

Combining TDoA and AoA with a particle filter in an outdoor LoRaWAN network

Michiel Aernouts*, Noori BniLam*, Nico Podevijn†, David Plets†,
Wout Joseph†, Rafael Berkvens* and Maarten Weyn*

*University of Antwerp - imec, IDLab - Faculty of Applied Engineering, Belgium, michiel.aernouts@uantwerpen.be

†Department of Information Technology, University of Ghent, imec - WAVES, Belgium

Abstract—Internet of Things (IoT) applications that value long battery lifetime over accurate location-based services benefit from localization via Low Power Wide Area Networks (LPWANs) such as LoRaWAN. Recent work on Angle Of Arrival (AoA) estimation with LoRa enables us to explore new optimizations that decrease the estimation error and increase the reliability of Time Difference Of Arrival (TDoA) methods. In this paper, particle filtering is applied to combine TDoA and AoA measurements that were collected in a dense urban environment. The performance of this particle filter is compared to a TDoA estimator and our previous grid-based combination. The results show that a median estimation error of 199 m can be obtained with a particle filter without AoA, which is an error reduction of 10 % compared to the grid-based method. Moreover, the median error is reduced with 57 % if AoA measurements are used. Hence, more accurate and reliable localization is achieved compared to the performance of other baseline methods.

Index Terms—IoT, LPWAN, LoRaWAN, TDoA, AoA, Particle filter

I. INTRODUCTION

As the Internet of Things (IoT) becomes more relevant for use cases such as electric metering, smart farming, manufacturing automation and asset tracking, an increasing amount of devices becomes interconnected. It is often required that these IoT devices operate continuously on a single battery for years on end. Additionally, users want to keep track of the devices' current location to add context information to their data, e.g by using Global Navigation Satellite System (GNSS) receivers. However, adding a GNSS receiver to a battery-powered device often opposes the low-power requirements of an IoT use case. Also, GNSS can not be used for localization in indoor environments. Therefore, energy-efficient alternatives such as wireless localization methods via Low Power Wide Area Networks (LPWANs), e.g. Sigfox and LoRaWAN, are being investigated. Although localization via these networks generally results in high estimation errors compared to GNSS solutions, they are a better option for IoT use cases that cover both indoor as outdoor scenarios which favor battery lifetime over location accuracy.

In order to estimate the location of a transmitting LPWAN device, wireless localization methods take advantage of the signal's physical characteristics. With Sigfox data from our openly available LPWAN dataset, we have evaluated methods that are based on Received Signal Strength (RSS) measurements [1]. This research has shown that widely used RSS-based methods such as trilateration and proximity result in a

mean estimation error of 721 m to 880 m when implemented in an large urban Sigfox network [2]. With RSS fingerprinting, a lower mean estimation error of 322 m was achieved, but it has to be noted that building and maintaining a fingerprinting database on such a large scale is a very laborious task [3]. Time-based localization methods can lead to lower estimation errors, because they are less susceptible to noisy measurements. Podevijn et al. evaluated Time Difference Of Arrival (TDoA) localization in multiple scenarios with a public LoRaWAN network, and found that it is feasible to obtain a median error of 200 m [4]. By taking previous location estimates into account, and adding a map-based filter, the median error decreased to 75 m. Moreover, the 90th percentile error of the walking scenario was reduced from 579 m to 177 m. This clearly illustrates that adding filtering algorithms to wireless localization improves the accuracy of the location estimates. A third localization option is Angle Of Arrival (AoA). With an antenna array at the receiving LPWAN gateway, it is possible to estimate the direction of a wireless signal and hence, two arrays suffice to compute a location estimate. Unfortunately, the cost and complexity of deploying AoA location estimation systems at the receiver side is considered a major challenge for AoA-based localization. However, a cost-efficient AoA unit for localization in IoT applications has been presented in our recent work [5]–[8].

Because of these recent AoA accomplishments, we are able to start experimenting with AoA localization for LoRaWAN [8]. In this paper, we present the implementation of a particle filter that combines TDoA measurements from a public LoRaWAN network with AoA estimates that were collected with our own unit. This way, we aim to achieve more reliable localization with lower estimation errors. The accuracy of the particle filter is compared to a grid-based approach which is introduced in previous work [9]. For the first time, the grid-based combination of TDoA and AoA is evaluated with real measurements instead of simulations. To the best of our knowledge, this is the first time that real AoA LoRaWAN measurements are evaluated in a localization algorithm.

The remainder of this paper is structured as follows: Section II lists related work regarding LoRaWAN, TDoA, AoA and particle filtering. Section III describes the collected dataset and the implementation of the particle filter. Also, we briefly recap the grid-based combination that will be used as a baseline in our comparison to the particle filter. Section IV

shows the results of this comparison, which are discussed in Section V. Finally, we conclude the paper and introduce our future work in Section VI.

II. RELATED WORK

In this section, more details are given about the current state of the art on TDoA and AoA localization with LoRaWAN. Additionally, a brief explanation about particle filtering is provided.

A. LoRaWAN

LPWANs allow IoT devices to communicate their sensor data over ranges of multiple kilometers while maintaining a low power consumption. This is possible because LPWAN transmitters are designed to send only small payloads, which limits the required operating bandwidth. Generally, a trade-off between communication range and data rate has to be considered when implementing an LPWAN in an IoT application.

LoRaWAN is an LPWAN which uses a Chirp Spread Spectrum (CSS) based modulation named LoRa, and operates in the license free 868 MHz and 915 MHz bands in Europe and the US respectively. With a so called 'spreading factor' that tunes the LoRa modulation, a balance between range and data rate is regulated. A low spreading factor allows a data rate of up to 37.5 kbps, but limits the communication range to 5 km. Increasing the spreading factor results in a range of up to 15 km, but decreases the data rate to 300 bps [10]. Of course, the communication also strongly depends on the environment type. Urban environments will have smaller ranges due to the limited Line Of Sight (LOS) between transmitter and receiver. LoRaWAN end devices do not connect to a single receiving gateway. Instead, all gateways within the devices' communication range receive and forward the signal to a LoRaWAN server. Consequently, localization techniques such as trilateration and multilateration can be applied because multiple measurements of one signal are available.

B. Time Difference of Arrival

For time-based localization methods, it is crucial that the time of arrival of a transmitted signal is accurately detected at the receivers. This is not possible for Ultra Narrowband (UNB) networks such as Sigfox, because the rising edge of the signal pulse can not be determined accurately enough [11]. LoRaWAN has a bandwidth of 125 kHz to 250 kHz, which makes it a suitable candidate for time-based localization. For LoRaWAN specifically, TDoA algorithms are a popular low-power approach. Since the transmitting device does not have to be in sync with the gateways, there is no need to add synchronization hardware that would drain the battery. Via a network of precisely synchronized gateways, the location of a LoRaWAN transmitter can be calculated based on the time difference of arrival relative to a reference gateway. This time difference between a pair of gateways can be represented by a hyperbolic curve that shows all possible transmitter locations. At least three curves are required to find a reliable intersection which represents the location estimate, therefore the time of

arrival from at least four receiving gateways is needed. The error of the TDoA location estimate strongly depends on the timestamp precision of the receiving gateways. Because radio signals propagate at light speed, the estimation error can go up to 300 m if a gateway provides a timestamp with a precision of 1 μ s.

Experiments in an outdoor public LoRaWAN network have shown that commercial TDoA solvers can locate a mobile device with a median error of 200 m without additional tracking algorithms [4]. Other research demonstrated that static devices can be located with an accuracy of around 100 m [12].

C. Angle of Arrival

As previously discussed, LPWAN devices operate with small bandwidths to transmit their payloads. Therefore, accurate timestamp detection for time-based localization is not guaranteed. Accordingly, the localization accuracy of the time-based approach can be very low. In contrast to time-based localization, AoA-based localization techniques are independent of the timestamp precision at the receiving gateways. Instead, they estimate the angle between the transmitter and the receiver by measuring the phase of the received signal at different points in space using array antennas. Multiple spatially distributed array antennas are required to provide a location estimate of the transmitting device. With an AoA estimation method, every array antenna system can provide the direction of the received signals with respect to its own location [13]. Combining the direction information, either by a triangulation process or probabilistic models, can provide a location estimate of the transmitting device [14], [15]. AoA estimation techniques have the potential to provide accurate localization solutions for IoT applications [15]. However, to the best of our knowledge, commercially available AoA-based localization systems do not yet exist. This absence can be attributed to the complexity that is associated with implementing them. Nevertheless, hardware and software solutions have recently been introduced to reduce the cost and complexity of deploying AoA-based localization techniques for IoT applications [5]–[7].

Recently, RTL-Array has been introduced as a low cost AoA estimation unit for IoT applications [8]. The angular estimation accuracy of the RTL-Array was verified in an anechoic chamber for 6 antenna elements. The estimated AoA accuracy was below 1 degree in the 868 MHz band. For the research that we present in this paper, a single RTL-Array with 8 antenna elements is deployed to estimate the AoA of the received LoRa signals in an outdoor Non Line Of Sight (NLOS) environment. Furthermore, the estimated AoA values are used as input data for a particle filter. It is worth mentioning that the conventional Beamforming (CBF) technique has been utilized to estimate the AoA of the received signal. More information regarding the different AoA estimation techniques can be found in our recent work [13].

D. Particle filtering

With Bayesian filtering, the state of a time-varying system that is observed with noisy measurements can be estimated by collecting dynamic variables such as speed, orientation and position. There is a very wide range of applications that benefit from Bayesian filtering, e.g. brain imaging, biological processes, telecommunication, GNSS, navigation with inertial sensors, etc. Bakkali et al. studied the performance of LoRaWAN for indoor localization with a Kalman filter, which is a type of Bayesian filtering. In a small indoor environment, a mean estimation error of 1.72 m was achieved with NLOS RSS measurements [16].

Particle filtering is another variation of Bayesian filtering that uses Monte Carlo approximations to represent a target distribution. This is done via a set of particles which are distributed according to the target density. Each particle in the distribution has a weight which is updated according to the probability that the particle corresponds with a measurement, e.g. the time of arrival at a receiving LPWAN gateway. Through Sequential Importance Resampling (SIR), particles with small weights are removed and particles with large weights are duplicated. [17].

For this research, we implement a bootstrap particle filter that uses TDoA and AoA measurements to update and resample the weights of the particles. The details of our implementation can be found in Section III-C.

III. METHODS

First, this section describes where and how we collected TDoA and AoA measurements that are used as input for the particle filter. Second, we explain the grid-based approach to which we later compare the results of the particle filter implementation. Third, we discuss the particle filter implementation itself.

A. Dataset

In our previous work, we collected large LPWAN datasets with ground truth information, to provide researchers with a tool to develop and evaluate their localization methods for IoT applications [1]. Although the latest version of this dataset contains precise timestamps that can be used for TDoA localization, it does not incorporate AoA measurements that are required for this research. Therefore, new data had to be gathered. Our initial plan was to do so via The Things Network (TTN), which is a global community-based LoRaWAN network [18]. We installed LoRaWAN gateways in and around Antwerp, Belgium and connected them to TTN, but unfortunately, our gateways do not always forward their received data. Hence, we collected 388 TDoA measurements in the city center of Antwerp using a public LoRaWAN network that is operated by Proximus, a national LoRaWAN network provider. In Figure 1a, gateway locations of the Proximus network in and around Antwerp are illustrated. The gateways that are shown in this figure have received one or more of our transmitted messages, and are spread in an area of over 426 km². The LoRaWAN transmission

locations of our experiment cover only a minor part of this large area. Specifically, the 19 locations that are depicted in Figure 1b cover a total area of 0.4 km². While walking a trajectory in this area, we stopped at each location and sent twenty LoRaWAN messages with spreading factor 12. For all messages, the public LoRaWAN network provides timestamps from the receiving gateways with nanosecond precision. These measurements are used in both the grid-based approach as in the particle filter, which are explained in Section III-B and Section III-C respectively.

Apart from the TDoA measurements, we gathered AoA measurements on the same 19 locations. The location of the AoA unit is displayed as a red marker in Figure 1b. We collected AoA measurements for each of location, and merged them with the TDoA measurements for the same location afterwards. Since the AoA unit is still in an experimental stage, it is not possible to continuously record and process LoRaWAN signals. Due to this limitation, only ten measurements were obtained per location. In order to have an angle estimate that can be merged with each of the TDoA measurements, we used each angle estimate twice. The focus of this research is to combine TDoA and AoA data to reduce estimation errors and increase estimation reliability. Details about the AoA estimation algorithm itself will not be discussed in this paper, but can be found in [13].

B. Grid-based combination of TDoA and AoA

In our previous work, we proposed a grid-based approach to combine TDoA and AoA measurements in a LoRaWAN network [9]. To demonstrate this method, we simulated 10 000 LoRaWAN messages which were received by two TDoA-enabled gateways 'RX 1' and 'RX 2', which are both illustrated in Figure 2. Additionally, we assumed that 'RX 1' is also able to calculate an AoA estimate. With this research, we illustrated that it is feasible to estimate the location of a transmitter using only a limited amount of receiving gateways. The grid-based combination includes three steps: first, the measurements of two synchronized gateways are used to calculate a TDoA probability density map. This map displays a long hyperbolic curve that represents all possible location estimates for the LoRaWAN transmitter. Second, the associated measurement at the AoA gateway draws another probability density map, where locations that match the AoA estimation get the highest probability. Third, the TDoA and AoA probability maps are added to each other, resulting in a new map in which the cells where the TDoA hyperbola intersect with the AoA estimate get the highest probability. Hence, a location estimate is obtained using a limited amount of receivers. The complete method is briefly summarized in Figure 2.

C. Particle Filter

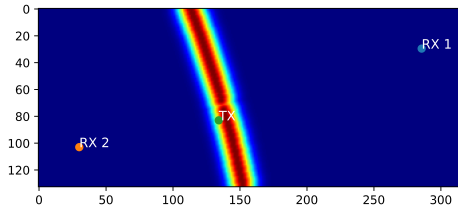
In order to combine the TDoA and AoA measurements, we apply a bootstrap particle filter. Our implementation consists of four main steps which are summarized in Algorithm 1, and explained in detail in this subsection.



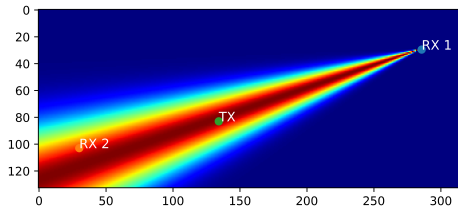
(a) The locations of the public LoRaWAN gateways that received the transmissions and provided us with timing information.

(b) The area where LoRaWAN messages were transmitted. Circles indicate the transmission locations, and the location pin shows the location of our AoA unit.

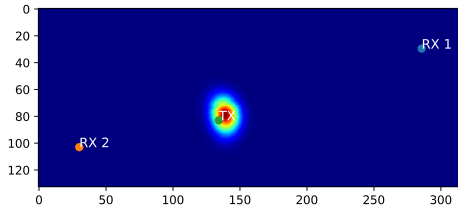
Fig. 1: (a) The area that is covered by the receiving TDoA gateways. (b) The measurement locations and the location of the AoA unit.



(a) TDoA



(b) AoA



(c) TDoA + AoA

Fig. 2: Illustration of the grid-based method. The TDoA map in (a) is combined with the AoA map in (b). The resulting map in (c) is used to calculate the final location estimate.

Algorithm 1 Particle filter

Input: N LoRaWAN messages

- 1: **Initialize** M particles
 - 2: **for** $i \leftarrow 1$ to N **do**
 - 3: **Predict** new particle states
 - 4: **Update** particle weights
 - 5: **Resample**
 - 6: **end for**
-

Firstly, a set of particles has to be initialized a single time. This step is summed up in Algorithm 2. A TDoA estimator that is introduced by Li et al. is applied to the first LoRaWAN message in the dataset [19]. This results in an initial location estimate μ that is used as the mean of a normal distribution. As we have seen in Section II-B, a median localization error of 200 m is feasible to obtain through TDoA localization with LoRaWAN. Therefore, we use a standard deviation of 300 m for the normal distribution, so that the initial estimate lies within a safe error margin. Subsequently, M particles are initialized, and a location ρ for each of them is randomly drawn from the distribution. Furthermore, each particle is assigned a speed ν , direction θ and a weight ω . Our dataset was collected on foot, so it is logical that the motion model for the particles is chosen accordingly. Hence, the speed of a particle is randomly sampled from a uniform distribution ranging from 0 m/s to 1.5 m/s. Similarly, a particle's direction is a uniform random variable between 0 degrees and 360 degrees. In this stage, all particles will have the same weight of $1/M$. Figure 3a is a visual representation of the initialization. The ground truth measurement location is displayed as a green location pin, and particles are shown as colored circles with lines. These lines illustrate the speed and direction of each



(a) Initialization of a set of 1000 particles around an initial TDoA estimate.

(b) The particle set after three iterations of updating and resampling.

(c) After the resampling step in (b), new particle states are predicted.

Fig. 3: A visual representation of the particle filter implementation, where particles are shown as colored circles with lines. The speed and direction of a particle are shown by its line, whereas color indicates its weight: the lighter the color, the higher the weight. The ground truth measurement location is shown as a green location pin. The location estimate after resampling, only shown in (b), is displayed as a red pin.

particle, whereas the weight is shown in the color of each particle. Initially, all particles will be shown in the same color, since they all have the same weight. In Figures 3b and 3c, which will be explained next, it becomes clear that lighter colors indicate particles with higher weights.

Algorithm 2 Initialize M particles

```

1: for  $i \leftarrow 0$  to  $M$  do
2:    $\rho \sim \mathcal{N}(\mu, 300 \text{ m})$ 
3:    $\nu \sim [0, 1.5] \text{ [m/s]}$ 
4:    $\theta \sim [0, 360] \text{ [degrees]}$ 
5:    $\omega = 1/M$ 
6:    $particles_i = [\rho, \nu, \theta, \omega]$ 
7: end for

```

Secondly, Algorithm 3 explains how the new state of each particle is predicted. We iterate over the LoRaWAN messages in the dataset and calculate the time difference Δt between the current and previous message. Because we have to take sudden changes in motion of the LoRaWAN transmitter into account, random noise is added to the speed and direction of each particle. The new speed of each particle is randomly chosen from a normal distribution with the previous speed as mean value, and a standard deviation of 0.5 m/s. The only limitation is that the new particle speed has to stay within the initial bounds of 0 m/s and 1.5 m/s. Similarly, a normal distribution with the previous particle direction as the mean value and $\Delta t * 10$ degrees as the standard deviation, is used to determine a new random direction. Now that both speed and direction are altered, a new particle location can be calculated based on Δt . Figure 3c shows how the prediction step looks like. New speeds and directions are chosen for the particles of Figure 3b, and particles are moved to new locations based on these new values and Δt .

Thirdly, the weights of all particles are updated, as clarified in Algorithm 4. This is done by comparing the list of

Algorithm 3 Predict new particle states

Input: Δt

```

1: for  $i \leftarrow 0$  to  $M$  do
2:    $particles_{i,\nu} \sim \mathcal{N}(particles_{i,\nu}, 0.5)$ 
3:    $particles_{i,\theta} \sim \mathcal{N}(particles_{i,\theta}, (10 * \Delta t))$ 
4:   Move  $particles_{i,\rho}$  in new direction  $\theta$  with a distance of  $\Delta t * \nu$ 
5: end for

```

timestamps as well as the AoA estimate α of a LoRaWAN message. First, a list of time differences $\Delta \tau$ between all receiving gateways and the first gateway is derived from the timestamps. Then, we look at the location of each particle and calculate the distance to the receiving gateways. Hence, a theoretical time of flight between the particle and each of the gateways can be calculated and subsequently, a similar list of time differences $\Delta \tau_p$ is acquired. With this information, we are able to determine the probability of a particle's predicted measurements matching with the actual time measurements. The probability is calculated by comparing with a normal distribution with mean $\Delta \tau$. The standard deviation of the distribution is set to $1.8 \mu\text{s}$, as our previous experiments with the public LoRaWAN network in Antwerp demonstrate that the timing error is normally distributed with a mean of $0 \mu\text{s}$, and a standard deviation of $1.8 \mu\text{s}$ [20]. The resulting value x_{TDoA} is multiplied with the current weight of the particle. In a similar way, the difference between the actual angle estimate α and the angle between a particle and the AoA unit α_p is calculated and used in Gaussian distribution to determine how likely it is that they match. We chose to set the standard deviation of the distribution to 3 degrees, since prior technical tests with the AoA unit indicated that it is feasible to obtain an angle estimate within this margin. Again, the resulting value x_{AoA} is multiplied with the particle's weight. After all

particles have been updated, we normalize them by dividing their weight by the sum of all particle weights.

Algorithm 4 Update particle weights

Input: $\Delta\tau, \alpha$

```

1: for  $i \leftarrow 0$  to  $M$  do
2:    $x_{TDoA} = \mathcal{N}((\Delta\tau - \Delta\tau_p), 1.8)$ 
3:    $particles_{i,\omega} = particles_{i,\omega} * x_{TDoA}$ 
4:    $x_{AoA} = \mathcal{N}((\alpha - \alpha_p), 3)$ 
5:    $particles_{i,\omega} = particles_{i,\omega} * x_{AoA}$ 
6: end for
7: for  $i \leftarrow 0$  to  $M$  do
8:    $particles_{i,\omega} / = sum(particles_{i,\omega})$ 
9: end for

```

Finally, the particles are resampled in Algorithm 5 based on their updated weights. A random variable β is selected from a uniform distribution between zero and twice the value of the largest weight in the particle set. Moreover, two values j and sum are initialized as zero. As long as β is larger or equal than the sum value, j is increased by 1 and sum is increased with the weight of the particle at index j . When sum exceeds β , the particle at index j is selected for resampling. As a result, particles with high weights will populate a new set of M particles more often than particles that have with lower weights. Additionally, a location estimate is computed in this step by taking the mean location of all particles in the new set. Figure 3b shows the particle set after a few iterations of predicting and resampling. Particles with a light color have higher weights than those with a darker color. The green location pin shows the ground truth TX location, whereas the red pin indicates the location estimate after resampling. In this figure, the spatial distribution of the particles clearly shows the effect of the AoA estimates on the particle filter's measurement model.

Algorithm 5 Resample

```

1:  $j = 0$ 
2:  $sum = 0$ 
3: for  $i \leftarrow 0$  to  $M$  do
4:    $\beta \sim ([0, 2 * max(particles_{i,\omega})] + i) / M$ 
5:   while  $\beta \geq sum$  do
6:      $j = (j + 1) \bmod M$ 
7:      $sum = sum + particles_{j,\omega}$ 
8:   end while
9:   Resample  $particles_j$ 
10: end for
11: return location estimate

```

There are a few important differences between this particle filter and the grid-based approach from Section III-B. Firstly, the particle filter employs all available TDoA and AoA information, whereas the grid method uses only two TDoA-enabled gateways and one AoA estimate. Secondly, the probabilistic combination of TDoA and AoA works in a different way. In the grid method, a hyperbolic curve is

calculated via a least-squares TDoA solver, and the geographic coordinates of this curve are used as the mean of a Gaussian distribution. Although the particle filter also applies a Gaussian distributions, it uses the time and angle measurements itself as the mean instead of the location estimate that resulted from a solver. This way, the uncertainty of the measurements is taken into account. Both methods will be evaluated and compared to each other in Section IV.

IV. RESULTS

In this section, we list the results of our research in some comprehensive plots and tables. A more detailed discussion of the results is provided in Section V.

As explained in Section III-A, the dataset contains 388 LoRaWAN messages which in turn have an amount of TDoA measurements that can vary from one message to another, and a single AoA estimate. In Figure 4, the amount of TDoA measurements per message, as well as the mean amount per location are shown. On average, the LoRaWAN messages are received by 6.32 gateways from the public LoRaWAN network.

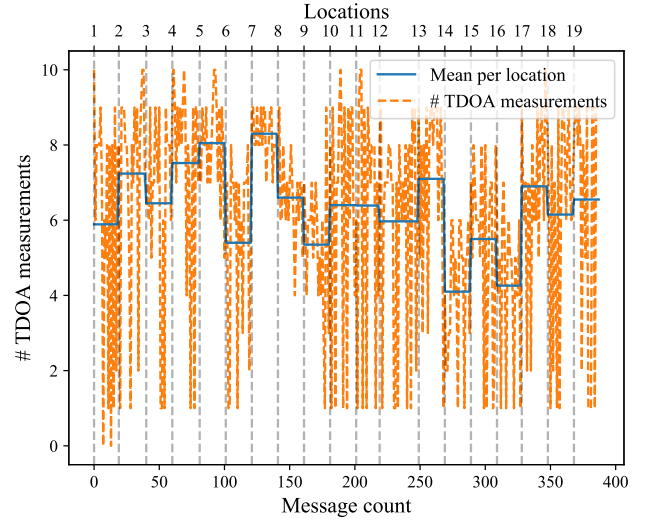


Fig. 4: The number of receiving TDoA gateways of each LoRaWAN message.

Every LoRaWAN message has exactly one AoA estimate, their errors are depicted in Figure 5. The mean AoA error is 6.37 degrees, with a standard deviation of 3.05 degrees. 95 % of the estimation errors lie within 12.02 degrees. This plot also illustrates the precision of the AoA estimations. We can see that the estimations at locations with LOS tot the AoA unit (i.e. locations 2, 18 and 19) are very stable, whereas NLOS locations such as location 7 and 16 have less precise estimates.

The dataset is used to compare three localization methods. First, TDoA localization is applied without adding AoA information. For this purpose, we employ the same solver that we use to initialize a particle set [19]. This method resulted in 274 locations estimates with a median error of 331 m, and a 95th percentile of 1167 m. Second, the grid-based approach

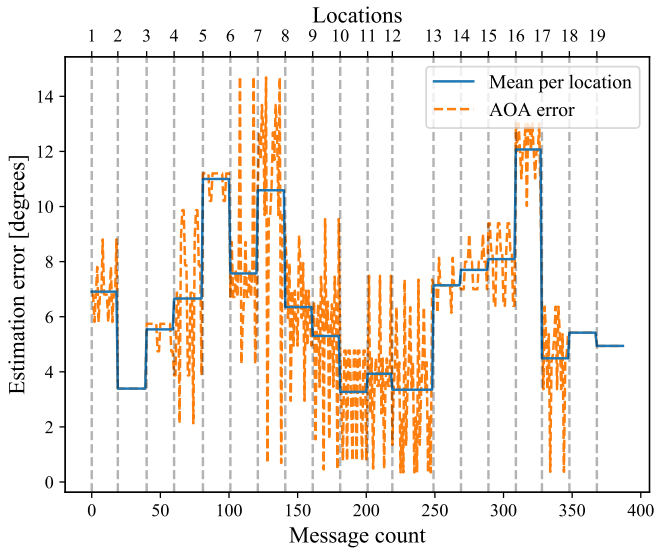


Fig. 5: The AoA estimation error of each LoRaWAN message.

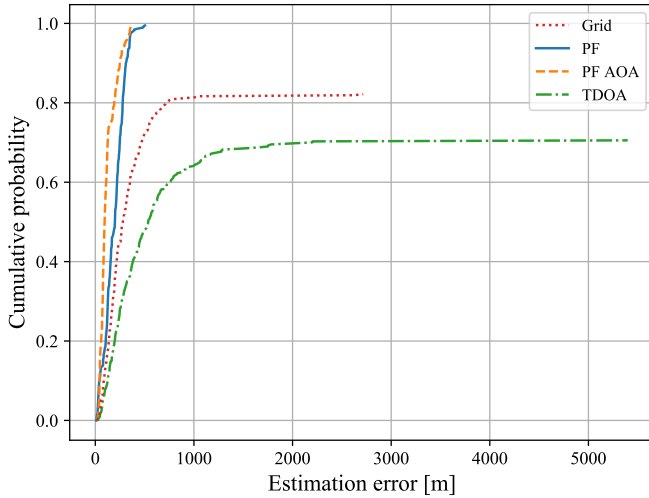


Fig. 6: Cumulative distribution of the estimation errors for each method.

of Section III-B returned 319 estimates with a median error of 221 m, and a 95th percentile of 657 m. Finally, we assess the results of the particle filter implementation. When no AoA data is used, only one message did not return a location estimate. The median error here was 199 m, and the 95th percentile error was 347 m. After adding AoA information, the median error drops to 94 m and the 95th percentile error to 297 m. Clearly, the particle filter with AoA outperforms the other methods in our comparison, both in accuracy as in reliability.

Figures 6 and 7 compare the estimation errors of these methods. Table I provides a concise overview of the amount of returned estimates per method as well as the mean, median and 95th percentile error of each method. Furthermore, we plotted the errors for each message in Figure 8 to show a time lapse of the estimation errors. The plot indicates that the TDOA and

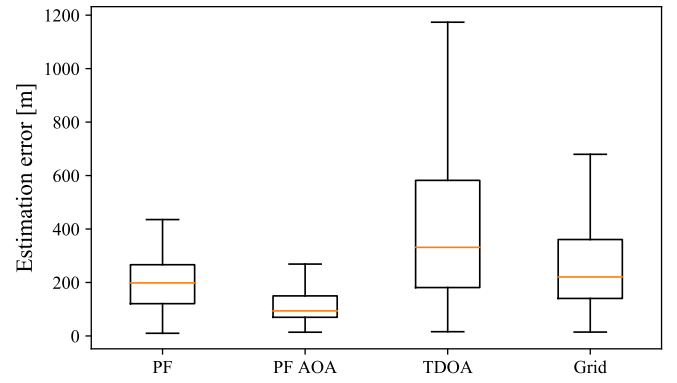


Fig. 7: Boxplot of the estimation errors for each method. Outliers are excluded from the plots.

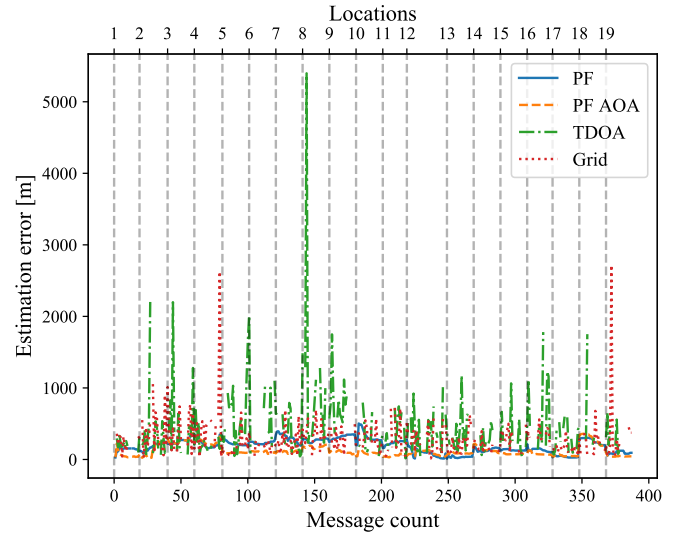


Fig. 8: A general comparison of the accuracy and reliability of each method by showing the estimation errors for each message in the dataset.

grid-based results have a lot of uncertainty compared to the particle filter results, this observation is confirmed by the 95th percentile errors in Table I. If a location estimate could not be obtained, the line plot is interrupted at that message. Because the outliers of the TDoA and grid-based methods obscure the results of the particle filter in Figure 8, a similar plot that only shows the particle filter results is given in Figure 9.

V. DISCUSSION

In Section III-A, we explained how we have collected TDoA measurements via a public LoRaWAN network. The dataset that we now use contains 388 LoRaWAN messages which were sent using spreading factor 12, only a few seconds between subsequent messages at a location. Because of the duty cycle limitations of LPWAN, it is not feasible to do this in a real IoT application. However, for now we only want to evaluate the localization algorithms.

TABLE I: The localization results of four methods.

	Amount of estimates	Mean error [m]	Median error [m]	95th percentile [m]
TDoA	274	455	331	1167
Grid-based TDoA + AoA	319	286	221	657
Particle filter without AoA	386	189	199	347
Particle filter with AoA	387	122	94	297

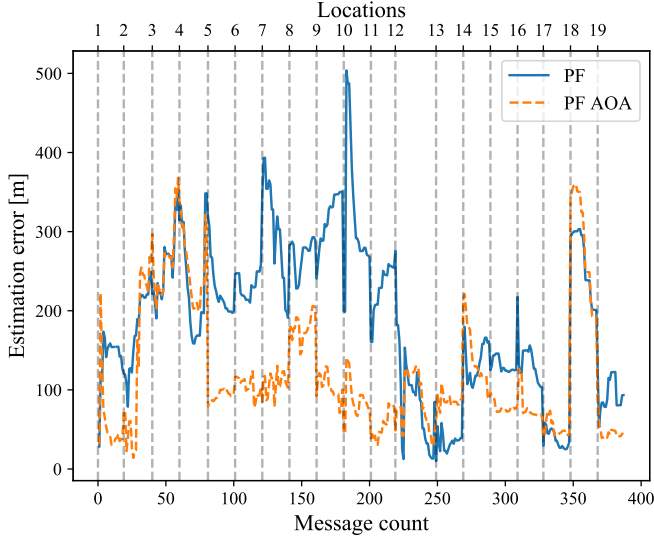


Fig. 9: We compare the performance of the particle filter by plotting the estimation errors for each message in the dataset. Both the errors with and without AoA data are shown.

Furthermore, we used our own AoA unit to obtain angle estimates for each of the measurement locations. This data was collected simultaneously with our first TDoA data via TTN, on the 19 measurement locations from Figure 1. The workflow of the AoA unit and the AoA estimation algorithms are not discussed here, since it does not lie within the scope of this paper. The interested reader is referred to our related work on AoA in Section II-C. Due to the aforementioned technical difficulties with our gateways, the initial dataset that we collected was unusable. Luckily, we were able to extract the AoA data and merge it with the new TDoA measurements that were collected at a different date at the exact same locations. For now, the AoA unit has some technical limitations that we have to take into account for our experiments. As mentioned before, it is not yet possible to continuously record and process LoRaWAN signals. Therefore, only ten AoA estimates are calculated for each location. Since we have twenty LoRaWAN messages per location, AoA estimates were re-used to complete the dataset. We believe that this will only have a limited impact on the results, since no AoA estimate is re-used more than once. Further analysis on this will be done after new datasets have been collected in our future work. Also, the AoA unit has a limited range in NLOS environments such as the area where we collected the data. This is why the measurement locations are all within 600m from the unit.

Next, we discuss the localization methods that we evaluated

with the dataset. First, we validated a TDoA estimator proposed by Li et al. [19]. For the 388 messages, the estimator returned 274 location estimates, which covers 71 % of the dataset. This is mainly because, like many TDoA solvers, this estimator requires at least four receiving gateways to compute a result. Figure 4 shows that not every entry in the dataset contains this many measurements. The TDoA solver resulted in a mean error of 445 m, and a 95th percentile error of 1167 m. We see in Figures 7 and 8 that this method is less accurate than the other methods in the comparison.

Secondly, we test the grid-based approach that we simulated in our previous work [9]. A mean error of 286 m and a 95th percentile error of 657 m were achieved with this method. Although the mean error is reduced with 37 % compared to the previous TDoA results, Figure 8 indicates that there is still a lack of accuracy. Also, only 82 % of the messages in the dataset resulted in a location estimate. Thus, the reliability of this method has increased with 11 % compared to the TDoA solver. The first reason is that at least two TDoA measurements have to be available to draw a hyperbolic curve, otherwise the method fails. The second reason is that it is not always possible to draw a hyperbolic curve because of geometric constraints. A few important comments must be made on these results. First, the grid-based method makes use of only the first two receiving TDoA gateways, and one gateway, whereas other methods employ as much input data that they can get. Second, the locations of the TDoA and AoA gateways are of importance for this method. In our previous simulations, we assume that the AoA gateway is at the exact same location as the TDoA reference gateway, which results in an optimal overlap of a hyperbolic curve and an angle estimate [9]. However, this is not the case for the dataset in this research, as we could not install the AoA unit at one of the public LoRaWAN gateway locations. As a result, the overlapping area on the probability density grid is larger, and the errors are higher as well. Therefore, we will re-evaluate this approach as soon as we have installed the AoA unit at one of our own gateway locations.

Lastly, the particle filter method with and without AoA estimates is evaluated. The estimation errors are significantly lower compared to the previous methods, especially if AoA data is used. As the first message in the dataset is used to initialize the particle set, no location estimate is calculated. Hence the particle filter with AoA returned an estimate for all messages except the first one. The particle filter without AoA returned one less estimate, since one of the messages in the dataset contains no TDoA measurements. Therefore, there was no information to update and resample the particles. Our current implementation does not return a location estimate in this

case. However, a particle filter could still return an estimate by updating the particles with uniformly distributed weights. Adding this functionality to our future implementation will require only a minor effort.

For the particle filter without AoA, a mean error of 189 m and a 95th percentile of 347 m was obtained. For the particle filter with AoA, a mean error of 122 m and a 95th percentile of 297 m was obtained. Hence, adding AoA to the particle filter leads to a 35 % decrease of the mean location error. Compared to the TDoA and grid-based methods, the mean location error was decreased with 73 % and 57 % respectively. Furthermore, the particle filter with AoA was able to calculate an estimate for all 388 LoRaWAN messages, whereas the TDoA solver could do this for only 274 of them. The grid-based method could calculate an estimate for 319 messages. Clearly, the particle filter is the most reliable method in this comparison.

An important factor that determines the performance of the particle filter is the initialization step. In the current implementation, a Gaussian distribution is applied around an initial TDoA solver estimate, with a standard deviation of 300 m. However, if the initial estimate would be closer to the solvers' 95th percentile error of 1167 m, there is a significant risk that many particles are initiated in the wrong area, or that they are at least very sparsely distributed, given a computationally feasible amount of particles. Additionally, particles miss the opportunity of being initiated close to the ground truth location if the location estimate has a low error. Therefore, instead of using a fixed initial standard deviation, a more reliable method should be implemented. One possible method is to calculate a confidence radius for the initial estimate based on the available measurements, and then using this confidence radius as the standard deviation of the normal distribution.

Because a particle filter takes the previous states of its particles into account, outliers occur far less frequently compared to the previous methods. As can be seen in Figure 9, outliers mostly arise at the first transmitted message of a location. A logical explanation for this is that there is a relatively big time difference between the last message of a location and the first message of the next location. Hence, the particles will move further away from each other, resulting in more uncertainty for the next location estimate. In many cases, the estimation errors decrease after the first estimate at a location, because the particles start to converge again. For some locations, errors increase after the first estimate. This could be caused by higher AoA errors, or a lower amount of receiving TDoA gateways at those locations. Further research is required to state this with certainty.

Finding a clear correlation between the location estimation errors of Figure 9, the amount of TDoA measurements and the AoA errors in Figures 4 and 5 is not self-evident. For example, the errors of locations 10 and 18 show that an above average amount of TDoA measurements and a below average AoA estimation error does not necessarily result in accurate estimation errors. On the other hand, less TDoA gateways or a high AoA error do not always result in the opposite, as seen

in the particle filter with AoA plot of location 5. Therefore, our future work includes studying the quality of our measurements to see if a more distinct correlation can be observed. For now, we do not filter any input data in the particle filter, so TDoA measurements with an inaccurate timestamp could have a negative effect on the resulting location estimate.

VI. CONCLUSION

As stated in the introduction of this paper, our goal was to achieve reliable localization with low estimation errors by combining LoRaWAN TDoA and AoA measurements by means of a particle filter. Compared to a regular TDoA solver, the mean error of the particle filter implementation with AoA is 73 % lower, whereas the 95th percentile error is 74 % lower. Moreover, the particle filter was able to obtain a location estimate for every LoRaWAN message, contrary to the TDoA solver which only returned a location estimate for 71 % of the messages. The grid-based combination performs better than the TDoA solver, but is still outperformed by the particle filter which has a mean location error reduction of 57 % and a reliability increase of 21 % compared to the grid-based method. Concisely, the particle filter surpasses the other methods in both reliability as accuracy, since estimates with lower errors are maintained even when no location can be estimated through the TDoA or grid-based methods. Another important achievement is the use of real AoA data instead of simulations. To the best of our knowledge, we are the first to implement real AoA data for localization in LoRaWAN networks.

Nevertheless, our future work includes a lot of possible optimizations that could further improve the results of the particle filter. First, the input data for the particle filter can be improved. Apart from TDoA and AoA, RSS will be added to the measurements. We intend to make use of our own gateways to collect these measurements, so that we could not only evaluate the localization algorithms, but also the effect of physical changes in the network, e.g. the locations of the gateways. As previously mentioned in the discussion, we intend to filter the TDoA data to remove inaccurate timestamps from the measurements. Also, we want to expand the AoA data, so that it not only includes an azimuth angle, but also the elevation.

Of course, the elevation estimate is only useful if the elevation angle between a particle location and the AoA gateway location can be calculated. To do this, map information has to be included to estimate the altitude of a particle. This map information will be used for many more optimizations. For example, the particles can be redistributed so that they are only located on roads, and their motion model can be adapted to speed limits and other traffic information. Additionally, we will analyze how to detect when the application context switches and change the measurement and motion models of the particle filter accordingly, e.g. when the context switches from walking to driving a vehicle. To do so, we will collect data in a realistic IoT application that keeps LPWAN limitations such as duty cycle regulations into account.

REFERENCES

- [1] 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, 4 2018.
- [2] M. Aernouts, B. Bellekens, R. Berkvens, and M. Weyn, "A Comparison of Signal Strength Localization Methods with Sigfox," in *2018 15th Workshop on Positioning, Navigation and Communications (WPNC)*. IEEE, 10 2018, pp. 1–6.
- [3] T. Janssen, M. Aernouts, R. Berkvens, and M. Weyn, "Outdoor Fingerprinting Localization Using Sigfox," in *IPIN 2018 - 9th International Conference on Indoor Positioning and Indoor Navigation*, 2018.
- [4] 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, pp. 1–9, 5 2018.
- [5] N. BniLam, J. Steckel, and M. Weyn, "Synchronization of Multiple Independent Subarray Antennas: An Application for Angle of Arrival Estimation," *IEEE Transactions on Antennas and Propagation*, vol. 67, no. 2, pp. 1223–1232, 2 2019.
- [6] J. Steckel, D. Laurijssen, A. Schenck, N. BniLamf, and M. Weyn, "Low-cost Hardware Platform for Angle of Arrival Estimation Using Compressive Sensing," in *12th European Conference on Antennas and Propagation (EuCAP 2018)*. Institution of Engineering and Technology, 2018, pp. 222 – 225.
- [7] N. BniLam, J. Steckel, and M. Weyn, "Synchronization of Multiple Independent Sub-Array Antennas for IoT applications," in *12th European Conference on Antennas and Propagation (EuCAP 2018)*. Institution of Engineering and Technology, 2018, pp. 125 – 130.
- [8] N. BniLam, D. Joosens, J. Steckel, and M. Weyn, "Low Cost AoA Unit for IoT Applications," in *2019 13th European Conference on Antennas and Propagation (EuCAP)*. Krakow, Poland: European Association on Antennas and Propagation, 2019, pp. 3–7.
- [9] M. Aernouts, N. BniLam, R. Berkvens, and M. Weyn, "Simulating a Combination of TDoA and AoA Localization for LoRaWAN," in *Advances on P2P, Parallel, Grid, Cloud and Internet Computing*, vol. 96. Springer, 2020, pp. 756–765.
- [10] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 855–873, 2017.
- [11] H. Sallouha, A. Chiumento, and S. Pollin, "Localization in long-range ultra narrow band IoT networks using RSSI," in *IEEE International Conference on Communications*, 2017.
- [12] B. C. Fargas and M. N. Petersen, "GPS-free geolocation using LoRa in low-power WANs," in *2017 Global Internet of Things Summit (GIoTS)*. IEEE, 6 2017, pp. 1–6.
- [13] N. BniLam, E. Tanghe, J. Steckel, W. Joseph, and M. Weyn, "ANGLE: ANGular Location Estimation Algorithms," *IEEE Access*, vol. 8, pp. 14 620–14 629, 2020.
- [14] A. Bensky, *Wireless Positioning: Technologies and Applications*. Artech House, 2016.
- [15] N. BniLam, G. Ergeerts, D. Subotic, J. Steckel, and M. Weyn, "Adaptive probabilistic model using angle of arrival estimation for IoT indoor localization," *2017 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2017*, vol. 2017-January, pp. 1–7, 2017.
- [16] W. Bakkali, M. Kieffer, M. Lalam, and T. Lestable, "Kalman filter-based localization for Internet of Things LoRaWAN™ end points," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, vol. 2017-Octob. IEEE, 10 2017, pp. 1–6.
- [17] S. Särkkä, "Bayesian filtering and smoothing," *Bayesian Filtering and Smoothing*, pp. 1–232, 2010.
- [18] "The Things Network." [Online]. Available: <https://www.thingsnetwork.org/>
- [19] S. Li and K. C. Ho, "Accurate and Effective Localization of an Object in Large Equal Radius Scenario," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8273–8285, 12 2016.
- [20] N. Podevijn, D. Plets, M. Aernouts, R. Berkvens, L. Martens, M. Weyn, and W. Joseph, "Experimental TDoA Localisation in Real Public LoRa Networks," in *IPIN 2019 - 10th International Conference on Indoor Positioning and Indoor Navigation*, Pisa, Italy, 2019.