

Compass Aided TDoA Tracking in LoRaWAN networks

Nico Podevijn*, Jens Trogh*, Michiel Aernouts**, Rafael Berkvens**,
Luc Martens*, Maarten Weyn**, Wout Joseph*, David Plets*

* Department of Information Technology, University of Ghent, IMEC, WAVES, Belgium, nico.podevijn@ugent.be

** Faculty of Applied Engineering, University of Antwerp, IMEC, IDLab, Belgium

Abstract—LoRaWAN nodes can be localized using the TDoA (Time Difference of Arrival) approach. A big advantage is that the geo-location does not consume a lot of energy when compared to GPS approaches. Depending on the use case, the accuracy of the geo-localization (200m) might be a disadvantage. Therefore we propose activating an E-compass in the LoRa node and frequently communicating this directional information over the LoRaWAN network. This extra information can then be fused with TDoA in a map matching algorithm to improve the estimations. We show that our sensor fusion technique is 8 times more accurate than raw TDoA at the cost of only embedding a low-cost e-compass.

Index Terms—LoRaWAN, Geo-location, TDoA, Tracking, Map Matching, Compass, Sensor fusion

I. INTRODUCTION

Currently a lot of objects are being connected to the internet by means of wireless sensors: the Internet of Things (IoT). Supporting wireless networks for IoT are NB-IoT, Sigfox and LoRaWAN. Typical sectors are smart cities, industry, logistics and agriculture. Although the networks are designed to provide communication for sensor devices, many of them also have geo-location abilities. These location based services (LBS) are mostly not as accurate as Global Navigation Satellite System (GNSS) but they do allow for low cost and low power localization without the need for a GNSS receiver. Some examples of LBS are localizing construction site materials, pallets (and their goods) and containers. For such assets an accuracy of, e.g., 200m is more than sufficient and a battery life of the attached sensor of 5+ years is a clear advantage.

LoRaWAN deployments come in different options. The deployment can be either done with gateways which only record the incoming received signal strength as metadata (option A) or they may also record the timestamps of the received packets (option B). The latter allows for localization using the Time Difference of Arrival (TDoA) method. In this method the sensor node transmits a packet to the network which is assumed to be received by 3 or more gateways. Each gateway is equipped with a GPS receiver and is therefore able to record the timing of when the packet was exactly received and recorded. The recorded timestamps and location of the gateways allow for localization of the sensor node which transmitted the packet. The accuracy of the method depends on many factors such as gateway density, environment, timestamp accuracy, Line-of-Sight (LOS) or Non Line-of-Sight (NLOS) scenario. The median accuracy is around 200m for moving

Figure 1: Measurement setup carried within a jacket: Left: TX LoRaWAN node for which we estimate its location. Right: Smartphone which logs the GPS ground truth

(single shot) TDoA estimates [1]. In some cases the accuracy can be improved when a stationary node is assumed: multiple measurements can be averaged and improved accuracies are possible. The disadvantage of LoRa localization is that it provides discrete points on a map. When an asset is to be retrieved, one may demand for a tracking mode but currently this mode is not available. To overcome this limitation, one may use for example higher uplink frequency LoRa transmissions (every 5 seconds) in combination with a Viterbi road map matching algorithm [1] that restricts the reconstructed trajectory to the road network and takes into account the limited mobility of the sensor. For example a maximum speed of 30km/h may be assumed for a sensor attached to a rental bike which needs to be tracked. In this work, we propose to embed an electronic compass into the transmitting LoRa node and transmit this data over LoRaWAN. The general idea is to combine the info from the compass with the TDoA estimates in a sensor fusion algorithm. In this manner, the most likely trajectory can be estimated with even a higher accuracy. The fusion operates by first estimating the inertial trajectory from compass data and an assumed constant speed. Then the ‘best fit’ with the TDoA updates is estimated for different speeds. The result gives the most likely trajectory. The improvement using this method versus raw TDoA is in most cases over 50%, enabling median accuracies of 100m and lower.



Figure 2: Ground truth (trace) and estimated TDoA locations (dots) from Semtech solver

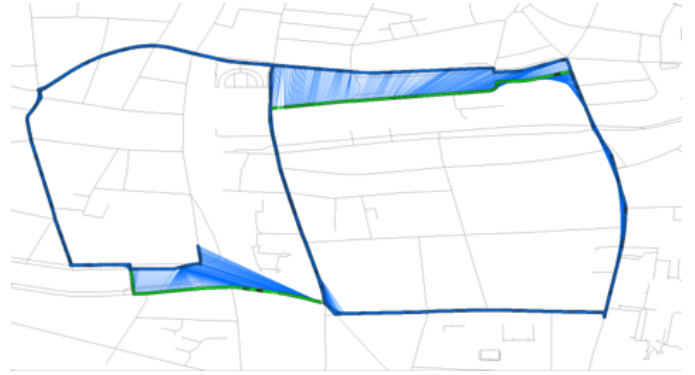


Figure 4: Ground truth (green) and estimated trajectory after map matching (blue) with (relative) compass heading

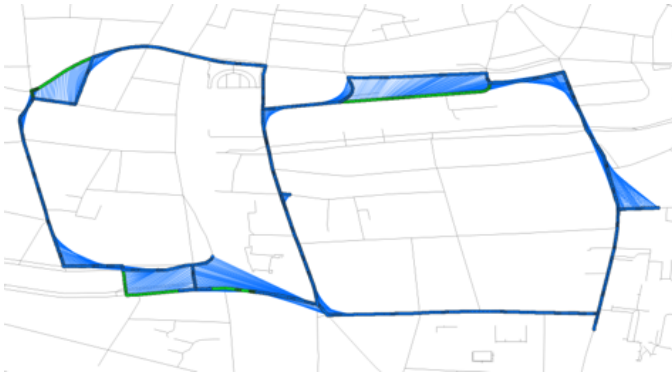


Figure 3: Ground truth (green) and estimated trajectory after map matching (blue) without using a compass.

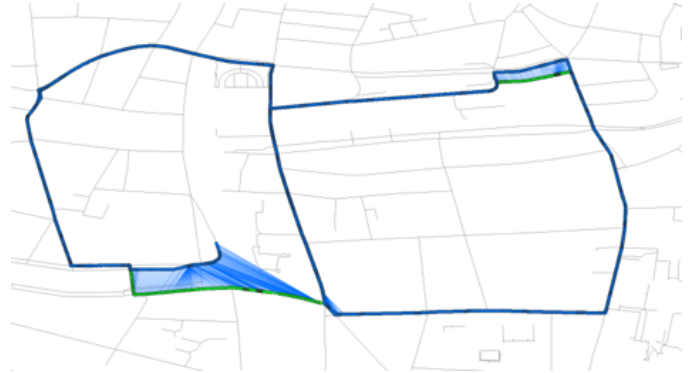


Figure 5: Ground truth (green) and estimated trajectory after map matching (blue) with (absolute) compass heading

The paper is structured as follows. In Section II we provide an overview of related work concerning LoRa-geolocation methods and improvements proposed in literature. Section III describes our materials and methods, describing our obtained data collection method, trajectory and algorithm implementation. The obtained results using our technique and method are presented in Section IV. These results are then discussed in Section V. We conclude our findings in Section VI where we provide recommendations and suggest directions for future work.

II. RELATED WORK

One way to locate sensor nodes in a LoRaWAN network is making use of the RSS metadata received on the gateways after a transmission. This has been studied in the works of [2] and [3]. In [2], the RSS received from up to 3 gateways was mapped to a location using different algorithms and an obtained median accuracy between 1250 and 2500 m was obtained. With known mobility, improvements are possible using map matching which can reduce the median error to 700 m. An improvement is possible using RSS fingerprinting [3] after collecting a large training database. The mean obtained accuracy for this approach was around 400 m. Similar results using other fingerprinting methods were reported in [4] where a mean accuracy of around 350 m was obtained. The downside

of such approaches is the fact it requires a lot of work to build and maintain a LPWAN fingerprinting database on a large scale.

Many deployed LoRaWAN networks are currently using gateways that can accurately determine the Time of Arrival (ToA) of an incoming packet sent by a node. The recorded time stamps and known locations of the gateways can be used as input to a Time-Difference-of-Arrival (TDoA) algorithm to estimate the location with a higher degree of accuracy. The median accuracy of using this method with the use of a maximum likelihood (ML) algorithm was around 200 m as reported in the works of [2], [5] and [1]. A new, improved TDoA algorithm is proposed in [6] and compared with a Least Squares (LS) approach. It was shown that the 95th percentile was improved from 2200 m to 840 m in a simulated environment. Correcting the received timestamps of a mobile node by the use of machine learning in combination with stationary reference nodes is reported in [7]. Using this method, the best reported accuracy was around 61 m. It remains unclear how many reference nodes are needed and this approach might not be economically feasible for large deployments.

In this work, we base on real performed experiments in public LoRaWAN networks. We consider moving nodes and compare our implementation with the best state-of-the-art

algorithms available.

III. MATERIALS AND METHODS

A. Data Collection and Trajectory

The sensor node we use for our localisation experiments is the MCS iTalks 1608. The device is provisioned on the Belgium Proximus LoRaWAN network. The firmware in the device has been adjusted to obtain a short interval between successive transmissions. Every 5 seconds, a 5-byte packet is transmitted on spreading factor (SF) 7. Airtime for each transmission is around 50ms and therefore we still comply with the ETSI regulation of a 1 percent duty cycle. Packets are received by the gateways deployed in the area around the node and are then forwarded to the network server (NS). The available data we use from the NS is as follows:

- A 5-byte payload from the device which contains the 5 compass values (also called heading or bearing) from the last 5 seconds. Due to the unavailability of the compass in the node, the compass values are simply emulated on a (SamSung Galaxy A20E) smart-phone application (OSMTracker for Android) which was carried together with the device.
- Nano-second Timestamp of the arrived packet on each gateway denoted with their ID

The network deployment of the gateways is known for the Proximus network (ID versus latitude/longitude coordinates) and together with the recorded data allows us to post process the data into estimated locations and/or a trajectory using a suitable algorithm. The ground truth (and time) was logged with a GPS application on the smart-phone (OSMTracker for Android).

Our LoRa device and smart-phone were carried in the front pocket of a jacket (See Fig. 1) and was carried along a walking trajectory in the city of Ghent. The ground truth trajectory is shown in a black trace in Figure 2. The length of the trajectory was 3.1 km and at an average walking speed of 4.6 km/h it therefore took 40 minutes to complete. The considered measurement area was around 500 m x 1200 m.

B. Scenarios

We consider different scenarios which are applicable for different use cases. Firstly we investigate possible improvement when using map matching without using a compass. Next we consider an (emulated) compass node for which heading values are available and use this info to improve the map matching.

Finally, we also distinguish between a device which is assumed to have no compass (Agnostic) and a device equipped with Compass sensor. To emulate an agnostic device without compass we simply ignore the compass data we receive in our algorithms. We denote the agnostic or compass-enabled device by the letters 'A' or 'C' respectively.

We investigate 4 scenarios

- Semtech solver: Our starting point is to use a third-party TDoA solver such as the one from Semtech [8]. The collected ToA timestamps and location of the gateways are

forwarded to their server which estimates the locations using a (TDoA) algorithm. These locations are shown as dots in figure 2.

- TDoA MM: Our TDoA map matching algorithm (without compass) which takes the road map, max mobility speed, timestamps and locations of the gateways as input to solve for the most likely trajectory.
- TDoA MM abs compass: This is our map matching technique with compass sensor fusion and it is assumed that the compass bearing gives absolute headings. eg. 0 degrees means the asset moves towards North. This implementation may be well suited to track bicycles for which one knows how the compass was installed relative to the bicycle.
- TDoA MM rel compass: This implementation is similar to the previous one but it only uses relative headings from the compass. eg. a 90 to 180 degree bearing or a 30 to 120 degree transition means a turn to the right. This method is more suitable to track parcels in transit for which one does not know the absolute orientation of the installed compass node relative to the direction of movement.

C. Algorithm

The pseudo-code of the map matching method is shown in Algorithm 1 and the variables and steps are discussed in the text below.

The map matching algorithm is initialized based on the first TDoA measurements (TM_0) for which a solver can calculate a location (L_0), e.g., if three gateways are visible. Then, a predefined number of other locations (MP) are selected around this location (from open street maps data) and their probability is initialized to one, e.g., the 1000 closest grid points to the current position. This ensures that the map matching algorithm can recover from initially noisy data, e.g., 1000 grid points and a grid size of 10 m results in covered surfaces of around 50 hectares (the exact area depends on the density of the road network). The initialization forms the starting point of all possible paths that are kept in the memory of the location tracking algorithm (*paths*).

Next, for the subsequent TDoA measurements (TM), all reachable positions (RGP) starting from the path's current endpoint (E) are determined for all *paths* in memory by making use of the surrounding road network, the time elapsed since last location update (Δt), the mode of transportation (MoT), and OpenStreetMap metadata, i.e., maximum speed, type of road, and one-way information. These reachable positions, which are also grid points, are the candidate positions for the next location update. The transitions between grid points are limited by the road infrastructure. Each candidate position (CP) retains a link to their parent (i.e., the previous endpoint E), a list with visited road segments RS , and a probability that represents this new branch along the road network. This new path (branch) and updated probability are added to the temporary list (tmp) as a tuple ($path_{new}, P_{new}$). The updated probability is the product of the previous probability P with a TDoA and compass contribution (P_{tdoa} and P_{comp}).

Algorithm 1: Map matching technique.

Input: compass data + TDoA measurements (TMS)**Result:** map matched trajectory (MMT)

```
1  $MoT \leftarrow$  mode of transportation
2  $TM_0 \leftarrow$  first TDoA measurements
3  $L_0 \leftarrow$  location based on  $TM_0$ 
4  $t_{prev} \leftarrow$  first timestamp
5  $MP \leftarrow 1000$  // maximum paths in memory
6  $paths \leftarrow$  list with  $MP$  grid points closest to  $L_0$ 
   initialized with probability 1
   // iterate over all TDoA measurements
7 for  $TM \in TMS$  do
8    $t \leftarrow$  current timestamp
9    $\Delta t \leftarrow t - t_{prev}$ 
10   $tmp \leftarrow$  empty list
11  for  $path \in paths$  do
12     $P \leftarrow$  current probability of  $path$ 
13     $E \leftarrow$  current endpoint of  $path$  (parent)
14     $RGP \leftarrow$  reachable grid points along roads with
       $MoT$  within time span  $\Delta t$  starting from  $E$ 
15     $CD \leftarrow$  compass data between  $t_{prev}$  and  $t$ 
      // update new path probabilities
16    for  $CP \in RGP$  do
17      // CP: candidate position
18       $RS \leftarrow$  road segments between  $E$  and  $CP$ 
      // compass probability
19       $P_{comp} \leftarrow$  probability of  $RS$  given  $CD$ 
      // tdoa probability
20       $P_{tdoa} \leftarrow$  probability of  $CP$  given  $TM$ 
       $path_{new} \leftarrow path + RS + CP$ 
21       $P_{new} \leftarrow P \cdot P_{tdoa} \cdot P_{comp}$ 
22      add  $(path_{new}, P_{new})$  to  $tmp$ 
23   $paths \leftarrow$  retain  $MP$  paths from  $tmp$  based on
    highest probability
24   $t_{prev} \leftarrow t$ 
25  $MMT \leftarrow$  reconstruct trajectory along  $path$  with
    highest probability in  $paths$ 
```

P_{tdoa} is based on the probability of the TDoA measurements given CP , the gateway locations, and the standard deviation of LoRa TDoA measurements, e.g., 383 m in our experimental validation. P_{comp} is based on the probability of the compass data between t_{prev} and t given the bearing of the visited road segments RS and the standard deviation of the compass, e.g., 29° in our experimental validation.

Lastly, the MP paths with highest probability are retained to serve as input for the next iteration. After all TDoA measurements are processed, the entire trajectory of the path with highest probability in memory is reconstructed. As mentioned in our scenarios we consider 3 implementations of this algorithm

- TDoA MM which does not take into account the compass values. The algorithm is the same but with $P_{comp} = 1$.

- TDoA MM abs compass: The probability P_{comp} is calculated as the likelihood (combined probability) of all segment headings in RS given the compass data CD.
- TDoA MM rel compass: The probability P_{comp} is calculated as the likelihood (combined probability) of all relative segment headings in RS given the relative compass data CD.

IV. RESULTS

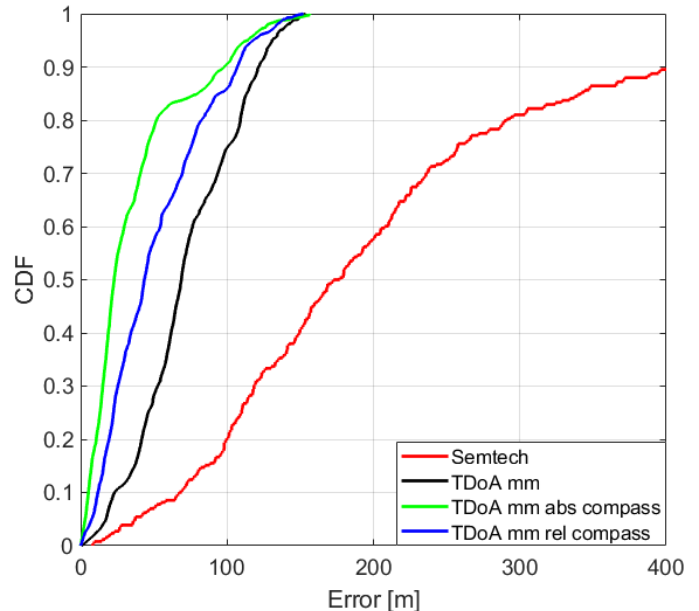


Figure 6: CDF of the different approaches

The obtained accuracy when using the 3rd party Semtech TDoA solver is shown in a red trace in Fig. 6. From the CDF we obtain a median accuracy of 175 m and in 90% of the cases the error is less than 413 m. When we use our TDoA map matching technique (which takes into account the road infrastructure and limited mobility of the considered asset), we obtain a large improvement as we see from the black vs red trace in Fig 6). The median drops to 69m and the 90th percentile drops to 121 m for our approach. The technique is therefore 60% more accurate when compared to Semtech. Additional improvement is further possible if the device has a compass embedded in the LoRa node and transmits its heading information to the network. Using the TDoA MM abs or TDoA MM rel compass algorithm, the median can be further reduced to 23 m and 43 m respectively. The 90th percentile errors using this approach were further reduced to respectively 98 m and 107 m. The error distribution is shown in respectively green and blue in Fig. 6. The additional improvement on the median accuracy is 67% for TDoA MM abs compass algorithm and is 38 % when using the TDoA MM rel compass algorithm. Table I gives an overview of all obtained accuracies and quantifies the improvement when using the Semtech solver as reference.

Method	p50 [m]	n50 [%]	p90 [m]	n90 [%]
Semtech	175	Ref.	413	Ref.
TDoA MM	69	61	121	71
TDoA abs compass	23	87	98	76
TDoA rel compass	43	75	107	74

Table I: Obtained accuracy results for the different approaches

V. CONCLUSION

Tracking assets without using a GPS receiver in the device attached to the asset is possible when using a transmitting LoRa node. LoRaWAN gateways from the surrounding network time-stamp the incoming packets and the network server calculates the position using a TDoA solver. This approach enables joint communication and localization using a single low cost and low power technology. During our test along a walking trajectory, the median accuracy of the Semtech TDoA solver was around 175 m. In this paper, we proposed a map matching technique which takes into account the road infrastructure to reduce this median error to 69m. Our algorithm was able to further reduce the error when the device would have an electronic compass embedded and communicates this info to the network. Fusing the compass data in the proposed TDoA map matching algorithm significantly reduced the error to 23m in our test. Further work will focus on changing the transmission interval from 5 seconds to higher intervals and re-evaluate its tracking accuracy. This would allow higher spreading factors to be used which are beneficial in less dense LoRa networks. Different types of mobility scenarios (e.g. cycling and driving trajectories) and repeating the tests to check the reproducibility of the results are also part of future work.

REFERENCES

- [1] N. Podevijn, D. Plets, J. Trogh, L. Martens, P. Suanet, K. Hendrikse, and W. Joseph, "Tdoa-based outdoor positioning with tracking algorithm in a public lora network," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.
- [2] D. Plets, N. Podevijn, J. Trogh, L. Martens, and W. Joseph, "Experimental performance evaluation of outdoor tdoa and rssi positioning in a public lora network," in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2018, pp. 1–8.
- [3] M. Aernouts, R. Berkvens, K. Van Vlaenderen, and M. Weyn, "Sigfox and lorawan datasets for fingerprint localization in large urban and rural areas," *Data*, vol. 3, no. 2, p. 13, 2018.
- [4] G. G. Anagnostopoulos and A. Kalousis, "A reproducible comparison of rssi fingerprinting localization methods using lorawan," *arXiv preprint arXiv:1908.05085*, 2019.
- [5] N. Podevijn, J. Trogh, A. Karaagac, J. Haxhibeqiri, J. Hoebeke, L. Martens, P. Suanet, K. Hendrikse, D. Plets, and W. Joseph, "Tdoa-based outdoor positioning in a public lora network," in *12th European Conference on Antennas and Propagation (EuCAP 2018)*. IET, 2018, pp. 1–4.
- [6] A. Ghany, B. Uguen, and D. Lemur, "A parametric tdoa technique in the iot localization context," 2019.
- [7] J. Cho, D. Hwang, and K.-H. Kim, "Improving tdoa based positioning accuracy using machine learning in a lorawan environment," in *2019 International Conference on Information Networking (ICOIN)*. IEEE, 2019, pp. 469–472.
- [8] Semtech. (2020) Loracloud geolocation. [Online]. Available: <https://www.loracloud.com/portal/geolocation>
- [9] D. Croce, D. Garlisi, F. Giuliano, A. L. Valvo, S. Mangione, and I. Tinnirello, "Performance of lora for bike-sharing systems," in *2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)*. IEEE, 2019, pp. 1–6.

- [10] N. Accettura, S. Medjiah, B. Prabhu, and T. Monteil, "Low power radiolocation through long range wide area networks: A performance study," in *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2017, pp. 1–8.
- [11] N. Podevijn, D. Plets, M. Aernouts, R. Berkvens, L. Martens, M. Weyn, and W. Joseph, "Experimental tdoa localisation in real public lora networks," in *Tenth International Conference on Indoor Positioning and Indoor Navigation (IPIN 2019)*, 2019, pp. 211–218.