Rural Communication in Outage Scenarios: Disruption-Tolerant Networking via LoRaWAN Setups

Franz Kuntke
Technical University of Darmstadt, PEASEC
kuntke@peasec.tu-darmstadt.de

Lars Baumgärtner
Technical University of Darmstadt, STG
baumgaertner@cs.tu-darmstadt.de

Christian Reuter
Technical University of Darmstadt, PEASEC
reuter@peasec.tu-darmstadt.de

ABSTRACT

Since communications infrastructure is subject to many impacts, e.g., destructive natural events, it can potentially collapse at any time. Especially in rural areas, the recovery of public network infrastructure can take some time, so a dedicated communication channel would be advantageous. We explore the possibility of transforming commodity LoRaWAN gateways into meshed network nodes for a digital emergency communication channel. In order to obtain the required parameters, we collected farm locations in Germany with OpenStreetMap. Based on the assumptions of LoRa communication range and considering our use case requirements, connecting farm communities seems theoretically feasible in many areas of our data set. To further analyze our idea, we ran simulations of two common DTN routing protocols with different scenarios. A proof-of-concept implementation allows smaller messages to be transmitted using real hardware and demonstrates that a decentralized communications infrastructure based on commodity hardware is possible.

Keywords: *Disaster Communication, Disruption-Tolerant Networking, Bundle Protocol Version 7, LoRaWAN gateways, LoRa Mesh Communication*

INTRODUCTION

The ability to communicate over long distances using technical devices is of great importance to modern society. In agriculture, communication plays a critical role when multiple farmers rely on shared labor and equipment to harvest cropland within tight time windows. Although easy to overlook, it should be noted that agriculture is generally considered a critical infrastructure with the responsibility to produce the required amount of food to sustain people's basic livelihoods. Serious efforts should be made to strengthen technology for this sector in many ways, as the technologies used in this sector are said to have comparatively poor resilience capacities (Kuntke, Linsner, et al. 2022).

Currently, the terms Agriculture 4.0 and Smart Farming are used to highlight several developments towards automated data generation and exchange between different stakeholders in the entire food production chain, by incorporating current trends in Information Technology (IT), such as the Internet of Things (IoT) and Cloud Computing (Rose and Chilvers 2018). As a logical consequence, the continuation of the vision of field robots also results in an increased need for communication between devices, such as autonomous vehicles, weather stations, sensors, and actuators. In order to meet the increasing demand for data exchange, while at the same time ensuring energy efficiency, so-called Low Power Wide Area Networks (LPWANs) have been established in certain areas of application, e.g., to connect a large number of sensors. A prominent representative of this technology category is the Long Range Wide Area Network (LoRaWAN), which allows the development of autarkic IoT networks. As already described, not only machinery depends on communication, but also farmers require reliable line communication between each other. An exemplary use case is the bundling of labor and machinery of neighboring actors during harvest. This use case is particularly important for efficient agriculture in small-structured agricultural regions, as e.g. in Germany.

In the event of major internet outages – which are not unlikely (Grandhi et al. 2020), although their duration and extent cannot be predicted – basic data exchange would still be possible in self-established LoRaWAN networks. This leads to the idea of using this technology in crisis situations – especially when the general communications infrastructure is broken. The possibility to change the usual LoRaWAN star-of-star topology to build multi-hop networks has already been investigated in various works (Centelles et al. 2021). Promising approaches utilize Disruption-Tolerant Networking (DTN) to increase the success rate of delivering messages in crisis situations with rather unpredictable networking resources (Baumgärtner et al. 2020). Two downsides, however, to the approaches most commonly described in literature are (1) the incompatibility with default LoRaWAN networks, leading to devices that only have the single-purpose of crisis-communication, and (2) the requirement of custom firmware for most developer devices, making the approaches hard to use for IT-laypeople. But as the distribution of LoRaWAN hardware increases, especially in the domain of agriculture, we can see the benefit of enhancing the software stack behind a commodity LoRaWAN gateway to allow messages to be exchanged between neighboring farms up to several kilometers apart. This approach would connect rural communities that have LoRaWAN hardware for common IoT applications in the event of a crisis. Since no *expert* hardware will be needed, the approach can be made to work with just installation of our software addition - which at best is already running in the background before a crisis event - and can thus be more inclusive than other approaches.

This core question of this work is therefore: *How can LoRaWAN-based IoT setups be utilized to allow DTN-based peer-to-peer communication?* As part of our work, we make the following contributions:

- A novel tool[1] for calculating geographic statistics for wireless network planning based on OpenStreetMap data
- A concept that allows to send/receive payloads in a LoRaWAN-conform manner via commodity LoRaWAN gateways, along with a prototypical implementation
- An evaluation of the concept through simulations of 40 farm neighborhoods in two scenarios, comparing performance of two DTN routing mechanisms
- A novel software library `chirpstack_gwb_integration`[2] as a companion to ChirpStack LoRaWAN Network Server, working with commodity hardware allowing to send/receive arbitrary payloads in a LoRaWAN-conform manner
- A novel software `spatz`[3] that builds a DTN routing, utilizing `chirpstack_gwb_integration`

The developed tools and evaluations were conducted with the application area of agriculture in mind, but can also be transferred to other areas – especially where IoT technology is already being used.

BACKGROUND

In this section, a brief overview of LoRa, LoRaWAN, DTN and the DTN Bundle Protocol are given and related work in the field of adapting LPWAN technologies is presented.

LoRaWAN and LoRa

LoRaWAN was standardized by the LoRa Alliance 2015. LoRaWAN is a popular LPWAN technology that adjusts and modulates signals using an exclusive proprietary spread spectrum technology in the sub-GHz-ISM band. The physical layer of LoRaWAN is LoRa, which stands for *Long Range*. LoRa operates in the unlicensed ISM band (e.g., in Europe 433/868 MHz, in North America 915 MHz). Depending on the region, a duty cycle regulation may apply, for example 1% in Europe for 868 MHz. As shown by Vejlgaard et al. 2017, interference issues are possible when the unlicensed bands are widely used in an area. The level of such interference issues is expected to grow with the deployment of more wireless IoT solutions. However, this problem mainly concerns urban deployments, while in this work we focus on rural areas, especially agricultural areas.

LoRa works with Chirp Spread Spectrum (CSS) as a modulation type. A coding rate indicates the rate of the Forward Error Correction (FEC), whereby the value 4/5 is used for standard LoRa frames. LoRa allows for six

[1] https://github.com/PEASEC/distance-statistics
[2] https://github.com/PEASEC/LoRaWAN-DTN/tree/main/chirpstack_gwb_integration
[3] https://github.com/PEASEC/LoRaWAN-DTN/tree/main/spatz

different spreading factors (SF7 to SF12) to balance the signal scattering factor (and thus the range), the data rate, and the energy consumption. The spreading factors define the number of symbols. A sinusoidal signal sequence or transmission pulse is referred to as a symbol. The number of bits that can be represented by a symbol corresponds to the SF. The maximum payload (`MACPayload`) capacity is 250 bytes (LoRa Alliance Technical Committee Regional Parameters Workgroup 2021).

The LoRaWAN standard uses LoRa as a transmission technology (with predefined settings on code rate, SF, bandwidth) and defines the used architecture, as well as LoRaWAN-compliant devices. A LoRaWAN setup is a stars-of-stars topology, with $1..n$ end-devices transmitting data (encapsulated into LoRa frames) to $1..m$ gateways, which itself are connected (via IP) to a single network server. For different regions, different specific transmission preferences exist, which are allowed and respect the local free ISM bands. LoRaWAN itself allows for different data rate configurations, that are a combination of SF and bandwidth, depending on the regional parameters (LoRa Alliance Technical Committee Regional Parameters Workgroup 2021). For Europe (SRD860, 863-870MHz), data rate 0 is the long range configuration with SF 12 and 125 kHz bandwidth, and data rate 6 is the fastest LoRa transmission configuration, with SF 7 and 250 kHz bandwidth. Of course, a wireless data transmission technology is also subject for security attacks, and despite the fact that LoRaWAN also takes into account several security aspects in the protocol design, there is still a known attack surface that should be taken into account when developing and deploying IoT systems based on this technology (Kuntke, Romanenko, et al. 2022).

Disruption-Tolerant Networking

Disruption-Tolerant Networking (DTN), also called Delay-Tolerant Networking, receives increasing attention for various applications, as it allows for a resilient and flexible data exchange in challenging network conditions. DTN solutions are commonly based on a *store, carry, and forward*-approach. Here, network participants act as *data-mules* and physically carry around and opportunistically exchange data with other nodes encountered. Therefore, it is not suitable for real-time applications such as video conferencing or other applications that require a stable end-to-end connection, but provides robustness and fault tolerance for applications that can tolerate delays in data dissemination, e.g., messaging, sensor data, or file sharing. Here, one area of application is disaster communication, in case of network infrastructure outages, e.g. after natural disasters (Setianingsih et al. 2018; Zobel et al. 2022). The Bundle Protocol Version 7 (BP7) is the most recent Internet Engineering Task Force (IETF) standard (Burleigh et al. 2022) for such a DTN architecture. Additionally, different routing algorithms, e.g., epidemic routing or Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) (Lindgren et al. 2012) can be used for distributing the bundles. This enables optimization for various properties such as fast/reliable bundle delivery or a minimum number of duplicates in the network. Besides advanced routing decisions that take into account, for example, geographic locations (Cheng et al. 2010; Baumgärtner et al. 2020; Sánchez-Carmona et al. 2016), there also exist other metrics which affect data dissemination across different convergence layers, e.g., duty-cycle restrictions when using LoRa (Msaad et al. 2021) or the workload of the involved nodes (Wang et al. 2021; Zhang et al. 2013).

RELATED WORK: ADAPTING LPWAN

Previous research approaches have already investigated multi-hop networks using LPWANs. For example, Abrardo and Pozzebon 2019 describe a LoRa network where the network topology is changed to bridge the route to the gateway through other nodes. The resulting sensor network based on LoRa was used to perform measurements in an underground environment that only allows for a maximum range of 200m. Zguira et al. 2018 utilize a 802.11p-based multi hop network to transmit sensor data of shared bikes to base stations.

Other publications, such as Abrardo, Fort, et al. 2019 or Dias and Grilo 2018 are concerned with increasing the range of the network while simultaneously saving the energy of the end devices by reducing the necessary transmission power by shortening the distance to the receiver, as the receiver is the closest sensor. To realize this, they rely on multi-hop networks. Furthermore, other contributions, such as the work of Ebi et al. 2019, describe using multi-hop LoRa-networks in other range-critical situations. Instead of a star or linear topology, they are based on a mesh network topology. Further studies such as Lee and Ke 2018 and Huh and Kim 2019 describe the extension of the network's coverage through a mesh network. However, the data from sensor nodes is always forwarded to the base station (gateway) via other sensor nodes in order to expand large sensor networks.

Other work is investigating the use of LPWAN-based networking technologies to increase resiliency, e.g., in the form of a long-range wireless data channel for TCP/IP-based network hardware (Kuntke, Sinn, et al. 2021). Vigil-Hayes et al. 2022 describe a system that combines high bandwidth networks with LPWAN to extend internet coverage. *"The key idea behind this paradigm is that a useful set of service calls can be partially completed with limited data* *rate transfers and then fully completed when high bandwidth access is available."* (Vigil-Hayes et al. 2022, p.196). The transmission range of their test setup was only 400m with line-of-sight in an urban region, which could be due to the fact that rather small chips were used for LoRa transmission.. The aspect of addressed communication between two end nodes or end node and gateway is also addressed in some other works, but communication between two gateways is not intended. The protocol on the data link layer (OSI-layer 2) is modified and extended in some studies. A communication system cannot be implemented using the procedures described above. Such a system would be based on the physical layer and would require a replacement of the previously used protocol on the data link layer.

A related approach for bidirectional communication is the Serval Project (Gardner-Stephen 2011). The underlying purposes of Serval Mesh are crisis communication and the provision of basic mobile communication for low-income or isolated communities (Gardner-Stephen and Palaniswamy 2011). While being independent of further hardware, the Serval Mesh application utilizes the WiFi function of Android-driven smartphones. It features a *store, carry and forward*-architecture through which text messages, calls, and data transmissions are made available. Therefore, an advantage of this approach is that a cost-effective physical layer is created that is detached from local providers. However, the use of WiFi technology in the Serval Project entails the disadvantages of incompatibility issues and reduced range compared to LoRa (Gardner-Stephen and Palaniswamy 2011). To resolve the range limitation, inexpensive and weatherproof extenders which use UHF to allow for long-distance connections have been designed (Gardner-Stephen, Farouque, et al. 2017).

Höchst et al. 2020 connect smartphones via Bluetooth with LoRa capable micro-controller boards. A specific chat application allows the smartphone user to send SMS-like messages as LoRa signals to other users. The developed system allows for device-to-device communication with an experimentally evaluated range of up to 2.89 km.

Baumgärtner et al. 2020 describe a similar application that differs from our approach in several ways. Firstly, it is based on the scenario of immediate crisis communication in environments without any ICT, while our goal is to build a communication network that can serve as a substitute for internet-based communication also in the medium and long term, where previous existing ICT is damaged. Secondly, a major difference lies in the choice of hardware needed for the implementation, which is also rooted in the scenario choice: While Baumgärtner et al. 2020 have developed additional battery-powered, low-cost relay nodes and pager devices, our project aims at utilizing only already installed commodity LoRaWAN gateways for communication purposes. By this way, our system does not need specific actions regarding crisis prevention, but is just available for all farms that use LoRaWAN IoT technologies.

Therefore, the aim of our work is to design and implement a concept that enables addressed communication between LoRaWAN gateway hardware in a multi-hop network without internet access. In doing so, the advantages offered by the physical layer of LPWAN technologies are to be utilized. This concept is intended to ensure the resilient transmission of messages and to develop a communication system. The use case and exemplary scenario are presented in the following section.

USE CASE AND SCENARIO: EMERGENCY COMMUNICATION FOR AGRICULTURAL AREAS

Farmers in developed countries are increasingly adopting smart farming technologies involving IoT solutions. To our understanding, LoRaWAN has a high standing in this domain, probably due to low-cost sensors and low sequential costs. To build up resilience capacities regarding communication infrastructure in this domain, we see an opportunity to leverage the increasing adoption of LoRaWAN setups for a self-operated communication network. Such a communication network could be used for emergency communication over long distances when the landline and cellular network is broken. It could also help to organize the farmers' workforce in situations of prolonged internet connectivity outages, or allow neighborhoods surrounding of these farms to communicate with other nearby communities. We have three kinds of possible messages in mind that could be exchanged in crisis scenarios and that differ in their time priority:

Time-critical communication There are numerous reasons for a need for time-critical communication, e.g. a medical emergency. In the farming context, there is often a need to coordinate multiple neighboring actors that are required to combine their workforce during harvest within ideal time windows. Such messages are short, but should arrive in seconds rather than minutes.

Time-relaxed communication In emergency situations, there is also a need for regular communication between people in a local community. This involves transmission of small-sized data like messages, medium-sized data like photos or small audio-files, or large-sized data like videos. This data exchange is not considered to be highly time-sensitive, but should, of course, be transmitted as fast as possible.

Sensor-related communication For technology-driven farming that enables optimal use of resources like water, fertilizer, fuel, and electric energy, the analysis of recent environmental data is of great importance. However, typically not every farm has all kinds of sensor stations. This applies particularly to small, family-driven businesses, which are predominant in Europe. Therefore, such small farms in particular have a specific need for sensor data exchange, as this could provide necessary data without the financial burden of having to invest in multiple sensor stations. Especially for weather analyses, aggregated data from multiple neighboring regions, in the best case high-quality data from meteorological services, could be of high importance to improve a farm's overall efficiency. Such data is likely to be extensive, but not as time-sensitive as the other communication.

In the next section, we elaborate on the possibility of connecting farms via LoRaWAN, that has a reliable coverage of several kilometers, using Germany as an example.

FARM-TO-FARM DISTANCES

To have a first estimate about the feasibility of connecting neighboring farms via wireless communication technologies, we evaluated distances between farms. As we have no access to a farm address database (perhaps there is no such database), we have chosen to evaluate available data provided by the OpenStreetMap project and developed a tool for this purpose.

Querying and Processing OpenStreetMap Data

The objective is to determine whether the given distances that a wireless setup has to bridge between individual farms can be achieved by LoRaWAN. For this purpose, we developed a tool in python[4]. The tool's jobs can be roughly grouped into three parts:

1. retrieving: query and filter OpenStreetMap data
2. processing: calculate distance matrix
3. presenting: generate statistics and graphs

To retrieve farms, we were faced with the problem that OpenStreetMap has an inconsistent level of detail in the mapped data, especially when comparing rural and urban areas. Next to large cities, farms are often tagged very accurately (`building=farm`), even with the company name. In such cases, a query for farm buildings retrieved a superset of current farm businesses' buildings. In rural areas, however, farms are not often tagged as such, resulting in low recall performance, i.e., there are many non-retrieved farms. We chose to use the tag `landuse=farmyard`: *"Area of land with farm buildings (farmhouse, sheds, stables, barns, etc.)"*[5] and to filter empty areas. Using this tag provides a better approximation of current farm business areas (more relevant elements), but requires additional filtering of the retrieved data (also more false positives). Filtering is done based on the child elements of the farmyards. In case there is no building inside a farmyard, we omit this area as we are only looking for buildings. As some neighboring farmyards were obviously part of the same farm business – sometimes as a result of a complex polygon that was split, sometimes because a street splits an area – we decided to merge nearby (up to 300m distance from geometric center to center) areas. Even in case this merges multiple farm businesses, they might share their communication link in case of an emergency situation. In the last step of the retrieving part, we selected a random building on each of the remaining areas as a representative farmhouse that may contain IT equipment, including a LoRaWAN gateway.

The processing is a much more straightforward task. Based on the filtered farmhouses, we calculated center points for each farm. These centers allowed us to compile a distance matrix that takes into account the curvature of the earth by using `geopy.distance`. We embedded comfort functions to store intermediate results to continue a distance matrix creation, which can take several hours depending on the node count and computational resources. Based on the distance matrix, we evaluated two properties: (1) minimum distances between farms, and (2) count of neighboring farms in a range of $[1, 2, 3, 4, 5]\ km$. We took these ranges as assumptions for typical real-world coverage of LoRaWAN hardware, respecting our experience for typical deployment ranges, as well as literature (El Chall et al. 2019).

Presentation of the statistics is done by using `geopandas`, `folium`, and `matplotlib`. Embedded in an jupyter notebook file, the statistics allow for further data analysis.

[4] https://github.com/PEASEC/distance-statistics
[5] https://wiki.openstreetmap.org/wiki/Item:Q4877

Table 1. Count of retrieved farms ($N = 117{,}744$) that have at least n neighboring farms in a specific range.

range [km]	1	2	3	4	5
$n = 1$	96,229	112,662	116,419	117,301	117,580
$n = 2$	72,758	104,620	113,455	116,288	117,192
$n = 3$	52,974	95,979	109,450	114,550	116,438
$n = 4$	37,084	87,739	104,914	112,405	115,429
$n = 5$	25,263	80,133	100,247	109,826	114,093

Table 2. Results of different clustering parameters.

ϵ	1000 m			2000 m			3000 m		
minPts	3	5	7	3	5	7	3	5	7
count	7881	3462	1743	3345	1910	1401	1076	916	745
mean	10.68	16.35	19.69	32.73	52.12	63.83	107.71	122.79	143.87
std	163.76	188.38	68.78	836.33	1045.44	1059.23	2904.94	3058.33	2034.03
median	4	7	10	5	8	11	6	8	11
max	14,028	10,768	1863	46,643	44,396	38,674	95,328	92,615	50,657
noise	33,545	61,143	83,427	8252	18,199	28,318	1848	5265	10,563

Analysis of Retrieved Data

We ran the tool for all federal states of Germany as an example for a large industrialized European country. Contacting colleagues from different parts of Germany allowed us to verify retrieved data on a random basis and check the data quality for their local neighborhood. Comparing the data with the statistics of the agricultural sector in Germany, we find that the number of buildings we retrieved ($N = 117{,}744$) is only 45% of the registered agricultural businesses in 2021 ($N = 259{,}200$) (Statistisches Bundesamt (Destatis) 2021). To our understanding, this is mainly due the incompleteness of OpenStreetMap data, being a voluntary tool without the claim for 100% correct data. Taking this into consideration, our data represent a rather lower bound of possible application. Nonetheless, our data evaluation shows that the principal idea can work for some areas: As shown in Table 1, most of the detected farms ($N = 80{,}133$; 68%) have five or more neighboring farms within a 2 km radius, which is supposed to be a feasible range for LoRaWAN devices with good antennas in rural areas (El Chall et al. 2019).

We also clustered the buildings using DBSCAN (Ester et al. 1996), which creates groups such that each member of a group has at least one neighboring member in the same group with the maximum distance set. In Table 2 clustering results are presented for the parameter combinations ϵ with 1000m, 2000m, and 3000m, and $minPts$ with 3, 5, and 7. Figure 1 shows three plotted configurations with colored clusters. As can be seen, there are very large differences in the mean size of the clusters in our data set, ranging from 10.68 to 143.87. As expected, increasing the maximum Euclidean distance between two points (ϵ) reduces the noise points, i.e., the coverage of the data set by all clusters is higher.

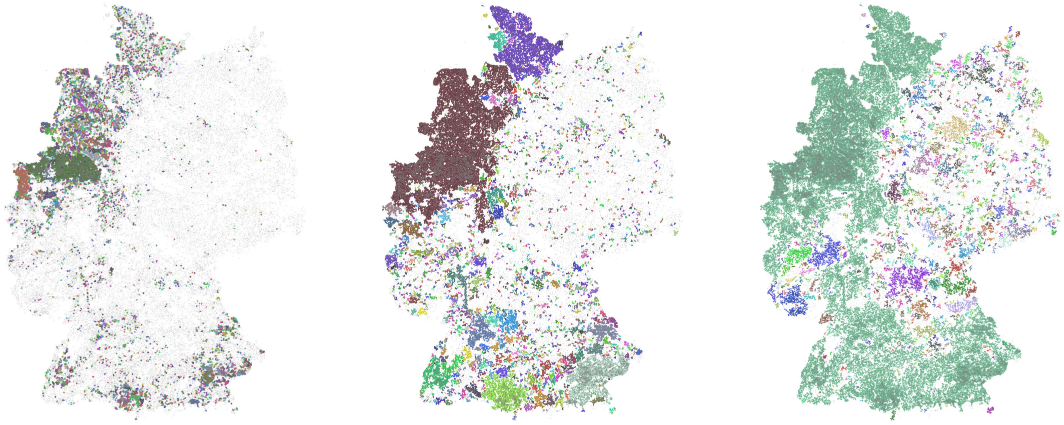
SIMULATION

Based on the results from the previous section Farm-to-Farm Distances, we see the opportunity of building networks that connect neighboring farms via LPWAN networks that could be used for small data exchange, e.g., messaging. In this section, we evaluate two DTN routing approaches by simulation using gathered real-world data from OpenStreetMap.

Setup

For simulation of the network approach we use the `ONE` (Keränen et al. 2009) DTN simulation software. We test two scenarios (Kuntke and Baumgärtner 2023): (1) complete static nodes using only LoRaWAN as a transmission channel, and (2) a mixed-mode with additional WiFi ad-hoc data exchange of mobile nodes.

The simulation is based on real geographic data extracted from the data generated as described in Section Farm-to-Farm Distances. For further processing we decided to use one of the previously described DBSCAN clusterings results with a moderate setting, i.e. $\epsilon = 2000$m and $minPts = 5$. The largest cluster of this data set has a size of 44,396 elements. As the scope of our work is the connection of farms and people inside a local community, we



(a) ϵ = 1000 m (b) ϵ = 2000 m (c) ϵ = 3000 m

Figure 1. Clustering results with maximum distance ϵ and at least five elements per cluster (*minPts* = *5*). Each element of a cluster is assigned a random color. All (including non-clustered) buildings are displayed as gray dots overlaid. With a point-to-point communication range of 2000 m or more, large parts of south and west of Germany could be covered.

Table 3. Statistics of the k-Means *post-processed* data set, used for picking simulation areas.

count	mean	std	median	max
2660	37.42	37.42	11	224

decided to further reduce the size of large clusters to better approximate the size of local communities. For this reason, we reduce each cluster c with $|c| > 100$ by using k-Means with $k = \lceil \frac{|c|}{100} \rceil$ to receive rather community-sized clusters. Figure 2 visualizes the resulting data set (base data set without k-Means visualized via Figure 1b), and table 3 presents the statistics. Based on the 95% confidence interval (35.74, 39.10), we decided to simulate 40 randomly chosen clusters, which should give us a good approximation of the complete data set.

The 40 selected clusters have an average size of 37.95 (std = 37.29). The largest chosen cluster includes 116 elements, and there are three clusters with the smallest size of 5 elements. For the mixed-mode scenario, we took mobile nodes into consideration. To have a more realistic simulation, we exported additional path geometries from OpenStreetMap to let our simulated pedestrians (mobile nodes) move on streets and ways. Figure 3 shows two exemplary clusters with their corresponding paths.

General Simulation Configuration

Both scenarios are simulated for all 40 clusters. The simulated time duration is 12 hours (43,200 seconds), with 0.05 second update intervals. Each cluster element is considered to be a static node, representing a farm. To evaluate the performance of two common DTN routing protocols for our scenario, we ran all configurations with both *PRoPHET* and *Epidemic* routing.

Scenario Related Configuration

We used the following settings for our two scenarios:

static A random static node sends a message to a random target within $\lfloor \frac{1800s}{|node|} \rfloor$. The message size is also random, between 80 and 500 Bytes. The LoRaWAN communication range is set to a maximum of 2000m, and the transmission speed is set to 7 kbps, which is between 5470 bps (data rate 5) and 11 kbps (data rate 6). These static nodes also have a WiFi interface; however, with a limited range of 100m, they are not used in the static scenario at all.

mobile Scenario *mobile* uses the same static nodes as scenario *static*, but adds mobile nodes representing pedestrians. We add as many mobile nodes as static nodes, i.e. one moving person is simulated per farm, traveling during the day. A random node (static or mobile) sends a message to a random target within $\lfloor \frac{3600s}{|node|} \rfloor$. The message

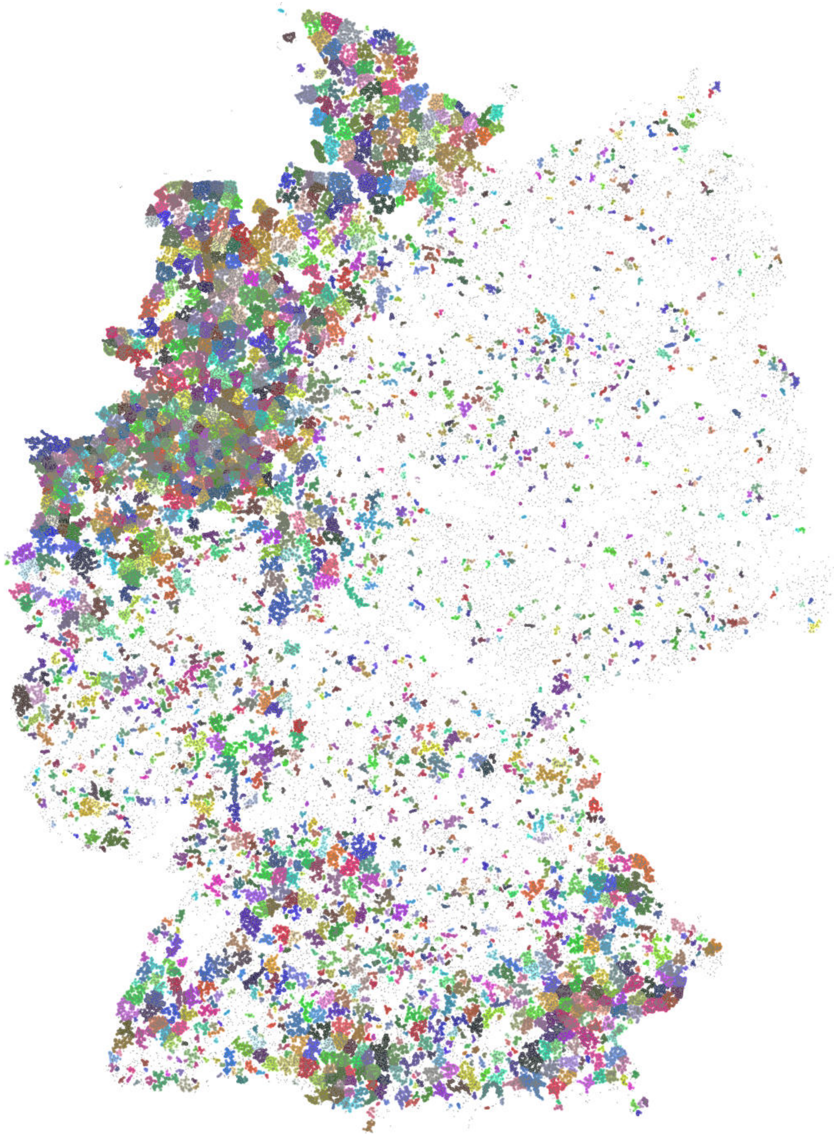
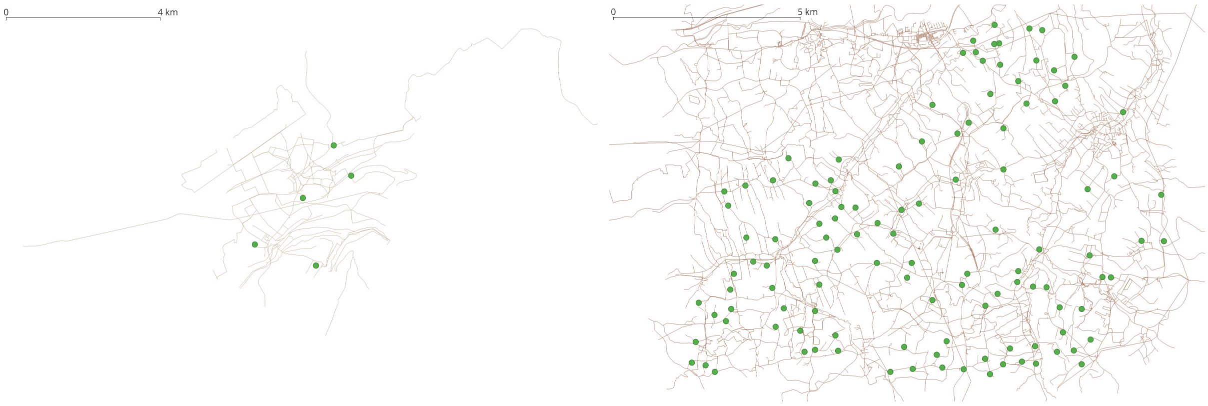


Figure 2. As we investigate on neighborhood communication, we reduced large clusters of our data set by k-Means to get more community-sized clusters for simulation.

size is also random, between 80 and 500 Bytes. These mobile nodes only have WiFi interfaces (smartphones) to exchange messages when in a range of 100m with another mobile node or with a static node. Each WiFi interface is set up with a transmission speed of 54 MBit/s.

Results

Statistics of the messages sent are presented in Table 4 and Table 5. The evaluation shows that in both scenarios, the flooding-based *Epidemic* routing achieves a higher delivery probability, at the cost of more routed messages. Figure 4 plots the message delivery rate over time for both scenarios. In *static*, *Epidemic* routing could deliver about 99% of the created messages. Both *PRoPHET* and *Epidemic* routing have an almost constant delivery probability after a few minutes. From the second hour, however, a gap builds up between *PRoPHET* and *Epidemic* routing. Interestingly, *static* with LoRaWAN-only communication achieves an overall higher delivery performance compared to *mobile*. From a technical point of view, this is obvious, since the mobile nodes must first come within WiFi reception range of another node. However, this could have practical implications: With regard to successful message delivery, it may make more sense to nudge users to rely less on mobile ad hoc connections via WiFi, and instead rely on static but connected LoRaWAN gateways.



(a) Cluster 7, one of the smallest clusters with 5 elements. **(b) Cluster 1, largest cluster with 116 elements.**

Figure 3. Static nodes (farms) overlaid on extracted OpenStreetMap paths, used for simulation of pedestrians.

Table 4. Message statistics of *static*. Mean over all 40 runs.

	created	started	relayed	delivered	delivery_prob	latency_avg [s]
EpidemicRouter	927.75	66,411.25	66,410.63	917.03	0.99	0.29
ProphetRouter	927.75	10,211.43	10,210.90	614.78	0.81	0.18

CONCEPT & IMPLEMENTATION

Based on the results from previous section Simulation, we see the opportunity of building networks that connect neighboring farms via LPWAN networks, that could be used for small data exchange, e.g., messaging. LoRaWAN itself offers a range of multiple km depending on the settings, hardware and geographic circumstances. In this section, we describe our concept and the proof-of-concept implementation.

Concept

We assume a farm building contains a small server for the purpose of running management software, as well its own LoRaWAN network server, to be able to collect and process the data without limitations and running expenses. When considering the challenges of using LoRaWAN for our goal, we are faced with high airtime of up to nearly three seconds per frame, a duty cycle restriction for most region/band combinations (e.g., 1% in the EU within the 868 MHz band), low payload and potentially unavailable network nodes (e.g., powered-down gateways), and additionally also the typical wireless problem that transmissions can fail in practice for various reasons (e.g., high noise in the used frequency band). Then again, we achieved a potentially high transmission range of up to several km by extending an existing software ecosystem.

Communication via LoRaWAN Gateways

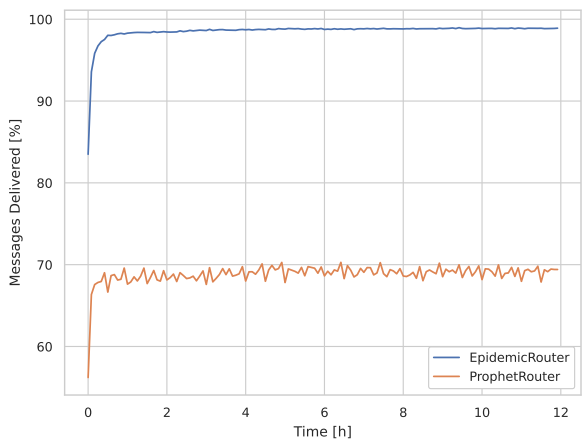
Our goal is to use neighboring LoRaWAN gateways to communicate with each other. A proxy is supposed to intercept the communication between the *LoRaWAN Network Server* and a gateway and forward our own frames to another processing pipeline. In this way we do not interrupt the IoT setup in its regular operations, but add our emergency communication layer on top. In our concept, the LoRaWAN network server software and our proxy are located on a physical mini server next to the gateway.

LoRaWAN Frames According to ISO/OSI

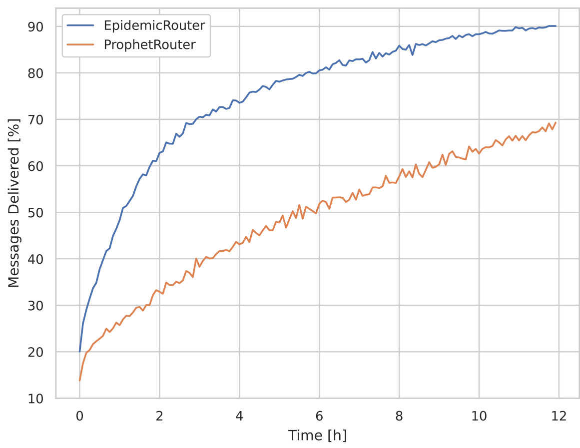
On the *Physical Layer*, we are bound to LoRa transmission, as we use off-the-shelf LoRaWAN gateways. On the *Data Link* and *Network Layer*, we differ from the plain LoRaWAN standard (LoRa Alliance Technical Committee

Table 5. Message statistics of *mobile*. Mean over all 40 runs.

	created	started	relayed	delivered	delivery_prob	latency_avg [s]
EpidemicRouter	927.75	123,095.88	123,095.40	835.18	0.88	3863.13
ProphetRouter	927.75	54,895.90	54,895.55	631.78	0.72	7326.52



(a) ***static***



(b) ***mobile***

Figure 4. Message delivery rates of both scenarios. Mean over all 40 runs.

2020) in that we use our own frames. However, we only differ in the fields `MACPayload` and `MIC` for our goal. In this way, we could be compatible with future LoRaWAN repeaters (LoRa Alliance Technical Committee 2020), which may enhance the usefulness of our approach. On the upper layers (*Transport, Session, Presentation*), we use BP7 (Burleigh et al. 2022) for message delivery allowing applications to send and receive bundles.

Routing Between End-Devices

The ability to send frames via a gateway with our proxy software, as well as receive and process frames sent by other gateways, allows us to integrate this into the bigger picture of creating a disruption-tolerant multi-hop communication network. For this purpose, we need a routing logic that processes bundles. In case of a received bundle there are two options: (1) the current gateway is the destination, meaning the bundle must be forwarded internally to an application/end-device; or (2) the bundle must be forwarded externally, meaning it has to be sent out by the gateway. By using the bundle protocol standard, we allow applications to exchange data through additional ways, e.g., via a smartphones' Bluetooth or WiFi. One important aspect is the address scheme. For this purpose, we use the phone number (E.123 notation) as the interplanetary network (IPN) endpoint identifier, as every user is expected to possess one main mobile device and a unique phone number. As we have a very limited payload, we use 4 bytes for the address by calculating the CRC-32 checksum of the phone number. The bundle itself is encoded as Concise Binary Object Representation (CBOR) (Bormann and Hoffman 2020), according to the standard.

Implementation

When inspecting the ChirpStack[6] (the de-facto standard open-source LoRaWAN network server), we saw that all necessary communication from and to gateways had already been converted into a message queuing protocol (MQTT), which we could use to read LoRa frames received by a gateway, as well as send messages via a LoRaWAN gateway. By sending commands to a gateway, we can specify the payload in our own way, respecting the limitations on maximum payload size depending on the set up data rate (SF and bandwidth). By doing this, we do not have to intercept the packet forwarder, as described in our concept, but we can do the same by just implementing a specific MQTT client, which reduces the complexity.

Due to the modular design, we separated the two functions: (1) being an MQTT client, reading LoRaWAN frames, and sending commands towards a gateway, and (2) parsing LoRaWAN frames into DTN bundles, containing routing logic, and connecting them with applications like our concept messenger app. We implemented[7] this (1) in Rust as a library, called `chirpstack_gwb_integration`, and (2) also in Rust as our convergence layer application `spatz` that uses `DTN7-RS`[8] as BP7 implementation. Additionally, we created a simple browser-based messaging client with VueJS that connects to a `spatz` instance via TCP/IP and allows us to send messages as an end-user. Figure 5 depicts the components of our proof-of-concept implementation.

[6]https://chirpstack.io
[7]https://github.com/PEASEC/LoRaWAN-DTN
[8]https://github.com/dtn7/dtn7-rs

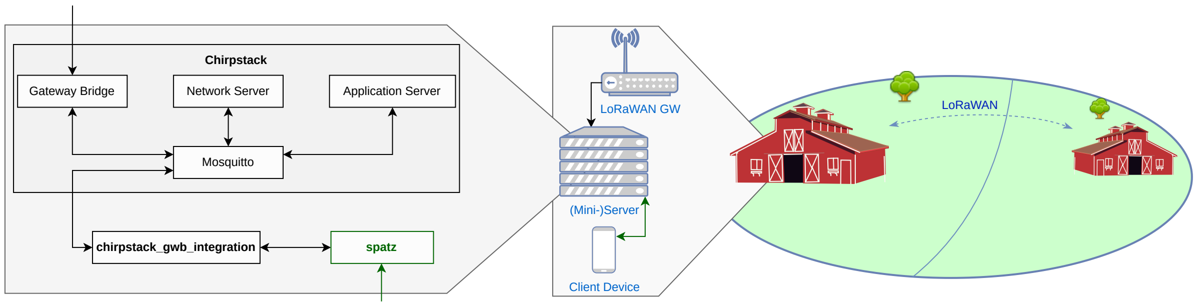


Figure 5. Technical concept: Regular LoRaWAN setup is extended by a LoRaWAN Packet Forwarder AddOn that allows to send and receive arbitrary LoRa(WAN) frames. The concept allows message exchange during network infrastructure outages.

Table 6. List of exemplary system component options with current prices (retrieved in January 2023).

Component Type	Name	Price
Mini-Server	Raspberry Pi 4 Computer Modell B, 4GB RAM Accessories (case, heat sink, power supply, 64 GB SDCard)	~€70 ~€26
Mini-Server	Intel NUC 8 Rugged Kit 4GB RAM, 64GB SSD	~€200
LoRaWAN Gateway (indoor)	Dragino LPS8N-868	~€160
LoRaWAN Gateway (indoor)	RAK 7268-N	~€180
LoRaWAN Gateway (outdoor)	Dragino DLOS8-868	~€320

chirpstack_gwb_integration The library's[9] main purpose is to be a Rust interface for directly interacting with a gateway, which is registered on a Chirpstack instance. The goal is not to interfere with the usual IoT setup of a LoRaWAN instance, but being able to independently send and receive LoRa frames via one or more connected LoRaWAN gateways. The library acts as a MQTT client and allows the creation of callbacks for incoming messages, as well as triggering `downlink` commands as outgoing LoRaWAN frames with specific transmission parameters like frequency and data rate, and payload.

spatz The main application[10] implements the bundle protocol convergence layer and the routing logic. It allows external user interfaces to connect to it by using websocket connections. `spatz` also handles the packet fragmentation, in cases a retrieved bundle could not be transmitted in one LoRaWAN frame. Due to the higher delivery probability in our simulation results, we decided to implement epidemic routing. For configuration settings, e.g. adding and deleting associated phone numbers (IPN endpoint identifier), `spatz` has a REST API.

Real World Setup

For our real world tests we use a Raspberry Pi 4, 4GB and three different LoRaWAN gateways: Dragino LPS8, Dragino DLOS8N, and RAK 7268-N. One node setup consists of a Linux server (e.g. Raspberry Pi 4) and one LoRaWAN Gateway (e.g. Dragino LPS8). Table 6 lists exemplary hardware costs. The cost of our evaluation setup hardware for one node starts at €256 (Raspberry Pi 4 + required accessories and a Dragino LPS8N-868). However, it should be kept in mind that these hardware requirements — at least a LoRaWAN gateway — are also necessary for regular LoRaWAN IoT setups, especially for farms that require long range and cost-effective wireless transmission of sensor data. We use Debian 12 as Raspberry Pi operating system, and Chirpstack v4 is installed according to the official *Quickstart Docker Compose* guide [11]. Our own software (`chirpstack_gwb_integration` and `spatz`) is compiled and executed directly on the Raspberry Pi. The browser-based messaging client is served by its own Docker container on the Raspberry Pi and allows it to be opened from a browser on a device (e.g. smartphone or laptop) on the same network. With this setup we were able to confirm the proper operation of our development with three nodes. As a limitation, we have not carried out a large-n scale test with real hardware, which will be part of follow-up research.

[9] https://github.com/PEASEC/LoRaWAN-DTN/tree/main/chirpstack_gwb_integration
[10] https://github.com/PEASEC/LoRaWAN-DTN/tree/main/spatz
[11] https://www.chirpstack.io/project/guides/docker-compose/

CONCLUSION

This work presents a novel approach for transforming commercial off-the-shelf LoRaWAN setups into DTN base stations for long range communication and looks into the feasibility of building communication networks in rural areas by leveraging these LoRa-DTN base stations located on farms. Current research has already shown how multi-hop networks based on LPWAN technology can be used to increase coverage (Abrardo and Pozzebon 2019; Ebi et al. 2019). Until now, the focus has mainly been on the data flow between the end device (e.g. the sensor) and the base station. We differ from the existing body of work on multi-hop communication and LPWAN improvements through our use case and design to provide support in crisis scenarios by DTN based message transmission. The existing approaches for extending IoT-communication described in the literature are not suitable for the design of a communication system that we focused on. We also differ by using commodity hardware from the existing works of LPWAN-based emergency communication technologies, as those rely on specific devices like self-made pagers or smartphone companion boards that might not being available in times of crisis event. By analyzing data from OpenStreetMap, we have obtained an approximation of positions of real farms in Germany. Even though the database is not complete, it gives a good indication of how well our idea could work in the European area if farms might enrich already existing LoRaWAN installations with our approach to be able to communicate across farms without external infrastructure in case of a crisis. Our simulation results have shown the feasibility, even if only LoRaWAN is in charge of message transmission. One possible application scenario for our development is to ensure the exchange of short messages during times of communication infrastructure failure in rural communities. Future work should identify the feasibility of concrete application cases under realistic conditions in order to prepare the technology for real crisis scenarios.

ACKNOWLEDGMENT

This work was supported by funds of the German Government's Special Purpose Fund held at Landwirtschaftliche Rentenbank in the projects Geobox-II and AgriRegio, as well as by the funds of the LOEWE initiative (Hesse, Germany) within the emergenCITY centre and the German Research Foundation (DFG) in the Collaborative Research Center (SFB) 1053 MAKI.

REFERENCES

Abrardo, A., Fort, A., Landi, E., Mugnaini, M., Panzardi, E., and Pozzebon, A. (2019). "Black Powder Flow Monitoring in Pipelines by Means of Multi-Hop LoRa Networks". In: *Proceedings of the Workshop on Metrology for Industry 4.0 and IoT*.

Abrardo, A. and Pozzebon, A. (2019). "A multi-hop LoRa linear sensor network for the monitoring of underground environments: the case of the Medieval Aqueducts in Siena, Italy". In: *Sensors* 19.2.

Baumgärtner, L., Lieser, P., Zobel, J., Bloessl, B., Steinmetz, R., and Mezini, M. (2020). "LoRAgent: A DTN-based Location-aware Communication System using LoRa". In: *Proceedings of the Global Humanitarian Technology Conference*.

Bormann, C. and Hoffman, P. E. (2020). *Concise Binary Object Representation (CBOR)*. Request for Comments 8949.

Burleigh, S., Fall, K., and Birrane, E. J. (2022). *Bundle Protocol Version 7*. Request for Comments 9171.

Centelles, R. P., Freitag, F., Meseguer, R., and Navarro, L. (2021). "Beyond the Star of Stars: An Introduction to Multihop and Mesh for LoRa and LoRaWAN". In: *Pervasive Computing* 20.2.

Cheng, P.-C., Lee, K. C., Gerla, M., and Härri, J. (2010). "GeoDTN+ Nav: geographic DTN routing with navigator prediction for urban vehicular environments". In: *Mobile Networks and Applications* 15.1.

Dias, J. and Grilo, A. (2018). "LoRaWAN multi-hop uplink extension". In: *Procedia Computer Science* 130.

Ebi, C., Schaltegger, F., Rüst, A., and Blumensaat, F. (2019). "Synchronous LoRa mesh network to monitor processes in underground infrastructure". In: *IEEE Access* 7.

El Chall, R., Lahoud, S., and El Helou, M. (2019). "LoRaWAN Network: Radio Propagation Models and Performance Evaluation in Various Environments in Lebanon". In: *Internet of Things Journal* 6.2.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*. Vol. 96. 34.

Gardner-Stephen, P., Farouque, S., Lloyd, M., Bate, A., and Cullen, A. (2017). "Piloting the serval mesh and serval mesh extender 2.0 in Vanuatu: Preliminary results". In: *Proceedings of the Global Humanitarian Technology Conference*.

Gardner-Stephen, P. (2011). "The Serval Project: Practical Wireless Ad-hoc Mobile Telecommunications". In: *Flinders University, Adelaide, Technical Report*.

Gardner-Stephen, P. and Palaniswamy, S. (2011). "Serval Mesh Software-WiFi Multi Model Management". In: *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*.

Grandhi, S. A., Plotnick, L., and Hiltz, S. R. (2020). "An Internet-less World? Expected Impacts of a Complete Internet Outage with Implications for Preparedness and Design". In: *Proceedings of the ACM on Human-Computer Interaction* 4.

Höchst, J., Baumgärtner, L., Kuntke, F., Penning, A., Sterz, A., and Freisleben, B. (2020). "LoRa-based Device-to-Device Smartphone Communication for Crisis Scenarios". In: *17th International Conference on Information Systems for Crisis Response and Management*.

Huh, H. and Kim, J. Y. (2019). "LoRa-based Mesh Network for IoT Applications". In: *Proceedings of the 5th World Forum on Internet of Things*.

Keränen, A., Ott, J., and Kärkkäinen, T. (2009). "The ONE Simulator for DTN Protocol Evaluation". In: *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*.

Kuntke, F. and Baumgärtner, L. (2023). *Simulation of 40 Selected German Farm neighborhoods with LoRaWAN-based DTN*. URL: https://doi.org/10.6084/m9.figshare.21082441.v1.

Kuntke, F., Linsner, S., Steinbrink, E., Franken, J., and Reuter, C. (2022). "Resilience in Agriculture: Communication and Energy Infrastructure Dependencies of German Farmers". In: *International Journal of Disaster Risk Science*.

Kuntke, F., Romanenko, V., Linsner, S., Steinbrink, E., and Reuter, C. (2022). "LoRaWAN Security Issues and Mitigation Options by the Example of Agricultural IoT Scenarios". In: *Transactions on Emerging Telecommunications Technologies* 33.

Kuntke, F., Sinn, M., and Reuter, C. (2021). "Reliable Data Transmission using Low Power Wide Area Networks (LPWAN) for Agricultural Applications". In: *Proceedings of the 16th International Workshop on Frontiers in Availability, Reliability and Security*.

Lee, H.-C. and Ke, K.-H. (2018). "Monitoring of large-area IoT sensors using a LoRa wireless mesh network system: Design and evaluation". In: *Transactions on Instrumentation and Measurement* 67.9.

Lindgren, A., Doria, A., Davies, E., and Grasic, S. (2012). *Probabilistic Routing Protocol for Intermittently Connected Networks*. Request for Comments 6693.

LoRa Alliance (2015). *LoRaWAN™ Specification V1.0*. URL: https://lora-alliance.org/resource_hub/lorawan-specification-v1-0/.

LoRa Alliance Technical Committee (2020). *LoRaWAN® L2 1.0.4 Specification (TS001-1.0.4)*. URL: https://lora-alliance.org/resource_hub/lorawan-104-specification-package/.

LoRa Alliance Technical Committee Regional Parameters Workgroup (2021). *RP002-1.0.3 LoRaWAN® Regional Parameters*. URL: https://lora-alliance.org/resource_hub/rp2-1-0-3-lorawan-regional-parameters/.

Msaad, M., Waleed, M., and Kosta, S. (2021). "Enabling LoRaWAN Communication with Out-of-coverage End Nodes in DTN Scenarios Through an Optimised Duty-cycle". In: *Proceedings of the 14th Critical ICT Infrastructures and Platforms International Conference*.

Rose, D. C. and Chilvers, J. (2018). "Agriculture 4.0: Broadening Responsible Innovation in an Era of Smart Farming". In: *Frontiers in Sustainable Food Systems* 2.

Sánchez-Carmona, A., Robles, S., and Borrego, C. (2016). "PrivHab+: A secure geographic routing protocol for DTN". In: *Computer Communications* 78.

Setianingsih, C., Nurjanah, R. S., Devi Gunawan, A., Nurjanah, R., and Murti, M. A. (2018). "ION-DTN based on UAV System for Emergency Communication During Natural Disaster". In: *Proceedings of the 21st International Symposium on Wireless Personal Multimedia Communications*.

Statistisches Bundesamt (Destatis) (2021). *Betriebsgrößenstruktur landwirtschaftlicher Betriebe nach Bundesländern*. URL: https://www.destatis.de/DE/Themen/Branchen-Unternehmen/Landwirtschaft-

Forstwirtschaft-Fischerei/Landwirtschaftliche-Betriebe/Tabellen/betriebsgroessenstruktur-landwirtschaftliche-betriebe.html (visited on 06/17/2022).

Vejlgaard, B., Lauridsen, M., Nguyen, H., Kovacs, I. Z., Mogensen, P., and Sorensen, M. (2017). “Interference Impact on Coverage and Capacity for Low Power Wide Area IoT Networks”. In: *Proceedings of the Wireless Communications and Networking Conference*.

Vigil-Hayes, M., Hossain, M. N., Elliott, A. K., Belding, E. M., and Zegura, E. (2022). “LoRaX: Repurposing LoRa as a Low Data Rate Messaging System to Extend Internet Boundaries”. In: *Proceedings of the Conference on Computing and Sustainable Societies*.

Wang, W., Bai, Y., Feng, P., Huang, J., Sha, M., and Tantai, J. (2021). “DTN-Balance: A Forwarding-Capacity and Forwarding-Queue Aware Routing for Self-organizing DTNs”. In: *Wireless Personal Communications* 118.1.

Zguira, Y., Rivano, H., and Meddeb, A. (2018). “IoB-DTN: A lightweight DTN protocol for mobile IoT applications to smart bike sharing systems”. In: *Wireless Days*. Dubai.

Zhang, S., Wu, J., and Lu, S. (2013). “Minimum Makespan Workload Dissemination in DTNs: Making Full Utilization of Computational Surplus Around”. In: *Proceedings of the Fourteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. Association for Computing Machinery.

Zobel, J., Kundel, R., and Steinmetz, R. (2022). “CAMON: Aerial-Ground Cooperation System for Disaster Network Detection”. In: *Proceedings of the 19th International Conference on Information Systems for Crisis Response and Management*.