Range-free Localization Algorithms with Mobile Anchors at Different Altitudes: A Comparative Study

Conference Paper · January 2020

DOI: 10.1145/3369740.3369766

3 authors:

Francesco Betti Sorbelli
Università degli Studi di Perugia
60 PUBLICATIONS **577** CITATIONS
SEE PROFILE

Cristina M. Pinotti
Università degli Studi di Perugia
214 PUBLICATIONS **1,901** CITATIONS
SEE PROFILE

Giulio Rigoni
Sapienza University of Rome
12 PUBLICATIONS **86** CITATIONS
SEE PROFILE

Range-free Localization Algorithms with Mobile Anchors at Different Altitudes: A Comparative Study

Francesco Betti Sorbelli
University of Perugia, Italy
francesco.bettisorbelli@unipg.it

Cristina M. Pinotti
University of Perugia, Italy
cristina.pinotti@unipg.it

Giulio Rigoni
University of Florence, Italy
giulio.rigoni@unifi.it

ABSTRACT

In this work, we compare the performance of range-free localization algorithms either radius-based or radius-free that use a mobile anchor (rover or drone). All the compared algorithms are based on the so called heard/not-heard method. Despite its simplicity, the localization accuracy of the heard/not-heard method heavily depends on the radius of the antennas. Usually the only information available for the antenna is the average radius issued by the manufacturer's technical datasheet. Thinking that the knowledge of the actual average radius can be of help for the localization, we observe in a real test-bed, at different altitudes, the antenna radiation pattern of the DM1001 antennas used in the commercial MDEK1001 kit from DecaWave. We simulate the localization algorithms using the average antenna radius tested during the experiments on the field and using the radius provided by the manufacturer's technical datasheet. In practice, however, the more precise information about the radius does not reduce the localization error. It only reduces the number of unlocalized devices. We conclude that the knowledge of the exact antenna pattern is the essential requirement for any range-free localization algorithm. However, since increasing the altitude the antenna radius becomes smaller and less dispersed, smaller errors occur localizing with a drone than with a rover.

CCS CONCEPTS

• **Computer systems organization → Sensor networks**; • **Hardware →** *Sensor applications and deployments*;

KEYWORDS

Drone; rover; localization mission; UWB antennas.

ACM Reference Format:
Francesco Betti Sorbelli, Cristina M. Pinotti, and Giulio Rigoni. 2019. Range-free Localization Algorithms with Mobile Anchors at Different Altitudes: A Comparative Study. In *Proceedings of 21st International Conference on Distributed Computing and Networking (ICDCN 2020).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

In this paper, we compare the performance of localization algorithms that substitute the fixed ground anchors with a single mobile anchor (MA) moving at different altitudes. We call *rover* a MA that moves on the ground, and *drone* a MA that flies on the sky. During the localization mission, the MA transmits a *beacon* and the ground device (GD) to be localized simply listens to these beacons. The beacon message contains the *waypoint*, that is, the current GPS position of the MA obtained by the coordinates of an on-board GPS module at the MA. The beacon is sent at regular intervals of time, and the distance of any two consecutive transmitted beacons is called *inter-waypoint* distance I_w, which depends on the MA current speed.

All the localization algorithms that we study are *range-free*, i.e., they localize without measurements (e.g., distances, powers) and all of them exploit the *heard/not-heard* method. The heard/not-heard method consists in detecting two consecutive beacons transmitted by the MA, one heard and one not-heard from the GD. From the heard/not-heard method, the GD learns that the MA is at the border of the transmission area of GD. Namely, if the GD hears the beacon B sent by the MA but does not hear the previous beacon B', the MA is inside the receiving area of the GD, at most at distance I_w from the border of such area. In general, as we will see in detail later, three applications of the heard/not-heard method are required for localizing the GD.

All the range-free localization algorithms that we consider have been described in the literature under the *ideal* model that assumes both the MA and the GD equipped with the same type of isotropic antenna whose transmission area is a perfect sphere. In such a case, the transmission and receiving areas projected on the ground are perfect circles, and since the position of the MA's is sampled at regular distance I_w, the GD and the MA learn from heard/not-heard method to be in a circular corona of width I_w. In the ideal case, only the I_w distance affects the localization accuracy: the smaller is I_w, the thinner is the corona, and hence the uncertainty is reduced bringing to get a more accurate estimation of the GD's position.

We remark that, in our understanding, the range-free algorithms not only ban the distance measurements between the antennas based on the measurement of the Time of Arrivals (TOAs) or Angle of Arrivals (AOAs), but also they forbid any explicit measurement of the Received Signal Strength Indicator (RSSI) [3].

Moreover, we point out that the antennas that we use have been calibrated by the manufacturer and no further calibrations have been undertaken during the experiments.

1.1 Motivations and Contributions

Recently, a test-bed using inexpensive UWB antennas (DecaWave's DM1001 UWB transceiver) operating at 6.5 Ghz and a 3DR Solo

The work has been partially supported by GEO-SAFE (H2020-691161), Project *NALP-SAPR* granted by FSE, and Project *NALP-SAPR2: Navigazione Autonoma, Logistica, e agricoltura di Precisione per Sistemi Aeromobili a Pilotaggio Remoto e Robot*, granted by Fondo Ricerca di Base, 2019, University of Perugia.

ICDCN 2020, January 4–7, 2018, Kolkata, India
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

drone is built to evaluate the localization accuracy of the range-free DRF algorithm proposed in [6]. Unfortunately, in practice, the localization error obtained by the implemented DRF algorithm is large and a modified version based on measurements (and, strictly speaking, no longer range-free) of the DRF algorithm is introduced to reach a good accuracy of the estimated position of the GD [4].

Another recent study in [7] shows that the irregularities of the hardware antenna radiation pattern can heavily affect the air-to-ground (A2G) link quality between a drone and a GD. The authors study the A2G link quality of BroadSpec UWB antennas from Time Domain Inc. by observing the RSSI. They show a dependency of the link quality from the antennas's orientations, their elevation, and their distance.

Considering that the range-free algorithms have been around for a long time, that the algorithms tested in [4] on the field use a drone, and that the irregularity shown in [7] are for air-to-ground links, we start thinking that the altitude is the cause of the poor results obtained for the range-free algorithm in [4]. In support of this hypothesis, we observe that the antenna gain changes with the polar angle. The polar angle depends on the difference of the altitude and on the distance of the two antennas. Hence, the polar angle continuously varies if the trajectory of the drone is not circular around the GD and the elevation is not zero. Moreover, since the drone is guided by the telemetry during the flight, it dynamically sets its position according to the next waypoint to be reached and frequently changes its pitch (inclination). Thus, also due to the pitch, the polar angle changes and possibly also the gain changes. We conclude that the 3D pattern of the antenna when the drone flies at a certain altitude is not shaped as a sphere (i.e., same gain in all the directions), but perhaps more likely as a "nibbled apple". Projecting on a plane a nibbled apple leads to irregularity in the 2D antenna pattern, as observed in the experimental tests in [4].

Then, the first goal of this paper is to verify our conjecture that the altitude is the main cause of the 3D, and thus 2D, pattern irregularity. We expect that, when the two antennas are both on the ground, the polar angle is null, and the range-free algorithms work much better. The test on the field conducted in this paper will show, however, that the 2D pattern of the UWB antennas of the MDEK1001 kit [8] is not regular even when the two antennas lay on the same plane, that is, when the MA is a rover.

The second goal, failed the previous conjecture, is to study how the antenna pattern varies with the elevation. For this purpose, we perform a campaign to collect the heard/not-heard beacons at different altitudes. Throughout this paper, the heard beacon of the heard/not-heard beacon pair will be called the *endpoint*. From the observed endpoints, knowing the position of the GD in our experiments, we derive the radii of the 2D antenna pattern when the drone flies at different altitudes[1]. Since it requires a lot of effort to collect the actual endpoints observed on-the-field, to reproduce synthetically a large set of endpoints on which to simulate our algorithms, we find the statistic distribution that fits the observed radii and we use the so derived statistic distribution for generating the endpoints in our simulations.

The third goal is to compare in the simulations the errors of the different range-free algorithms at different altitudes, assuming the radius distribution that fit the observed radii. We consider two kinds of range-free algorithms: those that only use the endpoints information, assuming no knowledge on the transmission's antenna radius but assuming that the 2D antenna patters are perfect circles, and those that, in addition to the endpoints, assume to know the radius r_0 of the 2D antenna pattern. As we will see in the algorithm description, fixed the radius r_0, we can infer different knowledge on the GD position. For the sake of simplicity, from now on, we identify the first and the second class of algorithms as *radius-free* and *radius-based*, respectively. The accuracy should be higher for the radius-based algorithms because they have more information. For example, they can more easily discard outliers among the endpoints. In practice, however, the assumed radius of the 2D antenna pattern (known by the algorithm) can be different from the actual one that has generated the endpoints due to the irregularity of the antenna. So, it happens that the algorithm interprets the endpoint in the light of the assumed radius, that is, the radius declared by the manufacturer's datasheet, which is indeed wrong. In our experiments, we show that the radius information given in input to the algorithms can create more damages than benefits, also in the ideal model.

Organization. The rest of this paper is so organized: Sec. 2 reviews the range-free localization algorithms that we compare. Sec. 3 describes the hardware/software architecture of the test-bed and the missions performed to collect the endpoints. Sec. 4 reports the technical datasheet information about our antennas and applies the goodness-of-fit method to derive the radius distribution. Sec. 5 evaluates all the algorithms under different scenarios, and also using the real set of endpoints. Finally, Sec. 6 offers conclusions.

2 THE RANGE-FREE ALGORITHMS

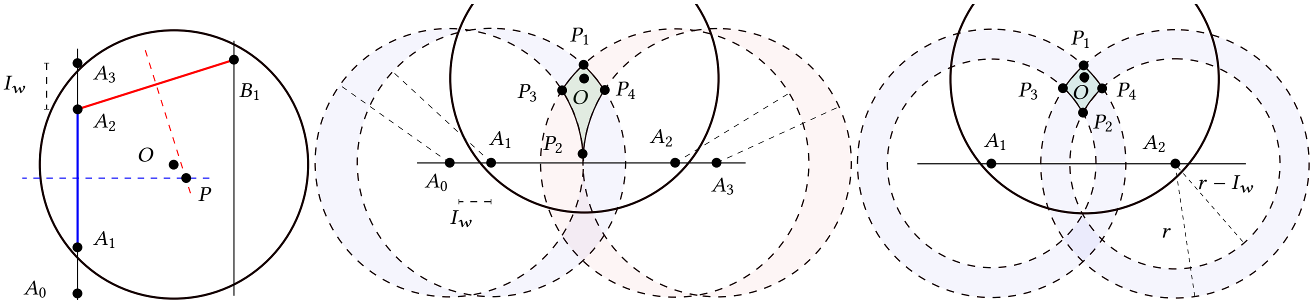
In this section, we describe the range-free localization algorithms that we evaluate in this paper. We also introduce a new variant of the DRF algorithm proposed in [5]. We start describing the algorithms under the ideal model in which all the 3D (resp., 2D) communication shapes are perfect spheres (resp., circles), and then we comment on the kind of errors that can afflict them. In addition to the endpoints, the radius-based algorithms receive in input a radius r, that they assume to be the 2D antenna pattern radius.

The DRF algorithm is radius-free, while all the remaining algorithms are radius-based.

2.1 The DRF algorithm

DRF has been the first range-free algorithm that localizes GDs using a drone as the MA [6]. DRF is a radius-free localization algorithm: it simply exploits the fact that if the GD detects two endpoints, it interprets the segment that connect them as a chord in its receiving disk. Then, the GD knows that it resides on the perpendicular bisector of such a chord. Thus, if the GD detects two non-parallel chords (or equivalently, three endpoints), it localizes itself at the intersection of the two perpendicular bisectors associated to such chords. Specifically, the implementation of DRF requires three non-collinear endpoints to estimate the position of the GD. In Fig. 1(a), the GD centered in O detects the first endpoint A_1 and continues to

[1] We use low altitude, $h = \{10, 20\}$, which is surely reasonable in a real scenario, for example, search and rescue after avalanches [11] or after an earthquake [10].



(a) Drf. **(b) Xiao.** **(c) Lee.**

Figure 1: Intersection areas.

hear the beacons until it detects A_2. Such pair of endpoints generates the chord $\overline{A_1A_2}$. After detecting the third endpoint B_1, the GD generates a second chord $\overline{A_2B_1}$. Applying the perpendicular bisector mechanism, the GD estimates its position P at the intersection of those two perpendiculars.

Accuracy in Drf. Under the ideal model, the performance of Drf is only affected by the distance I_w between two consecutive beacons. Namely, if all the three endpoints fall on the receiving disk of the GD, the estimated position for the GD is exact. From the heard/not-heard method, instead, the endpoints may fall inside the disk, at most at distance I_w from the border. An analytic bound of the accuracy of Drf that depends on I_w has been given in [6]. As long as the antenna pattern is ideal (i.e., a circle), Drf works with limited error and it does not use any information on the radius. Instead, assuming a real model (i.e., a not regular transmission area) as shown in Fig. 2, the perpendicular bisector $b1$ of $\overline{A_1A_2}$ can pass far from the GD. In our test-bed, we will see that Drf always localizes the GD because it receives three non-collinear endpoints, but it can incur in a large error if the antenna pattern is not ideal.

2.2 The Xiao algorithm

Xiao et al. propose a range-free radius-based localization algorithm that uses a rover [12]. As for Drf, the MA in Xiao broadcasts its current GPS position at regular intervals of time while it follows a predefined static path. The authors aim to build a constraint area that bounds the actual GD's position, assuming the same communication range r for both the MA and the GD. Note that the communication range r is simply the radius of the 2D antenna pattern. In Fig. 1(b), the endpoints A_1 and A_2, called arrival and departure position, respectively, are illustrated along with the beacons A_0 and A_3, called, pre-arrival and post-departure position, respectively. In other words, Xiao identifies the endpoints A_1 and A_2 applying the heard/not-heard method to the pairs A_0, A_1 and A_2, A_3. Four circumferences of radius r are drawn, centered at each of these four beacons.

Those circumferences create two symmetrical intersection areas (one of them is P_1, P_2, P_3, and P_4 in Fig. 1(b)) where the GD may reside. The GD has to detect another endpoint to disambiguate the proper symmetrical area where GD resides. The GD is estimated at the "center" of such intersection area.

Accuracy in Xiao. In the ideal model, assuming that the exact antenna radius r is known, as shown in Fig. 1(b), the accuracy in Xiao only depends on the distance I_w between the pair of beacons of the heard/not-heard method. However, the transmission radius information r is particularly critical. Indeed, two kinds of radius are involved: the communication range (or antenna radius) from which the endpoints depend and the assumed radius r_0 of the circumference drawn at the beacons A_i to find the expected position of the GD. The latter is given in input to the algorithm, the former depends on the actual antenna pattern. Since the former radius, say r, of the antenna pattern is difficult to be derived and may depend on the environment, it happens that the algorithm uses a default radius r_0 to draw the circle which is different from the actual radius r. For example, the radius r_0 used by the algorithm can be the one issued by the manufacturer's technical datasheet. Two different cases may occur. If $r_0 < r$, Xiao can potentially not localize the sensed GD because the intersection of the different circles centered at the collected endpoints could be empty or localize the GD in a wrong position, closer to an endpoint. Instead, if $r_0 > r$, the sensed GD is always localized farther away from the MA, getting a large localization error. Note that, if A_1 and A_2 are at distance greater than $2r_0$, they will be discarded.

Moreover, even in the case the manufacturer radius r_0 given in input to the algorithm is on average accurate, i.e., $r_0 = r$, the antenna pattern might be influenced by the environment and in practice, the actual radius that has generated the endpoints may be different from the assumed radius.

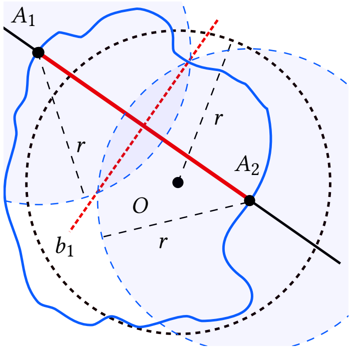


Figure 2: Localization errors: a bubble in A_1, and a hole in A_2.

For example, in Fig. 2, let us apply the Xiao algorithm at A_2: a circle of radius r centered in A_2 is drawn and the GD will be estimated on the dashed circumference centered at A_2. However, the GD is estimated farther than its actual position O because the radius r drawn around A_2 is longer than the current distance $\overline{A_2O}$ which represents the radius used to detect the endpoint A_2. This happens because the actual antenna pattern (the blue border) has a *hole* on that side. Due to the hole, the endpoint falls closer to the GD and the estimated position of the GD is pushed away from A_2.

Similarly, consider the endpoint A_1 in Fig. 2. Let us apply Xiao at A_1: a circle of radius r centered in A_1 is drawn and the GD will be estimated on the dashed circumference centered at A_1. That is, the GD is estimated closer than its actual position O because the radius r used to estimate the position of A_1 is shorter than the current distance $\overline{A_1O}$, which represents the radius used to detect the endpoint A_1. This is because the actual antenna pattern (the blue border) has a *bubble* on that side. Due to the bubble, the endpoint falls farther from the GD and the GD is attracted towards A_1.

2.3 The Lee algorithm

Lee et al. present a range-free radius-based localization algorithm that uses a rover as the MA [9]. Like Xiao, the Lee algorithm focuses on creating a constraint area where the GD could reside. The Lee algorithm uses three different endpoints to estimate the GD's position, as represented in Fig. 1(c). When the GD senses the endpoint in position A_1, it continues to listen to the MA until it registers the endpoint in position A_2. As shown in Fig. 1(c), for both the first and second endpoint, two circles with radius r and $r - I_w$ are drawn. The algorithm assumes that the GD to be localized is inside the area created by the intersections of the four points (P_1, P_2, P_3, and P_4 in Fig. 1(c)), and obviously the area is symmetrical with respect to the GD movement line, hence a third endpoint is required to serve as reference for the center.

2.4 The proposed DrfE algorithm

Now we present a new range-free algorithm, called DrfE, that shares the "chord" idea with the Drf algorithm, and exploits the radius information r.

When $I_w = 0$, DrfE is based on the following idea: given a chord of the receiving disk of the GD delimited by two endpoints, say A_1 and A_2, the GD resides on the point P of the perpendicular bisector that is at distance r from the two endpoints of the chord. There are two points on the perpendicular bisector that satisfy such properties: one on the "left" and one on the "right" of the chord. A second chord, built by using a third endpoint, is needed to disambiguate on which side of the chord the GD lays.

When $I_w \neq 0$, the DrfE algorithm logically repeats the above construction for all the chords obtained by extending $\overline{A_1A_2}$ of I_w above and below, as illustrated in Fig. 3. Repeating the construction above for the three chords $\overline{A_1A_2}$, $\overline{A_1A_3}$, and $\overline{A_0A_2}$, on the right of the chords we find three points P_1, P_2, and P_3. Recall that, such points form three isosceles triangles $\triangle(A_1A_2P_1)$, $\triangle(A_0A_2P_3)$, and $\triangle(A_1A_3P_2)$ with the two equal sides of length r.

The triangle $P_1P_2P_3$ on the right of the chord A_0A_3 and the symmetric triangle on the left of the chord A_0A_3 are the constraint areas where the GD can reside. Using the perpendicular bisector of the second chord, we disambiguate the triangle. The GD stays at the centroid of the found triangle.

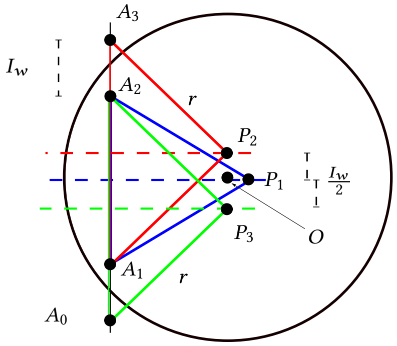


Figure 3: DrfE intersection area.

Accuracy in Lee and DrfE. The Lee and DrfE algorithms run into the same problems as Xiao. The only note is that it is relatively easier for DrfE to discover inconsistencies because, if the chord is very long with respect to r, the triangle cannot be built.

Since all the radius-based algorithms rely on the knowledge of the radius of the antenna pattern projected on the ground, in the next section we describe the test-bed that we have built to experimentally evaluate the antenna radius.

3 TEST-BED SETUP

In this section, we describe the hardware and software architecture of the test-bed. We fix a Cartesian coordinate system with origin at the special position Home $(0, 0, h_0)$, with $h_0 = 1$ m. At Home, we place the GD at the top of a tripod of height h_0.

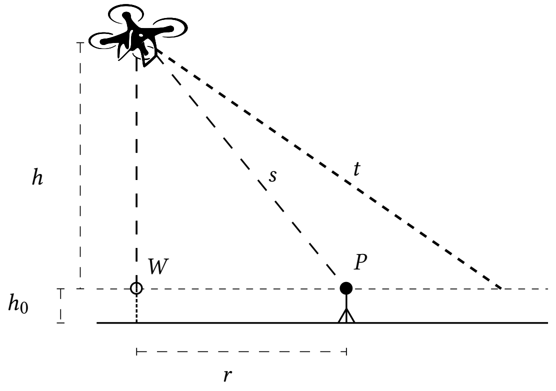


Figure 4: The drone and the ground device P.

When in our tests we set $h = 0$, we refer to a rover mission where the two antennas (one on the MA and another one on the GD) are placed at h_0, while when we refer to a mission at a certain altitude h, we refer to a drone that flies at an actual altitude $h_0 + h$. In the ideal model, the beacon sent by the drone when it is at the Cartesian coordinate W and altitude $h + h_0$ can be heard by any GD (at height h_0, see Fig. 4) at slant (i.e., 3D) distance $s \leq t$ from the drone, where t is the radius of the ideal sphere around the drone.

Consequently, on the ground, any GD at distance $r \leq \sqrt{t^2 - h^2}$ from the drone (at height $h_0 + h$, see Fig. 4) can hear.

In our test-bed, the rover is realized by a man who carries the antenna on a rigid support at height h_0. So, the antenna of the rover and that on the ground are at the same height.

3.1 Details of Implementation

In our test-bed, we employ different components: an UWB antennas kit MDEK1001, a Raspberry Pi, and a 3DR Solo drone.

3.1.1 Hardware Configuration. The UWB technology provides 10 cm precision for the ranging measurements [2]. Each DM1001 antenna has a 6.5 GHz center frequency, and has a point-to-point range up to 60 m on the line-of-sight [8]. Thus, in this paper, the datasheet radius will be set to $r_0 = 60$ m. The main component that pilots the MA and sends commands to the GD is the Raspberry Pi. The Raspberry can be used either for the ground experiments using a rover, or for the aerial ones employing a drone. For the previous case, we simulate the rover's behavior just walking in the field at a regular walking speed (about 3 km/h as measured by a smart-watch) keeping the Raspberry on the hands at $h_0 = 1$ m above the ground. Note that this speed is just an approximation, since even the walking speed of a human is affected by the turns, corners, and terrain. It is interesting to observe that, since the rover has to send the current GPS position, we relied on a very cheap USB GPS module connected to the Raspberry. Finally, in the latter case, in our flying experiments, we have used the 3DR Solo drone, (see Fig. 5), which is a very versatile quadcopter and it is able to fly up to 25 min [1].

Figure 5: 3DR Solo Drone.

3.1.2 Software Configuration. From the MA's side, the main program initially reads the input parameters setting the speed s, the fixed altitude h, and the Home point which coincides with the starting position. The pre-fixed speed s of the drone is not constant all the time, but is affected by the trajectory of the static path that the drone must follow on the sky (e.g., accelerations and decelerations at the corners) and by external factors as well (e.g., wind, air pressure). After that, the trajectory Π that consists of n segments S_i, $i = 0, \dots, n-1$, is built. Such list of n segments is made by generating $n+1$ random points in the deployment area. Each segment is delimited by two random points, and any two consecutive segments share one random point by setting the waypoints' coordinates (x_{W_i}, y_{W_i}, h) for each W_i, $i = 0, \dots, n-1$ (see Fig. 6). When all the waypoints are generated, the mission starts. Once the first waypoint of the trajectory Π has been reached, the drone/rover starts to send message beacons according to the current MA's position. In fact, the drone then converts its GPS coordinates in the local Cartesian (x, y) position. This process continues until the MA reaches the last waypoint of Π. When the mission is accomplished, the MA comes back to Home.

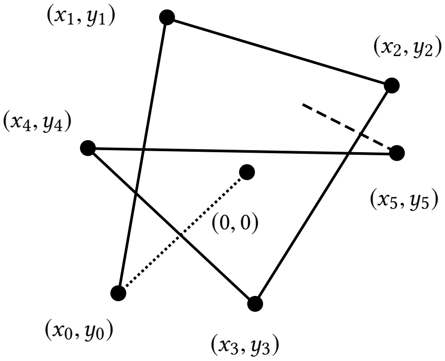
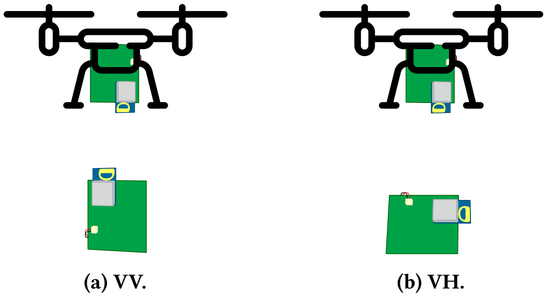


Figure 6: The drone's random path.

Concerning the GD's side, in our experiments, we set the antenna on the tripod, placed at Home, laying on two different sides, as sketched in Fig. 7. In the left case (Fig. 7(a)), the antenna lays on the short side, i.e., its xy side is parallel to the ground. In the right case (Fig. 7(b)), the antenna lays on the long side, i.e., its yz side is parallel to the ground. The antenna of the drone is always vertically fixed, with the short side parallel to the drone's body. We indicate the first configuration where the two antennas lay on the same side, but opposite direction, with VV (Vertical, Vertical); whereas we refer to the other configuration with VH (Vertical, Horizontal).



(a) VV.

(b) VH.

Figure 7: The implemented settings.

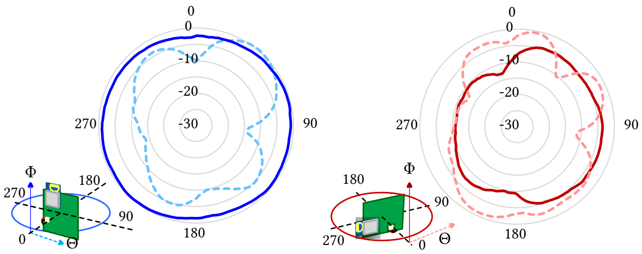
We are now ready to present the experimental results.

4 ANTENNA ANALYSIS

In this section, we first report the technical datasheet information of DecaWave about our antennas, and then we analyze the experimental data.

4.1 Datasheet Antenna Information

The solid dark blue line of Fig. 8(a) shows that, according to DecaWave, it is possible to obtain the same gain in all the directions in the xy-plane when an antenna vertically placed (i.e., on xy-plane) is observed by another antenna which shares the same *vertical* orientation (Φ *polarization*, or concordant). We recreate this situation putting the antennas as in Fig. 7(a). Thus, we expect that the VV configuration experiences the same gain at different angles, at least when the two antennas are at the same height.



(a) xy**-plane.** **(b)** yz**-plane.**

Figure 8: The D1001 radio pattern: dBm **vs angle.**

Both the dashed light blue line of Fig. 8(a) and the dashed light red line of Fig. 8(b) refer to the VH configuration in Fig. 7(b) because they show the gain when an antenna is observed by another antenna *perpendicularly* oriented (Θ *polarization*, or discordant). Both the dashed lines have nulls at certain angles that can limit the gain and can introduce holes or bubbles in the pattern. Thus, we expect that the VH configuration in Fig. 7(b) experiences different gains at different angles, and we also expect a relevant variability given that the dashed lines of Fig. 8(b) and Fig. 8(a) are different, although the relative position of the antennas seems to be the same. Since Fig. 8 does not mention elevation, these gains should refer to two antennas at the same height.

Analyzing these technical data, although there are many irregularities, it seems that the gain is omnidirectional at least when the two antennas are placed as in Fig. 7(a). According to our speculation that the altitude is the main cause of the pattern irregularity, we would expect that, when the two antennas are both on the ground (e.g., rover is the MA) in the VV configuration, the range-free algorithms work quite well.

Table 1: The DW1000 elevation gains (in dBm) **at** 6.2 GHz.

		Θ (discordant)	Φ (concordant)
yz-plane	peak	0.30	2.92
	average	−6.99	−3.04
xz-plane	peak	0.26	1.39
	average	−5.74	−3.90

We have not found, for the MDEK1001 kit, any data which correlates the gain and the polar angle, i.e., the elevation gain. However, in a previous document of a former antenna model called DW1000 [8], DecaWave gives the gain values on Tab. 1 for an antenna vertically placed as in Fig. 8(a) in an anechoic chamber. We report these values just to confirm the not negligible impact of the elevation[2]. We imagine the 3D radiation pattern is far from being a sphere, with the same gain in all the directions.

Then, we conjecture that, when the MA is a drone, the 3D antenna pattern is highly irregular and it can be sketched as a nibbled apple as reported in Fig. 9. Therefore, when such an antenna shape is projected on the ground, "holes" and "bubbles" can be found.

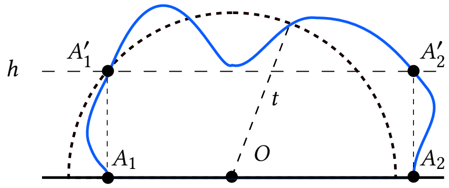


Figure 9: The ideal (dashed gray) and real (solid blue) antenna radiation profile in xz**-plane.**

In the next section, we report the antenna radii of our experiments that contradict our conjecture and also the DecaWave gain reported in Fig. 8(a). The antenna pattern is not regular when the antennas comply with the VV configuration and the MA is a rover.

4.2 Experiments for Antenna Radius

In our experiments, we wish to characterize the 2D antenna pattern observing the range of values of its radius.

In our experiments, the MA starts at Home in $(0, 0)$ at different altitudes $h = \{0, 10, 20\}$ m. As explained in Sec. 3, the MA traverses the deployment area with n segments that aim to cross the receiving shape of the GD. Moving along each random segment, the MA continuously broadcasts its current (x, y) position, and the GD registers the first and the last heard endpoints sent by the MA. Since we know in advance the position Home of the antenna, we are able to compute the actual 2D radius for each detected endpoint. Note that we observe the radius on the ground. It is important to recall that the beacons are sent at regular intervals of time, and so the observed radii have an intrinsic error of at most I_w. In this paper, we fix I_w to 0.40 m since we have experimentally observed this value, which clearly depends on the speed of the MA.

From the collected endpoints during the same experiment, we compute the mean μ and the standard deviation σ of the set of observed radii. Then, we apply the goodness-of-fit method in order to assess whether a given distribution is suitable to the built dataset[3]. We repeat the experiments for the two antenna configurations (VV and VH) and for different altitudes.

In the next, we report the results of the first experiment with the rover (i.e., $h = 0$) with the VV configuration (see Fig. 7(a)). According to DecaWave's datasheet (see the solid blue line in Fig. 8(a)), we expect an almost uniform radius in the experiments, at least when the two antennas are at the same height. We observed 38 different endpoints, with mean $\mu = 97.10$ and $\sigma = 39.74$. We observed the minimum and maximum radii, min $= 32.14$ and max $= 162.40$,

[2] Note also that the 3D antenna pattern is completely defined if we know its behavior in 3 planes: xy, xz, and yz.

[3] Fixed a distribution and a set of categories, each with at least 5 observed frequencies, we determine if there is a significant difference between the expected frequencies and the observed frequencies in one or more categories by using the chi-squared test.

respectively. According to the Pearson χ^2-test, at $h = 0$, the radii of the VV configuration have a uniform distribution.

As we will see, this is the only configuration with uniform distribution of the radii. Therefore, we agree with DecaWave that this configuration is somehow special. However, the radius cannot be considered really constant.

Table 2: $h = 0$, VV

class	radii		frequencies		
#	from	to	observed	uniform (U)	normal (N)
1	32.14	58.34	8	0.08	3.14
2	58.34	84.54	7	0.01	0.13
3	84.54	110.74	8	0.08	0.34
4	110.74	136.94	5	0.69	1.06
5	136.94	163.14	10	1.06	8.15
	likelihood			0.75	0.01

Here we report a second experiment at altitude $h = 10$, with the VH configuration (see Fig. 7(b)). We observed 28 different endpoints out of 38, thus confirming several null angles. The observed radii have mean $\mu = 66.34$, $\sigma = 22.81$, min = 16.52, and max = 121.66. According to the Pearson χ^2-test, at $h = 10$, the radii of the VH configuration have a normal distribution.

Table 3: $h = 10$, VH

class	radius		frequencies		
#	from	to	observed	uniform (U)	normal (N)
1	16.52	41.52	5	1.68	0.68
2	41.52	66.52	9	0.00	0.14
3	66.52	91.52	11	0.52	0.07
4	91.52	141.52	3	12.23	0.15
	likelihood			0.01	0.90

In Tab. 4, we summarize the statistic distributions that fit the observed radii of the experiments. For each distribution (i.e., U for Uniform and N for Normal), we give the values μ and σ observed in our experiments and the likelihood as well.

Table 4: The radii distribution with its parameters $D(\mu, \sigma)$ and its likelihood p.

	VV	VH
$h = 0$	$U(97.10, 39.74); 0.75$	$N(63.58, 33.01); 0.40$
$h = 10$	$N(84.97, 31.70); 0.81$	$N(66.34, 22.81); 0.90$
$h = 20$	$N(62.91, 34.06); 0.83$	$N(57.69, 24.91); 0.82$

In conclusion, except for the rover and the VV configuration, all the experiments show that the radius most likely follows a normal distribution, but with a large standard deviation. It is worthy to note that increasing h, the mean of the radii decreases. The mean decreases faster with the VV configuration while with the VH configuration it remains quite stable (see Tab. 4) The values of the radii are generally more concentrate with the VH configurations than the VV ones.

We conclude that, oppositely to our conjecture, VH seems better than VV and the radii obtained with a drone seem better than those obtained with a rover. Marginally, let us point out that organizing a localization mission is easier with a drone than with a rover because the drone is faster and less attention has to be paid to the terrain. Although the results are different from what we expected, we continue our investigation of the accuracy in localization algorithms. Thus, we use the results reported in Tab. 4 to generate a large synthetic set of endpoints that fit the estimated parameters of the radii distributions for testing the different algorithms surveyed in Sec. 2.

5 COMPARATIVE EVALUATION

In this section, we compare all the localization algorithms using first the set of synthetic endpoints and then the set of real endpoints collected during the experiments.

5.1 Setting

Our goal is to analyze the localization error and the percentage of unsuccessful localizations of the range-free Drf, Xiao, Lee, and DrfE algorithms, when:

- the exact radius used to generate the endpoints is given in input to the radius-based algorithms or not,
- the MA is a rover ($h = 0$), and
- the MA is a drone ($h = \{10, 20\}$).

From now on, the height h, with $h = \{0, 10, 20\}$, and the antenna configurations $\{VV, VH\}$ are referred to as a particular scenario.

For each simulated scenario, we run 200 localizations. Precisely, for 200 times, we randomly generate three endpoints, say A_1, A_2, and A_3, with the distribution and the parameters of the simulated scenario derived in Section 4.2 from the observed endpoints (see Table 4). For each triple of synthetic endpoints, we invoke the four algorithms Drf, Xiao, Lee, and DrfE, using either the radius $r = \mu$ used to generate the endpoints or the datasheet radius r_0. That is, if $r = \mu$, we test the behavior of the algorithms when the algorithm receives in input the actual radius, but still the endpoints can be affected by the antenna irregularity (i.e., σ). Instead, if $r = r_0$, we test the behavior of the algorithms when the algorithm receives in input a radius completely different from the actual one. It is important to point out that the three endpoints we use satisfy two constraints: the distances $d(A_1, A_2)$, $d(A_2, A_3)$, and $d(A_3, A_1)$ must be at least 60 m, and the minimum angle between the two segments $\overline{A_1A_2}$ and $\overline{A_2A_3}$ must be greater than 20 deg to enforce the fact that the selected endpoints represent the endpoints of two consecutive segments of the static path Π traced during the mission (see Fig. 6).

For the localizations on the set of real endpoints collected during the experiments, we extract three consecutive endpoints that belong to two consecutive segments of the mission. We repeat the localization as many times as we find three suitable endpoints.

We compare our algorithms under two metrics, i.e., the *localization error*, defined as the distance between the actual GD position and the estimated position outputted by the algorithms, and the *percentage of unlocalized*. Concerning the first metric, we report the localization error resumed into a boxplot that highlights the median (red line), the average value (blue circle), and the data between the first Q_1 and the third quartile Q_3 (box). Additionally, the extremes of the whiskers represent the $Q_1 - 1.5$ IRQ and $Q_3 + 1.5$ IRQ, respectively, where the interquartile range (IQR) is defined as IRQ $= Q_3 - Q_1$. Lastly, about the second metric, an unsuccessful localization is an application of the algorithm which does not return any constraint area or in general any geometrical intersection where the GD can reside, that is, the GD remains unlocalized.

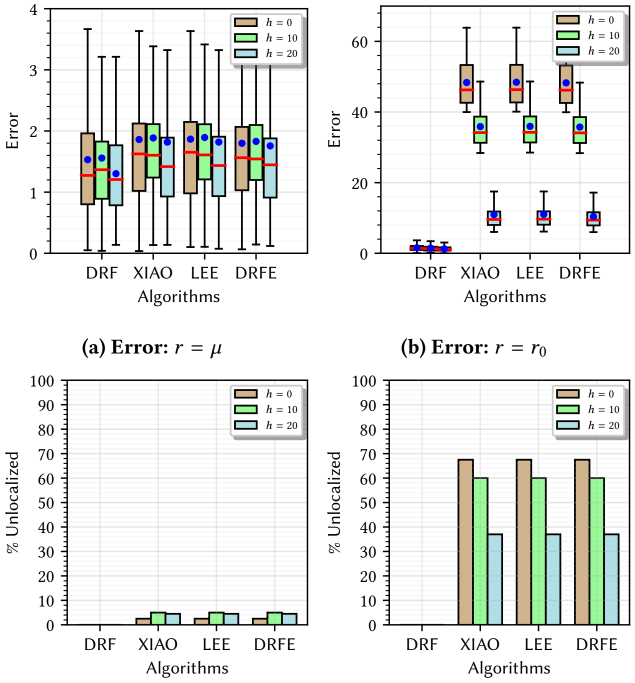
5.2 Results

In this section, we start considering synthetic set of endpoints. We evaluate the performance of the range-free algorithms on a nearly ideal model that assumes an isotropic antenna with almost constant radius (see Fig. 10). This set of simulated localizations supports the observation that, even if the antenna is isotropic, if two different radii are used, one to generate the endpoints and one to localize the GD, the localization error is large.

Moreover, we discuss the performance of the range-free algorithms on the synthetic set of endpoints generated according to the distributions in Tab. 4 (see Fig. 11).

Finally, we analyze the impact of using a rover instead of a drone in Fig. 12, and we report the comparison between the performance of DRF and DRFE in Fig. 13.

5.2.1 Comparison Under the Ideal Model. Let us start considering the nearly ideal model in Fig. 10 in which the endpoints are generated with μ given in Tab. 4 for the VV configuration but setting σ equal to 1. Since the dispersion is low, the radius can be considered almost constant. As expected, all the radius-based algorithms perform well when they use (i.e., they receive in input) a radius r equal to the radius μ used to generate the endpoints. In such a case, the localization error is on the order of a couple of meters or less for all the algorithms[4]. DRF is slightly better than the other algorithms at any altitude, while DRFE is worse than DRF. The number of unlocalized GDs is very small. The performance of the radius-based algorithms drastically drop down when they receive in input the datasheet radius $r_0 = 60$ m, which is not the radius used to generate the endpoints. The percentage on unsuccessful localizations is extremely large.



(a) Error: $r = \mu$

(b) Error: $r = r_0$

(c) Unlocalized: $r = \mu$

(d) Unlocalized: $r = r_0$

Figure 10: The impact of the radius on the ideal model.

These results confirm what we said in Sec. 2: to obtain an accurate localization, not only the antenna must be of good quality (i.e., with a very small value of σ), but also the antenna radius μ must be exactly known by the algorithm. The results show that the error due to the use of the wrong radius r_0 decreases when h increases because $\mu \rightarrow r_0$.

Alongside, note that since DRF is radius-free, the performance of DRF is not influenced by the radius selection, as shown in Fig. 10(b).

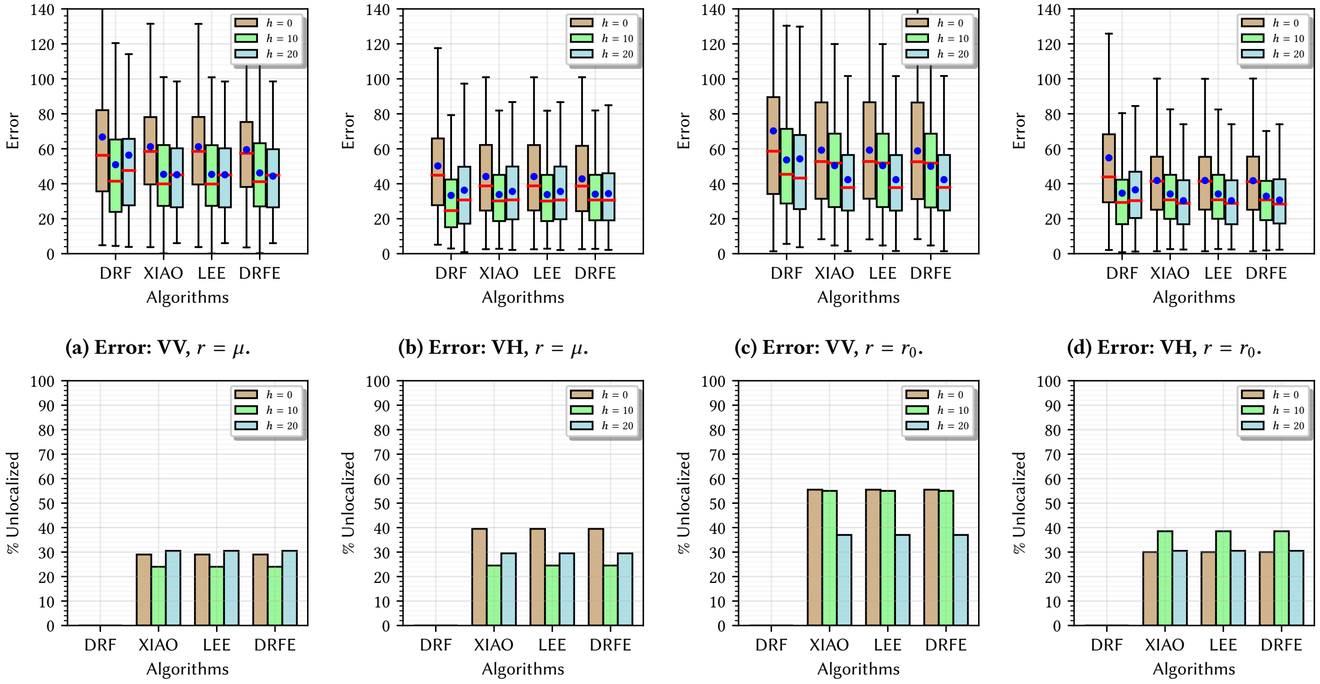
5.2.2 Comparison using the Synthetic Endpoints Set. Fig. 11 summarizes all our simulations on the synthetic set of endpoints generated according to the distribution in Tab. 4. The errors of the algorithms are compared at different altitudes, different antenna configurations, and different radii. For all the algorithms, the average error is large. The worst error occurs with the VV configuration of the antennas and $h = 0$: in such a scenario, the endpoints follow a uniform distribution. In general, the VH configuration makes an error smaller than the VV configuration probably because the radius that generated the endpoints for VH is less dispersed (i.e., σ is smaller) than that for VV. The three radius-based algorithms, XIAO, LEE and DRFE, exhibit the same performance. They are inspired by slightly different ideas, but they actually act the same.

The DRF algorithm experiences the worst average error, but the error is only slightly more than that of the radius-based algorithms. The whisker of the largest error of DRF is the longest whisker among all the algorithms probably because DRF finds a localization also in extreme cases when the other algorithms return an unsuccessful localization. As for the radius-based algorithms, the error in DRF decreases when the altitude increases, the error of the VV configuration is worse than that of the VH configuration.

Let now compare the performance of the algorithms when they receive in input the radius μ used to generate the endpoints or the datasheet radius r_0. Differently from the nearly ideal model, the knowledge of the radius μ does not really help in decreasing the localization error because the standard deviation σ is large in the synthetic set. In practice, as explained in Sec. 2, the antenna pattern has many holes and bubbles, and the algorithm returns errors similar to when it does not know the radius used to generate the endpoints. As witnessed by the whiskers of the boxplots in Figs. 11(a) and 11(c) or by Figs. 11(b) and 11(d), the error performances of the radius-based algorithms when they receive the two different values μ and r_0 differ only for the extreme values.

Instead, the knowledge of μ reduces percentage of unlocalized GDs as illustrated in Figs. 11(e) and 11(g) for the VV configuration.

[4] Please note that the scale of y axis in Fig. 10(a) is zoomed with respect to the scale of y axis in Fig. 10(b)



(a) Error: VV, $r = \mu$. **(b) Error: VH, $r = \mu$.** **(c) Error: VV, $r = r_0$.** **(d) Error: VH, $r = r_0$.**

(e) Unlocalized: VV, $r = \mu$. **(f) Unlocalized: VH, $r = \mu$.** **(g) Unlocalized: VV, $r = r_0$.** **(h) Unlocalized: VH, $r = r_0$.**

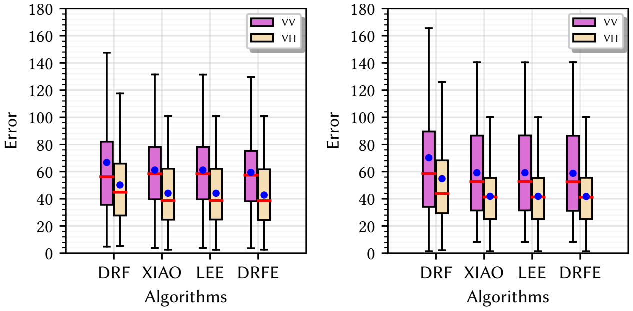
Figure 11: Comparisons between all the algorithms in the studied scenarios.

The improvement in Figs. 11(f) and 11(h) is weaker for the VH configuration, since the difference between the observed radii $\mu = \{63.58, 66.34, 57.60\}$ and $r_0 = 60$ is small.

In conclusion, the average localization error for VV and VH configurations is 50 m and 30 m, respectively, regardless of the fact that there is knowledge or not of the radius used for generating the endpoints. The radius dispersion is what matters for the error: the larger is the value of σ, the worse is the localization performance.

5.2.3 Comparison on the Ground. In Fig. 12, we report the simulations when a rover is used as the MA. All the algorithms incur in a large error. As illustrated in detail in Fig. 12, the conjecture (supported by Fig. 8(a)) that the antennas are isotropic at $h = 0$ with the VV configuration is clearly not verified. The VH configuration is a little bit better than the VV one. At $h = 0$, the antenna pattern is highly irregular because it does not benefit of the knowledge of the average radius μ used to generate the endpoints.

5.2.4 Comparison Between DRF and DRFE. In Fig. 13, we compare the performance of DRF, DRFE$_\mu$ (DRFE with observed radius μ), and DRFE$_{r_0}$ (DRFE with datasheet radius r_0). In general, DRFE$_\mu$ performs better than DRFE$_{r_0}$. Moreover, having the exact information about the radius that has generated the endpoints helps to improve the localization accuracy (DRFE$_\mu$ performs better than DRF), but the radius-based method has always an error not comparable with the error of the ideal model (see Fig. 10).

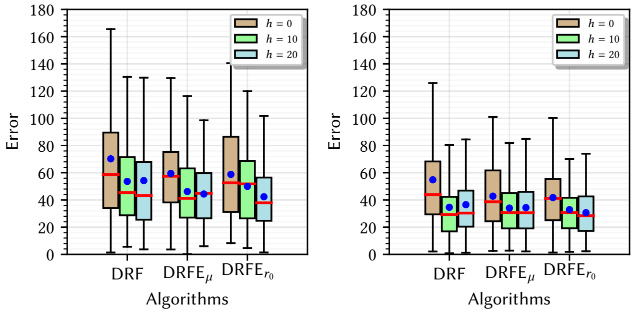


(a) Error: $h = 0$, $r = \mu$ **(b) Error: $h = 0$, $r = r_0$**

Figure 12: The rover error.

5.2.5 Results on Real Endpoints. In this subsection, in Fig. 14, we plot the results for the real endpoints collected on the field obtained during our test-bed. However, the reliability of these results is moderate because the number of valid triples is small.

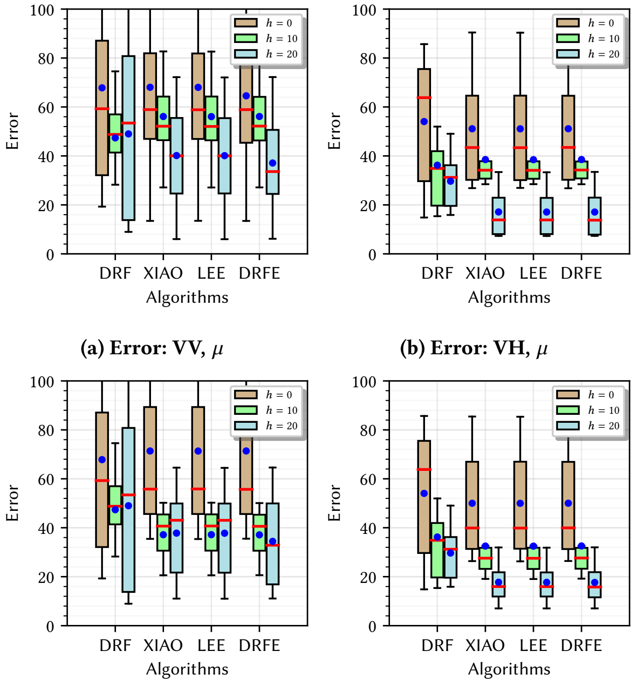
For the radius-based algorithms, in VV, the results of the localization error seems to follow the trend already seen on the synthetic generated set. In general, VV has a larger error compared to that of VH. In VH, the results on the real endpoints are slightly better than that obtained on the synthetic generated set, although the number of valid triples is small. Indeed, the average error at $h = 20$ is about 15 m although σ is greater than 20 m when the radius μ is used. The average error increases to 20 m when the radius r_0 is used. This can be explained with the fact that there is a much stronger correlation between the endpoints that are not fully captured by



(a) Error: VV, Drf vs DrfE

(b) Error: VH, Drf vs DrfE

Figure 13: Drf vs DrfE.



(a) Error: VV, μ

(b) Error: VH, μ

(c) Error: VV, r_0

(d) Error: VH, r_0

Figure 14: Error using real data.

the two constraints (distance and angle between the endpoints) that we imposed for the selection of the synthetic endpoints. Finally, the error of Drf is worse with real endpoints than with synthetic endpoints, especially for the VV configuration.

Concluding, these results on the real data-set give rise to a hope that the range-free radius-based algorithms, receiving a better model of the antenna pattern, can still be of interest.

6 CONCLUSION

In this paper, we compared the performance of heard/not-heard range-free localization algorithms, either radius-based or radius-free that use a MA (rover or drone) for localizing GDs, on a real test-bed using the UWB antennas DM1001. This work stems from a previous test-bed using such hardware in which a drone was the MA and which reported bad results. We initially conjectured that the bad performance was mainly affected by the altitude of the drone. Unfortunately, we verified in this work that the DM1001 antennas impact the localization performance even on the ground, despite the official datasheet's radiation pattern states an omnidirectional shape under specific conditions. We studied the antenna's pattern at different altitudes and relative VV and VH configurations in order to characterize its behavior. We concluded that the VH configuration is more accurate than the VV one, and that the results employing a drone as MA are more stable than that using a rover as MA. We simulated all the surveyed algorithms with a large data-set of endpoints obtained by the characterization of the antenna. In particular, we studied the impact of the knowledge or not of the exact radius of the antenna pattern (which is used to generate the endpoints) on the radius-based algorithms. We learned that accurate results with the range-free algorithms can be only obtained with a good omnidirectional antenna and the exact knowledge of its radius. Alternative solutions should be designed, for example using ranging measurements, to improve on the localization error.

REFERENCES

[1] 3D Robotics. *Solo Specs: Just the facts*, 2019 (accessed June 30, 2019). https://news.3dr.com/solo-specs-just-the-facts.

[2] Betti Sorbelli, F., Das, S. K., Pinotti, C. M., and Silvestri, S. On the accuracy of localizing terrestrial objects using drones. In *2018 IEEE International Conference on Communications (ICC)* (2018), IEEE, pp. 1–7.

[3] Betti Sorbelli, F., Das, S. K., Pinotti, C. M., and Silvestri, S. Range based Algorithms for Precise Localization of Terrestrial Objects using a Drone. *Pervasive and Mobile Computing 48* (2018), 20–42.

[4] Betti Sorbelli, F., and Pinotti, C. M. Ground localization with a drone and uwb antennas: Experiments on the field. In *2019 IEEE 20th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)* (2019), IEEE, pp. 1–7.

[5] Betti Sorbelli, F., Pinotti, C. M., and Ravelomanana, V. Range-free localization algorithm using a customary drone. In *2018 IEEE International Conference on Smart Computing (SMARTCOMP)* (2018), IEEE, pp. 9–16.

[6] Betti Sorbelli, F., Pinotti, C. M., and Ravelomanana, V. Range-Free Localization Algorithm Using a Customary Drone: Towards a Realistic Scenario. *Pervasive and Mobile Computing 54* (2019), 1–15.

[7] Chen, J., Raye, D., Khawaja, W. A. G., Sinha, P., and Güvenç, I. Impact of 3d uwb antenna radiation pattern on air-to-ground drone connectivity. *2018 IEEE 88th Vehicular Technology Conference* (2018), 1–5.

[8] DecaWave. *Product Documentation*, 2019 (accessed June 30, 2019). https://www.decawave.com/product-documentation/.

[9] Lee, S., Kim, E., Kim, C., and Kim, K. Localization with a mobile beacon based on geometric constraints in wireless sensor networks. *IEEE Transactions on Wireless Communications 8*, 12 (2009), 5801–5805.

[10] Qi, J., Song, D., Shang, H., Wang, N., Hua, C., Wu, C., Qi, X., and Han, J. Search and rescue rotary-wing uav and its application to the lushan ms 7.0 earthquake. *Journal of Field Robotics 33*, 3 (2016), 290–321.

[11] Silvagni, M., Tonoli, A., Zenerino, E., and Chiaberge, M. Multipurpose uav for search and rescue operations in mountain avalanche events. *Geomatics, Natural Hazards and Risk 8*, 1 (2017), 18–33.

[12] Xiao, B., Chen, H., and Zhou, S. Distributed localization using a moving beacon in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems 19*, 5 (2008), 587–600.