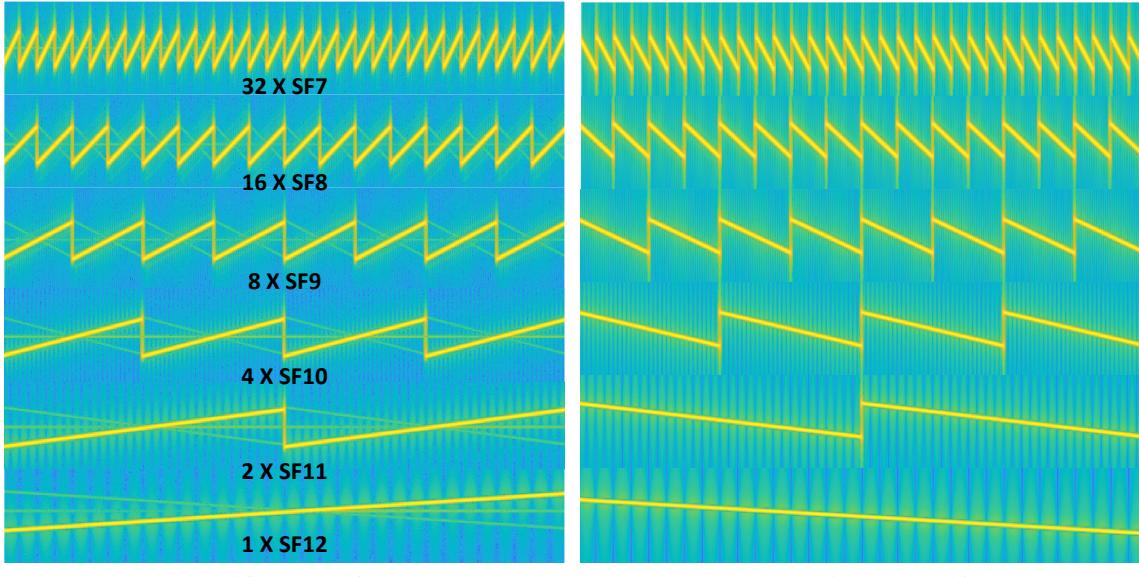
model does not see, degrading the SER. We can involve an online fine-tuning process to deal with those new noise patterns [100].

6.5 Case II: Network Throughput

In our second case study, we target to enable efficient concurrent transmission, allowing multiple LoRa nodes to transmit their packet simultaneously to enhance the network throughput and scalability [171, 229, 241]. Specifically, in standard LoRa, concurrent transmission is infeasible for multiple packets with the same SF configuration [171]. The six SFs create six quasi-orthogonal logic channels to enable concurrent transmission [242]. However, since the SNR threshold increases when SF decreases (§ 6.2.2), we cannot assign arbitrary SF to a LoRa node without noticing its SNR requirement, thus degrading the efficiency of concurrent efficiency. With ChirpTransformer, given the same SNR tolerance, we aim to design multiple orthogonal logic channels to achieve efficient concurrent transmission. Our key idea is to use different repeating chirp patterns with CRT configurations to create orthogonal logic channels.

6.5.1 CRT Encoder-Decoder Co-design

Encoding Method: For an orthogonal logic channel, the encoding method can be indicated as $(OT-k, IFO-i, CRT-(k - i), SH-0)$. $OT-k$ is the pre-configured on-air time. $IFO-i$ indicates that we use $SF-i$ chirps to construct the chirp repeating pattern and encode i data bits. The chirp



(a) Chirp Repeating

(b) Template Downchirp

Figure 6.16 Six orthogonal logic channels with OT-12.

repeating times are 2^{k-i} . i is in the range of $[7, \max\{k, 12\}]$, indicating k must be larger than i to create a diverse chirp repeating pattern, and i is not less than 7, the minimum SF chirp we have. Therefore, given the pre-configured on-air time OT- k , we have $k - 7$ orthogonal logic channels if $k \leq 12$. Otherwise, we have six orthogonal logic channels using SF-7 to SF-12 chirp repeating patterns. The larger the k is, the more orthogonal logic channels we can have. Figure 6.16a shows an example of the on-air time OT-12. We have six orthogonal logic channels with 32 SF-7 chirp repeating (OT-12, IFO-7, CRT-5, SH-0), 16 SF-8 chirp repeating (OT-12, IFO-8, CRT-4, SH-0), 8 SF-9 chirp repeating (OT-12, IFO-9, CRT-3, SH-0), 4 SF-10 chirp repeating (OT-12, IFO-10, CRT-2, SH-0), 2 SF-11 chirp repeating (OT-12, IFO-11, CRT-1, SH-0), and one SF-12 chirp (OT-12, IFO-12, CRT-0, SH-0).

Template Down-chirp based Decoding: Given the $(OT-k, IFO-i, CRT-(k-i), SH-0)$, we use a template down-chirp to find the energy peak in the FFT spectrum. The template down-chirp consists of 2^{k-i} SF- i base down-chirps. Figure 6.16b shows the template down-chirps for our six logic channels with OT-12. The basic observation is illustrated in Figure 6.17. Given an $(OT-12, IFO-11, CRT-1, SH-0)$ symbol with two SF-11 base up-chirps, we multiply two matched base down-chirps (called *template*) and coherently combine the derived signals in the two SF-11 windows

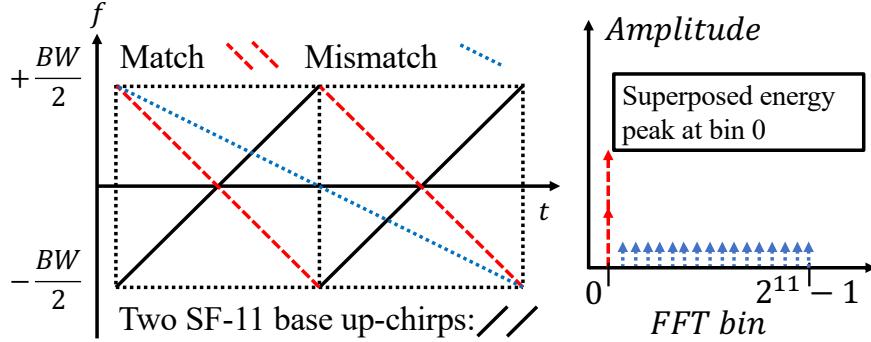


Figure 6.17 After multiplying the matched template down-chirp, a superposed energy peak appears.

together. After applying FFT, a superposed energy peak appears at bin 0, which is equivalent to the energy peak of an SF-12 base up-chirp due to the same on-air, thus supporting the same SNR tolerance ability. When using a mismatched template like an SF-12 base down-chirp, the energy is dispersed at all bins.

With this observation, the decoder first initializes $\max\{k - 7, 6\}$ down-chirp templates with different SF configurations. Then, for each template, due to the CFO and SFO among the repetitive chirps in the symbol, we compensate various phase offsets calculated by the CFO and SFO estimated during packet detection for alleviating the random initial phase problem [100], then apply the template to obtain the corresponding spectrum with FFT. With the phase compensation, we can make the obtained energy peak at frequency bin f_0 as accurate as possible. Suppose the derived energy peak with a template is larger than the current maximum energy peak. In that case, we update the expected initial frequency offset f_0 and IFO configuration of the symbol and the latest maximum energy peak. When traversing all templates, the maximum energy peak is selected, and the f_0 and IFO configuration are converted to data bits accordingly.

6.5.2 Implementation and Baseline

We implement the encoder and decoder on our campus testbed. Besides the standard LoRa, we select CurvingLoRa [171] as the state-of-the-art baseline from the perspective of LoRa encoder design. CurvingLoRa utilizes the orthogonal coding space created by different non-linear chirps to enable concurrent LoRa transmissions. The non-linear chirps have similar SNR tolerance compared to the linear chirp with the same on-air time. Moreover, besides SER, we use another metric

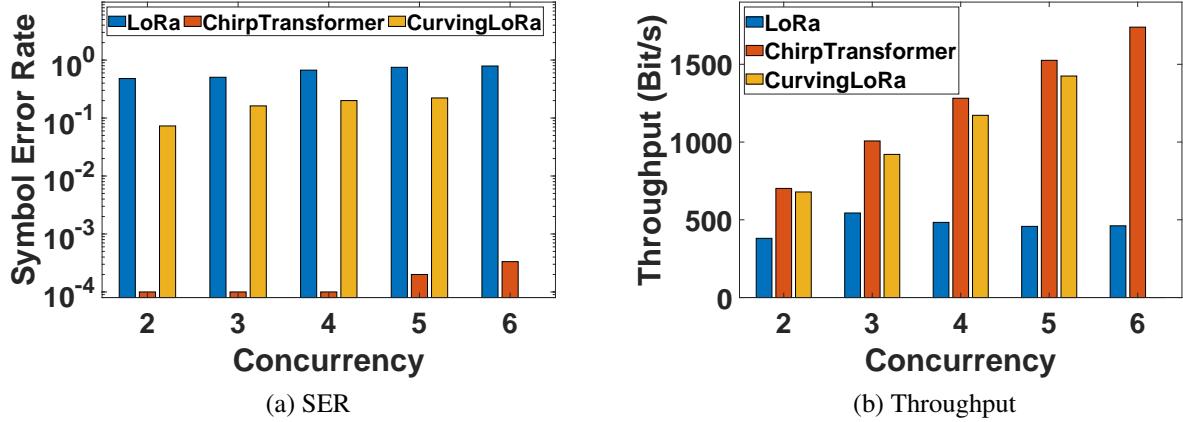


Figure 6.18 Collision performance of ChirpTransformer, CurvingLoRa and LoRa.

Throughput, which indicates the amount of data transmitted over a network in a unit of time.

6.5.3 Concurrent Transmission Evaluation

Setup: We collect high-SNR symbols from our campus testbed. We conduct trace-driven experiments to evaluate the SER and network throughput of ChirpTransformer, CurvingLoRa [171], and the standard LoRa during concurrent transmission. As the same operation in CurvingLoRa [171] to emulate signal collisions, we collect individual symbols in real environments separately and then add the symbols together to generate the overlapping patterns. To simulate diverse temporal patterns of symbol collisions, we randomly assign a symbol offset from [0,1] times the symbol on-air time. To evaluate the performance under near-far issues, we add a signal-to-interference ratio (SIR) ranging from -20 to 0 dB to concurrently transmitted symbols. The frequency bandwidth is 125 kHz. For the standard LoRa, we use SF-12 chirps for all concurrent transmitters. For CurvingLoRa, we adopt five types of chirps: quadratic1, quartic1, quadratic1, quartic1, and linear with on-air time OT-12. Since we assign concurrent transmitters to different orthogonal logical channels, the five logic channels only support five concurrencies at most. For ChirpTransformer, we use the six orthogonal chirp repeating logic channels (OT-12, IFO-12, CRT-0, SH-0), (OT-12, IFO-11, CRT-1, SH-0), (OT-12, IFO-10, CRT-2, SH-0), (OT-12, IFO-9, CRT-3, SH-0), (OT-12, IFO-8, CRT-4, SH-0), and (OT-12, IFO-7, CRT-5, SH-0). To calculate the average network throughput and SER. We repeat experiments for each concurrency configuration 2,000 times.

Results for Symbol Error Rate Performance: The SER for different concurrency values from 2 to 6 is shown in Figure 6.18a. 10^{-4} represents no error symbols. When concurrency is 2, LoRa fails to decode half of the symbols correctly. In contrast, the SER for CurvingLoRa is less than 10%, and there is no error symbol for ChirpTransformer. As concurrency increases, the SER for CurvingLoRa and ChirpTransformer also increases. However, CurvingLoRa exhibits significantly higher symbol decoding errors compared to ChirpTransformer. Despite a SIR range of [-20,0] dB and a concurrency of 6, ChirpTransformer can still maintain a SER less than 0.2%.

Results for Network Throughput Performance: As shown in Figure 6.18b, LoRa consistently has the lowest throughput due to its high SER, unaffected by concurrency values ranging from 2 to 6. In contrast, ChirpTransformer consistently outperforms CurvingLoRa in throughput due to our lower SER. Specifically, at a concurrency of 5, the throughput of ChirpTransformer is $3.33\times$ that of LoRa and $1.07\times$ that of CurvingLoRa, demonstrating ChirpTransformer's superior performance in concurrent transmissions.

6.6 Case III: Network Lifetime

In the standard LoRa, we configure SF to balance between SNR tolerance and data rate. After a LoRa node is deployed or its surrounding environment changes, a LoRa gateway will check the node's SNR and choose an SF configuration that can reach the highest data rate to enhance the energy efficiency while its SNR threshold is lower than the observed SNR to keep reliable packet delivery. Only six SFs are too coarse-grained to optimize network energy efficiency in complex environments [230, 231]. As such, we aim to develop fine-grained data rate adaptation with IFO configurations to achieve energy efficiency.

6.6.1 IFO based Encoder-Decoder Co-Design

Encoding Method: We adjust the available initial frequency offsets to obtain more encoding configurations. The encoding method can be indicated as $(OT-k, IFO-i, CRT-0, SH-0)$. For energy efficiency, the on-air time is not larger than that of an SF-12 chirp. Thus, we have $k \leq 12$ and $i \leq k$. Given the on-air time $OT-k$, we encode i bits of data. The data rate is $\frac{i \times BW}{2^k}$. To encode i bits data, we only need 2^i initial frequency offsets. Given the total 2^k initial frequency offsets, we

uniformly remove unused offsets and select the rest as the available initial frequency offsets for encoding. For example, for OT-12, the range of initial frequency offsets is $[0, \frac{2^{12}-1}{2^{12}}BW]$. For IFO-10, we uniformly select a quarter of these offsets as $\{0, \frac{4}{2^{12}}BW, \frac{8}{2^{12}}BW, \frac{12}{2^{12}}BW, \dots, \frac{2^{12}-4}{2^{12}}\}$. When k is from 7 to 11, i is in the range of $[\lceil \frac{k+1}{2} \rceil, k]$. When k is 12, i is in the range of [6, 12]. For example, when k is 7, the available i is 7, 6, and 5. If i is 4, the data rate is $\frac{4BW}{2^7}$ which is the same with $\frac{8BW}{2^8}$ represented by (OT-8, IFO-8, CRT-0, SH-0). In this way, we avoid the redundant data rate among different chirps with different OTs, resulting total 23 data rate configurations.

Selective Energy Peak Searching Decoding Method: According to the set of available initial frequency offsets with the configuration (OT- k , IFO- i , CRT-0, SH-0), we apply the standard dechirp for decoding and search the energy peak at specific FFT bins corresponding to the available initial frequency offsets. Since noises are random across the entire spectrum, the fewer the FFT bins we search for the energy peak, the less the energy peak of noises can influence our energy peak searching. Thus, when we limit the range of the searching space, we can lower the SNR threshold for successful decoding. Hence, we provide the tradeoff between noise tolerance and data rate.

6.6.2 Implementation and Evaluation

Setup: We collect a full set of standard LoRa chirp symbols with all available initial frequency offsets across SF-7 to SF-12 at high SNR levels (>30 dB) on our campus testbed. Then, we generate chirp symbols at multiple low SNR levels by injecting random Gaussian noise to simulate real-world scenarios [100]. We use the SNR threshold and data rate as metrics to compare the performance of different settings. The encoding configuration (OT- k , IFO- i , CRT-0, SH-0) is indicate as SF k - i . For example, SF7-6 indicates using SF-7 chirps to encode 6-bit data.

Moreover, to verify the battery life gain (BLG) of ChirpTransformer compared to other methods, we use real-world LoSee traces [95] for vehicle tracking in an urban environment. A mobile LoRa node on a car periodically travels across the roads between office and home, ranging from hundreds of meters to 3.2 kilometers in an urban area. The LoRa node transmitted packets with SF-12 configuration. With the idea of data rate adaption, we reassign the data rate according to the SNR to reduce transmission time and shorten energy consumption. One baseline method is the default

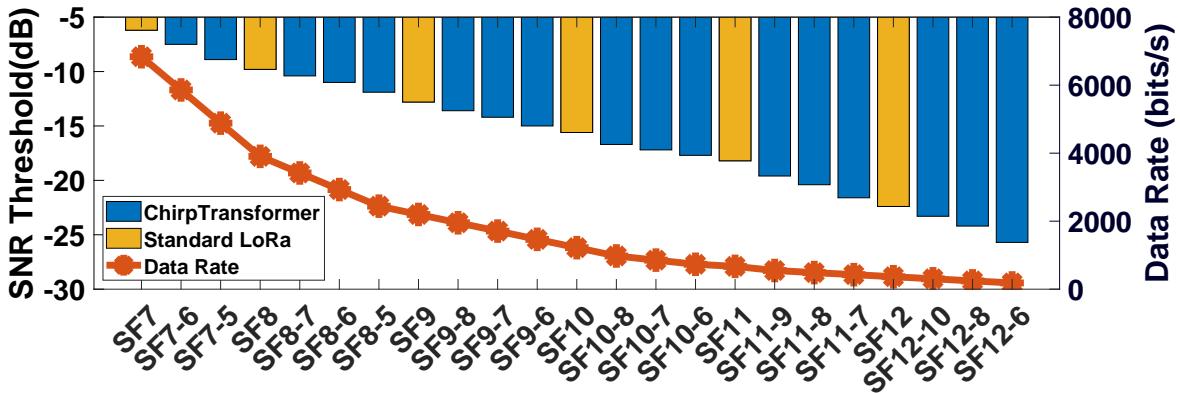


Figure 6.19 SNR threshold and data rate under various configurations for fine-grained data rate adaption.

LoRa data rate adaptation, which uses SF-7 to SF-11 configurations to replace SF-12 in some cases with better SNR conditions. However, ChirpTransformer offers a more precise data rate adaptation scheme with 23 configurations. We estimate the battery life assuming that the LoRa node is powered by two AA batteries (2200mAh) with 3.6 V voltage and sending packets of different lengths (from 10 to 30 bytes) 240 times per day, as the LoRa node would do in the morning and evening vehicle tracking. Based energy profiling on Figure 6.7, we calculate the energy consumption during two consecutive transmission beginning times by multiplying the currents across different phases with input voltages. The extra energy consumption caused by ChangeChannelFhss interruption on the node is neglectable. Then we accumulate the energy over time and calculate the battery life.

Results for SNR Threshold v.s. Data Rate: Figure 6.19 displays the SNR threshold and data rate in ChirpTransformer with various encoder configurations. The orange bars (e.g., SF-7, SF-8, SF-9, SF-10, SF-11, and SF-12) represent the SNR threshold of standard LoRa, while the blue bars show the complementary SNR threshold of ChirpTransformer. As the data rate (e.g., brown lines) gradually decreases from $\frac{7 \cdot BW}{2^7}$ to $\frac{6 \cdot BW}{2^{12}}$, the SNR threshold slowly decreases from -6.2 dB to -25.7 dB. The mean difference between adjacent data rate schemes is about 0.89 dB. Compared with the standard LoRa, we can see that ChirpTransformer provides a more flexible data rate scheme with a more diverse SNR threshold.

Results for Battery Life Gain (BLG): Figure 6.20 shows that LoRa data rate adaptation averagely

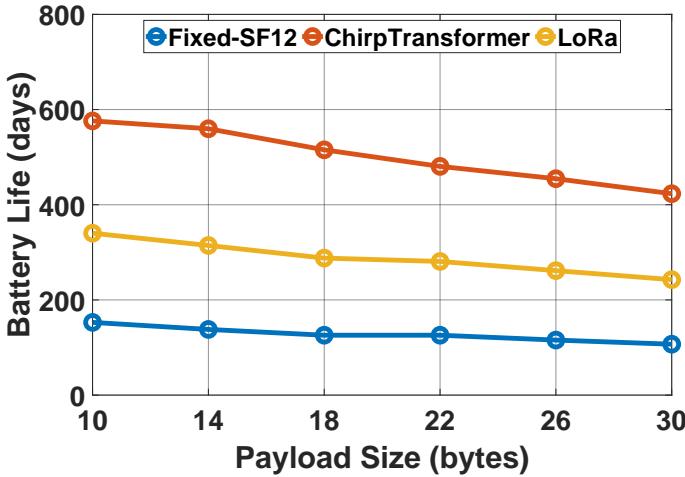


Figure 6.20 Battery life measurement with LoSee vehicle tracking traces.

extends battery life by 2.26 \times compared to the fixed-SF-12. On average, ChirpTransformer data rate adaption provides a battery life of 3.93 \times that of LoRaWAN with fixed-SF-12 and 1.74 \times that of LoRa data rate adaptation. The reason for this improvement is that ChirpTransformer reassigns a better data rate for the positions where the measured SNR is between the SNR threshold values of two standard LoRa SF configurations, whereas conventional rate adaptation in LoRa wastes the BLG improvement space. Moreover, the BLG trends remain stable as payload size increases.

6.7 Related Work

LoRa Reliable Decoding: By utilizing either multiple gateways and LoRa nodes, recent studies [155, 233–235, 241, 251] bring extra SNR gains for LoRa transmissions. Charm [155] coordinates multiple gateways to decode weak signals undecodable at any gateway by detecting the combined energy peak in the spectrum. Choir [241] exploits the correlation across co-located LoRa nodes, enabling a larger communication range than an individual one. NELoRa [100] develops a DNN decoder capturing multi-dimension features to obtain 1.84-2.35 dB SNR gains. Ostinato [251] uses repeated SF-12 chirps to achieve weak signal decoding beyond SF-12 configuration. XCopy [235] enhances signal strength by coherently combining retransmitted packets over weak links to boost weak signal decoding. In contrast, instead of developing a decoder at the gateway and server side, we obtain SNR gains from the encoder side. ChirpTransformer is parallel with these works.

Packet Collision Resolving: Previous research has focused on identifying collisions in the time or frequency domain. For instance, Choir [241] matches bits to each LoRa node by detecting the frequency changes caused by oscillator deficiencies. FTrack [240] identifies collisions by exploring distinct tracks on the spectrum and symbol edges in the time domain. Due to the near-far problem, existing work cannot implement concurrent transmission with high SIR among different devices in a large-scale deployment. CIC [229] decodes signals by combining the spectrum from different parts of a single symbol to cancel interference signals at gateways and support concurrent transmission. CurvingLoRa [171] enables efficient concurrent transmission with non-linear chirps, but it is not compatible with COTS LoRa nodes. LMAC [252] utilizes Channel Activity Detection (CAD) to implement carrier-sense multiple access protocol to avoid collision. Compared to these works, we propose an orthogonal encoding space at the encoder side to enable concurrent transmission, which can be realized on COTS LoRa nodes. The decoder design (e.g., CIC, Choir, FTrack) can be further combined with ChirpTransformer to further improve network performance.

Rate Adaptation in LoRa: To achieve an adaptive data rate [253], LoRa adjusts the data rate by using various SFs based on received SNR levels. FLoRa [254] focuses on dynamically managing link parameters to improve network scalability and energy efficiency. DyLoRa [243] establishes an energy model that associates link properties with transmission parameters. AdapLoRa [244] is another approach that periodically adjusts resource allocation based on a linear regression process estimating network lifetime. Current works rely on the standard LoRa encoding scheme and are unable to implement precise data rates necessary to adapt to diverse environments to extend battery life. Beyond the standard LoRa, ChirpTransformer provides fine-grained data rate adaption by selecting different sets of initial frequency to encode data bits with different on-air times.

6.8 Discussion

6.8.1 System Co-existence

ChirpTransformer and the standard LoRa encoder share the same group of basic chirps (e.g., SF-7 to SF-12), raising the concern of the co-existence issues when we apply ChirpTransformer in existing LoRa deployments. For example, when we utilize SH-[9-12] encoder to enhance com-

munication reliability, a ChirpTransformer symbol with the pattern of 8 repeated SF-9 chirps may interfere with other transmissions with SF-9 setting in co-existent standard LoRa deployment.

To measure the co-existence issues, we conduct experiments to emulate real-world signal collision between ChirpTransformer and the standard LoRa signals. First, we generate a ChirpTransformer symbol by selecting from one of three SH settings (i.e., SH-[7-10], SH-[8-11], SH-[9-12]). Then, the ChirpTransformer symbol is superposed with several standard LoRa symbols, characterized by one of the four different SF settings in the corresponding SH setting, with a random initial frequency offset. The on-air time of these standard LoRa symbols is the same as the ChirpTransformer symbol. We set a random time offset between the two superposed signals ranging from 0 to 1 of ChirpTransformer symbol on-air time. Moreover, the signal-to-interference ratio (SIR) [171, 247] of standard LoRa to ChirpTransformer indicates the signal strength of standard LoRa signals compared to ChirpTransformer signals. The higher the SIR is, the stronger the standard LoRa signals are. The lower the SIR is, the stronger the ChirpTransformer signals are. We divide the SIR values into 6 ranges: [-30, -20], [-20, -10], [-10, 0], [0, 10], [10, 20], and [20, 30]. With these emulation settings, we mimic the diverse collision conditions of ChirpTransformer and standard LoRa signals in various real-world scenarios. Finally, we apply our neural-enhanced decoder and the standard dechirp to decode ChirpTransformer and standard LoRa symbols from the collided signals. In each SIR range, we uniformly generate 1,000 collided signals across different SIR settings, SH settings in ChirpTransformer, and SF settings in the standard LoRa to calculate the average SER of decoding ChirpTransformer and standard LoRa symbols.

The impact of ChirpTransformer to existing LoRa deployments. As illustrated in Figure 6.21, it is evident that standard LoRa signals, when stronger than ChirpTransformer signals in the SIR ranges of [0,10], [10,20], and [20,30], can maintain a SER below 5%. However, when ChirpTransformer signals are stronger than standard LoRa signals as [-10,0] SIR, the SER of standard LoRa signals increases to 27.2%. This increase in SER is more pronounced as the SIR range decreases, highlighting interference from much stronger ChirpTransformer signals on existing LoRa.

The impact of the standard LoRa to ChirpTransformer. In Figure 6.21, we can also observe

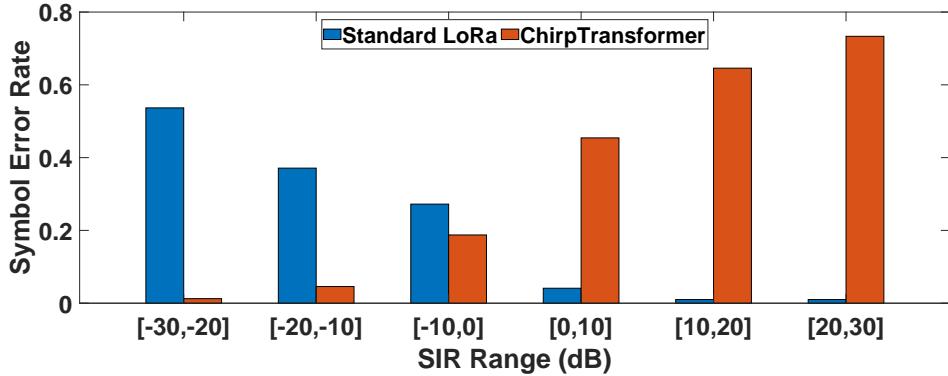


Figure 6.21 Concurrent decoding SER of collided ChirpTransformer and standard LoRa signals under various SIR settings.

that when standard LoRa signals are weaker than ChirpTransformer signals when SIR ranges are [-30,-20] and [-20,-10], the SER of decoding ChirpTransformer signals remains less than 5%. The SER rises up to about 20% at the SIR range [-10,0]. When standard LoRa signals are stronger than ChirpTransformer, the SER of ChirpTransformer increases quickly to 73% at the SIR range [20,30].

Remarks: The results imply a limitation of applying ChirpTransformer in practice that the co-existence issues are non-negligible. We could have three ways to alleviate the co-existence issues. Firstly, we can leverage the near-far effect. Since our SH encoder is more noise-resilient than the standard LoRa for weak signal decoding, ChirpTransformer is primarily adopted by LoRa nodes located beyond the reach of standard LoRa gateways. This ensures the high SIR of standard LoRa to ChirpTransformer and standard LoRa signals can be successfully decoded. Secondly, we can leverage LoRa carrier sense [55, 252] to avoid packet collision with transmission backoff. Thirdly, we can develop a new collision resolving method at the decoding side to enable successful decoding under low SIR conditions by borrowing the features from different LoRa packets [229, 238, 240]. For example, using multiple sub-symbol temporal windows of varying lengths [229] allows solving collision of ChirpTransformer and standard LoRa packets. By merging the spectral results from these windows, we retain the consistently appearing desired packet and eliminate intermittent interfering frequency peaks.

6.8.2 Efficient Adaptive Date Rate

ChirpTransformer supports a fine-grained adaptive data rate (ADR) with 23 configurations, allowing a LoRa nodes to adapt to its deployment scenario optimally. On the other hand, we need an efficient protocol to enable agile and accurate ADR with low control overhead. A feasible method is to inherit the current LoRaWAN ADR framework [21,253,255] in MAC commands. Specifically, LoRa gateways and network servers run an ADR algorithm that adjusts the encoding configuration of a LoRa node according to the SNR level of recently received packets. The LoRa node will open a receiving window after a packet transmission to receive the potential configuration request from LoRa gateways and network servers. When an encoding configuration change is necessitated, the LoRa gateways will initiate a request for data rate adjustment using *LinkADRReq* in the receiving window of the LoRa node. Then, the LoRa node will change its data rate correspondingly and respond with *LinkADRAns* for ADR acknowledgment.

By adopting our fine-grained ADR in this framework, a LoRa node will receive more requests from LoRa gateways and network servers and transmit more ADR acknowledgments. However, since the LoRa node keeps opening its receiving window after each packet transmission no matter whether a feedback packet is coming or not, receiving more requests will not bring extra energy consumption. Moreover, an ADR acknowledgment only contains two bytes of frame payload [253, 255], which is much shorter than a normal LoRa packet. Thus, the extra energy cost of sending more ADR acknowledgments is affordable regarding the benefits of our fine-grained SDR.

6.9 Conclusion

In this paper, we propose ChirpTransformer, a versatile LoRa encoding framework to enable the reliable connection between a LoRa node and gateways at extremely low SNR, achieve high network throughput with collision resolving, and improve energy efficiency in complex environments. Instead of using only one parameter, SF, to adjust the encoding method in the standard LoRa, ChirpTransformer develop four chirp features: 1) on-air time of a symbol; 1) available initial frequency offsets; 2) intra-symbol chirp repeating; 3) inter-symbol chirp pattern hopping, to enlarge the encoding feature space so that we can design different encoders to meet various LoRa deploy-

ment requirements. Different encoding methods can be represented with a four-factor tuple (OT, IFO, CRT, SH). Specifically, we have designed a symbol hopping based encoder for weak signal decoding, a chirp repeating based encoder for collision resolving, and a selective initial frequency offset based encoder for fine-grained data rate adaption. We implement ChirpTransformer with COTS LoRa nodes and SDR. Then, we conduct extensive experiments in both a campus testbed and real-world trace-driven studies to evaluate the performance of ChirpTransformer. The results show a $2.38 \times$ network coverage, $3.14 \times$ network throughput, and $3.93 \times$ battery lifetime compared with the standard LoRa.

CHAPTER 7

CONCLUSION

This dissertation systematically addresses critical challenges in deploying low-power, low-cost, and scalable LoRa-based wireless networking solutions for rural IoT. By developing practical techniques for robust satellite backhaul, improving signal coverage in complex rural environments, enabling scalable battery-free communication through innovative backscatter methods, and introducing flexible encoding mechanisms, my work significantly enhances the reliability, scalability, and efficiency of rural IoT deployments.

To build reliable and scalable LoRa networks for rural IoT, future research must go beyond local improvements and tackle full-system challenges in real-world conditions. The next step is to design systems that are more adaptive, intelligent, and well-integrated. First, space- and aerial vehicle-assisted networks can greatly improve coverage and flexibility but bring new challenges, such as handling intermittent links, supporting mobile gateways, and managing dynamic links. Second, embedding integrated sensing and communication into LoRa networks opens opportunities for joint environment monitoring and signal optimization. It requires new waveform designs and lightweight protocols with limited IoT resources. Third, to deal with the diversity of rural applications, we need cross-layer designs that coordinate the physical layer, network control, and application needs to dynamically balance throughput, energy, and coverage. These open the door to a new generation of LoRa systems that are not only technically robust but also deployable at scale across complex, underserved regions.

BIBLIOGRAPHY

- [1] C. Li and Z. Cao, “Lora networking techniques for large-scale and long-term iot: A down-to-top survey,” *ACM Computing Surveys*, vol. 55, no. 3, 2022.
- [2] A. Pagano, D. Croce, I. Tinnirello, and G. Vitale, “A survey on lora for smart agriculture: Current trends and future perspectives,” *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3664–3679, 2023.
- [3] C. Li and Z. Cao, “Lora networking techniques for large-scale and long-term iot: A down-to-top survey,” *ACM Computing Surveys (CSUR)*, 2022.
- [4] J. C. Liando, A. Gamage, A. W. Tengourtius, and M. Li, “Known and unknown facts of lora: Experiences from a large-scale measurement study,” *ACM Transactions on Sensor Networks*, vol. 15, no. 2, pp. 1–35, 2019.
- [5] C. Li, Y. Ren, S. Tong, S. I. Siam, M. Zhang, J. Wang, Y. Liu, and Z. Cao, “Chirptransformer: Versatile lora encoding for low-power wide-area iot,” in *Proceedings of ACM MobiSys*, 2024.
- [6] A. Research, “Nb-iot and lte-m issues to boost lora and sigfox near and long-term lead in lpwa network connections,” 2021.
- [7] Z. Sun, H. Yang, K. Liu, Z. Yin, Z. Li, and W. Xu, “Recent advances in lora: A comprehensive survey,” *ACM Transactions on Sensor Networks*, vol. 18, no. 4, pp. 1–44, 2022.
- [8] Y. Ren, L. Liu, C. Li, Z. Cao, and S. Chen, “Is lorawan really wide? fine-grained lora link-level measurement in an urban environment,” in *Proceedings of IEEE ICNP*, 2022.
- [9] Semtech, “Product details sx1262,” <https://www.semtech.com/products/wireless-rf/lora-connect/sx1262>, accessed 06-Jun-2025.
- [10] Y. Ren, A. Gamage, L. Liu, M. Li, S. Chen, Y. Dong, and Z. Cao, “Sateriot: High-performance ground-space networking for rural iot,” in *Proceedings of ACM MobiCom*, 2024.
- [11] Y. Ren, W. Sun, J. Du, H. Zeng, Y. Dong, M. Zhang, S. Chen, Y. Liu, T. Li, and Z. Cao, “Demeter: Reliable cross-soil lpwan with low-cost signal polarization alignment,” in *Proceedings of ACM MobiCom*, 2024.
- [12] Y. Ren, P. Cai, J. Jiang, J. Du, and Z. Cao, “Prism: High-throughput lora backscatter with non-linear chirps,” in *Proceedings of IEEE INFOCOM*, 2023.
- [13] Y. Ren, G. Li, Y. Liu, Y. Dong, and Z. Cao, “Aeroecho: Towards agricultural low-power wide-area backscatter with aerial excitation source,” in *Proceedings of IEEE INFOCOM*, 2025.
- [14] A. Pagano, D. Croce, I. Tinnirello, and G. Vitale, “A survey on lora for smart agriculture: Current trends and future perspectives,” *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3664–3679, 2023.

- [15] M. Gan, Y. Liu, L. Liu, C. Wu, Y. Dong, H. Zeng, and Z. Cao, “Poster: mmleaf: Versatile leaf wetness detection via mmwave sensing,” in *Proceedings of ACM MobiSys*, 2023.
- [16] S. Fang, L. Da Xu, Y. Zhu, J. Ahati, H. Pei, J. Yan, and Z. Liu, “An integrated system for regional environmental monitoring and management based on internet of things,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1596–1605, 2014.
- [17] M. T. Lazarescu, “Design of a wsn platform for long-term environmental monitoring for iot applications,” *IEEE Journal on emerging and selected topics in circuits and systems*, vol. 3, no. 1, pp. 45–54, 2013.
- [18] R. Wang, Y. Liu, and R. Müller, “Detection of passageways in natural foliage using biomimetic sonar,” *Bioinspiration & Biomimetics*, vol. 17, no. 5, p. 056009, 2022.
- [19] M. Grari, M. Yandouzi, I. Idrissi, M. Boukabous, O. Moussaoui, M. AZIZI, and M. MOUS-SAOUI, “Using iot and ml for forest fire detection, monitoring, and prediction: a literature review,” *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 19, pp. 5445–5461, 2022.
- [20] A. Sungheetha, R. Sharma *et al.*, “Real time monitoring and fire detection using internet of things and cloud based drones,” *Journal of Soft Computing Paradigm (JSCP)*, vol. 2, no. 03, pp. 168–174, 2020.
- [21] L. Alliance, “A technical overview of lora and lorawan,” https://lora-alliance.org/resource_hub/what-is-lorawan/, Retrieved Jun 06, 2025.
- [22] C. Li and Z. Cao, “Lora networking techniques for large-scale and long-term iot: A down-to-top survey,” *ACM Computing Surveys*, vol. 55, no. 3, pp. 1–36, 2022.
- [23] D. Yang, X. Zhang, X. Huang, L. Shen, J. Huang, X. Chang, and G. Xing, “Understanding power consumption of nb-iot in the wild: tool and large-scale measurement,” in *Proceedings of ACM MobiCom*, 2020.
- [24] A. Sørensen, H. Wang, M. J. Remy, N. Kjetstrup, R. B. Sørensen, J. J. Nielsen, P. Popovski, and G. C. Madueño, “Modeling and experimental validation for battery lifetime estimation in nb-iot and lte-m,” *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9804–9819, 2022.
- [25] D. Vasisht, J. Shenoy, and R. Chandra, “L2d2: Low latency distributed downlink for leo satellites,” in *Proceedings of ACM SIGCOMM*, 2021.
- [26] Y. Li, H. Li, W. Liu, L. Liu, Y. Chen, J. Wu, Q. Wu, J. Liu, and Z. Lai, “A case for stateless mobile core network functions in space,” in *Proceedings of ACM SIGCOMM*, 2022.
- [27] SpaceX, “Space x,” <https://www.spacex.com/>, accessed 06-Jun-2025.
- [28] SWARM, “Space x swarm - low cost, global satellite connectivity for iot,” <https://swarm.space/>, accessed 06-Jun-2025.
- [29] B. R. Elbert, *Introduction to satellite communication*. Artech house, 2008.

- [30] G. Maral, M. Bousquet, and Z. Sun, *Satellite communications systems: systems, techniques and technology*. John Wiley & Sons, 2020.
- [31] L. J. Ippolito Jr, *Satellite communications systems engineering: atmospheric effects, satellite link design and system performance*. John Wiley & Sons, 2017.
- [32] M. Centenaro, C. E. Costa, F. Granelli, C. Sacchi, and L. Vangelista, “A survey on technologies, standards and open challenges in satellite iot,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1693–1720, 2021.
- [33] L. J. Ippolito, *Radiowave propagation in satellite communications*. Springer Science & Business Media, 2012.
- [34] SWARM, “Product overview swarm m138 modem,” <https://swarm.space/wp-content/uploads/2022/09/Swarm-M138-Specifications.pdf>, accessed 06-Jun-2025.
- [35] Iridium, “Urb-iot: Lorawan outdoor gateway,” <https://www.iridium.com/products/urb-iot/>, accessed 06-Jun-2025.
- [36] LONESTAR, “Lorawan satellite gateway,” <https://www.lonestartracking.com/lorawan-satellite-gateway/>, accessed 06-Jun-2025.
- [37] DRYAD, “Fighting wildfires with long-range, low-power technologies,” <https://www.dryad.net/post/fighting-wildfires-with-lorawan>, accessed 06-Jun-2025.
- [38] SWARM, “Case study | dryad networks,” <https://swarm.space/wp-content/uploads/2022/02/ Swarm-Environmental-Case-Study-Dryad.pdf>, accessed 06-Jun-2025.
- [39] C. Li, Y. Ren, S. Tong, S. I. Siam, M. Zhang, J. Wang, Y. Liu, and Z. Cao, “Chirptransformer: Versatile lora encoding for low-power wide-area iot,” in *Proceedings of ACM MobiSys*, 2024.
- [40] J. Du, Y. Ren, Z. Zhu, C. Li, Z. Cao, Q. Ma, and Y. Liu, “Srlora: Neural-enhanced lora weak signal decoding with multi-gateway super resolution,” in *Proceedings of ACM MobiHoc*, 2023.
- [41] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J.-P. Vasseur, and R. Alexander, “Rpl: Ipv6 routing protocol for low-power and lossy networks,” Tech. Rep., 2012.
- [42] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, “Collection tree protocol,” in *Proceedings of ACM Sensys*, 2009.
- [43] F. Zhang, F. Yu, X. Zheng, L. Liu, and H. Ma, “Dfh: Improving the reliability of lr-fhss via dynamic frequency hopping,” in *Proceedings of IEEE ICNP*, 2023.
- [44] P. I. Parra, S. Montejo-Sánchez, J. A. Fraire, R. D. Souza, and S. Céspedes, “Network size estimation for direct-to-satellite iot,” *IEEE Internet of Things Journal*, vol. 10, no. 7, pp. 6111–6125, 2022.
- [45] Starlink, “Space x starlink,” <https://www.starlink.com/>, accessed 06-Jun-2025.

- [46] OneWeb, “Oneweb constellation,” <https://oneweb.net/>, accessed 06-Jun-2025.
- [47] Sateliot, “Sateliot,” <https://sateliot.space/en/>, accessed 06-Jun-2025.
- [48] Lacuna, “Lacuna space,” <https://lacuna.space/>, accessed 06-Jun-2025.
- [49] Semtech, “Semtech smart agriculture,” <https://www.semtech.com/lora/lora-applications/smart-agriculture>, accessed 06-Jun-2025.
- [50] SWARM, “Swarm eva kit,” <https://swarm.space/product/swarm-eval-kit/>, accessed 06-Jun-2025.
- [51] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. M. Montoya, J. C. M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff *et al.*, “Satellite communications in the new space era: A survey and future challenges,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 70–109, 2020.
- [52] S. Demetri, M. Zúñiga, G. P. Picco, F. Kuipers, L. Bruzzone, and T. Telkamp, “Automated estimation of link quality for lora: A remote sensing approach,” in *Proceedings of ACM/IEEE IPSN*, 2019.
- [53] Z. Cao, J. Wang, D. Liu, Q. Ma, X. Miao, and X. Mao, “Chase++: Fountain-enabled fast flooding in asynchronous duty cycle networks,” *IEEE/ACM Transactions on Networking*, vol. 29, no. 1, pp. 410–422, 2020.
- [54] Z. Li, M. Li, J. Wang, and Z. Cao, “Ubiquitous data collection for mobile users in wireless sensor networks,” in *Proceedings of IEEE INFOCOM*, 2011.
- [55] A. Gamage, J. C. Liando, C. Gu, R. Tan, and M. Li, “Lmac: efficient carrier-sense multiple access for lora,” in *Proceedings of ACM MobiCom*, 2020.
- [56] P. Levis, N. Patel, D. Culler, and S. Shenker, “Trickle: A self-regulating algorithm for code propagation and maintenance in wireless sensor networks,” in *Proceedings of USENIX/ACM NSDI*, 2004.
- [57] T. Connectivity, “Ant-916-cw-rcl-sma,” <https://www.te.com/usa-en/product-ANT-916-CW-RCL-SMA.html?q=CW-RCL&source=header>, accessed 06-Jun-2025.
- [58] ESPRESSIF, “Esp32,” <https://www.espressif.com/en/products/socs/esp32>, accessed 06-Jun-2025.
- [59] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in *Proceedings of ACM SIGKDD*, 2016.
- [60] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: synthetic minority over-sampling technique,” *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

- [61] R. Pi, “Raspberry pi 4,” <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>, accessed 06-Jun-2025.
- [62] Semtech, “Lora calculator,” <https://lora-developers.semtech.com/build/tools/calculator/>, Retrieved Sep 11, 2023.
- [63] B. Tao, M. Masood, I. Gupta, and D. Vasisht, “Transmitting, fast and slow: Scheduling satellite traffic through space and time,” in *Proceedings of ACM MobiCom*, 2023.
- [64] Y. Li, H. Li, W. Liu, L. Liu, W. Zhao, Y. Chen, J. Wu, Q. Wu, J. Liu, Z. Lai *et al.*, “A networking perspective on starlink’s self-driving leo mega-constellation,” in *Proceedings of ACM MobiCom*, 2023.
- [65] Z. Lai, H. Li, Y. Deng, Q. Wu, J. Liu, Y. Li, J. Li, L. Liu, W. Liu, and J. Wu, “{StarryNet}: Empowering researchers to evaluate futuristic integrated space and terrestrial networks,” in *Proceedings of USNEIX NSDI*, 2023.
- [66] B. Tao, O. Chabra, I. Janveja, I. Gupta, and D. Vasisht, “Known knowns and unknowns: Near-realtime earth observation via query bifurcation in serval,” in *Proceedings of USENIX NSDI*, 2024.
- [67] J. A. Fraire, S. Henn, F. Dovis, R. Garello, and G. Taricco, “Sparse satellite constellation design for lora-based direct-to-satellite internet of things,” in *Proceedings of IEEE GLOBECOM*, 2020.
- [68] Z. Zhang, Y. Li, C. Huang, Q. Guo, L. Liu, C. Yuen, and Y. L. Guan, “User activity detection and channel estimation for grant-free random access in leo satellite-enabled internet of things,” *IEEE Internet of Things journal*, vol. 7, no. 9, pp. 8811–8825, 2020.
- [69] Y. Qian, L. Ma, and X. Liang, “Symmetry chirp spread spectrum modulation used in leo satellite internet of things,” *IEEE Communications Letters*, vol. 22, no. 11, pp. 2230–2233, 2018.
- [70] V. Singh, T. Chakraborty, S. Jog, O. Chabra, D. Vasisht, and R. Chandra, “Spectrumize: Spectrum-efficient satellite networks for the internet of things,” in *Proceedings of USENIX NSDI*, 2024.
- [71] D. Vasisht, Z. Kapetanovic, J. Won, X. Jin, R. Chandra, S. Sinha, A. Kapoor, M. Sudarshan, and S. Stratman, “Farmbeats: An iot platform for data-driven agriculture,” in *Proceedings of USENIX NSDI*, 2017.
- [72] N. Islam, B. Ray, and F. Pasandideh, “Iot based smart farming: Are the lpwan technologies suitable for remote communication?” in *Proceedings of IEEE SmartIoT*, 2020.
- [73] R. Gebbers and V. I. Adamchuk, “Precision agriculture and food security,” *Science*, 2010.
- [74] N. Ahmed, D. De, and I. Hussain, “Internet of things (iot) for smart precision agriculture and farming in rural areas,” *IEEE Internet of Things Journal*, 2018.

- [75] EPQ, “Ground water,” <https://www.epa.gov/report-environment/ground-water#importance>, retrieved by Aug 15th, 2023.
- [76] USEPA, “Estimated nitrate concentrations in groundwater used for drinking,” <https://www.epa.gov/nutrient-policy-data/estimated-nitrate-concentrations-groundwater-used-drinking>, accessed 06-Jun-2025.
- [77] ——, “Overview of greenhouse gases,” <https://www.epa.gov/ghgemissions/overview-greenhouse-gases>, accessed 06-Jun-2025.
- [78] N. Quintana-Ashwell, D. Gholsom, G. Kaur, G. Singh, J. Massey, L. J. Krutz, C. G. Henry, T. Cooke III, M. Reba, and M. A. Locke, “Irrigation water management tools and alternative irrigation sources trends and perceptions by farmers from the delta regions of the lower mississippi river basin in south central usa,” *Agronomy*, 2022.
- [79] C. A. Carlson and J. L. Ingraham, “Comparison of denitrification by pseudomonas stutzeri, pseudomonas aeruginosa, and paracoccus denitrificans,” *Applied and Environmental Microbiology*, 1983.
- [80] D. E. A. Center, “Nitrate in drinking water,” https://www.dhd2.org/wp-content/uploads/2016/12/deq-wd-gws-ciу-nitratebrochure_270430_7.pdf, accessed 06-Jun-2025.
- [81] Doktar, “Filiz: Agricultural sensor station,” <https://www.doktar.com/en/products/agricultural-sensor-station-soil-temperature/>, accessed 06-Jun-2025.
- [82] Monnit, “Monnit agriculture and livestock monitoring,” <https://www.monnit.com/applications/agriculture-livestock-monitoring/>, accessed 06-Jun-2025.
- [83] HoBo, “Hobonet field monitoring system,” <https://www.onsetcomp.com/hobonet-field-monitoring-system>, retrieved by Aug 14th, 2023.
- [84] I. F. Akyildiz and E. P. Stuntebeck, “Wireless underground sensor networks: Research challenges,” *Ad Hoc Networks*, 2006.
- [85] A. R. Silva and M. C. Vuran, “Cps²: integration of center pivot systems with wireless underground sensor networks for autonomous precision agriculture,” in *Proceedings of ACM/IEEE ICCPS*, 2010.
- [86] NSF, “2022 signal in the soils program solicitation,” <https://www.nsf.gov/pubs/2022/nsf22550/nsf22550.htm>, accessed 06-Jun-2025.
- [87] Z. Sun and I. F. Akyildiz, “Connectivity in wireless underground sensor networks,” in *Proceedings of IEEE SECON*, 2010.
- [88] B. Zhou, V. S. S. L. Karanam, and M. C. Vuran, “Impacts of soil and antenna characteristics on lora in internet of underground things,” in *Proceedings of IEEE GLOBECOM*, 2021.
- [89] K. Lin and T. Hao, “Experimental link quality analysis for lora-based wireless underground sensor networks,” *IEEE Internet of Things Journal*, 2021.

- [90] Z. Sun and I. F. Akyildiz, “Magnetic induction communications for wireless underground sensor networks,” *IEEE transactions on antennas and propagation*, 2010.
- [91] A. R. Silva and M. C. Vuran, “Communication with aboveground devices in wireless underground sensor networks: An empirical study,” in *Proceedings of IEEE ICC*, 2010.
- [92] K. Yang, Y. Chen, X. Chen, and W. Du, “Link quality modeling for lora networks in orchards,” in *Proceedings of ACM/IEEE IPSN*, 2023.
- [93] A. R. Silva and M. C. Vuran, “Empirical evaluation of wireless underground-to-underground communication in wireless underground sensor networks,” in *Proceedings of DCOSS*, 2009.
- [94] M. C. Vuran and I. F. Akyildiz, “Channel model and analysis for wireless underground sensor networks in soil medium,” *Physical communication*, 2010.
- [95] Y. Ren, L. Liu, C. Li, Z. Cao, and S. Chen, “Is lorawan really wide? fine-grained lora link-level measurement in an urban environment,” in *Proceedings of IEEE ICNP*, 2022.
- [96] M. Mezzavilla, M. Zhang, M. Polese, R. Ford, S. Dutta, S. Rangan, and M. Zorzi, “End-to-end simulation of 5g mmwave networks,” *IEEE Communications Surveys & Tutorials*, 2018.
- [97] X.-f. Wan, Y. Yang, J. Cui, and M. S. Sardar, “Lora propagation testing in soil for wireless underground sensor networks,” in *Proceedings of IEEE APCAP*, 2017.
- [98] C. Ebi, F. Schaltegger, A. Rüst, and F. Blumensaat, “Synchronous lora mesh network to monitor processes in underground infrastructure,” *IEEE Access*, 2019.
- [99] G. Di Renzone, S. Parrino, G. Peruzzi, A. Pozzebon, and D. Bertoni, “Lorawan underground to aboveground data transmission performances for different soil compositions,” *IEEE Transactions on Instrumentation and Measurement*, 2021.
- [100] C. Li, H. Guo, S. Tong, X. Zeng, Z. Cao, M. Zhang, Q. Yan, L. Xiao, J. Wang, and Y. Liu, “Nelora: Towards ultra-low snr lora communication with neural-enhanced demodulation,” in *Proceedings of ACM Sensys*, 2021.
- [101] T. E. Berry, E. Lord, and C. Morgan, “Soil polarization data collected for the global undisturbed/disturbed earth (guide) program,” in *SPIE Proceedings of Polarization: Measurement, Analysis, and Remote Sensing XII*, 2016.
- [102] G. Hassan, K. Ismail, N. Persaud, and R. Reneau Jr, “Dependence of the degree of linear polarization in scattered visible light on soil textural fractions,” *Soil science*, 2004.
- [103] A. Sarkar, “Change of polarization of electromagnetic waves on reflection from planar interfaces,” *IETE Journal of Research*, 1981.
- [104] T. Tanikawa, K. Masuda, H. Ishimoto, T. Aoki, M. Hori, M. Niwano, A. Hachikubo, S. Matoba, K. Sugiura, T. Toyota *et al.*, “Spectral degree of linear polarization and neutral points of polarization in snow and ice surfaces,” *Journal of Quantitative Spectroscopy and Radiative Transfer*, 2021.

- [105] F.-M. Breon, D. Tanre, P. Lecomte, and M. Herman, “Polarized reflectance of bare soils and vegetation: measurements and models,” *IEEE Transactions on Geoscience and Remote Sensing*, 1995.
- [106] Semtech, “Semtech,” <https://www.semtech.com/>, accessed 06-Jun-2025.
- [107] T. A. Milligan, *Modern antenna design*. John Wiley & Sons, 2005.
- [108] M. Nilsson, “Directional antennas for wireless sensor networks,” in *In Proceedings of Adhoc Workshop*, 2009.
- [109] B. Großwindhager, M. S. Bakr, M. Rath, F. Gentili, W. Bösch, K. Witrisal, C. A. Boano, and K. Römer, “Poster: Switchable directional antenna system for uwb-based internet of things applications.” in *Proceedings of EWSN*, 2017.
- [110] Y. Wang, X. Zheng, L. Liu, and H. Ma, “Polartracker: Attitude-aware channel access for floating low power wide area networks,” in *Proceedings of IEEE INFOCOM*, 2021.
- [111] R. Li, X. Zheng, Y. Wang, L. Liu, and H. Ma, “Polarscheduler: Dynamic transmission control for floating lora networks,” in *Proceedings of IEEE INFOCOM*, 2022.
- [112] Semtech, “Sx1302,” <https://www.semtech.com/products/wireless-rf/lora-core/sx1302>, accessed 06-Jun-2025.
- [113] ———, “Sx1303,” <https://www.semtech.com/products/wireless-rf/lora-core/sx1303>, accessed 06-Jun-2025.
- [114] ———, “sx1302cf915gw1h,” <https://www.semtech.com/products/wireless-rf/lora-core/sx1302cf915gw1h>, accessed 06-Jun-2025.
- [115] L. Chen, W. Hu, K. Jamieson, X. Chen, D. Fang, and J. Gummesson, “Pushing the physical limits of iot devices with programmable metasurfaces,” in *Proceedings of USENIX NSDI*, 2021.
- [116] J. D. van der Laan, J. B. Wright, S. A. Kemme, and D. A. Scrymgeour, “Superior signal persistence of circularly polarized light in polydisperse, real-world fog environments,” *Applied Optics*, 2018.
- [117] J. Van der Laan, D. Scrymgeour, S. Kemme, and E. Dereniak, “Detection range enhancement using circularly polarized light in scattering environments for infrared wavelengths,” *Applied optics*, 2015.
- [118] B. Y. Toh, R. Cahill, and V. F. Fusco, “Understanding and measuring circular polarization,” *IEEE Transactions on Education*, 2003.
- [119] M. Nicolet, M. Schnaiter, and O. Stetzer, “Circular depolarization ratios of single water droplets and finite ice circular cylinders: a modeling study,” *Atmospheric Chemistry and Physics*, 2012.

- [120] A. Ishimaru, *Electromagnetic wave propagation, radiation, and scattering: from fundamentals to applications*. John Wiley & Sons, 2017.
- [121] J. D. van der Laan, J. B. Wright, D. A. Scrymgeour, S. A. Kemme, and E. L. Dereniak, “Evolution of circular and linear polarization in scattering environments,” *Optics Express*, 2015.
- [122] V. Temnov, K. Sokolowski-Tinten, P. Zhou, A. El-Khamawy, and D. Von Der Linde, “Multiphoton ionization in dielectrics: comparison of circular and linear polarization,” *Physical review letters*, 2006.
- [123] R. Galusca-OM6AA and P. Hazdra, “Circular polarization and polarization losses,” 2006.
- [124] L. Mottola, T. Voigt, and G. P. Picco, “Electronically-switched directional antennas for wireless sensor networks: A full-stack evaluation,” in *Proceedings of IEEE SECON*, 2013.
- [125] M. Nilsson, “Spida: A direction-finding antenna for wireless sensor networks,” in *Proceedings of Springer International Workshop on Real-World Wireless Sensor Networks*, 2010.
- [126] G. Tarter, L. Mottola, and G. P. Picco, “Directional antennas for convergecast in wireless sensor networks: Are they a good idea?” in *Proceedings of IEEE MASS*, 2016.
- [127] J. Schandy, S. Olofsson, N. Gammarano, L. Steinfeld, and T. Voigt, “Improving sensor network performance with directional antennas: A cross-layer optimization,” *ACM Transactions on Sensor Networks*, 2021.
- [128] N. G. Apergis, I. P. Inglesis, S. Vasileiadis, and S. A. Mitilinos, “Development of a small form factor switched-beam antenna for iot applications at the 868 mhz ism band,” in *Proceedings of APC*, 2019.
- [129] M. A. R. Hashmi and P. V. Brennan, “An antenna solution for glacial environmental sensor networks,” *IEEE Access*, 2023.
- [130] Y. Zhang, W. Sun, Y. Ren, S.-j. Lee, and M. Li, “Channel adapted antenna augmentation for improved wi-fi throughput,” *IEEE Transactions on Mobile Computing*, 2022.
- [131] pSemi, “pe44820,” <https://www.psemi.com/products/rf-phase-amplitude-control/phase-shifters/pe44820>, accessed 06-Jun-2025.
- [132] XRDS-RF, “3db xrds-rf 698-2700mhz passive splitter,” <https://www.amazon.com/Wide-Band-Splitter-3dB-Type-Female-50-XRDS-RF/dp/B07V6YH8BV>, accessed 06-Jun-2025.
- [133] WAVEFORM, “2x2 cross-polarized 600-2700 mhz outdoor antenna,” <https://www.waveform.com/products/2x2-mimo-outdoor-panel-antenna>, accessed 06-Jun-2025.
- [134] Semtech, “Sx1276 datasheet,” <https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>, accessed 06-Jun-2025.

- [135] ——, “Lora and lorawan,” <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/>, accessed 06-Jun-2025.
- [136] S. Tong, J. Wang, J. Yang, Y. Liu, and J. Zhang, “Citywide lora network deployment and operation: Measurements, analysis, and implications,” in *Proceedings of ACM SenSys*, 2023.
- [137] USDA, “Soil texture calculator,” Retrieved by Dec 27th 2023.
- [138] D. G. Groenendyk, T. P. Ferre, K. R. Thorp, and A. K. Rice, “Hydrologic-process-based soil texture classifications for improved visualization of landscape function,” *PloS one*, 2015.
- [139] USDA, “Web soil survey,” Retrieved by Dec 27th 2023.
- [140] PINO-TECH, “Soilwatch 10 – soil moisture sensor,” <https://pino-tech.eu/soilwatch10/>, accessed 06-Jun-2025.
- [141] S. Balivada, G. Grant, X. Zhang, M. Ghosh, S. Guha, and R. Matamala, “A wireless underground sensor network field pilot for agriculture and ecology: Soil moisture mapping using signal attenuation,” *Sensors*, 2022.
- [142] N. Chaamwe, W. Liu, and H. Jiang, “Wave propagation communication models for wireless underground sensor networks,” in *Proceedings of IEEE ICCT*, 2010.
- [143] C. L. Holloway, E. F. Kuester, J. A. Gordon, J. O’Hara, J. Booth, and D. R. Smith, “An overview of the theory and applications of metasurfaces: The two-dimensional equivalents of metamaterials,” *IEEE antennas and propagation magazine*, 2012.
- [144] S. Okamura and T. Oguchi, “Electromagnetic wave propagation in rain and polarization effects,” *Proceedings of the Japan academy, series B*, 2010.
- [145] K. Y. Bliokh and Y. P. Bliokh, “Conservation of angular momentum, transverse shift, and spin hall effect in reflection and refraction of an electromagnetic wave packet,” *Physical review letters*, 2006.
- [146] J. A. Shaw, “Degree of linear polarization in spectral radiances from water-viewing infrared radiometers,” *Applied optics*, 1999.
- [147] J. A. Stratton, *Electromagnetic theory*. John Wiley & Sons, 2007, vol. 33.
- [148] D. M. Pozar, *Microwave engineering*. John wiley & sons, 2011.
- [149] C.-I. Shie, J.-C. Cheng, S.-C. Chou, and Y.-C. Chiang, “Transdirectional coupled-line couplers implemented by periodical shunt capacitors,” *IEEE Transactions on Microwave Theory and Techniques*, 2009.
- [150] A. Salam and U. Raza, *Signals in the Soil*. Springer, 2020.
- [151] M. C. Vuran and A. R. Silva, “Communication through soil in wireless underground sensor networks–theory and practice,” *Sensor networks: Where theory meets practice*, 2009.

- [152] E. Research, “Usrp n210 datasheet,” <https://www.ettus.com/all-products/un210-kit/>, accessed 06-Jun-2025.
- [153] ———, “Sbx 400-4400 datasheet,” <https://www.ettus.com/all-products/sbx/>, accessed 06-Jun-2025.
- [154] J. Du, Y. Ren, M. Zhang, Y. Liu, and Z. Cao, “Nelora-bench: A benchmark for neural-enhanced lora demodulation,” *International Conference on Learning Representations (ICLR) Workshop on Machine Learning for IoT*, 2023.
- [155] A. Dongare, R. Narayanan, A. Gadre, A. Luong, A. Balanuta, S. Kumar, B. Iannucci, and A. Rowe, “Charm: Exploiting geographical diversity through coherent combining in low-power wide-area networks,” in *Proceedings of ACM/IEEE IPSN*, 2018.
- [156] Q. Wang, A. Terzis, and A. Szalay, “A novel soil measuring wireless sensor network,” in *Proceedings of IEEE I2MTC*, 2010.
- [157] J. Tooker, X. Dong, M. C. Vuran, and S. Irmak, “Connecting soil to the cloud: A wireless underground sensor network testbed,” in *Proceedings of IEEE SECON*, 2012.
- [158] X. Dong, M. C. Vuran, and S. Irmak, “Autonomous precision agriculture through integration of wireless underground sensor networks with center pivot irrigation systems,” *Ad Hoc Networks*, 2013.
- [159] G. Castellanos, M. Deruyck, L. Martens, and W. Joseph, “System assessment of wusn using nb-iot uav-aided networks in potato crops,” *IEEE Access*, 2020.
- [160] S. Yang, O. Baltaji, A. C. Singer, and Y. M. Hashash, “Development of an underground through-soil wireless acoustic communication system,” *IEEE Wireless Communications*, 2019.
- [161] J. Vasquez, V. Rodriguez, and D. Reagor, “Underground wireless communications using high-temperature superconducting receivers,” *IEEE Transactions on Applied Superconductivity*, 2004.
- [162] Q. Zhu and H. Lin, “Influences of soil, terrain, and crop growth on soil moisture variation from transect to farm scales,” *Geoderma*, 2011.
- [163] S. Datta, S. Taghvaeian, and J. Stivers, “Understanding soil water content and thresholds for irrigation management,” Oklahoma Cooperative Extension Service, Tech. Rep., 2017.
- [164] M. E. Holzman, R. Rivas, and M. C. Piccolo, “Estimating soil moisture and the relationship with crop yield using surface temperature and vegetation index,” *International Journal of Applied Earth Observation and Geoinformation*, 2014.
- [165] C. Nendel, M. Berg, K. C. Kersebaum, W. Mirschel, X. Specka, M. Wegehenkel, K. Wenkel, and R. Wieland, “The monica model: Testing predictability for crop growth, soil moisture and nitrogen dynamics,” *Ecological Modelling*, 2011.

- [166] Y. Peng, L. Shangguan, Y. Hu, Y. Qian, X. Lin, X. Chen, D. Fang, and K. Jamieson, “Plora: A passive long-range data network from ambient lora transmissions,” in *Proceedings of ACM SIGCOMM*, 2018.
- [167] V. Talla, M. Hessar, B. Kellogg, A. Najafi, J. R. Smith, and S. Gollakota, “Lora backscatter: Enabling the vision of ubiquitous connectivity,” *ACM IMWUT*, vol. 1, no. 3, 2017.
- [168] J. Jiang, Z. Xu, F. Dang, and J. Wang, “Long-range ambient lora backscatter with parallel decoding,” in *Proceedings of ACM MobiCom*, 2021.
- [169] M. Hessar, A. Najafi, and S. Gollakota, “Netscatter: Enabling large-scale backscatter networks,” in *Proceedings of USENIX NSDI*, 2019.
- [170] W. Mao, Z. Zhao, Z. Chang, G. Min, and W. Gao, “Energy-efficient industrial internet of things: overview and open issues,” *IEEE transactions on industrial informatics*, vol. 17, no. 11, pp. 7225–7237, 2021.
- [171] C. Li, X. Guo, L. Shangguan, Z. Cao, and K. Jamieson, “Curvinglora to boost lora network throughput via concurrent transmission,” in *Proceedings of USENIX NSDI*, 2022.
- [172] C. Li, Z. Cao, and L. Xiao, “Curvealoha: Non-linear chirps enabled high throughput random channel access for lora,” in *Proceedings of IEEE INFOCOM*, 2022.
- [173] X. Guo, L. Shangguan, Y. He, J. Zhang, H. Jiang, A. A. Siddiqi, and Y. Liu, “Aloba: rethinking on-off keying modulation for ambient lora backscatter,” in *Proceedings of ACM SenSys*, 2020.
- [174] Avago, “Hsms-285c datasheet,” <https://docs.broadcom.com/doc/AV02-1377EN>, accessed 06-Jun-2025.
- [175] T. Instruments, “Lpv7215mg datasheet,” <https://www.ti.com/store/ti/en/p/product/?p=LPV7215MG/NOPB>, accessed 06-Jun-2025.
- [176] STMicroelectronics, “Stm32l011d3p6 datasheet,” <https://www.st.com/resource/en/datasheet/stm32l011d4.pdf>, accessed 06-Jun-2025.
- [177] Maxim, “Max5530 datasheet,” [https://datasheets.maximintegrated.com/en/ds\(MAX5530-MAX5531\).pdf](https://datasheets.maximintegrated.com/en/ds(MAX5530-MAX5531).pdf), accessed 06-Jun-2025.
- [178] A. Devices, “Ltc6990is6 datasheet,” <https://www.analog.com/media/en/technicaldocumentation/data-sheets/LTC6990.pdf>, accessed 06-Jun-2025.
- [179] ——, “Adg902 datasheet,” https://www.analog.com/media/en/technicaldocumentation/data-sheets/ADG901_902.pdf, accessed 06-Jun-2025.
- [180] C. Li, H. Guo, S. Tong, X. Zeng, Z. Cao, M. Zhang, Q. Yan, L. Xiao, J. Wang, and Y. Liu, “Nelora: Towards ultra-low snr lora communication with neural-enhanced demodulation,” in *Proceedings of ACM SenSys*, 2021.

- [181] V. Liu, A. Parks, V. Talla, S. Gollakota, D. Wetherall, and J. R. Smith, “Ambient backscatter: Wireless communication out of thin air,” *ACM SIGCOMM computer communication review*, vol. 43, no. 4, p. 39–50, 2013.
- [182] R. Zhao, F. Zhu, Y. Feng, S. Peng, X. Tian, H. Yu, and X. Wang, “Ofdma-enabled wi-fi backscatter,” in *Proceedings of ACM MobiCom*, 2019.
- [183] Y. K. Chan, M. Y. Chua, and V. Koo, “Sidelobes reduction using simple two and tri-stages non linear frequency modulation (nlfm),” *Progress In Electromagnetics Research*, vol. 98, pp. 33–52, 2009.
- [184] S. Boukeffa, Y. Jiang, and T. Jiang, “Sidelobe reduction with nonlinear frequency modulated waveforms,” in *IEEE International Colloquium on Signal Processing and its Applications*, 2011.
- [185] N. Abramson, “The aloha system: Another alternative for computer communications,” in *Proceedings of the November 17-19, 1970, fall joint computer conference*, 1970, pp. 281–285.
- [186] R. Eletreby, D. Zhang, S. Kumar, and O. Yağan, “Empowering low-power wide area networks in urban settings,” in *Proceedings of ACM SIGCOMM*, 2017.
- [187] X. Xia, Y. Zheng, and T. Gu, “Ftrack: Parallel decoding for lora transmissions,” *IEEE/ACM Transactions on Networking*, vol. 28, no. 6, pp. 2573–2586, 2020.
- [188] M. O. Shahid, M. Philipose, K. Chintalapudi, S. Banerjee, and B. Krishnaswamy, “Concurrent interference cancellation: Decoding multi-packet collisions in lora,” in *Proceedings of ACM SIGCOMM*, 2021.
- [189] X. Wang, L. Kong, L. He, and G. Chen, “mLoRa: A Multi-Packet Reception Protocol in LoRa networks,” in *Proceedings of IEEE ICNP*, 2019.
- [190] S. Tong, Z. Xu, and J. Wang, “Colora: Enabling multi-packet reception in lora,” in *Proceedings of IEEE INFOCOM*, 2020.
- [191] Z. Xu, P. Xie, and J. Wang, “Pyramid: Real-time lora collision decoding with peak tracking,” in *Proceedings of IEEE INFOCOM*, 2021.
- [192] S. Tong, J. Wang, and Y. Liu, “Combating packet collisions using non-stationary signal scaling in lpwans,” in *Proceedings of ACM MobiSys*, 2020.
- [193] X. Xia, N. Hou, Y. Zheng, and T. Gu, “Pcube: scaling lora concurrent transmissions with reception diversities,” in *Proceedings of ACM MobiCom*, 2021.
- [194] A. Gamage, J. Liando, C. Gu, R. Tan, M. Li, and O. Seller, “Lmac: Efficient carrier-sense multiple access for lora,” *ACM Transactions on Sensor Networks*, vol. 19, no. 2, pp. 1–27, 2023.
- [195] N. Kouvelas, V. S. Rao, R. V. Prasad, G. Tawde, and K. Langendoen, “P-carma: Politely scaling lorawan,” in *Proceedings of ACM MobiCom*, 2020.

- [196] O. Elijah, T. A. Rahman, I. Orikuhi, C. Y. Leow, and M. N. Hindia, “An overview of internet of things (iot) and data analytics in agriculture: Benefits and challenges,” *IEEE IoT Journal*, vol. 5, no. 5, pp. 3758–3773, 2018.
- [197] “FarmBeats: AI, Edge, and IoT for Smart Farms,” <https://www.microsoft.com/en-us/research/project/farmbeats-iot-agriculture/>, Retrieved by Jul 28th 2023.
- [198] M. S. Islam and G. K. Dey, “Precision agriculture: Renewable energy based smart crop field monitoring and management system using wsn via iot,” in *Proceedings of IEEE STI*, 2019.
- [199] Y. Liu, M. Gan, H. Zeng, L. Liu, Y. Dong, and Z. Cao, “Hydra: Accurate multi-modal leaf wetness sensing with mm-wave and camera fusion,” in *Proceedings of ACM MobiCom*, 2024.
- [200] T. Chakraborty, H. Shi, Z. Kapetanovic, B. Priyantha, D. Vasisht, B. Vu, P. Pandit, P. Pillai, Y. Chabria, A. Nelson, M. Daum, and R. Chandra, “Whisper: Iot in the tv white space spectrum,” in *Proceedings of USENIX NSDI*, 2022.
- [201] Y. Ren, Y. Wang, Y. Dong, S. Chen, M. Zhang, J. Tang, and Z. Cao, “Demeter-demo: Demonstrating cross-soil lpwan with low-cost signal polarization alignment,” in *Proceedings of ACM MobiCom*, 2024.
- [202] D. Yang, G. Xing, J. Huang, X. Chang, and X. Jiang, “Qid: Robust mobile device recognition via a multi-coil qi-wireless charging system,” *ACM Transactions on Internet of Things*, vol. 3, no. 2, pp. 1–27, 2022.
- [203] P. Zhang, P. Hu, V. Pasikanti, and D. Ganesan, “Ekhonet: High speed ultra low-power backscatter for next generation sensors,” in *Proceedings of ACM MobiCom*, 2014.
- [204] M. Katanbaf, A. Weinand, and V. Talla, “Simplifying backscatter deployment: Full-duplex lora backscatter,” in *Proceedings of USENIX NSDI*, 2021.
- [205] Y. Ren, P. Cai, J. Jiang, J. Du, and Z. Cao, “Prism: High-throughput lora backscatter with non-linear chirps,” in *Proceedings of IEEE INFOCOM*, 2023.
- [206] Y. Song, L. Lu, J. Wang, C. Zhang, H. Zheng, S. Yang, J. Han, and J. Li, “ μ mote: Enabling passive chirp de-spreading and μ w-level long-range downlink for backscatter devices,” in *Proceedings of USENIX NSDI*, 2023.
- [207] S. Li, H. Zheng, C. Zhang, Y. Song, S. Yang, M. Chen, L. Lu, and M. Li, “Passive dsss: Empowering the downlink communication for backscatter systems,” in *Proceedings of USENIX NSDI*, 2022.
- [208] X. Guo, L. Shangguan, Y. He, N. Jing, J. Zhang, H. Jiang, and Y. Liu, “Saiyan: Design and implementation of a low-power demodulator for lora backscatter systems,” in *Proceedings of USENIX NSDI*, 2022.
- [209] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, and I. Moscholios, “A compilation of uav applications for precision agriculture,” *Computer Networks*, vol. 172, p. 107148, 2020.

- [210] *Unlicensed White Space Device Operations in the Television Bands*, Federal Communications Commission, 2023.
- [211] J. Du, Y. Liu, Y. Ren, L. Liu, and Z. Cao, “Loratrimmer: Optimal energy condensation with chirp trimming for lora weak signal decoding,” in *Proceedings of ACM MobiCom*, 2024.
- [212] E. W. Weisstein, “Heaviside step function,” <https://mathworld.wolfram.com/>, 2002.
- [213] G. S. GADGETS, “Hackrf one,” <https://greatscottgadgets.com/hackrf/one/>, accessed 06-Jun-2025.
- [214] Y. Sangar, Y. Biradavolu, K. Pederson, V. Ranganathan, and B. Krishnaswamy, “Pact: Scalable, long-range communication for monitoring and tracking systems using battery-less tags,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 4, pp. 1–27, 2023.
- [215] S. Tong, Z. Xu, and J. Wang, “Colora: Enabling multi-packet reception in lora,” in *Proceedings of IEEE INFOCOM*, 2020.
- [216] X. Xia, N. Hou, Y. Zheng, and T. Gu, “Pcube: scaling lora concurrent transmissions with reception diversities,” *ACM Transactions on Sensor Networks*, vol. 18, no. 4, pp. 1–25, 2023.
- [217] J. Luo, Z. Xu, J. Lin, C. Chen, and R. Xiong, “Ch-mac: Achieving low-latency reliable communication via coding and hopping in lpwan,” *ACM Transactions on Internet of Things*, vol. 4, no. 4, pp. 1–25, 2023.
- [218] G. Yang, R. Dai, and Y.-C. Liang, “Energy-efficient uav backscatter communication with joint trajectory design and resource optimization,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 926–941, 2020.
- [219] S. Yang, Y. Deng, X. Tang, Y. Ding, and J. Zhou, “Energy efficiency optimization for uav-assisted backscatter communications,” *IEEE Communications Letters*, vol. 23, no. 11, pp. 2041–2045, 2019.
- [220] R. Han, L. Bai, Y. Wen, J. Liu, J. Choi, and W. Zhang, “Uav-aided backscatter communications: Performance analysis and trajectory optimization,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 10, pp. 3129–3143, 2021.
- [221] “Lora by the numbers,” <https://www.semtech.com/lora>, accessed 06-Jun-2025.
- [222] “Amazon sidewalk,” <https://www.amazon.com/Amazon-Sidewalk/b?ie=UTF8&node=21328123011>, accessed 06-Jun-2025.
- [223] “Microsoft farmbeats,” <https://www.microsoft.com/en-us/research/project/farmbeats-iot-agriculture/>, accessed 06-Jun-2025.
- [224] T. Chakraborty, H. Shi, Z. Kapetanovic, B. Priyantha, D. Vasisht, B. Vu, P. Pandit, P. Pillai, Y. Chabria, A. Nelson, M. Daum, and R. Chandra, “Whisper: Iot in the tv white space spectrum,” in *Proceedings of USENIX NSDI*, 2022.

- [225] Y. Yao, Z. Ma, and Z. Cao, “Losee: Long-range shared bike communication system based on lorawan protocol.” in *Proceedings of EWSN*, 2019, pp. 407–412.
- [226] L. Liu, Y. Yao, Z. Cao, and M. Zhang, “Deeplora: Learning accurate path loss model for long distance links in lpwan,” in *Proceedings of IEEE INFOCOM*, 2021.
- [227] Semtech, “Sx1276/77/78/79 datasheet,” <https://www.semtech.com/products/wireless-rf/lora-transceivers>, accessed 06-Jun-2025.
- [228] B. Ghena, J. Adkins, L. Shangguan, K. Jamieson, P. Levis, and P. Dutta, “Challenge: Unlicensed lpwans are not yet the path to ubiquitous connectivity,” in *Proceedings of ACM MobiCom*, 2019.
- [229] M. O. Shahid, M. Philipose, K. Chintalapudi, S. Banerjee, and B. Krishnaswamy, “Concurrent interference cancellation: Decoding multi-packet collisions in lora,” in *Proceedings of ACM SIGCOMM*, 2021.
- [230] D. Mu, Y. Chen, J. Shi, and M. Sha, “Runtime control of lora spreading factor for campus shuttle monitoring,” in *Proceedings of IEEE ICNP*, 2020.
- [231] J. Yang, Z. Xu, and J. Wang, “Ferrylink: Combating link degradation for practical lpwan deployments,” in *Proceedings of IEEE ICPADS*, 2021.
- [232] A. Gadre, R. Narayanan, A. Luong, A. Rowe, B. Iannucci, and S. Kumar, “Frequency configuration for low-power wide-area networks in heartbeat,” in *Proceedings of USENIX NSDI*, 2020.
- [233] N. Hou, X. Xia, and Y. Zheng, “Don’t miss weak packets: Boosting lora reception with antenna diversities,” in *Proceedings of IEEE INFOCOM*, 2022.
- [234] A. Balanuta, N. Pereira, S. Kumar, and A. Rowe, “A cloud-optimized link layer for low-power wide-area networks,” in *Proceedings of ACM MobiSys*, 2020.
- [235] X. Xia, Q. Chen, N. Hou, Y. Zheng, and M. Li, “Xcopy: Boosting weak links for reliable lora communication,” in *Proceedings of ACM MobiCom*, 2023.
- [236] S. Tong, Z. Shen, Y. Liu, and J. Wang, “Combating link dynamics for reliable lora connection in urban settings,” in *Proceedings of ACM MobiCom*, 2021.
- [237] S. Yu, X. Xia, N. Hou, Y. Zheng, and T. Gu, “Revolutionizing lora gateway with xgate: Scalable concurrent transmission across massive logical channels,” in *Proceedings of ACM MobiCom*, 2024.
- [238] S. Tong, J. Wang, and Y. Liu, “Combating packet collisions using non-stationary signal scaling in LPWANs,” in *Proceedings of ACM MobiSys*, 2020.
- [239] S. Tong, Z. Xu, and J. Wang, “CoLoRa: Enabling Multi-Packet Reception in LoRa,” in *Proceedings of IEEE INFOCOM*, 2020.

- [240] X. Xia, Y. Zheng, and T. Gu, “FTrack: parallel decoding for LoRa transmissions,” in *Proceedings of ACM SenSys*, 2019.
- [241] R. Eletreby, D. Zhang, S. Kumar, and O. Yağan, “Empowering Low-Power Wide Area Networks in Urban Settings,” in *Proceedings of ACM SIGCOMM*, 2017.
- [242] F. Yu, X. Zheng, L. Liu, and H. Ma, “Enabling concurrency for non-orthogonal lora channels,” in *Proceedings of ACM MobiCom*, 2023.
- [243] Y. Li, J. Yang, and J. Wang, “DyLoRa: Towards Energy Efficient Dynamic LoRa Transmission Control,” in *Proceedings of IEEE INFOCOM*, 2020.
- [244] W. Gao, Z. Zhao, and G. Min, “AdapLoRa: Resource Adaptation for Maximizing Network Lifetime in LoRa networks,” in *Proceedings of IEEE ICNP*, 2020.
- [245] Z. Sun, T. Ni, H. Yang, K. Liu, Y. Zhang, T. Gu, and W. Xu, “Flora: Energy-efficient, reliable, and beamforming-assisted over-the-air firmware update in lora networks,” in *Proceedings of ACM/IEEE IPSN*, 2023.
- [246] K. Yang and W. Du, “Lldpc: A low-density parity-check coding scheme for lora networks,” in *Proceedings of ACM Sensys*, 2022.
- [247] C. Li, Z. Cao, and L. Xiao, “Curvealoha: Non-linear chirps enabled high throughput random channel access for lora,” in *Proceedings of IEEE INFOCOM*, 2022.
- [248] T. Elshabrawy and J. Robert, “Interleaved chirp spreading lora-based modulation,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3855–3863, 2019.
- [249] “Monsoon solutions inc. high voltage power monitor,” <https://www.msoon.com/high-voltage-power-monitor>, accessed 06-Jun-2025.
- [250] R. PI, “Raspberry pi 4,” <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>, Retrieved by Mar 28 2024.
- [251] Z. Xu, P. Xie, J. Wang, and Y. Liu, “Ostinato: Combating lora weak links in real deployments,” in *Proceedings of IEEE ICNP*, 2022.
- [252] A. Gamage, C. J. Liando, C. Gu, R. Tan, and M. Li, “Lmac:efficient carrier-sense multiple access for lora,” in *Proceedings of ACM MobiCom*, 2020.
- [253] Semtech, “Lorawan 1.0.4 specification in depth for end device developers,” <https://learn.semtech.com/course/view.php?id=6§ion=1>, accessed 06-Jun-2025.
- [254] M. Slabicki, G. Premsankar, and M. Di Francesco, “Adaptive configuration of lora networks for dense iot deployments,” in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2018, pp. 1–9.
- [255] Semtech, “Implementing adaptive data rate,” <https://learn.semtech.com/mod/book/view.php?id=174>, accessed 06-Jun-2025.