Abstract—This paper introduces a new localization algorithm called COM-LOC++ that improves the performance of a recently introduced distributed range-free algorithm called COM-LOC. COM-LOC and other existing range-free algorithms only process the information received from one reference node in order to estimate the distance to that reference node. COM-LOC++ improves this performance by processing the information received from all heard reference nodes to estimate the distance to one reference node. Simulations show that the use of this additional part of information decreases the localization error by 50...110% and the localization stability by 20...60%, relative to COM-LOC. Other comparisons with several RSS-based localization algorithms show that COM-LOC++ decreases the localization error by 20...180% and the localization stability by 70...210%, while keeping the communication costs equal.

I. INTRODUCTION AND RELATED WORK

In the recent years, there is a growing interest in locating devices in wireless communication networks. Several of these localization systems are based on connectivity measurements ([5], [6], [8], [9], [11], [12], [13] and [17]), as connectivity information can be obtained with no additional hardware and minimum energy costs. In general, other localization techniques can be more accurate than connectivity-based localization systems. These latter localization techniques often require specialized hardware (e.g. TOF, TDOA and UWB) that is not commonly available in a wireless communication network. The localization performance of these other techniques can be enhanced by processing connectivity information (as in [15]). Connectivity-based localization is still an attractive field of research.

In this paper, we focus on RSS-based localization. We distinguish three type of RSS-based localization algorithms, namely range-based, proximity-based and range-free localization.

Range-based localization algorithms assume that the signal strength decay over distance follows a distribution that is a priori known. This distribution is used for converting signal strength measurements into distance estimates. These distance estimates are then used for estimating the position (for example [13] and [16]).

Proximity-based localization algorithms assume that the signal strength decays over distance ([6] and [14]). The main difference with range-based algorithms is that they only use the order of RSS measurements. Therefore, proximity-based localization algorithms are not dependent on the goodness-of-fit of the RSS over distance distribution.

Range-free localization approaches use connectivity information ([5], [9], [11] and [12]). Existing localization algorithms based on connectivity assume that the transmission range is constant ([11]) or the deployment distribution is uniform and known a priori ([5], [9] and [12]). This means that the performance depends on the difference between the expected and real values of the transmission range and deployment distribution.

Most existing localization algorithms in wireless networks are designed with the assumption that certain localization specific information is available. Afterwards, a communication protocol is designed to obtain this information. As in [17], we do it the other way around. We first design a distributed range-free localization algorithm on the basis of a communication protocol commonly employed by localization algorithms (as in [5], [9], [11], [12] and [15]). This means that we construct the Maximum Likelihood Estimator (MLE) for localization on the basis of the communication protocol. Theoretically, this maximum likelihood estimator should provide optimum localization results for a given communication protocol. We then present a new algorithm called COM-LOC++ that improves a recently introduced distributed range-free algorithm called COM-LOC. COM-LOC++ uses the information received from all heard reference nodes to estimate the hop distance to one reference node, while COM-LOC and other existing range-free algorithms only evaluate the information received from each reference independently to estimate the hop distance.

This paper is organized as follows: after the problem formulation in Section II, Section III describes the model used for simulating the connectivity. Section IV shows how COM-LOC++ converts the information obtained during the communication phase into distance estimates and associated probabilities. Section V provides a description of COM-LOC++. Section VI analyzes the localization performance of COM-LOC++. In addition, this section compares COM-LOC++ with COM-LOC ([17]), ecolocation ([14]) and a modified version of the MLE described in [13]. Section VII presents the conclusion and future work.
II. PROBLEM FORMULATION

This section provides a formal description of the range-free localization problem. First consider a wireless network that consists of two type of nodes:

- **Reference nodes**: Reference nodes know their position in advance.
- **Blind nodes**: Blind nodes do not know their location in advance.

We address the problem of blind node localization on the basis of connectivity measurements using a communication protocol called sum-dist (as in [11]). First, each reference node sends one message with its position and a hop distance set to one. Each receiving blind node stores the received reference position and hop count. Then the hop distance is increased by one and the message is forwarded. This ends the communication phase. We keep the communication costs at the minimum for localization functionality in mobile wireless networks. At the end of the communication phase, blind nodes have the following information:

1. The blind nodes store a set of reference node positions that are one-hop-away. We represent this set by: \( S \subseteq R \).
2. The blind nodes store a set of reference node positions that are two-hops-away. We represent this set by: \( T \subseteq R \).
3. The blind nodes store the number of received messages from other blind nodes per reference node. We represent this number by: \( nr_{ref} \) and \( ref \in R \).
4. The blind nodes store the number of received messages from other blind nodes. We represent this number by: \( nr_{total} \).

We use these information components throughout this paper. Most existing distributed range-free localization algorithms use this communication protocol and only evaluate the shortest hop count for localization ([5], [9], [11] and [12]). Figure 1 shows an illustrative example of sum-dist. The black circles represent the nodes; \( r_1 \) represents a reference node and \( b_1 \ldots b_4 \) represent the blind nodes. The solid and broken lines indicate that the nodes can communicate with each other. The text above the communication lines shows whether the received messages are processed by the existing algorithms. The number of hops indicate how many hops the blind node is away from the reference node. COM-LOC distinguishes two type of messages that are not used by existing distributed range-free algorithms ([17]):

- **“redundant information”**: The messages from \( b_1 \ldots b_3 \) all indicate that \( b_4 \) is two-hops-away from reference node \( r_1 \). Therefore, two of these messages are considered redundant. Note that \( nr_{ref} = 3 \).
- **“false information”**: The messages from \( b_1 \ldots b_2 \) to \( b_3 \) indicate that \( b_1 \) is two-hops-away from reference node \( r_1 \), while the shortest hop-distance is one-hop. Hence, these messages are considered as false information. Note that \( nr_{ref} = 2 \).

This means that many received messages during the communication phase are considered useless and discarded. The main difference with shortest-hop localization algorithms is that COM-LOC processes these messages in order to increase the localization performance without increasing the communication costs. Note that shortest-hop distance algorithms only process information components 1 and 2, COM-LOC also evaluates information component 3. For details, we refer to [17].

COM-LOC++ further increases the performance by evaluating information component 4. Figure 2 shows an illustrative example what type of extra information COM-LOC++ processes in comparison with COM-LOC for node \( b_4 \):

- \( nr_1 = 3 \): node \( b_4 \) receives messages from \( r_1 \) via nodes \( b_1 \ldots b_3 \).
- \( nr_2 = 3 \): node \( b_4 \) receives messages from \( r_2 \) via nodes \( b_3 \) and \( b_5 \ldots b_6 \).
• $n_{\text{total}} = 5$: node $b_4$ receives messages from nodes $b_1 \ldots b_3$ and $b_5 \ldots b_6$.

This means that node $b_4$ does not receive messages from 2 nodes for reference nodes $r_1$ and $r_2$ ($n_{\text{total}} - n_r = n_{\text{total}} - n_r = 2$). In other words, $b_4$ is in transmission range of 5 blind nodes AND the reference nodes are NOT in transmission range of two of these nodes. COM-LOC++ uses this information to improve localization performance. Section IV describes how this information is processed.

III. Simulation Model

This section describes the used signal strength over distance distribution. As in COM-LOC ([17]), we adopt the Log-Normal Shadowing Model for modeling the signal strength over distance distribution ([11]), because both theoretical and measurement-based studies support this model in indoor and outdoor environments ([2] and [18]). The following formula represents the Log-Normal Shadowing Model:

$$P_d = P_{d_0} - 10 \cdot n \cdot \log_{10}(\frac{d}{d_0}) + X_d$$

Here $P_d$ represents the received signal power in dBm at distance $d$; $P_{d_0}$ represents the received signal power in dBm at reference distance $d_0$; $n$ represents the path loss exponent, representing the rate at which the path loss increases with distance; $X_d$ represents the error of the Log-Normal Shadowing Model and follows a zero-mean normal distribution with variance $\sigma_{d_{\text{dBm}}}^2$.

We use the Log-Normal Shadowing Model for estimating the packet delivery rate as a function of distance. In general the connectivity is determined by an RSS threshold (like in [13]). For simplicity, we assume that the distance estimates and associated probabilities are used for estimating the position of the blind node. Note that we use a similar method as described in [17] for estimating these probabilities.

A. One- and Two-Hop-Away Reference Nodes

For completeness, we show how we convert information component 1 and 2 into a distance over probability distribution (see Section II). We use Equation 2 to calculate the probability over distance distribution that blind node $b$ hears reference node $a$:

$$P(b \text{ hears } a|d_{a,b})$$

and we use Equation 3 to calculate the probability over distance distribution that blind node $b$ does not hear reference node $a$ (two-hop-away reference node information):

$$P(b \text{ does not hear } a|d_{a,b}) = 1 - P(b \text{ hears } a|d_{a,b})$$

We use these equations for processing information component 3 and 4.

B. Heard and Not Heard Blind Nodes

In this subsection, we describe how we process information components 3 and 4 as described in Section II. [17] describes how to process information component 3:

$$P(b \text{ hears } a \text{ via } nr_{\text{ref}} \text{ nodes}|d_{a,b})$$

In this paper, we are particularly interested in processing information component 4:

$$P(b \text{ did NOT hear } a \text{ via } nr_{\text{TOTAL}} - nr_{\text{ref}} \text{ nodes}|d_{a,b})$$

Note that the probabilities defined in Equations 5 and 6 are independent. Hence, both information components are processed by multiplying the calculated probabilities. We use a similar method as described in [17] to approximate this probability. We draw samples (Monte Carlo Simulations) as shown in Figure 3 to represent the position and distance distribution:

- Blind nodes that forward messages to node $b$ lie within transmission range from blind node $b$. Hence, we draw samples that lie within the transmission range from node $b$. We represent this set of possible positions by: $C$. Figure 3 represents position distribution $C$ by the crosses.
- The distance between reference node $a$ and blind node $b$ lies in the following interval: $[0 \ldots 2 \cdot tr]$. Hence, we draw samples in a straight line that starts at blind node $b$ and has a length of $2 \cdot tr$. We represent this set of possible positions and distances by: $A$. Figure 3 represents position distribution $A$ by the circles.

We use Equation 2 for estimating the probabilities between individual samples:

$$P(b \in B \text{ did NOT hear } a \text{ via } c \in C|d_{a,b}) =$$

$$P(c \in C \text{ did NOT hear } a|d_{a,c}) \cdot P(b \in B \text{ hears } c \in C|d_{c,b})$$

We use Equation 7 for estimating the probability that blind node $b$ did not hear reference node $a$ via one blind node:

$$P(b \in B \text{ did not hear } a \text{ via } c \in C|d_{a,b}) =$$

$$\sum_{c \in C} P(b \in B \text{ did not hear } a \text{ via } c \in C|d_{a,b})$$
Note that we assume that blind node \( b \) hears blind node \( c \) via another reference node than reference node \( a \), as explained in Section II. We use Equation 8 for estimating the probability that blind node \( b \) did not hear reference node \( a \) via \( n_{T_{\text{TOTAL}} - n_{r_{\text{ref}}}} \) blind nodes (Equation 5):

\[
P(b \text{ did NOT hear } a \text{ via } n_{T_{\text{TOTAL}} - n_{r_{\text{ref}}}} \text{ nodes} | d_{a,b}) = \prod_{i=1}^{n_r} P(b \in B \text{ did not hear } a \text{ via } C(d_{a,b}))
\]

Hence, we assume that the calculated probabilities are independent. Figure 4 shows Equation 8 as a function of the distance between the reference node \( a \) and blind node \( b \) for the following Log-Normal Shadowing Model parameter settings: \( n = \{3.5\} \), \( \sigma_{\text{dBm}} = \{6\} \) and \( P_{d0} = \{-40\} \) and different \( n_{T_{\text{TOTAL}}} \) and \( n_{r_{\text{ref}}} \) settings.

C. Final Probability over Distance Distribution

As in [17], we assume that the calculated probabilities associated with information components 1...4 are independent. Hence, the final probability over distance distribution is calculated by multiplying these probabilities:

- one-hop-away reference nodes \( s \in S \):
  \[
P(b \text{ hears } s | d_{a,b}) \cdot P(b \text{ hears } s \text{ via } n_{r_{\text{ref}}} \text{ nodes} | d_{a,b}) \cdot P(b \text{ did NOT hear } a \text{ via } n_{T_{\text{TOTAL}} - n_{r_{\text{ref}}}} \text{ nodes} | d_{a,b})
  \]
  (10)

- two-hops-away reference nodes \( t \in T \):
  \[
  (1 - P(b \text{ hears } t | d_{t,b})) \cdot P(b \text{ hears } t \text{ via } n_{r_{\text{ref}}} \text{ nodes} | d_{t,b}) \cdot P(b \text{ did NOT hear } a \text{ via } n_{T_{\text{TOTAL}} - n_{r_{\text{ref}}}} \text{ nodes} | d_{a,b})
  \]
  (11)

The required computations, described in Sections IV-A and IV-B, are too expensive to run on a blind node with limited computational capabilities. As in [17], we calculate the outcome of Equations 3, 8 and 5 (for \( n_{r_{\text{ref}}} = 1 \)) before deployment. Blind nodes store the results of these calculations in a table with a user defined distance resolution. Therefore, this implementation strategy significantly reduces the computational complexity, being linear for the required computations on the blind node. We refer to [17] for details.

D. Observations and Discussion

Increasing the number of heard and not heard nodes makes the probability over distance distribution steeper. Figure 4 shows two illustrative examples of this observation. A steeper probability distribution means that the distance estimate becomes more accurate. The number of heard and not heard blind nodes depends on the node density within a wireless network. Hence, the localization performance increases with increasing node density. Section VI further analyzes the influence of the node density on the localization performance.

V. RANGE-FREE LOCALIZATION ALGORITHM

This section describes how COM-LOC++ estimates a position using the probability over distance distributions of Section IV. As in [11] and [17], COM-LOC++ implements a grid-based Monte Carlo Localization approach. An overview of Sequential Monte Carlo methods can be found in [4]. COM-LOC++ consists of two phases:

- The “prediction phase” draws samples that represent the position distribution.
- The “filtering phase” weights the samples drawn in the prediction phase according to the observations.

COM-LOC limits the x- and y-coordinates of the position distribution on the basis of the calculated probability over distance distribution. We use this information to make a bounding box (as in [7]) and to keep the computational costs as low as possible. In the prediction phase we draw samples within the bounding box. After the prediction phase we filter/weight the samples by multiplying the computed probabilities of Section IV.
VI. SIMULATIONS

This section analyzes the localization performance of COM-LOC++. In addition, we compare COM-LOC++ with COM-LOC([17]), ecolocation ([14]) and a modified version of the MLE described in [13] and [17]. Ecolocation only processes one-hop information, therefore it requires less communication than the other localization algorithms.

A. Set-up

Throughout this paper we use the same set-up, except when stated otherwise. The set-up parameters are:

- The surface area is 100x100 m.
- The simulations simulate the RSS by using the model described in Equation 1. In general, the following parameter values are used: \( P_{d0} = -40 \text{ dBm}, n = 3.5, \sigma_{dBm} = 6 \).
- 36 reference nodes are randomly placed over the surface area.
- 400 blind nodes are randomly and uniformly placed over the surface area.
- The localization performance is given as the mean over 25 runs.

B. Comparison with Other Localization Algorithms

This subsection analyzes the performance as a function of \( \sigma_{dBm} \), as \( \sigma_{dBm} \) defines the performance of RSS-based localization algorithms. We express the performance in terms of two statistical quantities:

- The mean error over the localization surface, which we interpret as the localization error.
- The standard deviation of the localization error, which we interpret as the localization stability.

Typical values of \( \sigma_{dBm} \) are between 6 and 12 dBm ([2]). For completeness, we evaluate the RSS-based localization algorithms with \( \sigma_{dBm} \) values between 4 and 12 dBm. Figures 5 and 6 show the localization error and stability as a function of \( \sigma_{dBm} \). These figures show that COM-LOC++ outperforms COM-LOC and the RSS-based localization algorithms in terms of localization error and stability:

- COM-LOC++ decreases the localization error by 47...109% and the localization stability by 22...56% in comparison with COM-LOC. These results clearly show that the extra information processed by COM-LOC++ significantly increases the performance of COM-LOC.
- COM-LOC++ decreases the localization error by 15...175% and the localization stability by 67...206% in comparison with DV-PAT and ECOLOCATION. Note that DV-PAT and ECOLOCATION both use RSS measurements, while both COM-LOC++ and COM-LOC only use connectivity information.

Moreover, DV-PAT has a 5...28% smaller localization error than COM-LOC with small values of \( \sigma_{dBm} \) (\( \sigma_{dBm} = 4...6 \text{ dBm} \)), nevertheless COM-LOC has a significant better localization stability than DV-PAT with these small values of \( \sigma_{dBm} \) (32...56%).

C. Node Density

This subsection analyzes the performance as a function of the blind node density, as we expect that the node density influences the localization performance of COM-LOC++ (see Section IV-D). Figure 7 and 8 show this functional dependence on the node density. These figures show that:

- The localization performance of COM-LOC++ increases with an increasing node density. We refer to Section IV-D for an explanation.
- The localization stability of COM-LOC increases with an increasing node density.
- The localization error of COM-LOC, DV-PAT and ECOLOCATION remain more or less equal with an increasing node density.
- The localization stability of DV-PAT and ECOLOCATION remain more or less equal with an increasing node density.
• DV-PAT performs slightly better than COM-LOC++ with a low node density (50...100 blind nodes).

In addition, these figures show that COM-LOC++ decreases the localization error by 82% and increases the localization stability by 62% in a wireless network with a high node density.

VII. CONCLUSION AND FUTURE WORK

In this paper we introduced a new distributed range-free localization algorithm called COM-LOC++, which processes a new type of information. COM-LOC++ optimizes the localization performance for a communication protocol commonly employed by localization algorithms called sum-dist. Simulations show that the use of this new type of information increases the performance by 50...110% relative to its predecessor. In addition, comparative simulations of COM-LOC++ with two RSS-based localization algorithms show that COM-LOC++ performs 20...180% better than these algorithms over a wide range of conditions.

In the future we shall attempt to study the effect of erroneous calibration of the parameters of the Log-Normal Shadowing Model on the localization performance.

REFERENCES