Low-Cost Car Park Localization Using RSSI in Supervised LoRa Mesh Networks

Petr Gotthard
Konica Minolta Laboratory Europe
Konica Minolta
Brno, Czech Republic
petr.gotthard@konicaminolta.cz

Tomáš Jankech
Konica Minolta Laboratory Europe
Konica Minolta
Brno, Czech Republic
tomas.jankech@konicaminolta.cz

***Abstract*—In this paper, we present an RSSI-based localization solution that merely relies on the LoRa technology without requiring any anchors. It is primarily designed for large used car dealerships. Cars to be tracked are fitted with battery-powered tags that send each other plain LoRa messages and then report over LoRaWAN the observed RSSI to the server. The server estimates tag coordinates using multi-dimensional scaling and a novel corrective transformation based on tag clustering. We designed a low-cost tag for RSSI measurement and an energy efficient solution for managing a mesh of 1000 tags on one site using time division multiplexing. The tags are fully controlled and synchronized by the server. To stay synchronized for several hours until a next downlink the tags estimate and compensate the real-time clock error. Large-scale tests are yet to be performed, but lab-based and small-scale tests in real conditions show a maximal error of 8 meters and a battery lifetime of 5 years.**

Keywords*—*LoRa, localization, mesh, time-division multiplex, multi-dimensional scaling, clustering

I. Problem

We discovered that used car dealerships that have more than 400 cars on site (so called "car supermarkets") often lose track of the cars, so when a customer comes and wants to see a specific car it may take the staff even 20 minutes to find it. This might lead to customer loss, which in turn could cause a decline in dealership profit margin as shown in [1].

For a large used car dealer we designed a system that:

- Works both indoors and outdoors in all weather conditions;
- Can find a car to within 10-20 meters, so it can be then discovered using the beep sound when the wireless key fob is pressed;
- Can detect departed cars and determine location of newly arrived cars in 5-10 minutes;
- Requires a minimum infrastructure to deploy and minimum effort to maintain;
- Does not impact the way the staff is parking the cars.

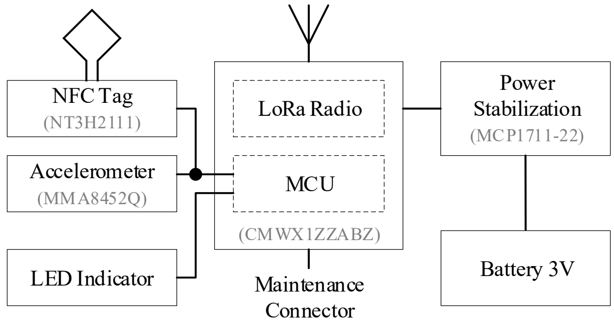

Fig. 2. Block diagram of the tag hardware

A. Solution

We designed a mechanism that uses the long-range LoRa technology to establish an ad-hoc network between wireless sensor devices (called "tags") and then applies the RSSI-based localization for Wireless Sensor Networks [2] to determine the location of each tag.

As we expect about 1000 tags on each site the tags need to have increased reliability and security. Tags are hence by design based on a simple low-cost hardware with no possibility for remote firmware upgrades. Furthermore, they are fully controlled by the server through a set of commands.

The solution works as shown in Fig. 1:

1) Each car to be tracked is fitted with a tag (see Fig. 2). The tags broadcast ("ping") their ID number to each other in regular intervals;
2) Each tag regularly reports the RSSI of the received signals to a central server. The server then responds with a command that determines the tag operations for the next cycle;
3) The server periodically recalculates relative tag-to-tag distances based on the recent RSSI data measured by the neighbouring tags and estimates absolute x-y coordinates of each tag.

Position of a car can be then displayed on a tablet, accessed via a REST API, or sent to a Dealer Management System.

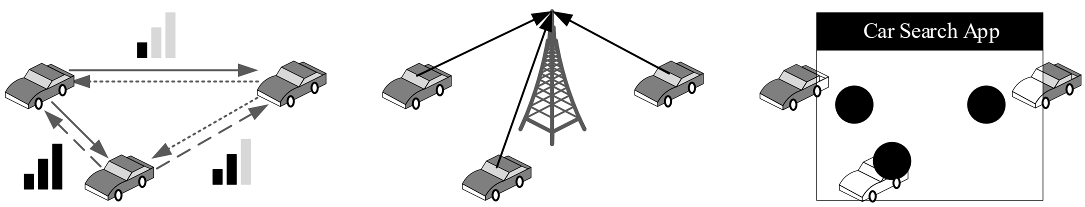

Fig. 1. Illustration of the localization system operation. The solution operates in three phases: 1) measurement, 2) reporting, 3) location estimation.

978-1-5386-6436-0/18/$31.00 ©2018 IEEE

II. MANAGEMENT OF LORA MESH

Four modes of a tag operation are defined:

- "Reporting" tags in dedicated slots periodically transmit and listen for "pings" and report the observed RSSI;
- When moved, the tag waits a random time (to avoid collisions when multiple tags have moved), then sends the Lost message and enters the "Moving" mode;
- "Moving" tag waits until the movement stops, then sends the Reset message and starts "Seeking";
- "Seeking" tags listen for pings from any reporting tags nearby and then send the Found message with results;
- "Detached" tags (not attached to any car) do not transmit nor listen for any messages, neither detect movements and just await tag-to-car attachment.

A. Slot Allocation Scheme

Time division multiplexing is used to streamline the tag transmissions. For each site the server uses two aligned time lines: Measurements and Reports, each with n slots in periodic cycles. A sample arrangement is illustrated in Fig. 3.

The slot count (n) is determined based on the required capacity of the site (the number of parking places). The length of one measurement slot (tm) and the length of one reporting slot (tr) can be modified to change the system refresh rate. The server randomly assigns a unique slot number to each tag arriving to the site and releases the slot again as soon as the tag leaves the site.

Reporting tags may only transmit (ping or report) in the assigned slot. The Init, Lost and Reset messages are however sent immediately to have a fast response time. Found results are sent immediately too, but the transmission is made in a second half of the actual reporting slot to minimize collisions.

All measurements (pings) use plain LoRa messages on a dedicated frequency (e.g. channel 9 at 868.8 MHz with 0.1% duty cycle). The messages only include the DevAddr of the transmitting tag. One "ping" is < 10 ms, but to save the energy we used tm = 2 s.

The reporting uses the standard LoRaWAN protocol [3] on a set of different frequencies (by default channels 0-2 at 868.1-868.5 with 1% duty cycle). The messages fit to the maximum payload size for the slowest Data Rate, so one reporting (Tx+Rx) is < 5 s. Since every uplink is followed by a downlink, the reporting period is much more constrained by the ISM band duty cycle limitations of the LoRaWAN gateway. The minimum recommended value for deployments with one gateway is tr = 30 s.

B. Tag Initialization

When first powered-on the tag:

1) Joins the LoRaWAN network. We rely on frequency hoping to avoid interferences with other Tx.
2) Sends an (empty) Initialisation message, requesting a first command from the server.
3) Receives the Detached message because the tag is not associated to any car.

A Near Field Communication (NFC) enabled device (e.g. a smart-phone) can be used to retrieve the tag DevEUI and associate it in a database with a selected car. When the user takes the attached tag away from the NFC field, the tag:

1) Sends the Reset message;
2) Receives the Seeking message.

Tags can be detached in any state using a similar procedure.

C. Seeking Tags

To quickly determine its approximate location the tag listens for a given amount of "pings" from any reporting tag nearby (see Fig. 4). Every Seeking tag:

1) Receives a Seeking message indicating the number of measurements to make, min/max number of slots to listen, and an RSSI threshold to omit measurements from distant tags, which are less accurate;
2) Listens for k successive measurement slots (k=tr/tm) and then checks the number of Rx slots opened so far and the number of RSSI values collected;
3) If not enough successful measurements was made and the max number of slots was not reached yet, the tag continues to listen;

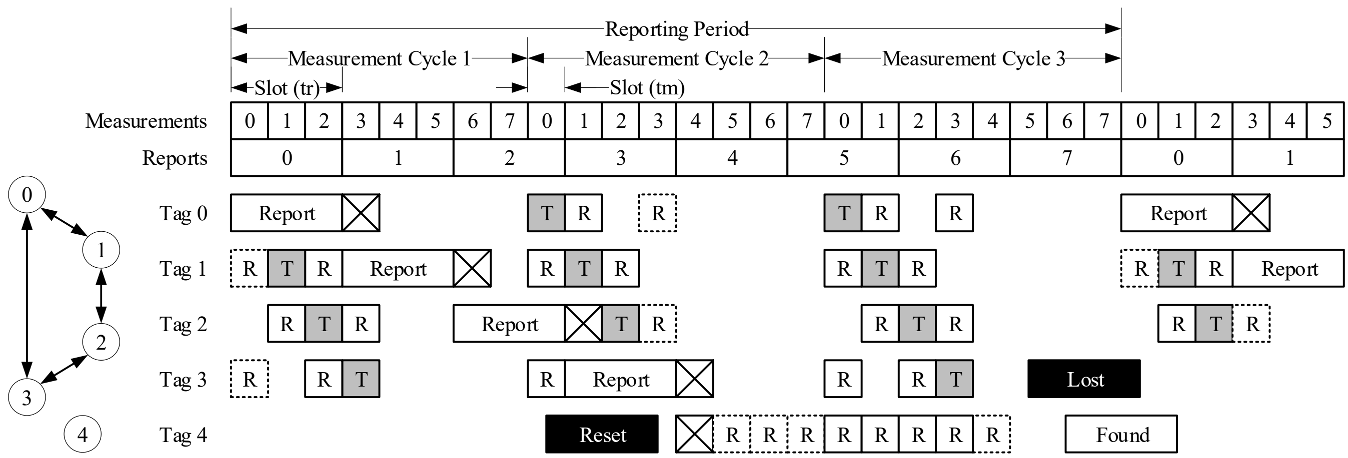

Fig. 3. Sample Tx and Rx slot arrangement for one site. The tag 0 is a neighbour to tags 1 and 3, so it listens for slots 1+3, tag 1 listens for 0+2, tag 2 for 1+3, and tag 3 for 0+2. The tag 4 newly arrived. The dashed "R" slots show unsuccessful reception when the neighborouing tag reports instead of sending a "ping".

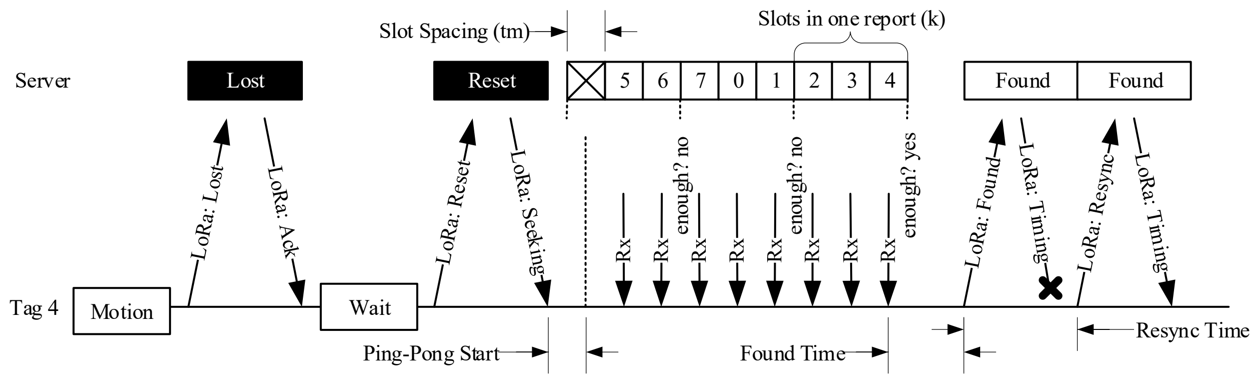

Fig. 4. Seeking tag operation within the slot arrangement introduced in Fig. 3.

4) Otherwise sends a Found message (possibly empty) indicating the DevAddr and RSSI of the heard neighbours. If no tag was heard, the seeking tag will be requested to perform another listening cycle with a lower RSSI threshold;

5) Receives a Timing message (and starts normal reporting). When not received, an empty Resync message is sent.

The server applies a uniform random distribution when assigning the slot numbers. To have a fast Seeking:

- Tags with slots n and n+1 must be in very different locations, so a seeking tag in any location quickly receives enough "ping" messages.
- The unallocated slots must be uniformly distributed to avoid having many similar indexes in one location when multiple tagged cars arrive.

Whether this assertion can be achieved at a large scale is yet to be validated at the customer site.

D. Reporting Tags

When a tag location is known the tag continues to measure and report the RSSI of its closest neighbours (see Fig. 5). Every Reporting tag:

1) Receives the Timing message, including:
   - o One Tx index corresponding to tag's slot and a requested number of measurement cycles;
   - o Up to 8 Rx indexes corresponding to slots of its neighbours and a list of 3 measurement cycles when the Rx windows shall be opened.

2) Sends and receives „pings" in the given measuring time-slots;

3) Sends a Report message that includes the DevAddr and RSSI of the heard neighbours;

4) Receives a next command from the server. When not received, stops "pings" and sends a Resync message.

Each tag keeps transmitting pings until the next reporting slot, but the Rx window is opened in few measurement cycles only. To save energy the server selects first 3 measurement cycles when all of the selected neighbours are transmitting. It does not schedule any Rx slots when some of the neighbours are reporting and thus do not "ping".

To ensure precise timing the server has to estimate the transmission time of the Timing message and subtract this value from the timing information sent to each tag.

The index numbers sent to the tag are relative to the start of the actual reporting period. The index 0 cannot be used as the tag needs some time to initialize the measurement.

To select the neighbours the server finds 8 closest tags in 8 different azimuth angles, preferring closer ones (as the precision is higher for lower distances) but aiming at having neighbours in all directions. As shown in Fig. 6 the server:

1) For each tag randomly selects an "azimuth angle" and then selects a closest neighbour in each of the 8 sectors, each 45° wide (tags 3,6,7 in Fig. 6).

2) If there is no neighbour in a given azimuth angle, selects a second neighbour from an adjacent sector, minimizing the heuristics φd, where d is a distance of the neighbour and φ is the angle (in degrees) between the neighbour and the segment centre (tag 1).

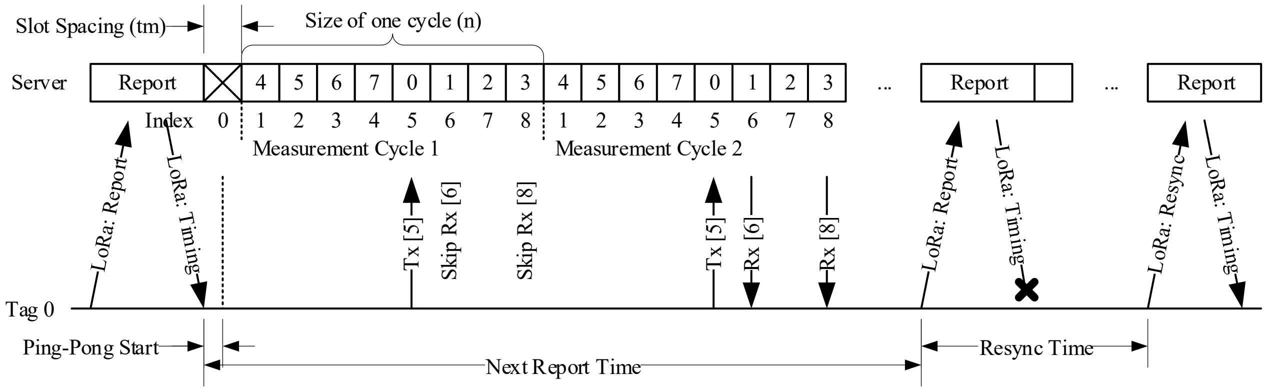

Fig. 5. Reporting tag operation. Tag 0 listens for slots 1+3. It however does not listen (skips Rx) in the first cycle because both neighbours are reporting.

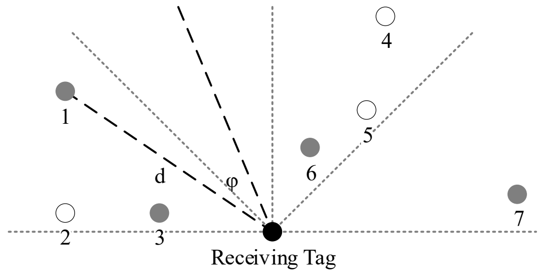

Fig. 6. Neighbour selection for a given tag (upper half only)

E. Clock Error Correction

The server uses GPS time received from the LoRaWAN gateway. To correctly schedule the measurement (ping) slots all tags on one site have to be synchronized with each other.

Each tag uses a 32.768 kHz clock for timing. The crystal used for that has ±20 ppm tolerance (*Ct*) at 25°C. This means that for a 1000-tag site and 21.6 second reporting slots we may get up to ±432 ms drift after one 6-hour reporting period. The maximum timing error *Emax* can be calculated as:

$$Emax = 2\ n\ tm\ Ct$$

Transmitting of one "ping" (*ta*) takes about 8 ms, but tags open their reception window for a longer time for higher chance to receive the message. The window is closed as soon as the ping transmission is received. The maximum width of the reception window (*tr*) is:

$$tr = Emax/3\ +\ 2\ ta$$

In case the real-time clocks are too desynchronized and a tag does not receive any "ping" in the scheduled slot, this tag modifies the start time of the next reception window for this neighbour and searches for the transmission (see Fig. 7).

In case the "ping" was successfully received, the tag checks the length of the waiting time *tw* before the neighbour's transmission was received (see Fig. 9). Too long wait is a waste of energy, while too short wait means a risk of further time drift. When $tw > 1.5\ Et$ or $tw < 3\ ta$ the tag updates the timing for this one neighbour as:

$$\text{next cycle offset} = n\ tm\ (1 - Ct) - 2\ ta$$

Both timing updates are applied only in consecutive measurement cycles. If a measurement cycle was skipped by the server, one of the neighbours was scheduled to receive new timing data from the server. The use of the new timing could misalign all upcoming transmissions from this neighbour.

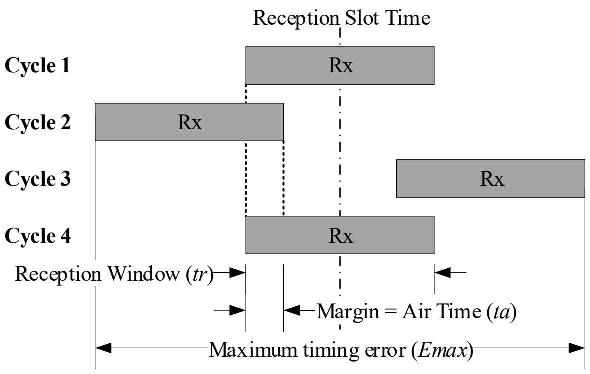

Fig. 7. Rx window shifts when searching for a Tx windows

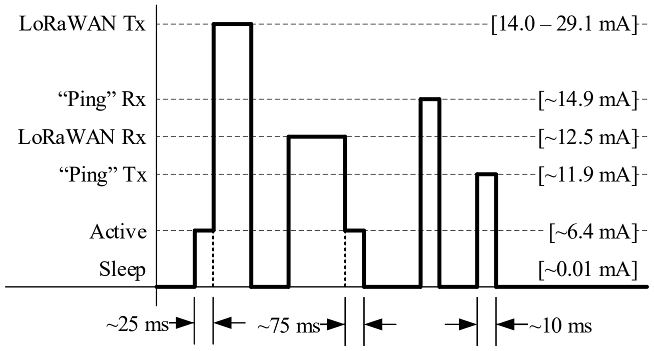

Fig. 8. Power consumption during tag operation (at 2.2 V): report Tx (-1 to 16 dBm), report Rx, ping Rx and ping Tx (-1 dBm).

F. Power Consumption

The five main phases of a tag operation are shown in Fig. 8:

- In the *Sleep* phase the device turns off all unnecessary hardware. The accelerometer stays active as a movement detector;
- In the *Active* phase the Microcontroller prepares the tag for a new report or measurements;
- The *LoRaWAN Tx/Rx* phases depend on the DataRate and the length of the message. The measured current consumption range for the LoRaWAN Tx corresponds to the output power in a range between -1 and 16 dBm;
- Measurement (*Ping*) phase consists of two steps (Tx and Rx). Transmission is always at -1 dBm for about 10 ms, but the width of the reception window varies as explained before.

III. Localization in LoRa Mesh

A. RSSI to Distance Conversion

The RSSI observed between tags gets converted into tag-to-tag distances. The best results have been reached with the equation:

$$\text{RSSI [dBm]} = -10p\ \log_{10}(d) + A,$$

where *A* is the received signal strength in dBm at 1 meter, *d* is the distance in metres and *p* is a propagation constant. The constant was selected empirically, based on calibration tests shown in Fig. 10.

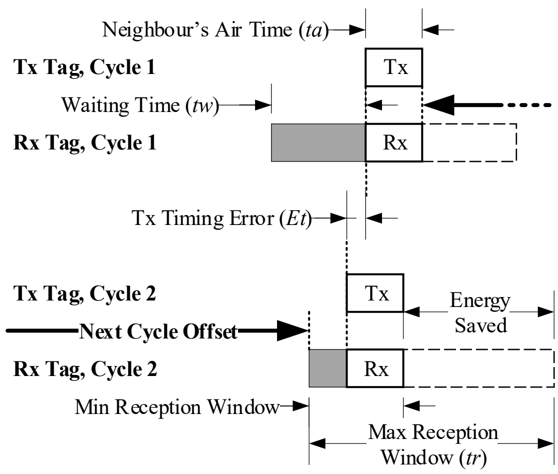

Fig. 9. Reception window optimization

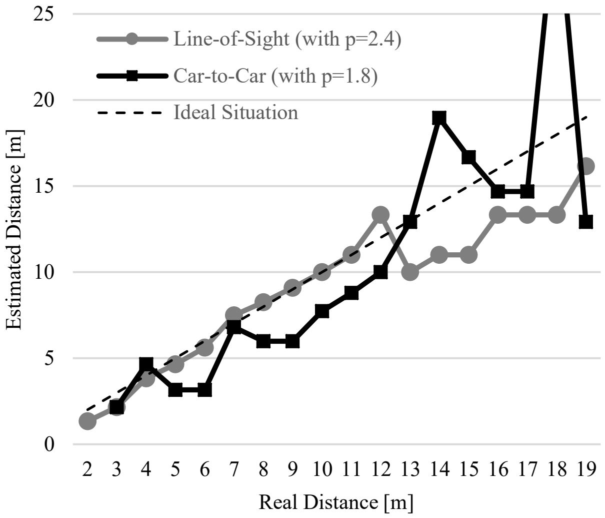

Fig. 10. Precision of the RSSI to distance conversion, tags 1m above ground. (For the line-of-sight, the bump around 13m is caused by a disruption of the Fresnel zone by the ground.)

B. *Methods for the RSSI stabilization*

The algorithm precision depends on how similar RSSI values we get between different tags at the same distance.

A plastic case for the tags was designed to be hang on a rear view mirror along the Z axis, which ensures that all tags are oriented the optimal way.

To achieve RSSI stability during the battery life-cycle the tags convert the varying battery voltage to stable 2.2 V. With the battery used is the supply voltage in a range 3.2 V to 2.2 V (the minimum supply voltage for the LoRa module), which affects the RSSI stability as shown in Fig. 11. The voltage stabilizer stabilizes the radio performance at the lowest level, which reduces the amount of interferences from other tags. The stabilizer also provides the short-circuit protection.

C. *Localization Algorithm*

The server converts the measured relative tag-to-tag distances to x-y coordinates of each tag using the standard MDS-MAP algorithm from [2] and [4] as follows:

1) Complete the distance matrix $\hat{D}$ by computing the shortest path between any two tags (Floyd–Warshall algorithm);
2) Compute $(-1/2)L\hat{D}L$, where $L = \mathbb{I}_n - (1/n)\, \mathbb{1}_n\, \mathbb{1}_n^{\mathrm{T}}$, $\mathbb{1}_n$ is the "all ones" vector and $\mathbb{I}_n$ is the $n{\times}n$ identity matrix;
3) Compute the best rank-2 approximation $U_2\, \Sigma_2\, U_2^{\mathrm{T}}$ (two largest eigenvalues) of $(-1/2)L\hat{D}L$;
4) Return the x-y coordinates $\hat{X} = U_2\, \Sigma_2^{1/2}$.

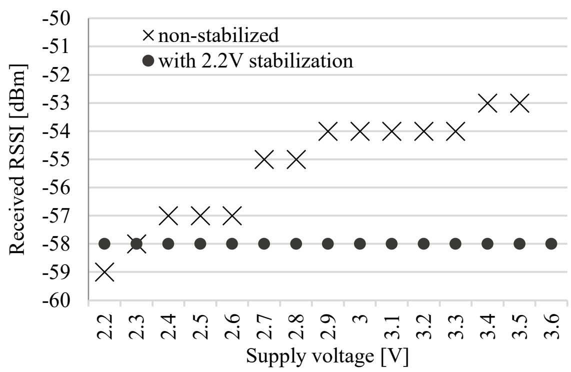

Fig. 11. RSSI stability at different supply voltages, measured between tags 80cm apart, for pings at SF7 and –2 dBm Tx ouput power.

The Multi-Dimensional Scaling (MDS) algorithm produces proportional coordinates $\hat{X}$ that are arbitrarily rotated, scaled or inverted. To estimate parameters of the corrective transformation and calculate the absolute coordinates $\hat{Y}$ in meters we designed the following approach:

1) Each site is divided into multiple (>3) zones and for each zone the absolute x-y coordinates $\hat{Z}$ (in meters) of the zone centre are determined;
2) Each Reporting tag is assigned to a cluster depending on in which zone it is physically located;
3) Centroid of each tag cluster $\hat{C} = \mathrm{mean}(\hat{X}_i)$ is calculated in the proportional coordinates and an affine transformation f is estimated such that $\hat{Z} = f(\hat{C})$;
4) The tag coordinates are transformed as $\hat{Y} = f(\hat{X})$.

The affine transformation $f(A) = cRA + T$ is estimated using the standard Umeyama algorithm [5] as follows:

1) Calculate a covariance matrix $\mathrm{cov}(\hat{C},\hat{Z})$ and a variance $\mathrm{var}(\hat{C})$;
2) Calculate a singular-value decomposition UDV^{T} of $\mathrm{cov}(\hat{C},\hat{Z})$;
3) Calculate the estimated rotation matrix $R = USV^{\mathrm{T}}$, scale factor $c = \mathrm{tr}(DS) / \mathrm{var}(\hat{C})$, and translation $T = \mathrm{mean}(\hat{Z}) - cR\, \mathrm{mean}(\hat{C})$, where $S = \begin{cases} 1, & \det(U)\det(V^{\mathrm{T}}) \geq 0 \\ -1, & \text{otherwise} \end{cases}$.

The initial clustering has to be done manually during system initialization. Once there is enough tags in each cluster the newly arriving tags are clustered automatically. New tags have no zone assigned, so they are not included into the centroid calculation. Once the corrective transformation is established and applied to all tag coordinates the new tags get assigned to the nearest cluster based on their coordinates.

The stability of this concept in large-scale deployments is yet to be investigated.

IV. RESULTS AND DISCUSSION

We designed a RSSI-based car localization system that solely relies on the LoRa technology and does not use any anchors. The system requires only a minimum infrastructure (a server and one LoRaWAN gateway per site). Two gateways per site are recommended though to achieve higher system availability. Tags use the ARM Cortex-M0+ microprocessor and consume 5.5kB RAM and 57.4kB flash.

The system is subject to the following constraints, which are however fully acceptable for car dealerships:

- Most of the cars must be stable to keep the mesh interconnected, only few may move at a given time;
- To keep the mesh inter-connected two neighbouring cars must not be more than 20 m from each other;
- To calculate absolute coordinates there must always be some cars in at least three zones (clusters). If all cars leave the system must be manually re-initialized.

Δ 5 m | Δ 8 m | Δ 7 m | Δ 5 m | Δ 3 m | Δ 2 m | Δ 3 m

Fig. 13. Results of a small-scale test. The white rectangle shows the real car location, the black circle shows the result of the location algorithm.

Although plain LoRa is used for the tag-to-tag pings, the solution is fully compliant to the LoRaWAN specification [3]. It only requires an extended application interface. For the Timing and Seeking downlinks, the server has to announce the amount of FOpts to be sent and the DataRate to be used, so the application can precisely estimate the transmission time and calculate the timing information.

A. Lab Test

The system performance in an ideal environment was verified by a lab test using 4×3 tags in a grid 150×150 mm as shown in Fig. 12. The average error was 29.7±17.7 mm, max 71.8 mm, which is about 50% of the tag-to-tag distance.

Interferences from other LoRa devices, joining tags or the environment are causing occasional ping and report failures. Typical failure rate was 1% for reports and 4% for pings. To keep the mesh network always sufficiently interconnected the RSSI data from past 10 reports from stable tags were stored.

B. Small-Scale Test in an Outdoor Environment

The system was also continuously tested for 4 days on 7 cars at a company parking in Prague. The maximal error Δ was 8 m, average error 4.71 m as shown in Fig. 13.

The system accuracy was heavily impacted when a high percentage of non-tagged cars was parked between the tagged neighbours. In such case the error was more than 15 m, but we expect a much lower percentage for the target deployment.

C. Estimated Large-Scale Performance

Scalability of the system is limited by the radio spectrum and related duty cycles. The more cars are on the parking area

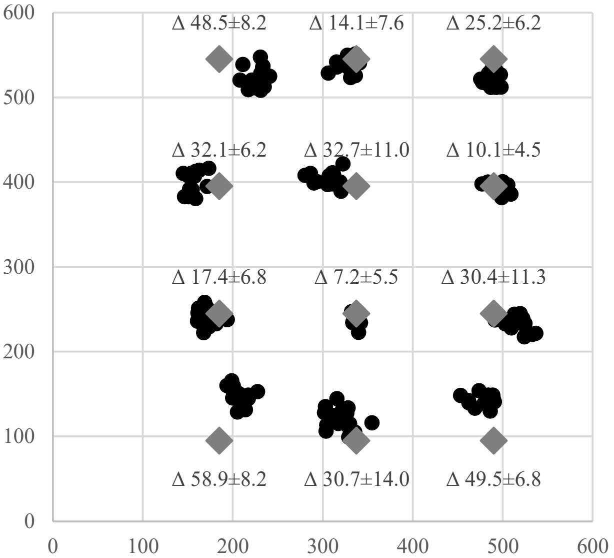

Fig. 12. Layout [mm] of the lab test with 3×4 tags showing the location error during 3 days of continuous operation with reports every 4 hours.

the longer will be the reporting and seeking cycles. To meet the requirement for fast localization of newly arrived cars the system should follow these settings:

For a 1000-tag site with *tm*=2.7 s and *tr*=21.6 s (each tag will report 6 times a day) a seeking tag can hear 60 nearby tags (6%) in one seeking cycle. Assuming there are 50 other tags in the 20m radius the seeking tag will hear 3 pings in one cycle. This means that after two cycles (6 min) the server will have enough data to estimate the location of the seeking tag.

Estimated life time of an 800 mAh CR2 battery with these settings is about 5 years. All tag transmissions use 0.083% of its allowed duty cycle and one gateway uses 68.75% of its allowed duty cycle, assuming the LoRa spreading factor SF7.

D. Research Questions for the Large-Scale Testing

Although the results of the small-scale test are encouraging, more tests are needed. As the system assumes a large amount of tags an at-scale test in the target environment needs to be performed, particularly to verify:

- Performance of the RSSI to distance conversion on different parking areas and the need for calibration;
- The amount of interferences caused by other objects and their impact on the system accuracy;
- The amount of collisions from moving and seeking tags in the environment of a large car dealership;
- The ability to keep the random slot distribution and the related speed of Seeking for newly appearing tags;
- The ability to auto-cluster the newly appearing tags on a large site.

Acknowledgment

Many thanks to John Shah (Midas) from Konica Minolta Business Innovation Centre (BIC) for discovering the UK customer and helping with the business aspects of the project.

References

[1] J. Shah, P. Gotthard, and T. Jankech, "How the IoT Can Improve Customer Experience Profits and More in Automotive Retail", in IEEE IoT Newsletter, September 2018.

[2] A. Karbasi and S. Oh, "Robust Localization From Incomplete Local Information," in IEEE/ACM Transactions on Networking, vol. 21, no. 4, pp. 1131-1144, Aug. 2013.

[3] LoRa Alliance, "LoRaWAN Specification", V1.0.2, July 2016.

[4] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, „Localization from Mere Connectivity," in Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing, pp. 201-212, June 2003.

[5] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 4, pp. 376-380, Apr 1991.