

# Lawrence Berkeley National Laboratory

## LBL Publications

### Title

TDD LoRa and Delta Encoding in Low-Power Networks of Environmental Sensor Arrays for Temperature and Deformation Monitoring

### Permalink

<https://escholarship.org/uc/item/91n1n03s>

### Journal

Journal of Signal Processing Systems, 95(7)

### ISSN

1939-8018

### Authors

Wielandt, Stijn  
Uhlemann, Sebastian  
Fiolleau, Sylvain  
[et al.](#)

### Publication Date

2023

### DOI

10.1007/s11265-023-01834-2

### Copyright Information

This work is made available under the terms of a Creative Commons Attribution-NonCommercial License, available at <https://creativecommons.org/licenses/by-nc/4.0/>

Peer reviewed



# TDD LoRa and Delta Encoding in Low-Power Networks of Environmental Sensor Arrays for Temperature and Deformation Monitoring

Stijn Wielandt<sup>1</sup> · Sebastian Uhlemann<sup>1</sup> · Sylvain Fiolleau<sup>1</sup> · Baptiste Dafflon<sup>1</sup>

Received: 31 May 2022 / Revised: 18 October 2022 / Accepted: 27 December 2022

This is a U.S. Government work and not under copyright protection in the US; foreign copyright protection may apply 2023

## Abstract

Densely distributed sensor networks can revolutionize environmental observations by providing real-time data with an unprecedented spatiotemporal resolution. However, field deployments often pose unique challenges in terms of power provisions and wireless connectivity. We present a framework for wirelessly connected distributed sensor arrays for near-surface temperature and/or deformation monitoring. Our research focuses on a novel time division duplex implementation of the LoRa protocol, enabling battery powered base stations and avoiding collisions within the network. In order to minimize transmissions and improve battery life throughout the network, we propose a dedicated delta encoding algorithm that utilizes the spatial and temporal similarity in the acquired data sets. We implemented the developed technologies in a AA battery powered hardware platform that can be used as a wireless data logger or base station, and we conducted an assessment of the power consumption. Without data compression, the projected battery life for a data logger is 4.74 years, and a wireless base stations can last several weeks or months depending on the amount of network traffic. The delta encoding algorithm can further improve this battery life with a factor of up to 3.50. Our results demonstrate the viability of the proposed methods for low-power environmental wireless sensor networks.

**Keywords** LPWAN · Collision avoidance · TDD · LoRa · Delta encoding · Environmental sensing

## 1 Introduction

The quantification, prediction, and management of natural resources require multi-scale observations of near-surface environmental processes. Existing remote sensing and ground-based observations alone lack the spatial and/or temporal resolution needed to capture the heterogeneity in complex environments, like mountainous watersheds [1]. Densely distributed environmental sensor networks can overcome some of these limitations, but the concept remains

largely unused for near-surface monitoring [2]. The slow adoption can be attributed to the unique challenges posed by field deployments. This includes a lack of power provisions and network infrastructure, climatic and topographic impacts on radio frequency propagation, and mechanical issues related to humidity, moisture, pressure, and even wildlife damage. Commercial devices for environmental monitoring (as used in [3]) are usually the result of a dedicated and costly design process, and low-cost solutions usually consist of prototypes with general purpose hardware (e.g., Arduino, Raspberry Pi) [4]. The cost, replicability, and reliability of these devices prevents system scale-ups to hundreds or thousands of distributed sensors. Another limiting factor in large-scale deployments of environmental monitoring systems, is the wireless connectivity of sensor nodes. Wireless networks of densely distributed sensors can provide data with an unprecedented spatial and temporal resolution for environmental research, providing new insights that improve the predictive understanding of ecosystems and their hydro-biogeochemical processes, and of natural hazards and their triggering processes. Real-time data transmission and remote access is useful for sensor management and for scheduling maintenance, reducing system downtime, and preventing

---

✉ Stijn Wielandt  
stijnwielandt@lbl.gov

Sebastian Uhlemann  
suhlemann@lbl.gov

Sylvain Fiolleau  
sfiolleau@lbl.gov

Baptiste Dafflon  
bdafflon@lbl.gov

<sup>1</sup> Earth and Environmental Sciences Area, Lawrence Berkeley National Laboratory, 1 Cyclotron Rd, Berkeley 94720, California, USA

loss of data. In the context of natural resource management and infrastructure monitoring, real-time data can drive crucial decision making processes (e.g., water management, landslide early warning systems) [5, 6]. Despite the wide range of available low-power wide area network (LPWAN) technologies, the unique requirements for environmental monitoring applications result in a non-trivial wireless technology implementation. Despite the long range and low power consumption of NB-IoT and LTE-M [7], these cellular internet of things (IoT) technologies do not provide a solution for environmental sensor networks in remote field sites due to their lack of coverage and operation in licensed spectrum. Alternatives in unlicensed spectrum include Sigfox and LoRaWAN [8]. For sensor nodes, these technologies offer an ultra-low power consumption. However, their base stations were never designed for power constrained applications, and backhaul connectivity is usually required [9]. Wielandt and Dafflon [10] presented an implementation of the LoRa Physical layer in a novel LPWAN protocol stack that offers low power consumption for sensor nodes as well as base stations. Because all devices in the network are time synchronized, packet reception and transmission can be scheduled on each side of a link, resulting in an optimized power and spectrum usage through time division duplex (TDD) and frequency hopping spread spectrum (FHSS). The resulting network protocol avoids collisions in the network and allows the use of single-channel radios in the nodes and base stations, drastically reducing system complexity, cost, and power consumption of the latter [11]. Given the operation in unlicensed frequency bands, collisions with packets from external networks are still possible. However, the proposed network protocol is aimed towards environmental monitoring applications in remote locations where wireless technologies are generally absent.

In comparison to common internet-of-things devices with few sensors, environmental monitoring systems often produce larger amounts of data, requiring particular attention to data transmission and compression. Here, we adopt the network protocol from [10], and we add a presentation layer that handles data compression, effectively reducing network traffic and power consumption of nodes and base stations. Most LPWAN devices employ standard payload formats (e.g., CayenneLPP [12]) and/or compression techniques (e.g., Huffman encoding) [13]. For the presented implementation, a custom payload format and a lossless compression technique is proposed, taking advantage of the temporal and spatial similarity in environmental sensor data.

The proposed protocol stack is implemented in a custom hardware platform for densely distributed, vertically resolved environmental measurements. Each sensor node comprises a battery powered data logger that is connected to a sensor probe. The sensor probes consist of a tube with an array of cascaded, discrete sensors; here we present two use cases, one with temperature sensors, and one with alternating

temperature sensors and accelerometers. When temperature sensors are used, the probes can be vertically deployed above the ground for snowpack monitoring, or below the ground for identifying subsurface thermal regimes, quantifying soil thermal parameters, and estimating heat and/or water fluxes. When temperature sensors are alternated with accelerometers, the probes can be used to measure soil deformation and related thermo-hydrological properties and processes (e.g., soil movement driven by permafrost thaw). The mechanical design, thermal modeling, and the use case scenarios of these sensor arrays have been described and validated by Dafflon et al. [14] and Wielandt et al. [15]. This paper focuses on the system design, the implementation of the novel network protocol stack and data compression techniques, and the effect on power consumption for sensor nodes and base stations.

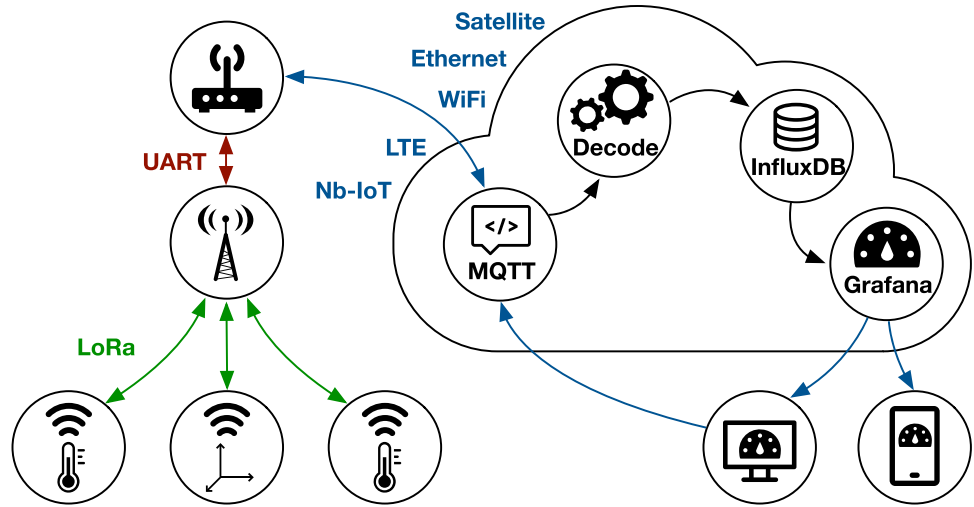
## 2 Wireless Networks for Environmental Monitoring

Figure 1 illustrates the architecture and services in a real-time environmental monitoring application. Environmental sensors connect through LoRa to a low-power base station, which communicates over UART with a data publisher. The publisher is connected to a server through a wired or wireless communication technology (e.g., Ethernet, LTE, NB-IoT, WiFi). The server hosts Mosquitto [16], a service that supports the open publish/subscribe MQTT protocol [17], specifically designed for low-cost, low-power operation over bandwidth constrained networks. The incoming sensor data is processed by a Python script, which decodes all messages and feeds data into a real-time data base. InfluxDB is an open-source database specifically targeted at handling time series data. [18]. Grafana is the open source service we suggest for data visualization [19]. The result is a real-time database and data visualization platform that can be accessed by users. Furthermore, users can use the MQTT service to send commands to the sensors for network and/or sensor management.

Most of the described protocols and technologies have been widely documented and their implementation requires little to no research efforts. However, the implementation of an LPWAN for remote environmental research applications poses a unique set of technical requirements that does not entirely align with current IoT technologies. Environmental sensors require accurate real-time clocks for regular time stamped measurements. Delays in data transmission are generally acceptable, but data loss should be avoided. Furthermore, remote network deployments require not only low-power sensor devices, but also low-power infrastructure.

LoRaWAN relies on LoRa as an underlying physical layer technology that provides long-range, low-power

**Figure 1** Architecture and services enabling real-time environmental monitoring. Environmental sensors connect through LoRa to a base station, which communicates over UART with a data publisher. The publisher is connected to a server with a real-time database and a visualization platform for users.

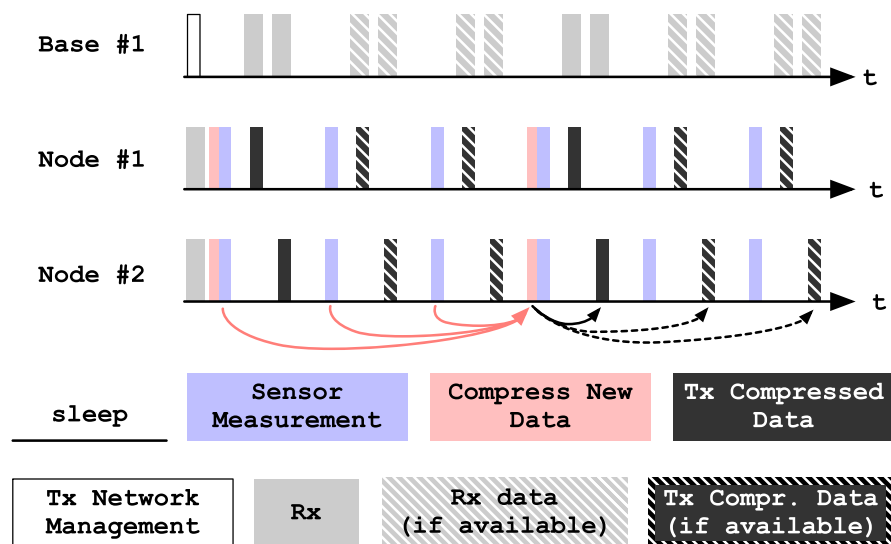


connectivity. However, LoRaWAN requires power intensive multi-channel base stations and backhaul connectivity [9]. The technology reduces collisions by using orthogonal spreading factors, however this does not eliminate collisions in the network. Furthermore, its fair-use policy limits the number of transmissions per day, or at least requires a listen-before-talk approach [20]. These limitations can be overcome with a novel LoRa based network protocol stack, as presented in [10]. The presented stack schedules transmissions in a TDD implementation, preventing package collisions. Pseudorandom network parameters (e.g., channel, bandwidth (*BW*), spreading factor (*SF*), code rate) are calculated on both sides of the link based on unique parameters (i.e., network name, device ID, time stamp). The protocol is particularly useful for environmental monitoring applications, since

regular measurements, transmissions, and receptions can be implemented in the TDD scheme. Figure 2 illustrates the different states of a network with one base station and two sensor nodes. The base station can either sleep, wake up to receive data from sensors, or broadcast a network management packet that includes the current time and measurement interval. The sensor nodes are mostly asleep, but wake up for a variety of reasons:

- When a scheduled network management packet is expected, the nodes will enter a LoRa receive (Rx) state.
- Once every measurement interval, all (*m*) sensors in the array are read out quasi simultaneously and their data is added to a buffer (i.e. data aggregation).
- After *t* measurement intervals, the buffer is full and a compression algorithm is executed, compressing the

**Figure 2** Scheme of TDD transmissions, sensor measurements, and data compression in a network with one base station and two sensor nodes (*t* = 3).



$t \cdot m$  sensor values. The algorithm reduces the number of transmissions required, resulting in a lower overall power consumption for sensor nodes and base stations.

- The compressed data is transmitted in as many packets as required ( $\leq t$ ). There is one transmission window per measurement interval for each node. When all compressed data is transmitted, no new transmissions will occur until a new compressed data set is available.

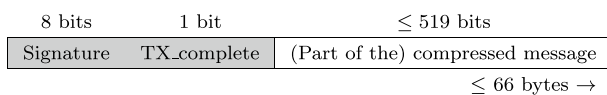
$$T_S = \frac{2^{SF}}{BW} \tag{1}$$

$$T_{\text{preamble}} = (n_{\text{preamble}} + 4.25) \cdot T_S \tag{2}$$

$$T_{\text{payload}} = T_S \cdot \left( 8 + \max \left\{ 0, (CR + 4) \cdot \left\lceil \frac{8PL - 4SF + 28 + 16CRC - 20H_{\text{en}}}{4SF - 2R_{\text{opt}}} \right\rceil \right\} \right) \tag{3}$$

$$T_{\text{packet}} = T_{\text{preamble}} + T_{\text{payload}} \tag{4}$$

Following FCC requirements [21], air time ( $T_{\text{packet}}$ ) is limited to 0.4 s. The air time depends on the duration of the packet’s preamble ( $T_{\text{preamble}}$ ) and payload ( $T_{\text{payload}}$ ). As expressed in (1-4), these values are determined by the duration of a LoRa symbol ( $T_S$ ). For sensor data transmissions,  $SF$  is a trade-off between range and throughput. In this research,  $SF = 9$  leads to a maximum payload ( $PL$ ) of 66 bytes following (1-4) [22], assuming an 8 bit preamble ( $n_{\text{preamble}} = 8$ ),  $BW = 125\text{e}3$ , 4/5 code rate ( $CR = 1$ ), cyclic redundancy check (CRC) enabled ( $CRC = 1$ ), explicit header ( $H_{\text{en}} = 0$ ), and low data rate optimization disabled ( $R_{\text{opt}} = 0$ ). The package structure is layed out in Fig. 3. The packet starts with a 9 bit packet header, containing an 8 bit signature for authentication (a pseudorandom bit sequence based on the network name, device ID, and time stamp). The subsequent ‘TX\_complete’ bit indicates if all compressed data has been transmitted, or if another packet will follow. When a transmission is complete (TX\_complete = 1), the base station and the corresponding sensor node will sleep through the remaining time slots to conserve power, as indicated in Fig. 2. Because of the small header size (9 bits vs. 117 bits for LoRaWAN), payloads of up to 519 bits can be transmitted in each packet. When the size



**Figure 3** Packet structure for compressed sensor data. When the size of a compressed message exceeds 519 bits, it is split into multiple LoRa packets.

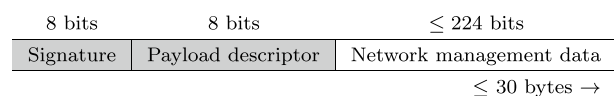
of the compressed message exceeds 519 bits, it is split into multiple LoRa packets.

Base station transmissions are performed with  $SF = 12$  and  $BW = 500\text{e}3$ , given the limited amount of network management data and the importance of network range. Following (1-4), this results in a maximum package size of 30 bytes, as presented in Fig. 4. The message starts with an 8 bits pseudorandom signature for identification, followed by a payload descriptor to identify the payload contents. In the most common scenario, the network management payload consists of a UTC time stamp (7 bytes, with epoch 00:00:00, 01/01/2000) and a measurement sample interval (2 bytes).

### 3 A Low-Power Hardware Platform with Sensor Node and Base Station Functionality

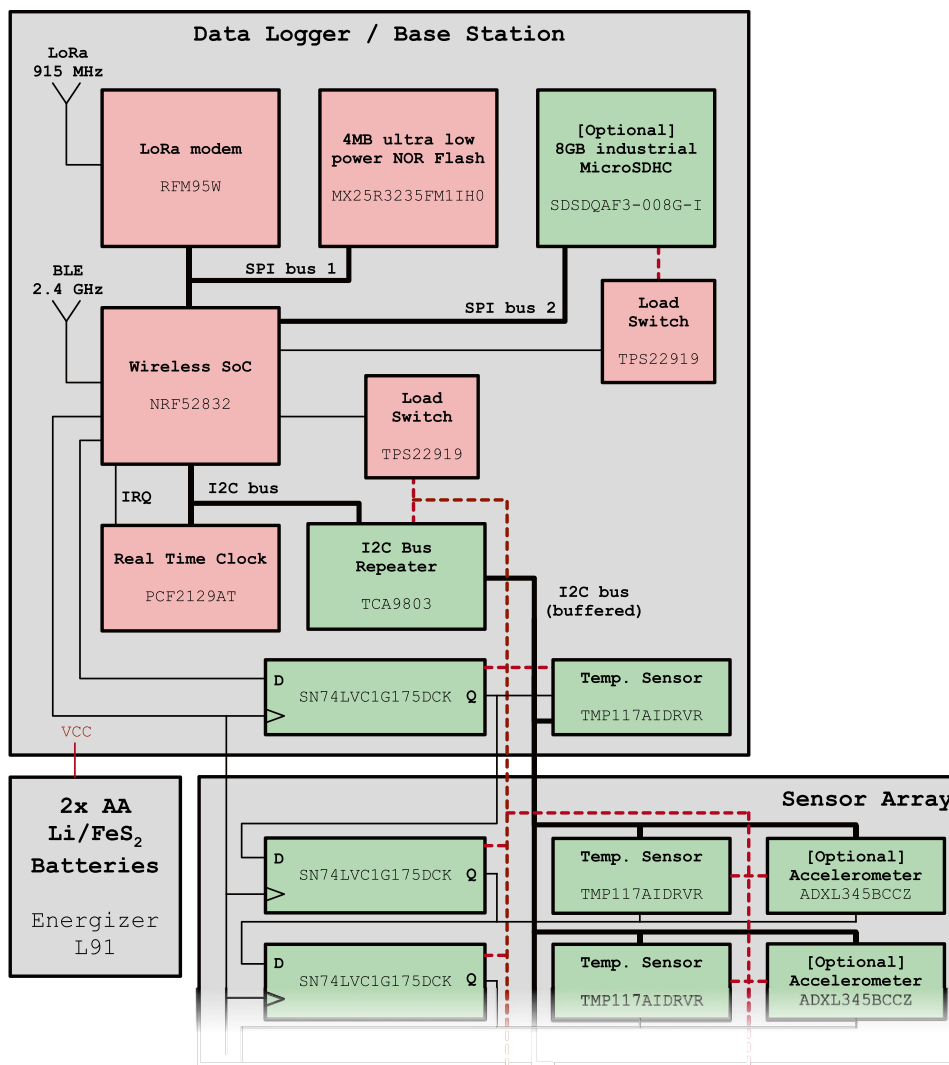
Figure 5 presents a hardware platform that provides the previously described functionality and meets the requirements of a low-cost, low-power, and reliable system for densely distributed, wirelessly connected thermal and deformation sensor arrays. Each device consists of a main board that can be used as a data logger or wireless base station. In order to allow standalone as well as networked sensor deployments, two wireless technologies are implemented. Bluetooth low energy (BLE) is selected for local connectivity to a computer or smartphone for data offloading and sensor configuration. The choice is motivated by the low power consumption and widely demonstrated technological maturity of BLE [23]. For long-range networking, we use LoRa technology, as previously discussed. In order to implement these technologies, we selected an NRF52832 ARM Cortex M4 as a low-power system-on-chip (SoC) with BLE provisions for short-range communications. This SoC controls a LoRa modem (RFM95W) for single channel communications and can be programmed to operate as a base station or wireless data logger. The PCF2129AT real-time clock takes care of the time sensitive tasks as presented in Fig. 2. A 4 MB low-power flash chip is used for storing sensor data on-device, which enables data logger functionality in offline configurations or data redundancy in online data loggers. In base station mode, received sensor data is stored on an optional microSDHC memory card, and data is shared with an external data publisher over a serial interface, as presented in Fig. 1.

The temperature sensor array consists of a series of TMP117A sensors with a resolution of 0.0078125 °C and



**Figure 4** Packet structure for network management transmissions.

**Figure 5** Block diagram of the hardware platform with LoRa sensor node or base station functionality for environmental sensor arrays. Red blocks are directly connected to the power supply, green blocks can be switched off.



a factory-assured accuracy of  $\pm 0.1$  °C [24], which was further improved to  $\pm 0.015$  °C using a calibration technique described in [14]. The high resolution and accuracy of the sensors enables observations of important environmental processes, e.g., freeze-thaw interfaces. For the temperature/deformation array each temperature sensor is accompanied by an ADXL345 accelerometer, allowing deformation measurements with a 0.390 mm resolution and a 95% confidence interval of  $\pm 0.73$  mm per meter of probe length [15]. The sensors are connected to the SoC through an I<sup>2</sup>C bus. In order to enable sensor arrays of up to 2 m long, the bus speed is limited to 100 kHz and an I<sup>2</sup>C bus buffer with integrated current sources is used (TCA9803). Each temperature sensor on the bus is accompanied by a D-type flip-flop, creating a shift register along the sensor array that is used for individually addressing each sensor.

All components in the system are selected for their operation in the 1.8 V - 3.6 V range, enabling the use of two AA batteries without a need for voltage regulators. Our

application uses Energizer L91 Li/FeS<sub>2</sub> cells, which have been rated at 3500 mAh for temperatures between  $-40$  °C and 60 °C [25]. The red blocks in Fig. 5 mark all the components that are directly connected to the power supply. The green blocks indicate the parts that are powered down with a load switch (TPS22919) whenever possible. This eliminates the impact of the MicroSDHC card and the entire sensor array on the device's sleep power.

## 4 Data Compression for Environmental Sensor Arrays

### 4.1 Use Case 1: Thermal Sensor Arrays

The thermal sensor arrays consist of a series of  $m$  TMP117A sensors [24]. Each sensor produces an output of 2 bytes, expressing the temperature in two's complement format with a 0.0078125 °C resolution. For each measurement, a time

**Table 1** Uncompressed temperature data points aggregated over time and space.

$$\begin{array}{cccc}
 & & j \in \{1, \dots, m\} & \\
 & & \leftarrow \text{Space} \rightarrow & \\
 & T_{1,1} & T_{1,2} & \dots & T_{1,j} \\
 \leftarrow \text{time} \rightarrow & T_{2,1} & T_{2,2} & \dots & T_{2,j} \\
 i \in \{1, \dots, t\} & \vdots & \vdots & \ddots & \vdots \\
 & T_{i,1} & T_{i,2} & \dots & T_{i,j}
 \end{array}$$

stamp (4 bytes) is stored, along with a 10 bits value of the battery voltage (stored as 2 bytes), followed by  $m$  temperature values of 2 bytes each. For a temperature probe with 17 sensors, this means that each sample takes up 40 bytes. Given the weight of LoRa transmissions in a device’s power budget [26] and the application’s tolerance towards delayed sensor data, we follow an approach of data aggregation over time, followed by compression, as proposed in [27] and illustrated in Fig. 2. Both data aggregation and compression can minimize energy consumption and network traffic, especially in tree-based network topologies where relay nodes reduce traffic by e.g., eliminating redundant data [28]. In this research, a star topology is employed, limiting opportunities for tree-based data aggregation. Other aggregation strategies are application driven, e.g., event based reporting, feature extraction. The application domain for our technology is environmental science and model development, hence we adopt a regular data reporting strategy without data reduction. Table 1 shows the aggregated temperature data, which forms a data set that exhibits similarity over time and space, because the sensors are deployed in a linear array to measure near-surface temperature gradients, and measurements are repeated regularly. While LPWAN payloads are often transmitted with lossy compression [29] or no compression at all (e.g., CayenneLPP [12]), some studies have investigated lossless data compression in LPWAN networks [13, 30]. Dictionary based compression techniques (e.g., Huffman encoding) can be a solution for efficient lossless compression, but lost packets can impact decoding in the long term and the exchange of dictionaries would cause prohibitive overhead for our monitoring system. The required reliability and the limited number of dynamic, noisy, and

high-resolution temperature data points, would result in frequent dictionary updates [31]. In this work we employ a delta encoding scheme, which allows the exploitation of spatial and temporal similarity with limited computational efforts. Instead of transmitting the entire collection of 16 bit temperature values, only  $T_{1,1}$  is transmitted as a reference value, followed by a series of  $\Delta T$  values. First, the variation over space is captured by the  $\Delta T_{1,j}$  values, as expressed in (5). Next, the temporal variability is captured according to (6).  $k$  expresses the required number of bits for the spatial variability ( $\Delta T_{1,j}$ ) values, and  $l$  represents the required number of bits for the temporal variability.

$$\Delta T_{1,j} = T_{1,j-1} - T_{1,j} \quad j \in \{2, \dots, m\} \tag{5}$$

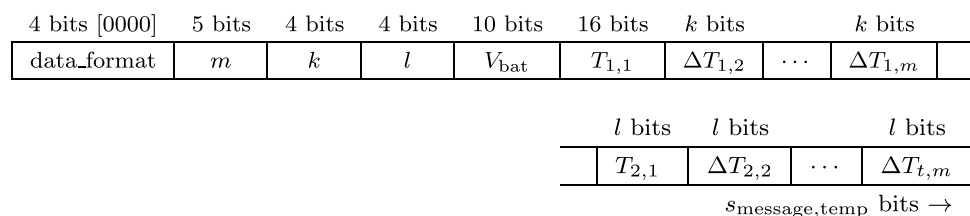
$$\Delta T_{i,j} = T_{i-1,j} - T_{i,j} \quad i \in \{2, \dots, t\} \quad j \in \{1, \dots, m\} \tag{6}$$

In order to transmit the compressed data set, a message is composed, as presented in Fig. 6. The first 4 bits represent a data format identifier that defines the message format and provides flexibility in terms of future sensors and compression protocols. Next,  $m$ ,  $k$ , and  $l$  are specified (respectively 5, 4, and 4 bits), as well as the battery voltage at the time of compression  $V_{\text{bat}}$  (10 bits). Finally, the compressed temperature data set is included starting with a 16 bits value for  $T_{1,1}$ . The total size of the message with compressed data ( $s_{\text{message,temp}}$ ) is determined by (7).

$$\begin{aligned}
 s_{\text{message,temp}} &= 4 + 5 + 4 + 4 + 10 + 16 \\
 &+ k \cdot (m - 1) + l \cdot m \cdot (t - 1) \quad [\text{bits}] \tag{7}
 \end{aligned}$$

Depending on  $m$ ,  $k$ , and  $l$ , the length of the composed message can exceed the maximum length of a data package. However,  $t$  time slots are available for transmitting the message, and the message can be split up into parts of  $\leq 519$  bits, as demonstrated in the packet structure in Fig. 3. The proposed compression algorithm is lossless, which means that all received data can be decoded without any loss of accuracy. However, under extreme conditions (e.g., a large number of sensors, significant spatial and or temporal variability in measurements) the size of a compressed message could exceed the available network throughput ( $t \times 519$  bits). In this case all incoming data will be decodable, but the tail of the compressed message would be lost.

**Figure 6** Message structure for compressed temperature data.



### 4.2 Use Case 2: Temperature/Deformation Sensor Arrays

The temperature/deformation probes contain an array of  $m$  pairs of TMP117A temperature sensors and ADXL345 accelerometers [15]. The accelerometers are used in static conditions, regularly sampling the Earth’s gravitational vector along three axes ( $x, y, z$ ). These measurements enable the calculation of each sensor’s tilt and thus the shape of the entire probe and its deformation over time. Each temperature value is represented as a 2 bytes value, and each accelerometer generates three 10 bits values ( $a_x, a_y, a_z$ ) to indicate the acceleration in two’s complement format with a resolution of 0.0039 g [32]. For a probe with 16 temperature/acceleration sensor pairs, this results in a total sample size of 736 bits (92 bytes). Given the maximum packet size of 519 bits, data compression does not only result in a reduced power consumption, but it is actually required to ensure that all data can be transmitted using the proposed protocol. In order to compress the measurement values,  $t$  samples are accumulated over time, in analogy to the compression algorithm for thermal sensor arrays. Table 2 presents the resulting data set, with temperature values exhibiting the same temporal and spatial similarity as explained before, so we employ the same delta encoding scheme for these values. Acceleration values exhibit different characteristics over time and space, justifying a modified approach for data compression. Unlike temperatures, which exhibit daily variations, the soil movements of interest are usually small, slow, and not reversible, resulting in a lot of potential for data compression. However, spatial similarity is not guaranteed, as soil deformation can be highly heterogeneous. Therefore, we generate a set of reference values ( $a_{i,ref,x}, a_{i,ref,y}, a_{i,ref,z}$ ) for all  $m$  accelerometers. These reference values are used to calculate each sensor’s delta values ( $\Delta a_{j,i,x}, \Delta a_{j,i,y}, \Delta a_{j,i,z}$ ). Given the slow variations in deformation measurements, reference values do not need to be updated in every message. Instead, reference values can be updated gradually and spread over multiple messages.

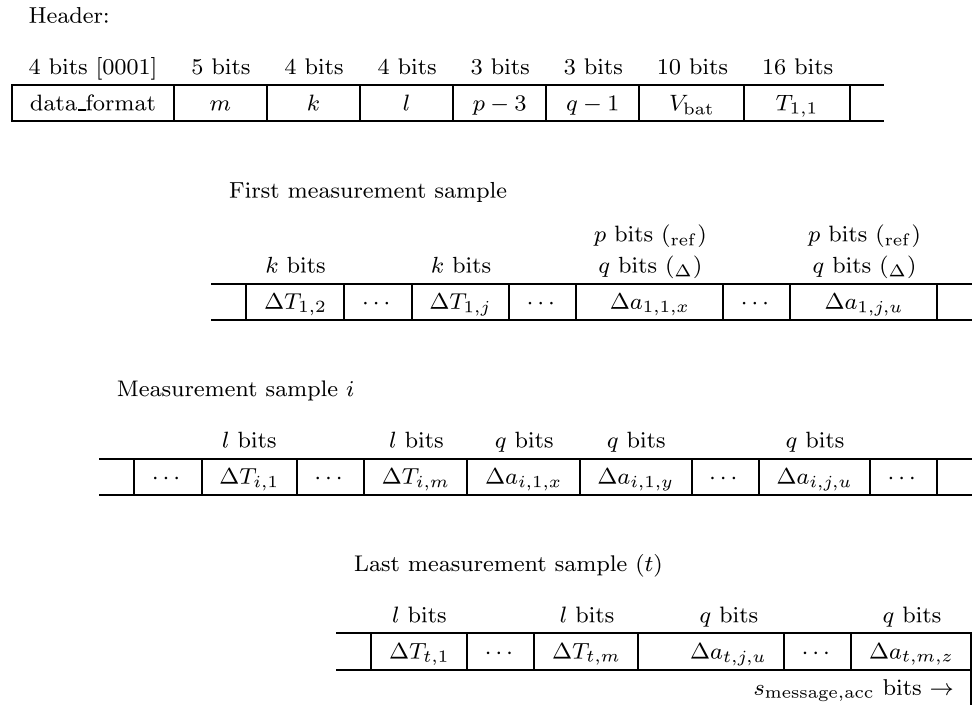
The number of reference values included in each message is a trade-off between compression rate and possible data loss: having all reference values included in each message results in low compression rates, but it guarantees that every received message can be decoded. When reference values are spread over multiple messages, higher compression rates can be achieved, but a single packet loss can affect multiple message decodings. In this research, we arbitrarily chose to include the reference values of two accelerometers in each message. For an  $m = 16$  probe with  $t = 4$  and a sampling interval of 15 min, this means that the entire reference data set will be refreshed every 8 hours. The calculation of reference values and delta values is presented in 8. In order to determine which reference values to update and transmit, a daily sample count ( $c$ ) is used. Any network device with a synchronized clock and knowledge of the sampling interval can calculate  $c$ , so this value does not need to be included in the message. For deformation measurements, the expected acceleration values are  $a_x = 1$  g,  $a_y = 0$  g and  $a_z = 0$  g because the probes are deployed vertically in the soil [15]. We use these expected values in the calculation of reference values, which allows a reduction of the number of bits per reference value ( $p$ ). The number of bits per delta value is defined as ( $q$ ). Assuming that  $p \geq 3$  and  $q \geq 1$ , each value can be encoded in the message header using just 3 bits, as presented in Fig. 7. This table shows the data format (designated as ‘0001’) for temperature/deformation arrays, consisting of a 49 bits header, followed by the first measurement sample’s encoded temperature and acceleration values (which includes two accelerometers’ updated reference values, according to 8). Subsequently, the temperature and acceleration values for the remaining samples are appended. This approach allows the receiver to start decoding the oldest measurement samples in messages that have not yet arrived completely (i.e. messages that have been split into multiple packets because they exceeded the 519 bits packet payload size). The total size of a message ( $s_{message,acc}$ ) is expressed by 9.

**Table 2** Uncompressed temperature and acceleration values aggregated over time and space.

	$j \in \{1, \dots, m\}$ ← Space →			$j \in \{1, \dots, m\}$ ← Space →						
$i \in \{1, \dots, t\}$ ← Time →	$T_{1,1}$	⋯	$T_{1,j}$	$a_{1,1,x}$	$a_{1,1,y}$	$a_{1,1,z}$	⋯	$a_{1,j,x}$	$a_{1,j,y}$	$a_{1,j,z}$
	$T_{2,1}$	⋯	$T_{2,j}$	$a_{2,1,x}$	$a_{2,1,y}$	$a_{2,1,z}$	⋯	$a_{2,j,x}$	$a_{2,j,y}$	$a_{2,j,z}$
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	$T_{i,1}$	⋯	$T_{i,j}$	$a_{i,1,x}$	$a_{i,1,y}$	$a_{i,1,z}$	⋯	$a_{i,j,x}$	$a_{i,j,y}$	$a_{i,j,z}$



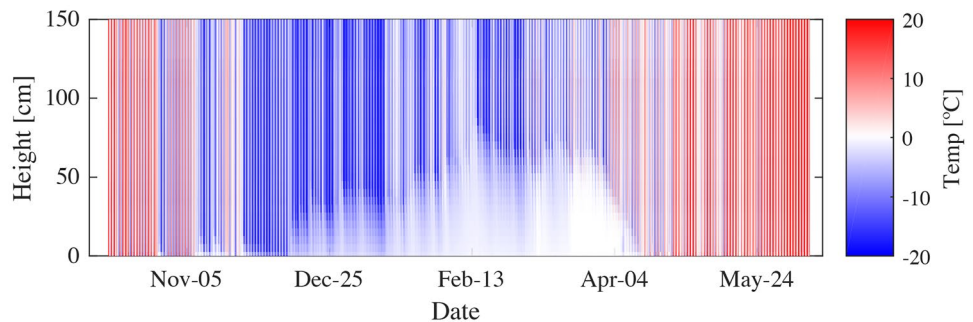
**Figure 7** Message structure for compressed temperature and acceleration data with  $i \in \{2, \dots, t\}$ ,  $j \in \{1, \dots, m\}$ ,  $u \in \{x, y, z\}$ .



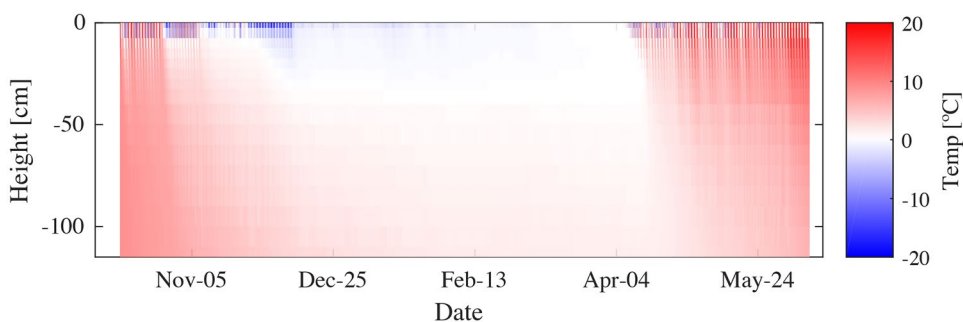
$$\left. \begin{array}{l} \forall i \in \{1, \dots, t\} \\ \forall j \in \{1, \dots, m\} \end{array} \right\} \left\{ \begin{array}{l} \left. \begin{array}{l} a_{i,\text{ref},x} = \Delta a_{i,j,x} = a_{i,j,x} - 1 \\ a_{i,\text{ref},y} = \Delta a_{i,j,y} = a_{i,j,y} \\ a_{i,\text{ref},z} = \Delta a_{i,j,z} = a_{i,j,z} \\ p = \max_{u \in \{x, y, z\}} (\lceil \log_2 |\Delta a_{i,j,u}| \rceil + 1) \end{array} \right\} \begin{array}{l} \text{if } i = 1 \text{ and} \\ \lfloor \frac{j}{2} \rfloor \equiv \frac{c}{t} \pmod{\frac{m}{2}} \\ \text{(Ref. values)} \end{array} \\ \\ \left. \begin{array}{l} \Delta a_{i,j,x} = a_{i,j,x} - a_{i,\text{ref},x} \\ \Delta a_{i,j,y} = a_{i,j,y} - a_{i,\text{ref},y} \\ \Delta a_{i,j,z} = a_{i,j,z} - a_{i,\text{ref},z} \\ q = \max_{u \in \{x, y, z\}} (\lceil \log_2 |\Delta a_{i,j,u}| \rceil + 1) \end{array} \right\} \begin{array}{l} \text{otherwise} \\ \text{(\Delta values)} \end{array} \end{array} \right. \quad (8)$$

$$s_{\text{message,acc}} = 49 + k \cdot (m - 1) + l \cdot m \cdot (t - 1) + 2 \cdot 3 \cdot p + (m \cdot t - 2) \cdot 3 \cdot q \quad [\text{bits}] \quad (9)$$

**Figure 8** Thermal data series for an above ground sensor array ( $m = 29$ ). The array was deployed at the 300 km<sup>2</sup> East River watershed, Colorado, and temperatures were sampled every 15 min.



**Figure 9** Thermal data series for a subsurface sensor array ( $m = 17$ ). The array was deployed at the East River watershed, Colorado, and temperatures were sampled every 15 min.



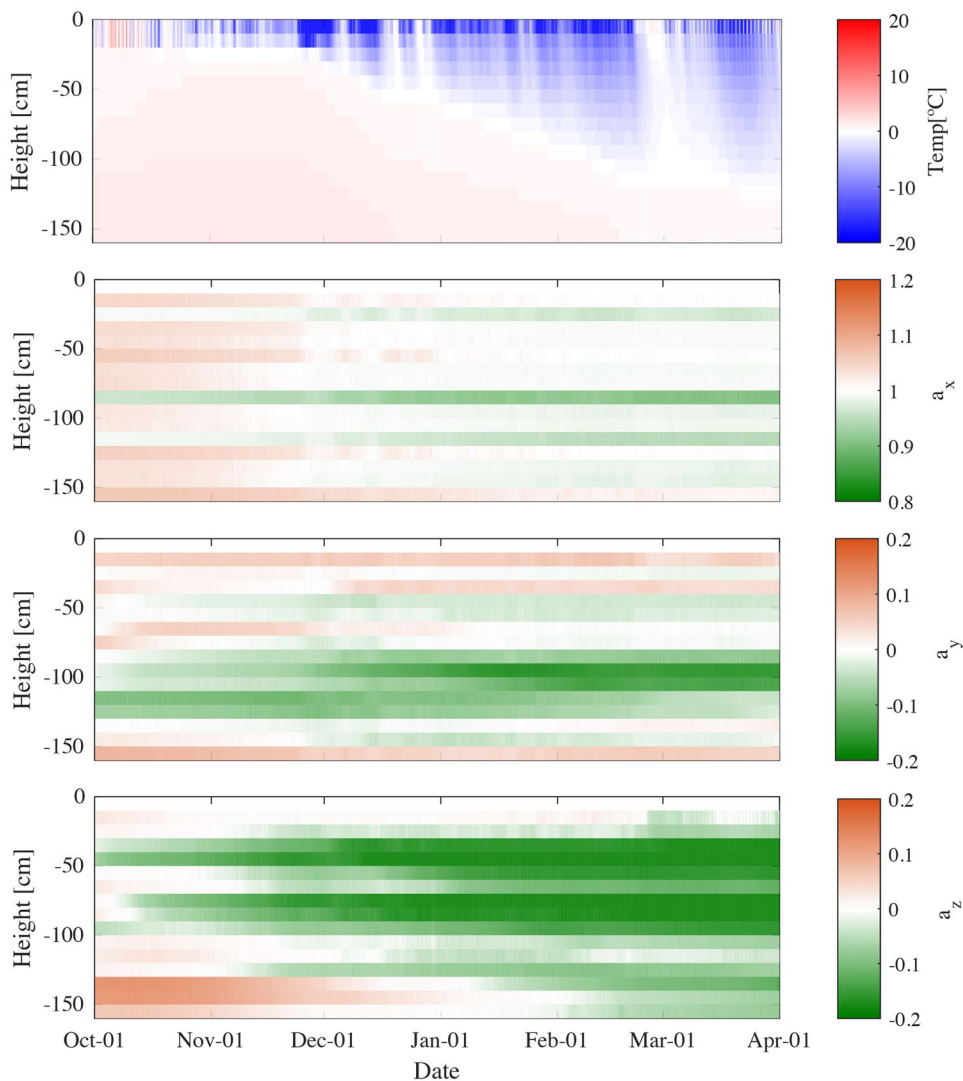
## 5 Results

### 5.1 Data Compression

To evaluate the efficiency of the proposed compression algorithm and its impact on the battery life of nodes and base stations, three data sets were acquired. The spatial and temporal profiles for these use cases are considerably

different, so we evaluate the compression algorithm for each scenario. Figure 8 presents a 10 months data series with 15 min measurement intervals for an above ground thermal sensor array ( $m = 29$ ) for snowpack monitoring in the East River watershed, Colorado [33]. The data are characterized by pronounced diurnal temperature cycles that fade under snow cover, enabling an estimation of snow thickness over time [34]. Figure 9 depicts subsurface temperature

**Figure 10** Data series for a subsurface sensor array for deformation and temperature sensing ( $m = 19$ ). The array was deployed at a watershed near Nome, Alaska, and data were sampled every 15 min.



**Table 3** Performance of the compression algorithm for two 10-month datasets, and the impact on power consumption.

<b>Temperature - Subsurface <math>m = 17</math></b>										
$t$			None*	2	3	4	5	6	7	8
Size data set [KB]			936	419	380	364	355	350	348	346
Compression ratio			1.000	0.448	0.406	0.388	0.379	0.374	0.372	0.370
# Transmitted packets			23,298	11,699	9,458	8,806	7,267	6,962	6,889	6,692
Power	Node	Battery life [days]	1,734	2,654	2,958	3,060	3,330	3,389	3,404	3,444
		Power savings factor	1.00	1.53	1.71	1.76	1.92	1.95	1.96	1.99
	Base	Battery life** [days]	32	64	79	85	103	107	108	112
		Power savings* factor	1.00	2.00	2.47	2.66	3.22	3.34	3.38	3.50
<b>Temperature - Snowpack <math>m = 29</math></b>										
$t$			None*	2	3	4	5	6	7	8
Size data set [KB]			1,606	810	801	804	807	811	815	819
Compression ratio			1.000	0.536	0.530	0.532	0.533	0.537	0.539	0.542
# Transmitted packets			23,617	20,540	15,739	15,973	14,418	14,819	14,039	14,204
Power	Node	Battery life [days]	1,734	1,908	2,259	2,240	2,380	2,343	2,418	2,401
		Power savings factor	1.00	1.10	1.30	1.29	1.37	1.35	1.39	1.38
	Base	Battery life** [days]	32	37	49	48	53	52	54	54
		Power savings* factor	1.00	1.17	1.52	1.50	1.66	1.61	1.70	1.68
<b>Temperature/Deformation - Subsurface <math>m = 19</math></b>										
$t$			None*	2	3	4	5	6	7	8
Size data set [KB]			1,873	1,257	1,200	1,173	1,159	1,148	1,143	1,138
Compression ratio			1.000	0.671	0.641	0.626	0.619	0.613	0.610	0.608
# Transmitted packets			39,730	24,413	20,393	19,975	19,754	19,283	19,074	18,831
Power	Node	Battery life [days]	1,023	1,496	1,703	1,728	1,742	1,771	1,785	1,800
		Power savings factor	1.00	1.46	1.67	1.69	1.70	1.73	1.75	1.76
	Base	Battery life** [days]	16	26	32	32	33	33	34	34
		Power savings* factor	1.00	1.62	1.94	1.98	2.00	2.05	2.08	2.10

\* $t = \text{None}$  represents the results for uncompressed data

\*\*In the base station power projections we assume a scenario with 100 connected nodes that exhibit the same compression ratios

data ( $m = 17$ ) for the same location, measurement interval, and time frame as the snowpack sensor array. Diurnal temperature variations can be observed near the surface, but temperatures at greater depth show more temporal stability. Figure 10 shows the temperature and acceleration values for a subsurface probe deployed on the Seward Peninsula, near Nome, Alaska, measuring every 15 min over a 6 months period. The temperature data show diurnal variations near the surface, as well as a deepening freezing front over time. As the freezing front evolves, we can also observe a change in acceleration values, indicating soil deformation.

For each data set, Table 3 compares several performance indicators for compressed and uncompressed data. The compressed data is evaluated as a function of  $t$ : as more data is aggregated over time, the efficiency of the compression algorithm can be expected to increase. However, higher values of  $t$  result in larger delays and a larger impact of packet loss. In order to assess the performance of the algorithm we consider

the number of transmitted packets, and the total size of the data set in KB, which leads to an easier to assess compression ratio (compressed to uncompressed data set size).

Table 3 clearly indicates the impact of  $t$  for all data sets. Low values of  $t$  result in a strong compression of the data, but in the case of snowpack measurements this does not necessarily lead to a proportional decrease in transmitted packets. This can be explained by the potentially inefficient filling of LoRa packets: higher values of  $t$  lead to larger  $s_{\text{message}}$  values, which means that more LoRa packets contain the maximum payload of 66 bytes. One can also remark that the impact of  $t$  diminishes as its value increases. Given the larger delays and the increased impact of packet loss, we advise a compromise of  $t \in \{3, \dots, 5\}$ .

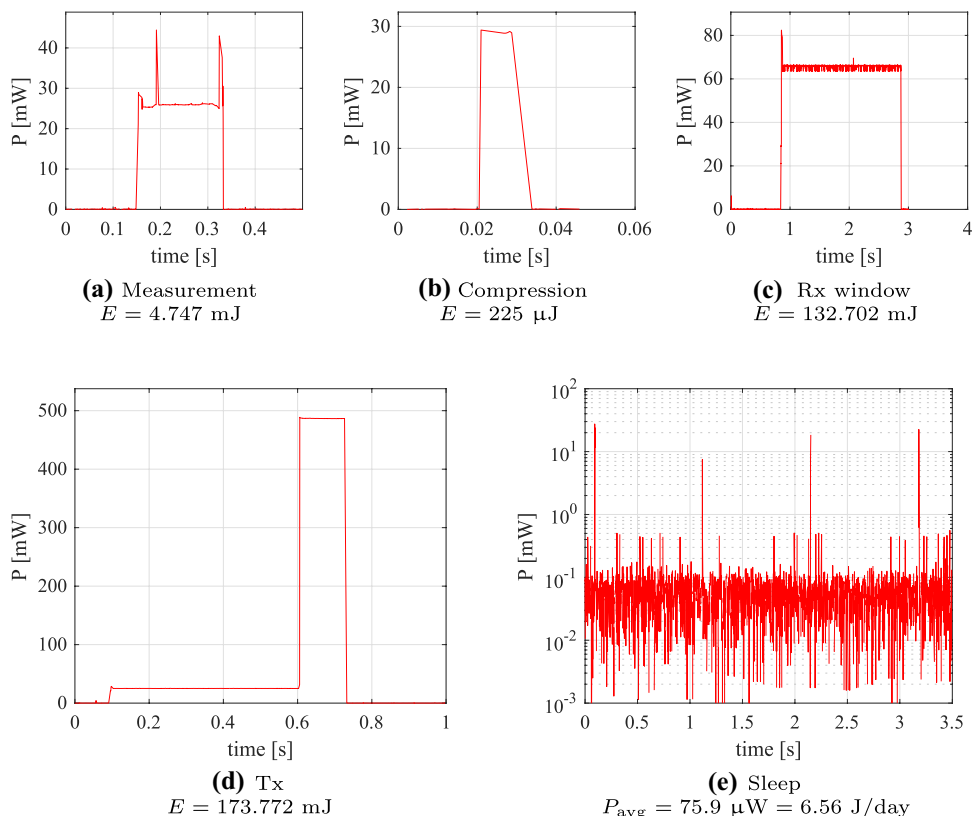
When we compare the different scenarios, it is clear that subsurface temperature data offers the greatest potential for compression. This can be attributed to the single data type (there is only thermal data and no acceleration

measurements along x, y, and z axes), and the low spatial and temporal variability that can be observed in Fig. 9. The compression ratios for snowpack temperature measurements are higher because of the strong diurnal variation in above ground temperatures as shown in Fig. 8, leading to a larger entropy. The results for subsurface temperature/deformation measurements indicate compression ratios as low as 0.608. Despite the low spatial and temporal temperature variability, the compression ratio is significantly higher than for the other scenarios. This can be linked to the fact that each measurement contains three 10 bits acceleration values, in comparison to a single 16 bits temperature value. Despite the highest compression ratios for the temperature/acceleration values, the impact of the compression algorithm is the most significant for this application. In a scenario without compression the entire data set would require 39,730 packet transmissions, equalling two transmissions per measurement. The compression algorithm reduces the number of packet transmissions to less than one per measurement, not only improving battery life, but ensuring the usability of the proposed network protocol. In related work [30], compression ratios between 0.48 and 0.62 are reported. The difference in data sets prevents a direct comparison of algorithm performance, but the results presented in this paper demonstrate the potential of spatial and temporal similarity for compression purposes.

### 5.2 Power Consumption

The power consumption of the nodes and base stations is a function of the presented hardware layout, communication protocol, compression algorithm, and acquired data. In order to assess the battery life of each device in the network, a power profile was recorded for each system state. For the calculation of these profiles, we measured the DC current along the VCC line under a constant voltage of 3.3765 V using a Keithley DMM6500 6.5 digit digital multimeter. Figure 11 presents the results of these measurements, along with the required energy for each event. The sleep state is characterized by an average power consumption  $P_{avg}$ , which takes into account the 1 Hz power spikes that are associated with BLE advertising. The presented results emphasize the impact of LoRa communications on a device’s energy budget. The aggregated data can not be reduced due to the nature of the application and network topology, but the presented data compression technique can reduce network traffic with a minimal energy impact of 225 $\mu$ J per compression, indicating the computational simplicity of the algorithm. The energy cost of a LoRa transmission is measured at 174 mJ, which means that the compression algorithm pays off when LoRa transmission can be reduced at least 1/772 (0.13%), a goal that is significantly exceeded as indicated in Table 3. As a result, trade-offs in the configuration of the compression

**Figure 11** Power profile for each system state, as presented in Fig. 2.



algorithm in our applications are solely related to data delays and potential impacts of packet loss, not energy cost.

The acquired power profiles are used to estimate the battery life for each configuration of the compression algorithm in Table 3. For these calculations we consider 15 min measurement intervals and we do not take user initiated BLE data transfers into account. The selected AA batteries provide an energy supply of 41,580 J [25]. Since the power consumption of base stations depends on the amount of network traffic [10] we assume a scenario of a single base station with 100 sensor nodes, all exhibiting the same compression ratios. The results in Table 3 demonstrate that thermal sensor nodes can last for > 4.5 years on a single pair of AA batteries without data compression. For temperature/deformation arrays ( $m = 19$ ), a battery life of 2.8 years can be expected, since each uncompressed measurement sample results in two LoRa packet transmissions. LoRa transmissions account for 70% of the energy, the sleep state is responsible for 27%, and sensor measurements use 2%. This indicates the potential benefit of the compression algorithm. As can be seen in Table 3 a node's battery life can even be doubled under favorable conditions (subsurface temperature sensors with  $t = 8$ ). For base stations, the potential battery life improvement is even more significant. For the scenario without data compression the battery life is only one month in networks of temperature sensors, or 16 days in networks of temperature/deformation sensors. For base stations, 98% of the energy budget is associated with the Rx state. This can be improved considerably if all sensor nodes in the network perform data compression. In the case of subsurface temperature arrays, base station battery life can be extended with a factor of up to 3.50. This enables autonomous operation in remote field sites for multiple months or even a year for base stations that use a small solar panel or multiple pairs of AA batteries. One can remark that the compression algorithm has a stronger impact on the battery lifetime of the base stations than the sensor nodes. This is explained by the higher relative importance of the sleep state in a sensor node's power budget, whereas base station power consumption is almost solely attributed to the Rx state. In [29], Väänänen et al. present a LoRaWAN sensor platform with various lossy data compression methods. The platform exhibits a  $467\mu\text{W}$  sleep power consumption (i.e. 6 times more than the platform presented in this paper), which results in a total battery lifetime of 492 days without compression. The best performing compression algorithm realizes a power savings factor of 1.28 (versus 1.99 in our research). Another solution is presented in [13], which reports a power savings factor of 1.45 for a lossless compression method. In related work, the power consumption of an iC880a LoRaWAN concentrator is reported. Consuming at least 1.44 W, this would result in a battery life of less than 8 hours when operating on a couple of AA batteries, which is significantly lower than any scenario presented in this paper.

## 6 Conclusions

Distributed environmental sensor arrays for above and below surface sensing require specific provisions in terms of power and connectivity. In this study, we presented a framework that covers the network protocol, a data compression technique, and a hardware platform that can be used as a wireless base station or a data logger for environmental sensor arrays. The presented wireless interface uses LoRa in a TDD implementation, enabling battery operated base stations and eliminating limitations concerning package collisions and fair use policies. A lossless data compression method was developed, relying on spatial and temporal similarity in the acquired data. This algorithm was implemented in a custom AA battery powered hardware platform and the power consumption was analyzed for various scenarios. As a wireless sensor node, the device exhibits a battery life of > 4.5 years, which can even be doubled with the proposed compression algorithm. As a LoRa base station, the platform's battery life is limited to several weeks or months, depending on the network configuration. However, the proposed compression algorithm can increase the battery lifetime with a factor of up to 3.50, making environmental distributed sensor deployments a viable option. In future work, we will further investigate the impact of measurement sampling intervals, the number of nodes, the number of sensors per node, etc. We will also develop relay nodes and deploy a sensor network in a mountainous watershed to study performance as a function of topography, vegetation, and antenna siting.

**Acknowledgements** This work was supported by the U.S. Department of Energy, Office of Biological and Environmental Research, Next-Generation Ecosystem Experiments-Arctic Project (NGEE-Arctic), and the Laboratory Directed Research and Development Program of Lawrence Berkeley National Laboratory under U.S. Department of Energy Contract No. DE-AC02-05CH11231.

**Availability of Data and Materials** The data presented in this study are openly available in the Next Generation Ecosystem Experiments Arctic Data Collection at <https://doi.org/10.5440/1907397>

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Ho, C. K., Robinson, A., Miller, D. R., & Davis, M. J. (2005). Overview of sensors and needs for environmental monitoring. *Sensors*, 5(1), 4–37. <https://doi.org/10.3390/s5010004>

2. Strachan, S., Kelsey, E. P., Brown, R. F., Dascalu, S., Harris, F., Kent, G., Lyles, B., McCurdy, G., Slater, D., & Smith, K. (2016). Filling the Data Gaps in Mountain Climate Observatories Through Advanced Technology, Refined Instrument Siting, and a Focus on Gradients. *Mountain Research and Development*, 36(4), 518–527. <https://doi.org/10.1659/MRD-JOURNAL-D-16-00028.1>
3. Cable, W. L., Romanovsky, V. E., & Jorgenson, M. T. (2016). Scaling-up permafrost thermal measurements in western alaska using an ecotype approach. *The Cryosphere*, 10(5), 2517–2532. <https://doi.org/10.5194/tc-10-2517-2016>
4. Léger, E., Dafflon, B., Robert, Y., Ulrich, C., Peterson, J. E., Biraud, S. C., Romanovsky, V. E., & Hubbard, S. S. (2019). A distributed temperature profiling method for assessing spatial variability in ground temperatures in a discontinuous permafrost region of alaska. *The Cryosphere*, 13(11), 2853–2867. <https://doi.org/10.5194/tc-13-2853-2019>
5. Ramesh, M. V. (2014). Design, development, and deployment of a wireless sensor network for detection of landslides. *Ad Hoc Networks*, 13, 2–18. <https://doi.org/10.1016/j.adhoc.2012.09.002>
6. Intrieri, E., Gigli, G., Mugnai, F., Fantì, R., & Casagli, N. (2012). Design and implementation of a landslide early warning system. *Engineering Geology*, 147–148, 124–136. <https://doi.org/10.1016/j.enggeo.2012.07.017>
7. Liberg, O., Sundberg, M., Wang, E., Bergman, J., & Sachs, J. (2017). *Cellular internet of things: Technologies, standards and performance*. Oxford, UK: Academic Press.
8. Mekki, K., Bajic, E., Chaxel, F., & Meyer, F. (2018). Overview of cellular LPWAN technologies for IoT deployment: Sigfox, LoRaWAN, and NB-IoT. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (percom Workshops)* (pp. 197–202). IEEE.
9. Sornin, N., & Yegin, A. (2017). Lorawan® specification v1.1. Technical report, LoRa Alliance.
10. Wielandt, S., & Dafflon, B. (2020). A local LoRa based network protocol with low power redundant base stations enabling remote environmental monitoring. In *2020 54th Asilomar Conference on Signals, Systems, and Computers* (pp. 520–523). <https://doi.org/10.1109/IEEECONF51394.2020.9443344>
11. Wielandt, S., & Dafflon, B. (2021). Minimizing power consumption in networks of environmental sensor arrays using TDD LoRa and delta encoding. In *2021 55th Asilomar Conference on Signals, Systems, and Computers* (pp. 318–323). <https://doi.org/10.1109/IEEECONF53345.2021.9723227>
12. Niles, K., Ray, J., Niles, K., Maxwell, A., & Netchaev, A. (2021). Monitoring for analytes through LoRa and LoRaWAN technology. *Procedia Computer Science*, 185, 152–159.
13. Săcăleanu, D., Popescu, R., Manciu, I., & Perișoară, L. (2018). Data compression in wireless sensor nodes with LoRa. In *2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)* (pp. 1–4). IEEE.
14. Dafflon, B., Wielandt, S., Lamb, J., McClure, P., Shirley, I., Uhlemann, S., Wang, C., Fiolleau, S., Brunetti, C., Akins, F. H., Fitzpatrick, J., Pullman, S., Busey, R., Ulrich, C., Peterson, J., & Hubbard, S. S. (2022). A distributed temperature profiling system for vertically and laterally dense acquisition of soil and snow temperature. *The Cryosphere*, 16(2), 719–736. <https://doi.org/10.5194/tc-16-719-2022>
15. Wielandt, S., Uhlemann, S., Fiolleau, S., & Dafflon, B. (2022). Low-power, flexible sensor arrays with solderless board-to-board connectors for monitoring soil deformation and temperature. *Sensors*, 22(7). <https://doi.org/10.3390/s22072814>
16. Light R. A. (2017). Mosquitto: server and client implementation of the MQTT protocol. *Journal of Open Source Software*, 2(13).
17. Naik, N. (2017). Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. In *2017 IEEE International Systems Engineering Symposium (ISSE)* (pp. 1–7). IEEE.
18. Naqvi, S. N. Z., Yfantidou, S., & Zimányi, E. (2017). Time series databases and influxdb. Studienarbeit, Université Libre de Bruxelles 12.
19. Beermann, T., Alekseev, A., Baberis, D., Crépe-Renaudin, S., Elmsheuser, J., Glushkov, I., Svatos, M., Vartapetian, A., Vokac, P., & Wolters, H. (2020). Implementation of atlas distributed computing monitoring dashboards using InfluxDB and Grafana. In *EPJ Web of Conferences* (Vol. 245, p. 03031). EDP Sciences.
20. AloS, A., Jiazi, Y., Thomas, C., et al. (2016). A study of LoRa: Long range & low power networks for the internet of things [j]. *Sensors*, 16(9), 1466–1475.
21. Part, F. R. (1997). 15.247: Operation within the bands 902-928 MHz, 2400-2483.5 MHz, and 5725-5850 MHz. Part15: Radio Frequency Devices.
22. Sx1276/77/78/79 datasheet rev. 7. (2020). Technical report, Semtech Corporation.
23. Gomez, C., Oller, J., & Paradells, J. (2012). Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. *Sensors*, 12(9), 11734–11753. <https://doi.org/10.3390/s120911734>
24. TMP117 high-accuracy, low-power, digital temperature sensor with smbus and i2c-compatible interface. (2021). Data sheet, Texas Instruments.
25. Crell, C. (2018). *Product data sheet - Energizer L91 Ultimate Lithium (191gl1218)*. Data sheet, Energizer.
26. Thoen, B., Callebaut, G., Leenders, G., & Wielandt, S. (2019). A deployable LPWAN platform for low-cost and energy-constrained IoT applications. *Sensors*, 19(3). <https://doi.org/10.3390/s19030585>
27. Callebaut, G., Leenders, G., VanMulders, J., Ottoy, G., DeStrycker, L., & Vander Perre, L. (2021). The art of designing remote IoT devices—technologies and strategies for a long battery life. *Sensors*, 21(3). <https://doi.org/10.3390/s21030913>
28. Yanhua, H., & Zhang, X. (2016). Aggregation tree based data aggregation algorithm in wireless sensor networks. *International Journal of Online and Biomedical Engineering (iJOE)*, 12(06), 10–15. <https://doi.org/10.3991/iJOE.v12i06.5408>
29. Väänänen, O., & Hämäläinen, T. (2021). LoRa-based sensor node energy consumption with data compression. In *2021 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0 & IoT)* (pp. 6–11). IEEE.
30. Hanumanthaiah, A., Gopinath, A., Arun, C., Hariharan, B., & Murugan, R. (2019). Comparison of lossless data compression techniques in low-cost low-power (LCLP) IoT systems. In *2019 9th International Symposium on Embedded Computing and System Design (ISED)* (pp. 1–5). IEEE.
31. Borda, M. (2011). *Fundamentals in Information Theory and Coding*. Berlin, Germany: Springer.
32. 3-axis,  $\pm 2\text{ g}/\pm 4\text{ g}/\pm 8\text{ g}/\pm 16\text{ g}$  digital accelerometer - ADXL345 rev. e. (2015). Data sheet, Analog Devices.
33. Hubbard, S. S., Williams, K. H., Agarwal, D., Banfield, J., Beller, H., Bouskill, N., Brodie, E., Carroll, R., Dafflon, B., Dwivedi, D., Falco, N., Faybishenko, B., Maxwell, R., Nico, P., Steefel, C., Steltzer, H., Tokunaga, T., Tran, P. A., Wainwright, H., & Varadharajan, C. (2018). The East River, Colorado, watershed: A mountainous community testbed for improving predictive understanding of multiscale hydrological-biogeochemical dynamics. *Vadose Zone Journal*, 17(1). <https://doi.org/10.2136/vzj2018.03.0061>
34. Dafflon, B., Wielandt, S., Lamb, J., McClure, P., Shirley, I., Uhlemann, S., Wang, C., Fiolleau, S., Brunetti, C., Akins, F. H., Fitzpatrick, J., Pullman, S., Busey, R., Ulrich, C., Peterson, J., & Hubbard, S. S. (2022). A distributed temperature profiling system for vertically and laterally dense acquisition of soil and snow temperature. *The Cryosphere*, 16(2), 719–736. <https://doi.org/10.5194/tc-16-719-2022>