Grid-Based Position Discovery

M. Onur Ergin, Vlado Handziski, Adam Wolisz
Telecommunication Networks Group, Technische Universität Berlin, Germany
Email: {ergin, handziski, wolisz}@tkn.tu-berlin.de

***Abstract*—Wireless embedded systems are becoming an increasingly important part of our lives. Information from such systems is almost exclusively meaningful when it is coupled with location information, or the location information itself is the desired knowledge to obtain. In this paper we explore a special sub-problem of location discovery: determining relative node positions when the nodes to be discovered are in a 2D grid configuration. We developed customized algorithms that leverage this information and compute the mapping between node IDs and the grid cells reliably and accurately, by using available "Received Signal Strength" information that common radio chips provide. We compare our approach to two classical 2D location discovery approaches, fingerprinting and MDS-MAP, on the same constrained configurations. Our simulations show that our methodology outperforms the competitors in various scenarios.**

I. INTRODUCTION

Discovering device location has long been a popular topic for indoor communication systems due to the real and increasing demand for location information for many kinds of applications. There are many proposals to discover locations of wireless devices, however, many of them do not take the natural constrains of the indoor deployment environment into consideration. For example, placing things inside a building often imposes a structure into deployment that allows us to develop customize localization solutions.

In this work we achieve mapping n node IDs to n potential node positions in spaces that are partitioned as a grid with regular cells along x and y axes. We aim to find the single correct matching amongst $n!$ possible mappings. This is a common case in many indoor environments, such as offices, hospitals or warehouses. We assume that each cell of the grid is occupied with one node and the sizes of the cells are relatively small, i.e. 5 or 10 meters on one side which are typical for indoor deployments. An example scenario of this kind of application is locating containers inside a warehouse. Containers are usually stored on multiple rows at roughly equal distances in regularly partitioned spaces. When one needs to locate a particular container, he or she needs to identify its placement relative to the other ones. Or in a hospital, one might need to find out where a particular bed is, without having to visit each bed separately. This can be applied to many other areas, such as identifying the sequence of train cars or seat placement in an auditorium.

There are plenty of algorithms that suggest solutions using techniques like ultra-wide band signals, visual landmarkings, acoustic signals, fingerprinting or chirp shifting. These solutions often require special hardware, extensive training campaigns or some sort of pre-configuration. A small group of algorithms use the connectivity information of nodes spread around a relatively bigger area, which do not handle dense environments well. Today an algorithm with some level of success is not difficult to find. However, using only Received Signal Strength (RSS) without a long training campaign remains a big challenge.

In order to compute the positions of the nodes, our system requires the IDs of the reference nodes at corners of the grid. In fact, this is all the information we need prior to assigning node IDs to grid cells. Then we iteratively select the closest node to a node whose position is already discovered. To detect the closeness information, we leverage the RSS observed in multiple channels and statistical reasoning.

We evaluate the performances of our algorithms by comparing the empirical probability of perfect mapping within a set of experiments with two well-known approaches, MDS-MAP and RSS-based Fingerprinting, using simulation.

As an added quality metric, we assign a "reliability" value to our results. Rather than associating an average localization error to our results, we detect how likely it is that the computed grid maps all of the nodes to correct positions. Our reliability metric has three discrete values: high, medium and low.

The rest of the paper is structured as follows: the model of the targeted system is given in Section II. In Section III we present an algorithm for finding potentially closest nodes (candidates) to another node and provide mathematical definitions that we use in our grid mapping algorithms given in Sections IV and V. We elaborate on reliability assessment in Section VI. We then explain our simulation design and evaluation of the system in Section VII. Finally we give a brief summary of the related work in Section VIII and conclude in Section IX.

II. SYSTEM MODEL

We assume the nodes are placed on a grid, each cell of which is occupied with one node only. We consider that the grid has regular rows and columns in a rectangular shape. Each cell of the grid is a potential position for any node and there are no empty positions. The nodes are approximately, but not strictly, at equal distances to each of their immediate neighbours and these distances can be as small as half a meter to a few meters. A small subset of these nodes are taken as "Reference Nodes" and their positions in the grid are known to us. Any pair of nodes may or may not be within the communication range of any other node. In other words, full connectivity is allowed but not required, as long as each node can communicate with any other node in a multihop manner.

The radio chips that the nodes are equipped with, must be able to support multiple frequencies.

III. Discovering the Closest Node using Multi-channel RSS measurements

The spatial power density of a transmitted wireless signal is reduced as it travels farther in space. Under ideal conditions, a node that is closer to the source of a signal should measure a stronger RSS value than another node which is farther away. However, especially indoors, multipath phenomena challenge this expectation and might cause a node to measure smaller RSS values than another one located farther from the same transmitter. In our previous study [6], we showed how frequency diversity can be used to alter the propagation paths and how combined measurements of RSS from one transmitter to multiple receivers could be leveraged to detect the node that is most likely to be the closest receiver to that particular transmitter.

In this paper we present a procedure to detect a set of possibly closest nodes (candidates) to another node. This procedure runs between a "sender node" and a set of "receiver nodes". The sender node, N_i, transmits t number of packets at e number of channels, preferably at all the channels available to the radio hardware. The receivers listen to these transmissions and record the measured RSS values for each transmission. At the end of the transmission campaign, each receiver returns a summary of the measurements, which for each channel is the mean value of measured RSS values that are greater than θ percentile of all values measured on that channel. A sender transmits $e \times t$ packets and the receivers transmit e summaries back for each channel. Using control messages after each t transmissions, the transmitters can trigger the channel change, so other nodes will relay this control message and change to the new channel within a given time-window. We elaborated on the channel synchronization in our previous paper [6]. The sender node N_i compares the collected RSS summaries for each channel and puts the ID of the node that reports the biggest RSS summary for each channel into a vector, V_i. Also, the ID of any other node that reports RSS summary within $\pm$ S dbm of the maximum value of RSS summary in V_i is added into that vector V_i, as well.

Eventually, each sender generates a vector of candidates with a minimum size of e elements. This procedure is summarized in Algorithm 1. The mode (most frequent element) of this vector, $mode(V_i)$, is the node that is most likely to be the closest node to that particular sender. In our algorithms we also use the second mode (second most frequent element) of the candidate vector as well, which we denote with $mode_2(V_i)$.

The ratio of the frequency of a node in the vector to the length of the vector gives the probability of that node being the closest node, as in Equation 1.

$$P(N_r|N_i) = \frac{\text{frequency of } N_r \text{ in candidate vector } V_i \text{ of } N_i}{\text{size of } V_i} \quad (1)$$

Algorithm 1 findCandidates(Node N_i, threshold=θ, sensitivity=S, includeList = L)

Step 1: N_i transmits t packets at each one of the e channels $(t \geq 1)$

Step 2: Each receiver node, $N_r \in L$, in the communication range measures RSS at each e channels and reports the average of $\overline{\text{RSS}}$ measurements that are greater than θ percentile (RSS^{θ}) at each channel to N_i

Step 3: N_i produces a vector, $\overline{V_i}$, of receiver nodes that reported the $\pm S\ dbm\ of\ max(\overline{RSS^{\theta}})$ value at each channel

Return: V_i

IV. Grid-Based Position Discovery using two reference nodes: GBPD-2

A. Assumptions:

We investigate the possibility of detecting node positions on a grid setting, while minimizing the amount of a priory knowledge. First assumption is that the nodes of the grid can form a connected graph. In other words, a node may reach any other node through multihop communication. Two nodes on two corners of the grid that share one edge are selected as reference nodes ($R1, R2$), such as shown in Table I. The number of rows and columns are not known to us. We call this system Grid Based Position Discovery with 2 reference nodes (GBPD-2).

$\boldsymbol{R1}$	$G_{1,2}$	$G_{1,3}$	...	$G_{1,m}$
$G_{2,1}$	$G_{2,2}$	$G_{2,3}$	...	$G_{2,m}$
$G_{3,1}$	$G_{3,2}$	$G_{3,3}$	...	$G_{3,m}$
		...		
$G_{(n-1),1}$	$G_{(n-1),2}$	$G_{(n-1),3}$	...	$G_{(n-1),m}$
$\boldsymbol{R2}$	$G_{n,2}$	$G_{n,3}$	...	$G_{n,m}$

TABLE I
Example node placement in Grid **G**

B. Approach:

The system first discovers the nodes along the common edge shared with reference nodes. The discovery is initiated by one of the reference nodes selecting two candidates as the immediate neighbor along the edge. If the node that selects the candidates is positioned at $G_{r,c}$ of the grid G (cell at r^{th} row and c^{th} column), then the candidates should be on cells $G_{r+1,c}$ and $G_{r,c+1}$ as they are the spatially closest cells. Then the other reference node selects one of these two candidates as the node closer to itself, indicating that it belongs to the common edge. This procedure starts from the first edge (column) of the grid (hence c is always 1), and iterates until all nodes between two reference nodes along the same edge are discovered. Each candidate-finding includes the set of nodes that are not yet discovered in the grid, which we denote with G'. This procedure is defined in Algorithm 2.

Knowing that this edge is the first column of the grid, each consequent column is discovered one cell at a time, in a row-by-row basis. During the procedure, a node in $G_{r,c}$ computes its candidate list, L1, using the Algorithm 1. If L1 is bi-modal (having two *modes*), then the node in the next row of the same column, $G_{r+1,c}$ computes another candidate list, L2, and its mode is put in $G_{r+1,c+1}$. In this case $G_{r,c}$ again computes its candidate list L1 and eventually the mode of L1 is placed in $G_{r,c+1}$. This procedure is iterated for each row in a column, until all nodes are placed in the grid. The whole procedure is summarized in Algorithm 3. At each update of the grid with a node placement, the current state of the grid G is distributed to all nodes (using a protocol mechanism independent of our algorithm). Therefore, distributed iterations can continue.

Algorithm 2 findEdge(Node $N1$, Node $N2$)

```
Initiate EdgeList
M1 = M2 = null
while (M1 OR M2) ≠ N2 do
    V_1 ← findCandidates(N1, θ, S, includeList=G')
    M1 ← mode(V_1)
    M2 ← mode_2(V_1)
    V_2 ← findCandidates(N2, θ, S, includeList={M1,M2})
    EdgeList.add(mode(V_2))
end while
Return: EdgeList
```

Algorithm 3 buildGrid(Node $R1$, Node $R2$)

```
firstColumn = findEdge(R1, R2)
G(..., 1) ← firstColumn
c ← 2                                ▷ starting from second column
while unplaced nodes exist  do
    for r in Each Row do
        L1 ← findCandidates(G_{c−1,r}, includeList=G')
        if L1 is bi-modal and G_{c−1,r+1} exists then
            L2 ← findCandidates(G_{c−1,r+1}, includeList=G')
            G_{c,r+1} ← mode(L2)          ▷ skip next iteration
            L1 ← findCandidates(G_{c−1,r}, includeList=(G'))
        end if
        G_{c,r} ← mode(L1)
    end for
end while
```

V. Grid-Based Position Discovery using three reference nodes: GBPD-3

A. Assumptions:

We assume we have a connected graph and **three** reference nodes, $R1$, $R2$ and $R3$, are at positions $G_{1,1}, G_{1,2}$ and $G_{2,1}$ of the Grid G as a priori information, as shown in Table II. The size of at least one of the dimensions and the total number of nodes are also known to us.

B. Approach:

Using 3 reference nodes ($R1$, $R2$ and $R3$), first the inner corner of the reference nodes (namely, $G_{2,2}$ in Table II) is

$\boldsymbol{R1}$	$\boldsymbol{R2}$	$G_{1,3}$	...	$G_{1,m}$
$\boldsymbol{R3}$	$G_{2,2}$	$G_{2,3}$	...	$G_{2,m}$
$G_{3,1}$	$G_{3,2}$	$G_{3,3}$	...	$G_{3,m}$
...				
$G_{(n-1),1}$	$G_{(n-1),2}$	$G_{(n-1),3}$	...	$G_{(n-1),m}$
$G_{n,1}$	$G_{n,2}$	$G_{n,3}$	...	$G_{n,m}$

TABLE II
EXAMPLE NODE PLACEMENT IN GRID **G**

discovered, completing the mapping of nodes to a 2×2 square grid. Next, the adjacent neighbors on either sides of the square is discovered: 2 nodes for the vertical side ($G_{1,3}$ and $G_{2,3}$), 2 nodes for the horizontal side($G_{3,1}$ and $G_{3,2}$). Now the system has 8 nodes of a 3×3 grid discovered. The iteration continues with the discovery of the missing corner node ($G_{3,3}$) of the square grid, until one of the edges is completely covered. Then the discovery is iterated with edges only, until all the nodes are positioned. For this procedure, we define three methods: **fc** (find corner), **fte** (find top edge), **fe** (find edge), which are given in Algorithms 4, 5 and 6 respectively. The procedure itself is given in Algorithm 7 and illustrated in Figure 1.

Algorithm 4 fc(Node $N1$, Node $N2$, Node $N3$)

```
V_1 ← findCandidates(N1, θ, S, includeList = G' )
V_2 ← findCandidates(N2, θ, S, includeList = V_1)
V_3 ← findCandidates(N3, θ, S, includeList = V_1)
Return:
cornerNode ← mode(V_2 + V_3)
```

Algorithm 5 fte(Node $N1$, Node $N2$)

```
V_1 ← findCandidates(N1, θ, S, includeList = G')
c1 ← mode(V_1)
c2 ← mode_2(V_1)
Return:
arg max(P(c1|N1) × P(c1|N2), P(c2|N1) × P(c2|N2))
```

Algorithm 6 fe(Node $N1$, Node $N2$)

```
V_1 ← findCandidates(N1, θ, S, includeList = G')
if V_1 is bi-modal and N2 exists then
    V_2 ← findCandidates(N2, θ, S, includeList = G')
    V_1 ← findCandidates(N1, θ, S, includeList = (G'−V_2))
end if
Return:
mode(V_1)
```

An example iteration of buildGrid() algorithm is shown in Table III. The cells are tagged with labels {iteration_number:function}.

Algorithm 7 buildGrid()

$n_x \leftarrow$ number of Grid rows
$n_y \leftarrow$ number of Grid columns
Assumption: $n_x < n_y$
$G_{2,2} \leftarrow fc(R1, R2, R3)$ ▷ 2×2 square shape is achieved
for d in 3 to n_x **do**
  $G_{1,d} \leftarrow fte(G_{1,d-1}, G_{1,d-2})$
  $G_{d,1} \leftarrow fte(G_{d-1,1}, G_{d-2,1})$
  for j in 2 to $d-1$ **do**
    $G_{j,d} \leftarrow fc(G_{j-1,d-1}, G_{j-1,d}, G_{j,d-1})$
    $G_{d,j} \leftarrow fc(G_{d-1,j-1}, G_{d-1,j}, G_{d,j-1})$
  end for
  $G_{d,d} \leftarrow fc(G_{d-1,d-1}, G_{d-1,d}, G_{d,d-1})$
end for
for d in $n_x + 1$ to n_y **do**
  for r in 1 to n_x **do**
    $G_{r,d} = fe(G_{r,d-1}, G_{r+1,d-1})$
  end for
end for

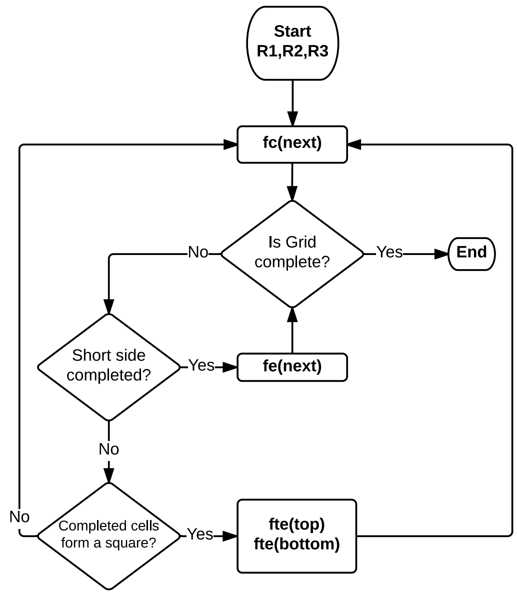

Fig. 1. Flowchart of GBPD-3, showing how *find corner (fc)*, *find top edge (fte)* and *find edge (fe)* procedures cooperate. *next*, *top* and *bottom* stand for the next cells to be discovered.

R1	**R2**	2:fte	7:fte	14:fe
R3	1:fc	4:fc	9:fc	15:fe
3:fte	5:fc	6:fc	11:fc	16:fe
8:fte	10:fc	12:fc	13:fc	17:fe

TABLE III
ITERATIONS AND FUNCTIONS

VI. Reliability Analysis

Due to the iterative nature of the process, an error in the midst of the grid would effect discovery of all the remaining nodes from that error on. We therefore verify the first and the last edges for consistency and assess how likely it is that all the nodes are positioned in the grid at the right places. We use Algorithm 2 ($findEdge$) to discover the mapping of a set of nodes to one edge within a two dimensional grid. This algorithm takes the two ends of the edge as input and works iteratively from one end to the other end.

First, we discover one edge, which contains two of the reference nodes, starting from one corner node towards the direction of the reference node $R1$. Then, we take the nodes at the opposite corners of the grid ($G_{1,m}$ and $G_{n,m}$) and re-discover the last column using Algorithm 2 to compare with what has already been found in the grid. Finally, we re-discover the last column in the reverse node order and apply the same comparison. Each of these three comparisons have a "score" value of 1. At the end, the sum of scores corresponds to a reliability classification. This algorithm is designed for grids with more than 2 rows.

$$\begin{aligned}
&score = 0\\
&\text{if } findEdge(G_{n,1}, G_{1,1}) == Grid[G_{1,1}, G_{n,1}], \quad score{+}{+}\\
&\text{if } findEdge(G_{1,m}, G_{n,m}) == Grid[G_{1,m}, G_{n,m}], \quad score{+}{+}\\
&\text{if } findEdge(G_{n,m}, G_{1,m}) == Grid[G_{n,m}, G_{1,m}], \quad score{+}{+}
\end{aligned}$$

Where the $++$ operand increments the preceding variable by 1, and **Grid**[x,y] is the set of nodes already mapped into one edge of a grid G by a GBPD algorithm.

And the reliability value will be:

$$reliability = \begin{cases} high & \text{, if } score == 3 \\ medium & \text{, if } score == 2 \\ low & \text{, if } score \leq 1 \end{cases}$$

VII. Evaluation

A. Simulation Design

To analyze our algorithms we used a simulation model that generates RSS values between any two nodes as a function of distance, which represents an indoor environment. We previously verified the validity of our simulation model by comparing it to real world measurements [6]. Our model assumes large-scale variation of RSS to be caused by change in distance and we did not consider shadow fading. Small scale variation of RSS is due to the fact that the wireless signals arrive at the receiver from multiple paths and therefore are affected by multipath fading. As each channel bandwidth is smaller than coherence bandwidth of usual office environments with around 100 ns delay spread, we assume a frequency flat fading for each channel but the fading gain is expected to change from one channel to the other channel which provides the frequency diversity of the system. Finally, we assume that the environment is static and changes slowly over time, which means that the coherence time is very large. RSS is also affected by the thermal noise of the receiver.

The propagation effects are modeled using statistical channel models including path loss [15], [7]. The signal strength values are produced based on the previous assumptions. The transmission power is constant and the path loss exponent α

is assumed to be static in the environment. We used Rayleigh fading model to model small scale fading, where the channel fading gains are from a Rayleigh distributed random variable. The fading gains are assumed to be constant in a channel due to a slowly fading assumption. However, it is assumed that the fading gain is different in each channel. The relation between received power P_R and transmit power P_T, both in $milliwatts$, is presented in the following equation:

$$P_R = |h|^2 \times P_T \times \left(\frac{\lambda}{4 \times \pi \times D}\right)^{\alpha} + P_n \qquad (2)$$

where h is the channel gain, which is a Rayleigh distributed random variable. λ is the wavelength at 2.4GHz, α is the path loss exponent and D is the distance between transmitter and receiver antennas in meters. P_n is the power of additive white Gaussian noise in the environment. We have considered two channel gain conditions, namely **moderate noise** (with variance of -45dbm) and **high noise** (with variance of -30dbm). In our model, -45dbm of noise variance causes RSS measurements between two static nodes at the same distance to deviate about 1-2dbm, which is also the precision of the CC2420 radio chip that is commonly used in wireless embedded systems. A noise variance of -30dbm causes the readings under the same conditions to deviate about 4-5dbm, which is possible in very noisy indoor environments, just like we observed in our previous measurement-based study [6]. The fading gain variance h was taken as 15. Both random variables have zero mean. Path loss exponent α was taken as 3, which is typical for indoor office environments.

In Figure 2, we show a set of generated RSS values using the above formula and parameters. Each produced RSS value corresponds to the reading at different transmitter-receiver distances. The graphs show that the signal strength value is a function of the distance, however it is not always monotonically decreasing with distance due to multipath fading.

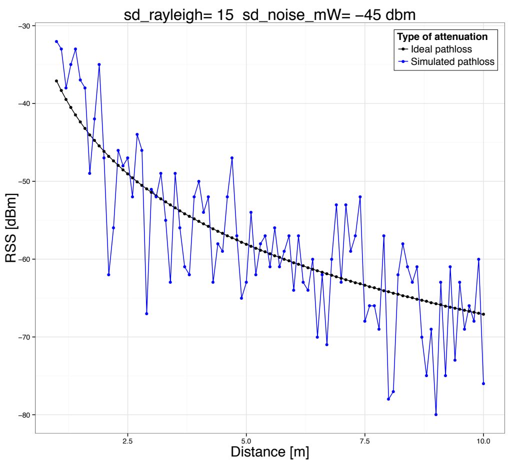

Fig. 2. Produced RSS values

B. Simulation results

For our evaluations we have generated RSS values according to the above explained simulation model. We fed these generated RSS values to our algorithm, to MDS-MAP and to Fingerprinting (Nearest Neighbor) algorithms. The reason we compare our proposal with MDS-MAP is because MDS-MAP does not require anchor points, but only reference points to correct the orientation of its output. Therefore, the amount of required information is the same as in our system. The same amount of information can also be used for applying fingerprinting, which is frequently used for indoor localization systems. So it is natural to ask, whether the fingerprinting might work for this kind of problems.

We identify our results as either a "success" or a "fail". "Success" indicates correct discovery of positions of **all nodes in the grid setting** and "fail" indicates otherwise. This is a very strict evaluation criteria in contrast to the typical metrics used in evaluation of localization systems.

In [11], the authors defined MDS-MAP as a connectivity based approach that utilizes classical Multi-Dimensional Scaling (MDS). Feeding connectivity data to MDS gives a relative map of the node locations. By selecting a subset of the nodes as anchors and applying linear transformations to these nodes for fitting them into the known locations, the absolute map can be generated. We used the popular MDS implementation of Smacof [4]. For the linear transformations of anchor nodes, we applied procrustes analysis [8], which is implemented in the Vegan package for R [10].

When we talk about node position discovery in a static setting with definite partitions, RSS-based fingerprinting is a very popular method that finds wide use. We have used the original fingerprinting approach [1], since the measurement points are always same as the training points and we do not consider mobility of the nodes.

First we evaluated a topology of 20 simulated nodes, placed into the cells of a 5x4 grid, as shown in Table IV. Each node was placed 3 meters away from its immediate neighbor. We applied both GBPD-2 and GBPD-3, as well as MDS-MAP and Fingerprinting algorithms, on the same set of RSS values. The results are plotted in Figure 3 for two-reference-node case and in Figure 5 for three-reference-node case. For two-reference node case, we assumed that the positions of the corner nodes, N0 and N15, were known to us. Likewise, N0, N1 and N5 were assumed to be known to us for the three-reference-node case and all the rest of the system was to be discovered. Other algorithms in our comparisons were also given the same set of reference (anchor) nodes for the sake of fairness.

In addition to the success ratio, we also computed the "reliability" of the results. Figure 4 shows the classification of position discovery verdicts for the 5 x 4 topology using 2 reference nodes with GBPD-2 algorithm. Figure 6 shows the same information for GBPD-3 algorithm, which uses 3 reference nodes. Figures 7 and 9 show the performance of GBPD-2 and GBPD-3 on a 10 x 5 topology with 50 nodes, also in comparison with MDS-MAP and Fingerprinting algorithms.

The reliability analysis results for 10 x 5 topology are given in Figures 8 and 9 respectively.

N0	**N1**	N2	N3	N4
N5	N6	N7	N8	N9
N10	N11	N12	N13	N14
N15	N16	N17	N18	N19

TABLE IV
5X4 GRID OF NODE PLACEMENT FOR SIMULATIONS

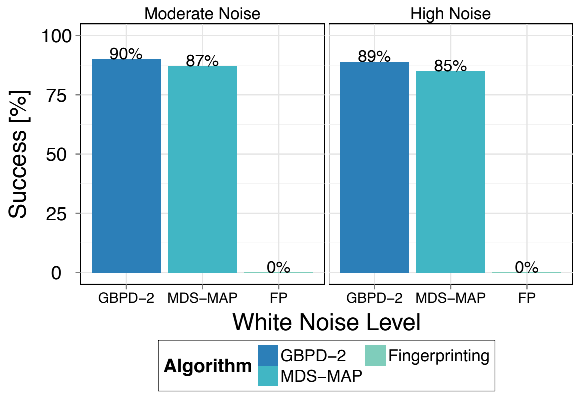

Fig. 3. GBPD-2: Grid based position discovery using two reference nodes, 5 x 4 topology

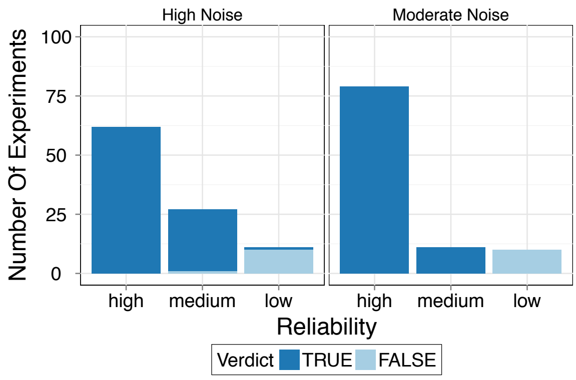

Fig. 4. GBPD-2: Reliability analysis of the results, 5 x 4 topology

Results show that using such a low number of anchor points (reference nodes) for Fingerprinting is not sufficient. In certain cases, such as 5 x 4 topology which is closer to a square shape, MDS-MAP offers a satisfactory performance but fails to maintain stability when the topology diverges from the square shape, such as for a 10 x 5 shaped topology. Also, it is more sensitive to the noise level of the wireless medium than GBPD approaches.

Here we need to mention a special case of MDS-MAP performance: if it was given three anchor nodes at three corners of a grid that is closer to a square shape, such as the 5 x 4 setting, it was able to produce 100% correct results

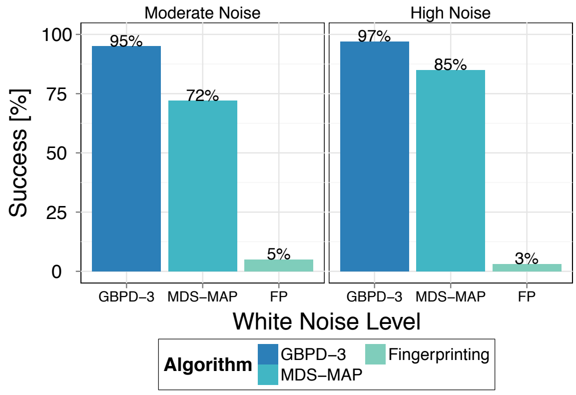

Fig. 5. GBPD-3: Grid based position discovery using three reference nodes, 5 x 4 topology

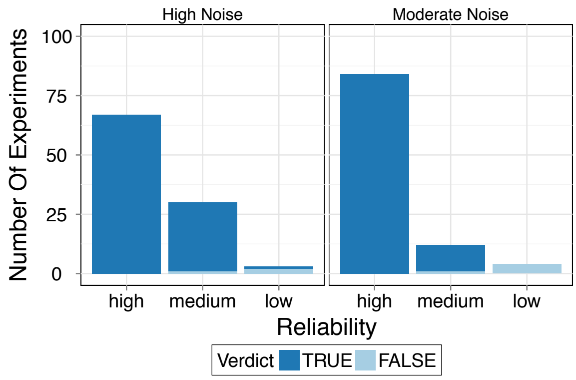

Fig. 6. GBPD-3: Reliability analysis of the results, 5 x 4 topology

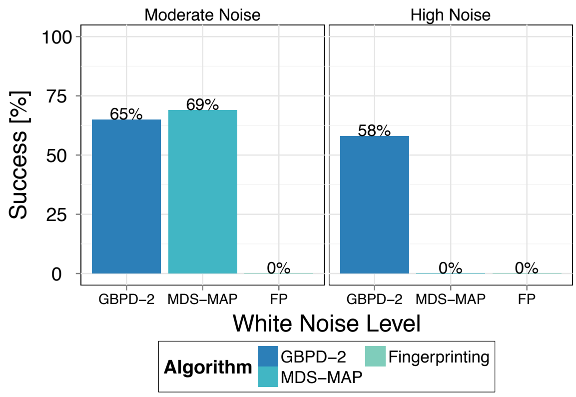

Fig. 7. GBPD-2: Grid based position discovery using two reference nodes, 10 x 5 topology

in moderate-noise conditions. For higher noise or lengthy topology settings, such as a 10 x 2 grid, the performance of MDS-MAP did not improve by providing 3 anchor nodes at three different corners.

The reliability analysis show that we can successfully categorize imperfect mappings (Verdict: FALSE) to "low"

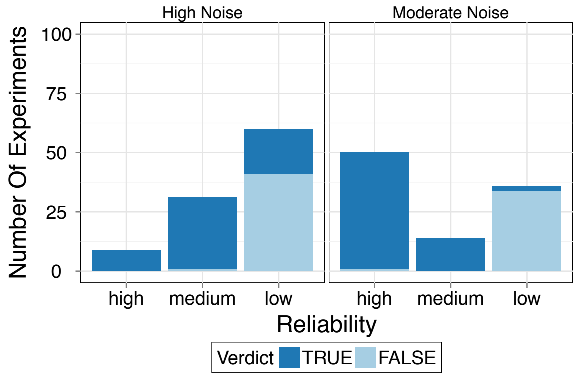

Fig. 8. GBPD-2: Reliability analysis of the results, 10 x 5 topology

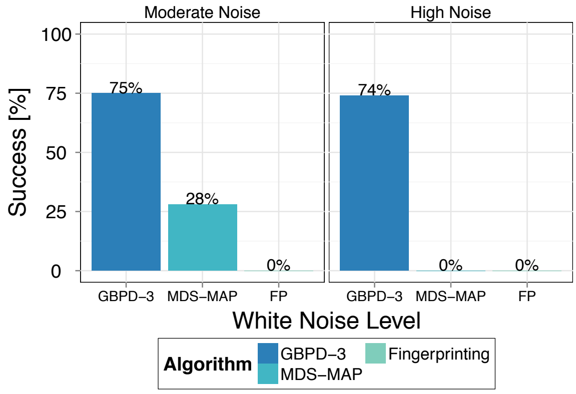

Fig. 9. GBPD-3: Grid based position discovery using three reference nodes, 10 x 5 topology

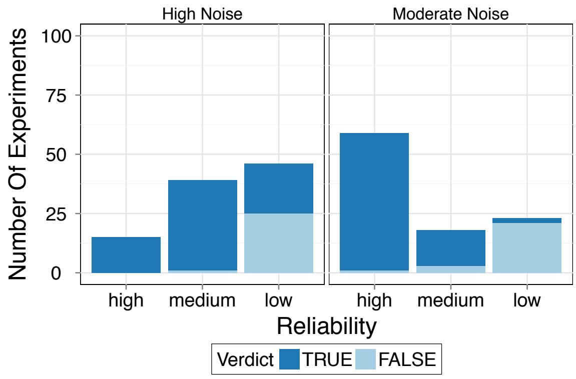

Fig. 10. GBPD-3: Reliability analysis of the results, 10 x 5 topology

reliability level, whereas "high" reliability level contains about 100% of correct mappings (Verdict: TRUE).

VIII. Related Work

RSS has been widely exploited for extracting location information from wireless systems. Various studies have concluded that fading related distortions cause significant range estimation errors when attempting to compute geographic distance [16],[18],[3]. Other systems, such as centroid, APIT or DV-Hop, which do not rely on direct inversion of the path-loss function to inter-node distances, show better performance like our method in this paper.

Instead of compensating for the distortion effects of fading by complex mechanisms, we benefit from the diversity in the frequency domain which enables us to observe different fading behaviors for the nodes that remain static the entire time of operation [20]. Averaging RSS measurements across different channels, in an attempt to stabilize the RSS estimates, has also been investigated by others [2], [19].

A 2-D range-free localization approach was introduced in [17], which extracts location sequences from distance ranks from a set of reference nodes by splitting the space into unique regions. Although they face the same challenges as we do in our work due to the RSS distortions, leveraging frequency diversity is not among the techniques they used to distance rank sequences. Also their method implicitly requires a higher number of reference nodes.

Relative signature distance is another range-free localization technique that calculates expected proximity between immediate neighbors by leveraging node connectivity [21]. Although this approach shows good usability outdoors, it lacks any mechanisms to deal with indoor conditions.

Another method presented in [5] deals with heterogeneous wireless sensor networks with different transmission capabilities, ie. ranges. They derive the node locations locally from expected hop progress, in which the nodes broadcast their own transmission capabilities, and use the anchor locations that they are aware of. However, the accuracy of the localization is within half of the a node's transmission range. In comparison, our work can not only deal with much denser settings, but also implicitly supports heterogeneous networks without nodes needing to announce their transmission capabilities, since the closest node selection mechanism uses one sender and multiple receivers.

An anchor-free localization method is proposed in [12] that is comparable to MDS-MAP. The authors use estimated distances between the neighboring nodes and generate the locations as a result of a central non-linear optimization process. They claim that they can increase the localization accuracy if MDS output is given to their algorithm as the initial state. However, they rely on measured ranges from one node to another, which in practice is sensitive to the challenging propagation effects elaborated in Section III, resulting in erroneous location estimates.

The surveys [13], [14], [9] commonly concluded that range-free location discovery systems fall short in accuracy. In contrast, our system provides high accuracy and an assessment of whether the perfect accuracy has been reached. This fact qualifies our study as being more applicable for a range-free position discovery system, even for challenging and dense indoor environments.

IX. Conclusion

In this study we investigated a solution for detecting relative two-dimensional geographic positions of multi-channel wireless systems under challenging indoor conditions. The requirements of our system are a radio unit that can communicate at different radio signal frequencies, installed on the wireless nodes that are deployed in a regular two-dimensional grid setting, and the initial position information of two or three nodes at any corner of that grid.

We have shown that, by combining frequency diversity with statistical reasoning, the relative positions of wireless capable devices can be detected at a high precision in a dense deployment. Based on our previous work, and the real-world measurements we made in that work, we have developed a simulation to reflect the behavior of an indoor medium and we have shown how the suggested system outperforms its best competitors MDS-MAP and Fingerprinting.

Our two proposals, GBPD-2 and GBPD-3 show that, this system is extendable, or even replaceable with alternative and potentially improved methods for discovering the deployment grid using our procedure *findCandidates* for detecting the closest node to another node. This flexibility allows users to develop customized approaches to their systems.

Finally, as an added feature, we successfully assess the outcome of the grid building algorithms to tell how reliable they are (high, medium or low), which tells the user whether a particular measurement needs to be repeated.

References

[1] P. Bahl and V. Padmanabhan. Radar: an in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, 2000.

[2] A. Bardella, N. Bui, A. Zanella, and M. Zorzi. An experimental study on ieee 802.15.4 multichannel transmission to improve rssi-based service performance. In *Proceedings of the 4th international conference on Real-world wireless sensor networks*, REALWSN'10, 2010.

[3] J. Cavers. *Mobile Channel Characteristics*. Kluwer Academic, 2000.

[4] J. de Leeuw and P. Mair. Multidimensional scaling using majorization: Smacof in r. *Journal of Statistical Software*, 31(3):1–30, 8 2009.

[5] A. El Assaf, S. Zaidi, S. Affes, and N. Kandil. Hop-count based localization algorithm for wireless sensor networks. In *Mediterranean Microwave Symposium (MMS), 2013 13th*, pages 1–6, Sept 2013.

[6] M. O. Ergin, V. Handziski, A. Behboodi, and A. Wolisz. Determining node sequence in a linear configuration. In *Indoor Positioning and Indoor Navigation (IPIN), 2014 International Conference on*, pages 534–543, Oct 2014.

[7] A. Goldsmith. *Wireless Communications*. Cambridge University Press, New York, NY, USA, 2005.

[8] J. Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[9] A. Mesmoudi, M. Feham, and N. Labraoui. Wireless sensor networks localization algorithms: a comprehensive survey. *CoRR*, abs/1312.4082, 2013.

[10] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

[11] Y. Shang, W. Ruml, Y. Zhang, and M. P. Fromherz. Localization from mere connectivity. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 201–212. ACM, 2003.

[12] S. Shioda and K. Shimamura. Anchor-free localization: Estimation of relative locations of sensors. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*, pages 2087–2092, Sept 2013.

[13] S. P. Singh and S. Sharma. Range free localization techniques in wireless sensor networks: A review. *Procedia Computer Science*, 57:7 – 16, 2015. 3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015).

[14] R. Stoleru, T. He, and J. Stankovic. Range-free localization. In R. Poovendran, S. Roy, and C. Wang, editors, *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks*, volume 30 of *Advances in Information Security*, pages 3–31. Springer US, 2007.

[15] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, New York, NY, USA, 2005.

[16] K. Whitehouse, C. Karlof, and D. Culler. A practical evaluation of radio signal strength for ranging-based localization. *SIGMOBILE Mob. Comput. Commun. Rev.*, 11(1), Jan. 2007.

[17] K. Yedavalli and B. Krishnamachari. Sequence-based localization in wireless sensor networks. *Mobile Computing, IEEE Transactions on*, 7(1), 2008.

[18] G. Zanca, F. Zorzi, A. Zanella, and M. Zorzi. Experimental comparison of rssi-based localization algorithms for indoor wireless sensor networks. In *Proceedings of the workshop on Real-world wireless sensor networks*, REALWSN '08, 2008.

[19] A. Zanella and A. Bardella. Rss-based ranging by multichannel rss averaging. *Wireless Communications Letters, IEEE*, 3(1):10–13, February 2014.

[20] D. Zhang, Y. Liu, X. Guo, M. Gao, and L. Ni. On distinguishing the multiple radio paths in rss-based ranging. In *INFOCOM, 2012 Proceedings IEEE*, Mar. 2012.

[21] Z. Zhong and T. He. Rsd: A metric for achieving range-free localization beyond connectivity. *Parallel and Distributed Systems, IEEE Transactions on*, 22(11):1943–1951, Nov 2011.