



HAL
open science

Scalability of LPWAN for Smart City Applications

Renato Caminha Juaçaba Neto, Pascal Merindol, Antoine Gallais, Fabrice Theoleyre

► **To cite this version:**

Renato Caminha Juaçaba Neto, Pascal Merindol, Antoine Gallais, Fabrice Theoleyre. Scalability of LPWAN for Smart City Applications. International Wireless Communications and Mobile Computing Conference (IWCMC), 2021, Harbin (virtual), China. pp.74 - 79, 10.1109/IWCMC51323.2021.9498698 . hal-03241742v2

HAL Id: hal-03241742

<https://hal.science/hal-03241742v2>

Submitted on 16 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Scalability of LPWAN for Smart City Applications

Renato Caminha Juacaba Neto, Pascal Merindol
ICube Laboratory
CNRS / University of Strasbourg
Email: caminha,merindol@unistra.fr

Antoine Gallais
UPHF, France
Email: antoine.gallais@uphf.fr

Fabrice Theoleyre
ICube Laboratory
CNRS / University of Strasbourg
Email: theoleyre@unistra.fr

Abstract—Smart Cities rely on a broad variety of services generating an heterogeneous traffic load, and having diverse privacy and performance requirements. In this paper, we focus on mobility services for an Intelligent Transportation System (ITS), where several applications inter-operate by sharing information. In particular, we consider three main services: ride-hailing, smart parkings and traffic regulation. To investigate the load generated by these mobility services, we rely on two real datasets. Our goal is to study whether a given technology can scale with realistic conditions related to the envisioned smart mobility scenario. In particular, we analyze here how many LoRa gateways are necessary and how they should be deployed to support these mobility services. We highlight the bottlenecks occurring in most dense areas of the network and argue that heterogeneous deployments can efficiently handle such hot spots.

Index Terms—Smart Cities; Intelligent Transportation Systems; LPWAN; LoRa; scalability.

I. INTRODUCTION

The plethora of smart city services highlights the various benefits that Internet of Things (IoT) can bring to the population, companies and local authorities [1]. A collection of sensors is typically disseminated in a smart city, and monitors a large set of environmental parameters such as traffic speed, air quality, weather, noise, etc. These data streams help to make the city more reactive and resilient. One of the most promising services is about road traffic detection sensors to re-route vehicles through less congested roads and indicate them best spots to park.

Nowadays, smart city services are deployed as specific applications, that emerge independently like as many proofs of concepts (e.g., charging stations for electric vehicles, waste management) for the future smart cities. But overall, smart cities expect that the different applications cohabit with each other exchanging data. Indeed, developing a silo architecture is inefficient: applications are isolated and have to deploy their own sensors even if they process the same kind of data. Besides, such isolation has a cost [2]: the same radio resources have to be shared among a larger number of flows, which is particularly prejudicial in the ISM band.

While most deployments have demonstrated the interest of connecting a myriad of sensors and actuators, we aim to investigate the scalability of Low Power Wide Area Networks (LPWAN) to handle such services. It is absolutely necessary to avoid the tragedy of the commons [3]. Indeed, while each object has only a few packets to transmit infrequently, the competition for the medium access involves many devices.

LPWAN have already highlighted their limits when the load (aka *bit flux*) increases [4], limiting the amount of traffic that an application can generate.

In this paper, we investigate the scalability of LPWAN solutions, and more specifically LoRa, a leading technology for LPWAN, when considering a challenging smart mobility solution. We consider both mobile devices, in particular to monitor the actual speed through many road segments, and static devices used in smart parking. We also consider a ride hailing service co-existing in the same city. The contributions of this paper are as follows:

- 1) first, we define a global scenario for Intelligent Transport System (ITS): distinct applications share data to offer a globally efficient transportation system for smart cities;
- 2) then, we take advantage of two real and large cities datasets for our global system. That way, we can generate packets that mimics a realistic situation, with all the heterogeneity (e.g., downtown vs. suburbs) and temporality of large scale ITS. The traffic model we derive from these datasets is useful to assess the communication performances in real smart cities;
- 3) finally, to fully exploit our construct, we rely on a LoRa model inferring the Packet Error Rate (PER) of an arbitrary deployment of LoRa gateways; we thus estimate the scalability of such LPWAN solutions in realistic deployments. We highlighted the presence of hot spots, where the density of required LoRa gateways can be quite high to decently deal with some traffic peaks.

II. RELATED WORK

We detail here a few applications defined for smart cities, and how to characterize them. Then, we discuss the existing LPWAN scalability evaluations.

A. Service characterization

Realistic service definitions are crucial in order to assess suitability of proposals. It is common to characterize a service by the amount of messages generated by device and/or geographical area [5]. Fortunately, more and more open data projects have emerged [6] and allow us to design specific scenarios. Zebranet [7] represents an early example of such projects which has experimentally investigated the accuracy of Wireless Sensor Networks for wildlife monitoring. The Zebranet dataset has then been widely used, to assess the performance of protocols and algorithms. In Smart Cities,

Kong *et al.* [8] exploit a floating car dataset in New York City (NYC) to simulate a Vehicular Social Network.

However and unfortunately, most of these deployments focus only on a single service and, as such, remain silo proofs of concepts. Mixing different services having distinct requirements is a more interesting challenge and certainly represents most common scenarios to come.

B. LPWAN and Scalability

Low Power Wide Range technologies enable smart cities by providing the ability to connect a large number of spread devices required by these services. In particular, LoRa [9] is capable of reaching several kilometers of range and thousands of low power devices per base station. The deployment then looks easier: a few antennas, aka. LoRa gateways, are sufficient in theory to provide a large scale connectivity.

LoRa is based on a Chirp Spread-Spectrum (CSS) modulation technique, that uses pulses to encode information. More precisely, a pair of transmitter/receiver has to select a spreading code, also designated as a Spreading Factor (SF) in LoRa. By selecting the right SF, each device can choose the highest bit rate adapted to its signal strength. More importantly, the SFs are mutually orthogonal, to multiplex the transmissions. That is, transmissions on different SFs never collide and SFs have different ranges, air times and bit rates. However, handling a very large number of devices (even generating only a few packets each) with few antennas is challenging [4]. Collisions are frequent and network capacity may be rapidly reached.

Characterizing the services is not enough: one needs to estimate the reliability provided by the wireless infrastructure to be certain that it matches the requirements. Bankov *et al.* [10] provide a model that computes the PER based on the signal strength of all the devices, the amount of packets they generate, and the allocation of SF to each device. The capture effect can significantly impact the reliability of communications in large geographical areas [11], it should also be modeled to provide more realistic results.

III. INTELLIGENT TRANSPORTATION SERVICES

Intelligent Transportation represents a key enabler for smart cities. We consider here the following mobility services:

Ride hailing a platform connects clients with vehicle drivers (aka. on-demand taxi). In a smart city, this service could be provided locally.

Smart parking where sensors monitor occupied parking spots. Vehicles can then be guided to find an available parking spot, and are billed by the system on departure.

Traffic monitoring where the ITS measures the average speed for each road segment. This information helps for traffic regulation (e.g., vehicle routing problem).

Figure 1 illustrates our scenario, where the different applications and entities share information with each other.

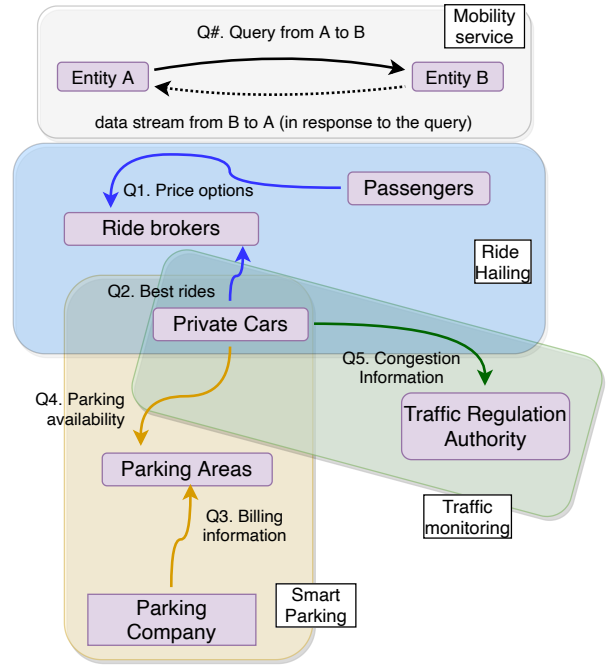


Fig. 1: Overview of services

A. Query Language

We rely on a slightly extended version of SQL to explicitly define the queries (i.e., the arrows in Fig. 1) that trigger the streams of data between entities (in the inverse direction of the queries). Furthermore, we map relational to IoT data by having data sources, samples and attributes acting as tables, rows and columns, respectively. We then simply add the following keywords in our query language extending SQL to fully describe the IoT streams:

COMPUTE indicates a list of computations to apply to the data in sequence.

EVENT describes condition(s) that may activate/deactivate a stream (anything, from local sensors to remote messages).

EVERY indicates when new data must be generated to the stream. Events or constant time intervals may apply.

UNTIL indicates the condition to stop the stream. Analogous to **EVERY**, events or specific duration may apply. Without this keyword, the query is assumed to hold indefinitely.

B. Ride Hailing

Recently, the portion of taxi trips being handled by private cars has been steadily increasing. A common platform serves as a broker between passengers and drivers, so that all the requests can be fulfilled. We detail here streams 1 and 2 illustrated in Fig. 1.

1) *Query 1 (ride selection for passengers)*: passengers request companies (i.e., brokers) for best quotes to perform a ride by specifying the time of departure, and the geographical location of departure and arrival. The reply data stream consists of a continuous query remaining active until the passenger has validated one of the proposed quotes. The data stream stops when the passenger accepts a quote.

```

1 SELECT pricing_type, cost_to($destination) as
   ride_cost
2 FROM ride_brokers
3 ORDER BY ride_cost DESC
4 LIMIT 5
5 EVERY minute
6 UNTIL EVENT client_decision

```

Query 1: Passenger request sent to a set of broker(s).

2) *Query 2 (ride selection for drivers)*: for the other part of the service, drivers must select their next ride. More precisely, the drivers trigger a continuous query to get unsatisfied rides when they are close to complete their current fare. We assume here that brokers implement a scoring function (*ride_score*) which ranks rides according to characteristics such as driver preferences, proximity, and expected payment. As soon as a driver accepts a quote, the corresponding query stops.

```

1 SELECT ride_begin, ride_end, estimated_price
2 FROM ride_brokers
3 ORDER BY ride_score(*) DESC
4 LIMIT 10
5 EVERY 5 minutes
6 UNTIL EVENT driver_decision

```

Query 2: A driver asks for the passenger requests.

C. Smart Parking Service

Searching for a parking spot is particularly expensive in terms of time and fuel [12]. Thus, smart parking solutions help drivers to find a parking spot, and to reduce their waiting time. We can also make the system more efficient, with an automatic billing solution.

1) *Query 3 (Smart Billing)*: A vehicle can be automatically billed when exiting its parking spot. We consider that a sensor is capable of detecting the occupation of one or multiple parking spots and then sends notifications whenever a vehicle enters or departs of a spot. This notification is sent to the company so that the vehicle can be charged accordingly. These notifications include what event occurred (arrival or departure), timestamp of the event, and identifications of vehicle and spot.

```

1 SELECT event_type, time_stamp, car_plate,
   parking_spot_id
2 FROM parking_parking
3 EVERY EVENT park_begin, park_end

```

Query 3: Tracking vehicles entrances.

2) *Query 4 (Guidance of cars to available parking)*: Drivers ask for the availability of nearby parking areas to find an available parking spot. A filter selects areas with a sufficient number of available spots (*THRESHOLD_VALUE*), else the available spots may be occupied by the time the driver arrives to the location. Then, results are ordered by proximity and limited to prevent verbose answers. The continuous query 4 holds from the time of departure to the time of arrival (end of trip), and the list is updated every minute.

```

1 SELECT parking_area_id, available_count, distance
   (arrival_location, parking_location) as
   geo_distance
2 FROM parking_areas
3 WHERE available_count > THRESHOLD_VALUE

```

```

4 ORDER BY geo_distance DESC
5 LIMIT 10
6 EVERY minute
7 UNTIL EVENT end_of_trip

```

Query 4: Parking availability.

D. Smart Road Traffic Routing Service

Traffic jams induce increasing costs in most major cities, both in terms of time, money and pollution¹. Not to mention the frustration caused to drivers and consumers and thus the ensuing damages it may lead to.

Vehicle routing solutions help drivers to find better routes [13] to mitigate these effects. It requires road surveillance in order to identify the level of congestion of each road segment.

1) *Query 5 (Real-time Road Speeds)*: A smart mobility service may collect the location and speed measurements from a large collection of vehicles (e.g., bus, private cars, taxis) to profile congestion in real time. Smart cities may exploit this information to make some interpolations and infer congestion in each road segment. The real identity of each vehicle should be hidden (e.g. with a hash) for obvious privacy concerns.

```

1 SELECT hash(car_plate), latitude, longitude,
   current_speed
2 FROM consenting_vehicles
3 EVERY 3 minutes

```

Query 5: Road congestion information.

IV. OPEN MOBILITY DATASETS

In this section, we detail how we exploit two public datasets to model the global smart mobility service we envision. We rely on this data for performance evaluation: can an ITS relying on LoRa gateways can scale in these realistic conditions?

A. Datasets Description

We exploited the two following datasets (cf. Table I).

Ride hailing: we use a dataset of rides performed by the companies Uber, Juno, Lyft and Via in NYC². Each sample describes the pick-up and drop-off information (i.e., timestamp and geographical location) of a trip. To respect privacy concerns, the city employs pre-defined zones and zone identifiers instead of exact geographical coordinates. We model here mobile sensors (i.e., vehicles);

Parking usage: the city of Melbourne provides measurements of parking sensors³. Each sample of this dataset describes individual occupations of parking spots in the city and includes timestamps of arrival and departures in addition to multiple identifiers of the parking spot. We have here static sensors with an event-based traffic.

¹<https://inrix.com/press-releases/2019-traffic-scorecard-us/>

²<https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page>

³<https://data.melbourne.vic.gov.au/Transport/On-street-Car-Parking-Sensor-Data-2019/7pgd-bdf2>

TABLE I: Datasets used in this study.

Service	Parking usage	Private-driver hailing
Location of sampling	Melbourne, Australia	New York City, USA
Number of samples	37.7 million samples	234 million samples
Sampled Period	Jan 2019 to Dec 2019	Feb 2019 to Dec 2019
Data available	Individual Parking occupations	Taxi rides provided by private cars

B. Dataset Preparation

Since the service usage described by these two datasets are deployed in different cities, we need to merge both to mimic a global service running in the same city. We map Melbourne’s dataset to NYC since more information is openly available for the latter. We gather a list of parking lots in NYC with the aid of the TomTom API⁴. Since we have more parking lots in Melbourne than in NYC, one-to-one mapping cannot be used. We first cluster the 4,876 parking lots in Melbourne into 1,212 clusters (the number of parking lots we found in NYC) using the k-means algorithm according to geographical location. With this, each cluster represents a virtual parking lot with the data from its members.

Now we can look for a parking bijection from Melbourne to NYC. Assuming that parking lots closer to Manhattan (commercial center of NYC) will have more demand, we map our virtual parking lots of Melbourne with most traffic into the parking lots of NYC which are closer to Manhattan.

To mimic vehicle displacement, we expand the NYC dataset by computing the trajectory between pick-up and drop-off locations for each trip. For the sake of simplicity, we consider here straight line trajectories under constant speeds. In our case, an approximated geographical location is sufficient to model the LoRa traffic.

C. Database-based Network Traffic Model

We infer the network traffic from the datasets (cf. Table II that summarizes the parameters used in our models):

1) *Ride Hailing*: we model both queries 1 and 2 by using the time and location information of taxi pick-ups and drop-offs in NYC. Each trip record emulates a query at time and location of pick-up and drop-off. We model the duration of the query with a normal distribution (see Table II).

2) *Smart Parking*: we model the `park_begin` and `park_end` events of queries 3 with the arrival and departure data observed in the Melbourne dataset. More specifically, for each record, at time of arrival and time of departure, one message is sent from the parking lot location.

We model the inter-query time of query 4 by assuming that there was a search for available parking every time a parking spot is taken. Since available parking can change constantly, one message is sent every minute until the car parks. For same reasons as before, we again rely on a normal distribution to model the time to find a parking spot (see Table II).

⁴<https://developer.tomtom.com/>

TABLE II: Service parameters.

Q	Parameter	Value
1	Periodicity	1 minute
	Passenger decision	Normal ($\mu=1, \sigma=1$) min
2	Periodicity	5 minutes
	Driver decision	Normal ($\mu=5, \sigma=3$) min
3	Event-based	One shot query
4	Periodicity	1 minute
	Limit	10
	Trip duration	Normal ($\mu=8, \sigma=3$) min
5	Periodicity	3 minutes
	Vehicles subset	10% of available cars

3) *Smart Road Traffic Routing*: in query 5, we consider that a subset of vehicles sends its speed information (e.g., a subset of vehicles is equipped). More precisely, we consider that only 10% of the dataset of taxi trips is considered by this sub-service. Messages are sent periodically along the trajectory we generated for each trip.

V. SCALABILITY ANALYSIS WITH A LORA DEPLOYMENT

We discuss here whether a wireless technology like LoRa could be used for such Smart Mobility services. We assume that LoRa will be the only means of communication.

A. LoRa Analysis Methodology

To conduct our scalability analysis, we consider the LoRa analytical model described in [10]: for a given amount of network traffic, it predicts the PER of each device associated with a LoRa gateway. This model measures collision probabilities for each SF from the number of devices tuned to it and assuming a Poisson process for packet generation⁵.

In our calculations we use the EU 863–880 MHz ISM band, with the receiver sensitivity of the Semtech SX1276 LoRa receiver. We aim here to understand how many gateways are required to ensure an efficient large-scale mobility service deployment with a reasonable PER.

We consider two different SF allocation methods:

Distance-based: some devices are too far from the LoRa gateway and we must consider their signal strength when selecting a SF. Thus, we allocate to each device the lowest SF with respect to its signal strength. A lower SF means that the bitrate is higher.

⁵All formal details are given in [10].

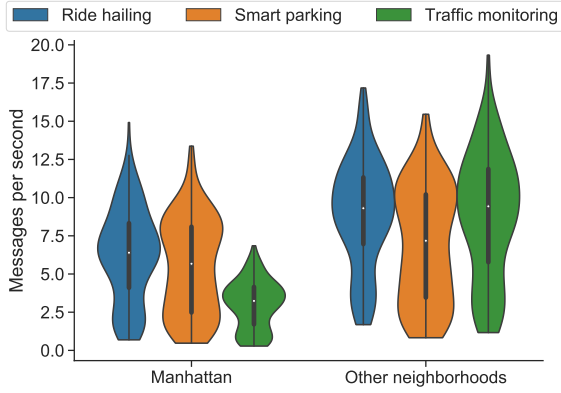


Fig. 2: Message density per service and neighborhood.

Airtime-based: with very dense deployments, devices are closer to the gateway and the signal strength is not a constraint anymore. In that case, we balance the load for each SF. More precisely, we assign the data rates so that all the data rates have the same cumulative air time. The devices with the largest SF consume more energy to stay awake longer, but we reduce the number of collisions.

We use the density of base stations deployed 5km apart as the threshold to use the fair airtime allocation; we indeed observed that at this point, the airtime allocation becomes a better strategy. To simplify our study, we also assume that any message can be sent within a single LoRa transmission.

We can now rely on the ITS design along with the real datasets introduced in sections III and IV. Since traffic can greatly vary in time, we pick 200 random time intervals of one hour, and use the average number of messages per second during each of these intervals. More specifically, for each randomly picked interval, we count the number of messages sent by each device and infer the average number of messages sent per second by each device. By considering independent enough time samples, we are able to investigate the scalability of LoRa for these smart mobility services.

We consider uniform grid deployments for LoRa gateways. To assess the scalability of such LoRa deployments, we however investigate several grid sizes, according to the density. Overall, we consider each borough separately since each area does not generate the same amount of traffic.

B. Results: Adapting the Deployment Density Looks Enough

We measured the intensity of traffic per mobility service and per borough (Fig. 2). The graph presents the distribution of the number of messages for all the 200 time intervals. The violin plots are stretched out: the number of messages is obviously much larger during peak hours. Besides, if we count the sum of messages, we can see that Manhattan concentrates most of the traffic, whatever the mobility service. For instance, 65% of ride hailing messages are generated in Manhattan while the

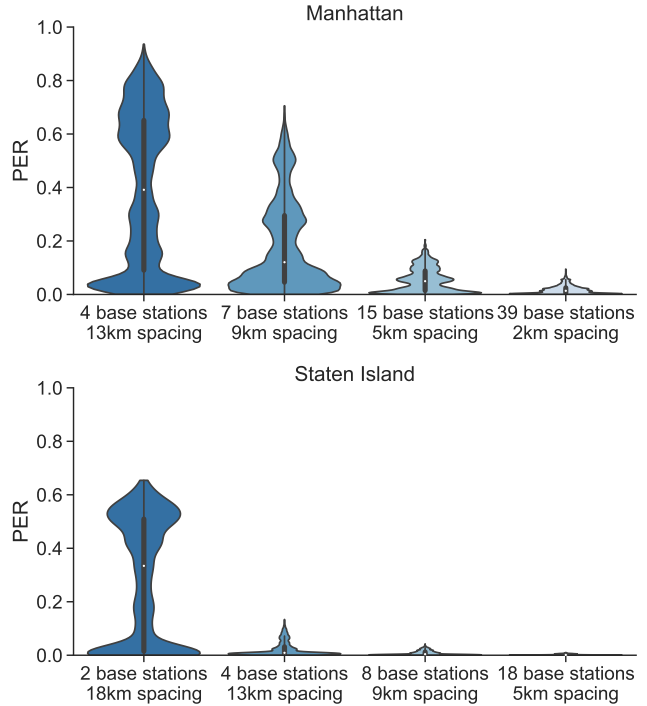


Fig. 3: PER of streams under different base station densities.

other boroughs only stretch over 7.5% of NYC. The traffic monitoring service is more balanced.

Relying on this data, we can now compute the distribution of the PER independently for each borough (Fig. 3). Due to space limitations, we limit our plots to Manhattan and Staten Island (with respectively the largest and the lowest number of messages per second in NYC). While each gateway is theoretically capable of serving nodes that are up to 9km away, the traffic is too high and the network is in that case congested. For instance, we can see that two gateways are not enough to efficiently cover Staten Island. However, 4 gateways seem enough to cover the whole area, and to also decently limit the number of collisions. Similar patterns can be seen for other boroughs, except for Manhattan.

For Manhattan, the volume of packets being specially large, many collisions occur. We see that only very dense deployments of LoRa gateways are capable of providing an acceptable PER. As a direct result, devices are very close to the LoRa gateways, and the lowest SF may be used by all devices, creating congestion. Thus, we need a more balanced allocation among the different SFs, which motivates a further evaluation of the two SF allocation schemes.

Fig. 4 shows how our two allocation strategies behave under various network densities. For extremely dense deployments (i.e., one gateway every kilometer), the number of devices associated with a given gateway is quite low and collisions seldom occur: the PER is maximal. For all the other densities, the airtime-based strategy is more efficient: devices are well balances among SFs, and the system minimizes the number

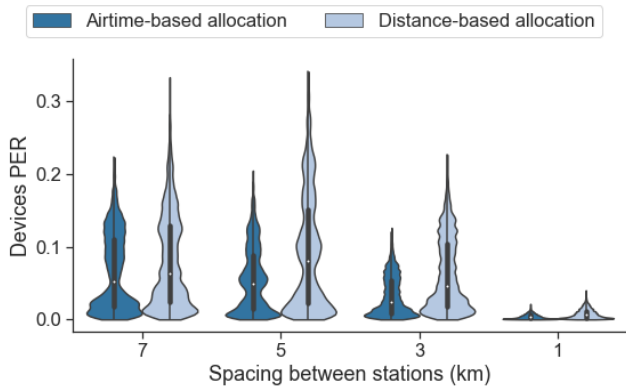


Fig. 4: Impact of the SF allocation strategy in Manhattan.

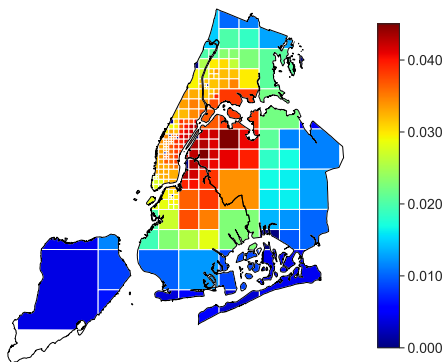


Fig. 5: PER under mixed deployment.

of messages lost (i.e., PER) due to both collisions and a low signal strength.

Finally, we look at the geographical distribution of the PER value (Fig. 5). More precisely, for each borough, we select a grid size to obtain for each device an acceptable reliability. Thus, we consider 5km spacing for Manhattan, 13 km spacing for Staten Island and 9km spacing for all other boroughs. We represent the results with a quadtree: the size of each square is selected so that each square has the same number of devices. Then, each square is colored according to the average PER obtained by the corresponding devices. As we can see, the resulting PER reaches a maximum of 4% in the center of the map: squares are smaller, denoting a huge number of devices that generate traffic. On the contrary, sparse grids are enough for Staten Island, since the traffic is very small in regards of Manhattan. If the PER has to be decreased even further, an efficient optimization scheme can be used to deploy extra base stations in the congested hop spot areas (red ones).

VI. CONCLUSION & PERSPECTIVES

We define here a global smart mobility scenario, where several services cohabit. We then emulate the traffic generated by such scenario by using two datasets, collected respectively

in NYC and Melbourne. In particular, we consider both event-based traffic from static devices, and periodical traffic from mobile ones. Finally, we analyze how a LPWAN deployment (LoRa in practice) may be sized to support these mobility services. While our scalability analysis tends to indicate that relying only on LoRa seems enough and so conceivable for an unified ITS deployment, we note that the number of LoRa gateways should be carefully tuned according to the traffic. Overall, the network capacity is reached quite fast, and may create scalability issues for large-scale scenarios.

In a future work, we plan to investigate how to refine the placement of the LoRa gateways. We propose here a simple heuristic, using a grid density adaptive to each borough. We should go further, by identifying the most accurate and favorable positions, depending on the location of hotspots. Unfortunately, the location of gateways being a continuous 2D variable, the global optimization becomes almost intractable. We may also explore how different technologies (medium vs. long range) may be complementary in such situations, handling differently static and mobile devices. Typically, smart parkings concentrate many devices in restricted areas that can result in congested hotspots.

ACKNOWLEDGMENT

This work was supported by the French National Research Agency (ANR) project Nano-Net under contract ANR-18-CE25-0003.

REFERENCES

- [1] J. Lin *et al.*, “A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications,” *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125–1142, Oct 2017.
- [2] C. Marquez *et al.*, “How Should I Slice My Network? A Multi-Service Empirical Evaluation of Resource Sharing Efficiency,” in *MobiCom*. ACM, 2018, pp. 191–206.
- [3] L. M. Feeney and P. Gunningberg, “Avoiding an IoT ‘Tragedy of the Commons’,” in *MobiSys*. ACM, 2018.
- [4] B. Ghena *et al.*, “Challenge: Unlicensed LPWANs Are Not Yet the Path to Ubiquitous Connectivity,” in *MobiCom*. ACM, 2019.
- [5] I. T. Union, “Guidelines for evaluation of radio interface technologies for IMT-2020,” 2017.
- [6] B. Ahlgren *et al.*, “Internet of Things for Smart Cities: Interoperability and Open Data,” *IEEE Internet Computing*, vol. 20, no. 6, pp. 52–56, 2016.
- [7] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi, “Hardware Design Experiences in ZebraNet,” in *ACM SenSys*, 2004, pp. 227–238.
- [8] X. Kong *et al.*, “Mobility Dataset Generation for Vehicular Social Networks Based on Floating Car Data,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, 2018.
- [9] N. Sornin *et al.*, “LoRaWAN Specification,” *LoRa alliance*, 2015.
- [10] D. Bankov, E. Khorov, and A. Lyakhov, “Mathematical model of LoRaWAN channel access with capture effect,” in *IEEE PIMRC*, 2017.
- [11] R. Fernandes *et al.*, “On the Real Capacity of LoRa Networks: The Impact of Non-Destructive Communications,” *IEEE Communications Letters*, vol. 23, no. 12, 2019.

- [12] R. Mangiaracina *et al.*, “Smart parking management in a smart city: Costs and benefits,” in *IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, 2017.
- [13] M. A. Falek *et al.*, “To Re-Route, or not to Re-Route: Impact of Real-Time Re-Routing in Urban Road Networks,” *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 2020.