

Research Article

Address Assignment in Indoor Wireless Networks Using Deterministic Channel Simulation

Edgar Holleis¹ and Christoph Grimm²

¹Institute of Computer Technology, Vienna University of Technology, Gußhausstraße 2729, 1040 Wien, Austria

²Department of Computer Science, University of Kaiserslautern, Postfach 3049, 67663 Kaiserslautern, Germany

Correspondence should be addressed to Christoph Grimm; grimm@cs.uni-kl.de

Received 27 May 2013; Accepted 8 July 2013

Academic Editors: A. Bonastre, M. Ekström, and L. Reggiani

Copyright © 2013 E. Holleis and C. Grimm. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A crucial step during commissioning of wireless sensor and automation networks is assigning high-level node addresses (e.g., floor/room/fixture) to nodes mounted at their respective location. This address assignment typically requires visiting every single node prior to, during, or after mounting. For large-scale networks it also presents a considerable logistical effort. This paper describes a new approach to automatically assign high-level addresses without visiting every node. First, the wireless channel is simulated using a deterministic channel simulation in order to obtain node-to-node estimates of path loss. Next, the channel is measured by a precommissioning test procedure on the live network. In a third step, results from measurements and simulation are condensed into graphs and matched against each other. The resulting problem, identified as weighted graph matching, is solved heuristically. Viability of the approach and its performance is demonstrated by means of a publicly available test data set, which the algorithm is able to solve flawlessly. Further points of interest are the conditions that lead to high quality address assignments.

1. Introduction

Building automation is not a new technology. However, with the advent of smart environments and the Internet of Things, the topic is again the focus of increased attention. This is especially true for *wireless* building automation systems, overlapping strongly with wireless sensor networks. This work deals with address assignment in wireless (building) automation networks, an underresearched topic, emerging primarily from experience with large-scale installations.

By *address assignment* we mean attaching a high-level address which conveys locality (e.g., floor/room/fixture) to the nodes of the wireless network. The goal is, in other words, to produce a mapping from the addresses used by the MAC layer onto logical addresses based on location or role.

Some protocols are explicit in their distinction between low-level address and high-level address. Other protocols, such as ZigBee, rely on low-level addresses for communication but support high-level roles with group membership or binding tables. High-level addresses become explicit again if dealt with in the context of a deployment plan or professional

commissioning tool. Address assignment is a necessary step of commissioning of automation networks.

While this work deals with both location and wireless channel measurements, it is different from the field of sensor node localization. Localization is typically a continuous problem; address assignment is a combinatorial problem. It is range free; that is, it does not assume any empirical relation between signal attenuation and distance. It does not rely on fingerprinting the wireless channel and does not need anchor nodes like indoor GPS. Instead, it uses a deterministic channel simulation to deal with the problems of multipath propagation. There is an ample body of literature showing that indoor localization based on channel measurements alone is not able to provide the necessary precision to reliably and unambiguously identify mounting locations with acceptable error rate.

This work is also not about autoconfiguration protocols. The latter are the consumers of the address mapping produced by address assignment. A device without flash memory can use an autoconfiguration protocol to download its configuration from network management. But network

management needs to have prior knowledge about the mapping between hardware addresses and configuration records.

Address assignment is relatively expensive because it traditionally requires manual intervention on a per-node basis. This work presents a novel way to conduct address assignment without manipulating nodes individually. The presented technique therefore bears the potential of substantially lowering the cost of commissioning wireless building automation networks.

1.1. Outline of the Presented Method. The presented technique for address assignment works on a freshly installed network. The nodes are powered and an initial, maybe temporary, wireless network has been formed. No further parametrization is needed; the only prerequisite is that nodes are reachable by hardware address.

We present the following three-step technique.

- (1) Channel estimation: using the deployment plan of the network and the floor plan of the building as inputs, a three-dimensional deterministic channel simulation produces the *simulated connectivity graph*. It is attributed with the nodes' logical addresses.
- (2) Channel measurement: all participating nodes cooperate to perform measurements of the wireless channel, such as RSSI measurement. The measurements are transmitted to the management workstation to assemble the *measured connectivity graph*. It is attributed with the participating nodes' hardware addresses.
- (3) Matching algorithm: simulated and measured connectivity graphs are isomorphic. They are thus matched against each other, producing the sought after mapping between hardware addresses and logical addresses.

Measured and simulated connectivity graphs are depicted in Figure 1. For the channel estimation, ray optical ray tracing is used. The deviation between simulation and measurement is modelled as noise. Even for challenging indoor environments, it is shown that noise levels are small enough to produce reliable address mappings with small error rate.

For the purpose of demonstrating the presented technique, the publicly available data set of King et al., recorded in 2008 at the University of Mannheim, is adapted and used [1]. The resulting address mapping is free of errors.

This work assumes the prior availability of detailed computer readable plans of the building, as well as a deployment plan of the network. This requirement is easily satisfiable for large, professional installations because the necessary plans are provided by the planning agency as part of their contractual obligations. The plans need to be available before the responsible contractor can begin installation of the automation network. Indeed, it is a stated goal of this work to devise an address assignment technique that naturally fits into established work flows.

2. State of the Art

Traditionally, address assignment happens prior to, during, or after mounting [2, 3].

- (i) Address assignment prior to mounting: the mapping is produced as soon as the hardware addresses of the components become known. From there, a plan is devised that documents which of the individual components should be installed into exactly which position. The components and the plan are handed to the installer who is responsible for the plan to be followed without errors [4].
- (ii) Address assignment during mounting: the mapping is created on the fly by the installer. He may be supported by a commissioning tool that can automatically read and record hardware addresses from printed labels. Another option is that components do not require hardware addresses at all and instead the logical addresses are programmed by the installer on the fly [5].
- (iii) Address assignment after mounting: components are mounted flexibly in any order. As part of the commissioning procedure, a technician visits every mounting location and records the hardware address of the component thereby creating the mapping. The important downside of this approach is that components may already be covered by panels, or components may be mounted in inaccessible locations. Luminaires, on the other hand, can locally transmit their MAC address out of band by modulating their light source [6].

All of the above is executed by workers on a per-devices basis, which is the major downside of the approach. It is a cost driver, a monotonous exercise, error prone, and requires extra training. The presented approach, by contrast, only requires few manual steps—all of those with network-wide impact.

Node localization is a related problem. It assumes no prior knowledge of mounting locations or, at least, does not make use of it. The field of node localization is of substantial breadth. Based on the taxonomies of [7–9], the presented work uses the following.

- (i) Active localization—all system components emit signals. The cooperation of localization target with infrastructure is needed to compute locations. The algorithm is centralized.
- (ii) Sensing via RF RSS—measurements of received signal strength (RSS) or alternatively time of flight (TOF). In contrast to many systems based on RSS measurements, no empirical relationship between measured data and distance is assumed. The system needs no dedicated hardware other than what is available in industry standard sensor nodes.
- (iii) Multihop localization based on radio connectivity, multilateration, without anchors—physical locations are calculated based on the devices position in the networks connectivity graph. In contrast to other works, the usage of anchors—special nodes with preconfigured location—is optional.

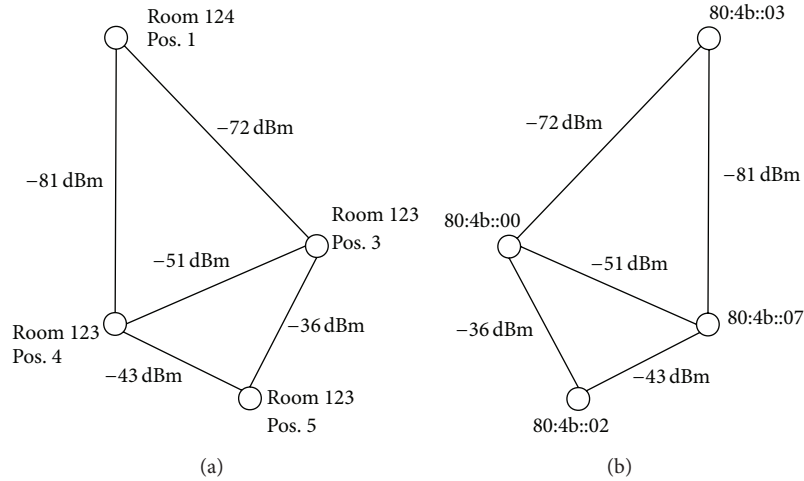


FIGURE 1: The connectivity graphs for illustration purposes. Real ones are dramatically bigger and weights have added noise. (a) Simulated connectivity graph attributed with logical addresses based on device location. (b) Measured connectivity graph attributed with MAC addresses.

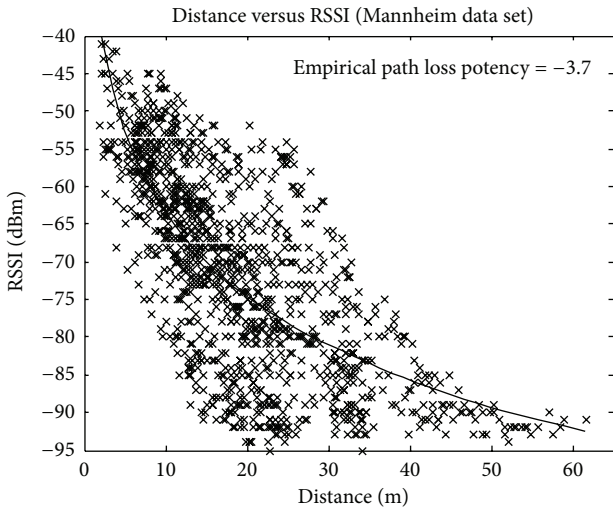


FIGURE 2: Distance versus RSSI of the test data set. The solid line is an empirical inverse-square-law model fitted to the data, with an inverse square of -3.7 .

- (iv) Locations are expressed symbolically (e.g., floor/room/fixture) and relatively to the particular structure or building.

For indoor localization based on signal attenuation, there are several approaches for overcoming the challenges presented by multipath propagation. Fingerprinting relies on building a detailed channel map of the building [10–12]. This relies on first installing a number of anchors that broadcast regular beacons and then taking systematic measurements based on a regular, dense grid. During the operation phase of the system, locations are calculated by measuring the beacons and comparing measurements to the map. King et al. observed distance errors of less than 3 m at the 90th percentile using a dense network of WiFi access points as beacons [12].

This approach is not well applicable to address assignment, since building the required channel map requires a higher effort than manual address assignment.

Alternatively, statistics can be used to combat the problems of multipath propagation by using a very dense network of anchors, up to several per room. While multipath propagation breaks the inverse-square-law relationship between signal attenuation and distance for any particular measurement, it still applies on average. Zanca et al. compared several localization algorithms [13–15]. In the case of 5 beacons per room, the average localization error is no better than 2 m. This may be adequate for address assignment; the overhead of installing several anchors per room, however, is not.

Anchor selection algorithms, finally, combat the effects of multipath propagation by sorting anchors into those having line of sight to the localization target and those that do not have [16, 17]. The high number of required anchors, again, makes the approach impractical for address assignment.

3. A New Method for Automatic Address Assignment

The informal description of the three-step process of address assignment is given in Section 1.1. This section builds on the previous definitions of the *simulated* and *measured connectivity graphs*. The problem of *matching* the graphs is thus formalized as follows:

$$\mathbf{G}_M = \mathbf{P}^\top \mathbf{G}_S \mathbf{P} + \text{off } \mathbf{J}_{SP} + \mathbf{N}_V \mathbf{J}_{SP} + \mathbf{J}_{SP} \mathbf{N}_V + \mathbf{N}_E \quad (1)$$

$$\mathbf{A}_{na,M} = \mathbf{P}^\top \mathbf{A}_{na,S} \mathbf{P},$$

where \mathbf{G}_M and \mathbf{G}_S are adjacency matrices of the measured and simulated connectivity graphs, symmetric, representing negative attenuation loss of the channel in dB (e.g., -95 dB), $\mathbf{A}_{na,M}$ and $\mathbf{A}_{na,S}$ are diagonal matrices holding one of na vertex attributes common to measured and simulated connectivity graphs (e.g., device types), \mathbf{P} is the permutation

matrix with the sought after mapping (address assignment), \mathbf{J}_{SP} is a logical matrix indicating the nonzero elements of $\mathbf{P}^T \mathbf{G}_S \mathbf{P}$, $off \mathbf{J}_{SP}$ is the global offset between weights in measured and simulated connectivity graph; off is scalar, in dB; $\mathbf{N}_V \mathbf{J}_{SP} + \mathbf{J}_{SP} \mathbf{N}_V$ is the *vertex noise*, a systematic offset affecting only edges connected to a particular vertex; \mathbf{N}_V is diagonal, in dB; $\mathbf{N}_V = \text{diag}(n_{V1}, \dots, n_{Vn})$, \mathbf{N}_E is the *edge noise*, any deviation between graph weights not better captured by global offset or vertex noise.

All matrices are square matrices $n \times n$, with n being the number of vertices, that is, the number of nodes to be matched. Given the high dynamic range of physical quantities in wireless technology, the logarithmic notation seems adequate, also given typical measurement accuracy. \mathbf{P} , off , \mathbf{N}_V , and \mathbf{N}_E are unknowns with respect to address assignment. This formalization is not directly usable for finding \mathbf{P} ; it serves however to explain the different steps of the algorithm.

The global offset off is a convenience device to keep vertex and edge noise zero mean. While in theory it is possible to reach the same goal by eliminating all systematic sources of error from measurement and simulation, it also means a lot of effort. For practical reasons, the channel simulation does not cover all physical phenomena but only the most important. Conveniently, the global offset is relatively simple to estimate.

Vertex noise captures all deviations between measurement and simulation affecting a specific node. The physical phenomena underlying those deviations are related to the node itself and its immediate vicinity, most importantly the antenna and its misalignment or mistuning due to inappropriate placement or defects. The term $\mathbf{N}_V \mathbf{J}_{SP} + \mathbf{J}_{SP} \mathbf{N}_V$ with $\mathbf{N}_V = \text{diag}(n_{V1}, \dots, n_{Vn})$ adds the scalars n_{V1}, \dots, n_{Vn} to the respective columns and rows of the adjacency matrix, such that edges connected to the particular vertex receive the noise figure associated with the vertex. Section 3.3 discloses a method of cancelling the effect of vertex noise.

Vertex noise and edge noise as defined previously are expected to be Gaussian. This is due to the central limit theorem in the logarithmic domain. The delogarithmized equivalents of the adjacency matrices $\exp(\mathbf{G}_M)$ and $\exp(\mathbf{G}_S)$ are the result of a multiplicative product of uncertain quantities (material coefficients, physical dimensions, and reflections). $\exp(\mathbf{G}_M)$ and $\exp(\mathbf{G}_S)$ are expected to adhere to the log-normal distribution and hence \mathbf{G}_M and \mathbf{G}_S , as well as \mathbf{N}_V and \mathbf{N}_E , Gaussian.

Inferring distance directly from the values in \mathbf{G}_M and \mathbf{G}_S is essentially futile, even in the LOS case. Ignoring multipath antenna gains and insertion losses, the RSSI reported as the result of a 0 dBm transmission at distance d in the 2.4 GHz band is given by $\text{RSSI} = -20 \log(d) - 40.23$. While it is possible to distinguish between transmissions 1 m and 2 m away ($\Delta \text{RSSI} = 6$ dB), it is not possible to distinguish between 10 m and 11 m ($\Delta \text{RSSI} = 0.8$ dB), which is below the reporting accuracy of typical transceivers. Taking into account effects such as NLOS or wave guiding along corridors, we arrive at Figure 2. It is the test data of King et al., as extensively used in Section 4 [1].

Sections 3.1 and 3.2 treat, respectively, channel simulation and channel measurement, Section 3.3 the noise model

and noise cancellation, and Section 3.4 the weighted graph-matching algorithm.

3.1. Model of the Indoor Channel. This section discusses the first step (channel estimation) of the three-step process outlined in Section 1.1.

Since this work assumes availability of building and deployment plan, there is nothing to impede application of the most sophisticated channel models available, like ray-optical ray tracing. For the prototype presented in this work, AWE Software's WinProp channel modelling software is used. Unless otherwise noted, we primarily consider the 2.4–2.48 GHz ISM band.

The channel is assumed to be static, that is, static enough that time variations need not be considered and cannot be taken advantage of. At commissioning time of the building automation network, the building is assumed uninhabited and network components are mounted in their respective fixtures.

The two main challenges for static indoor channels are line of sight (LOS) versus non line of sight (NLOS), as well as fading due to multipath propagation. Doppler fading is no concern for static channels. Polarization is dependent on antenna choices and, wherever known, can be adequately represented in the simulation. Badly oriented antennas of single nodes result in vertex noise and can be dealt with; confer Section 3.3. Systematic misorientation of antennas will likely result in a low-quality address assignment. This can in turn be rectified correcting either the physical deployment or the simulation.

In the NLOS channel, good estimates for signal attenuation and reflection properties of the different materials are crucial. Signal attenuation varies between 1 dB for some types of dry walls and up to 60 dB for the thick walls of historical buildings. Metal plating acts as near-perfect shielding up to the sensitivity limit of measurement equipment, although in practice gaps in the plating allow some of the signal through. Attenuation due to reflection on surfaces is in the order of 15 dB for walls and furniture and in the order of 0.3 dB for metal-plated surfaces. Because the matching algorithm can tolerate a certain amount of noise, empirical estimates based on approximate wall thickness and material class are sufficient. It is, however, essential to model large metal surfaces, such as radiators, metal suspended ceilings, metal doors, and metallized surface coating on windows and glass doors. Easily overlooked metal surfaces are air and cable conduits above suspended ceilings. In most cases, modelling of those surfaces is not necessary because their overall contribution to the field configuration is limited enough—as long as they do not obstruct crucial propagation paths. (The point becomes apparent if the radar equation is considered. A reflection on relatively small metal surfaces does not change the overall distribution of RF energy within the indoor channel.) The Winprop software comes with a material database that adequately covers common cases of building materials.

Fading due to multipath propagation needs to be considered in any meaningful discussion of indoor channels. It is a highly localized (centimetres) phenomenon, which is geometry and frequency dependent, and causes local variations

of the signal strength in the order of 30 dB [18]. Fading is commonly modelled statistically, in particular by the Rayleigh distribution in the NLOS case and the Rician distribution in the LOS case. This is not sensible for the problem at hand because there is too little variability in the static channel with respect to time and space due to the fixed mounting locations. The statistical-inference approach is incompatible with deterministic channel simulation.

Still, the effects of fading are severe enough that they need to be accounted for. This can either be done in the channel simulation or in the channel measurements. The Winprop software can do coherent simulations although in the 2.4 GHz band, where the free-space wavelength (λ_0) is merely 12 cm, the required precision with regard to entering dimensions into the simulation and the required precision with regard to placement of node antennas make the approach impractical.

The alternative approach is to measure the channel while avoiding fading effects. This is most easily done by measuring the channel at different frequencies. Wireless nodes compliant with IEEE 802.15.4 support 16 channels in the 2.4 GHz band, 5 MHz apart. If the coherence bandwidth (B_c) of the static indoor channel is smaller than the frequency range covered by IEEE 802.15.4, then the maximum observed RSSI over all channels is a good estimate for a nonfading channel. Janssen observes this in 85% of NLOS cases and 55% of LOS cases [19]. Even if this strategy cannot fully compensate for fading in all cases, the maximum of several measurements is still a better estimate for the fading-free channel than any particular single measurement.

The described approach to fading compensation is not possible in sub-GHz radio because the available frequency bands are too narrow to observe uncorrelated fading behaviour. Whether the larger wavelength makes coherent channel simulation practical is the subject of future testing.

Ray tracing and ray casting are Monte-Carlo methods for predicting wave propagation in cluttered environments. Taken together, the rays represent an unbiased sample approximating the field configuration. Clever sampling strategies, high quality geometry data, and realistic material parameters lead to high-quality channel models. Landstorfer reports standard deviations between measurements and simulations between 4.2 dB and 11.9 dB [20]. Valenzuela reports standard deviations better than 3.2 dB for LOS and better than 8.8 dB for NLOS [21]. In both cases, no specifics concerning the parametrization are given.

The most important phenomena to be modelled by raytracing are as follows.

- (i) Reflection—especially specular reflection. Diffuse reflections (scattering) are of secondary concern in the microwave bands.
- (ii) Transmission—optionally considering refraction. Both reflection and transmission may be modelled based on Fresnel equations or based on empirical (measured) data.
- (iii) Diffraction—different approximations exist, with the ones receiving the most attention being the geometrical and the unified theory of diffraction [22].

Assembling \mathbf{G}_S from the simulation output is straightforward.

3.2. Assembling the Measured Connectivity Graph. This section discusses the second step (channel measurement) of the three-step process outlined in Section 1.1.

While the simulated connectivity graph can be created ahead of time, assembling the measured connectivity graph is an on-site activity. The wireless network is assumed noncommissioned, but the nodes are powered and an initial, maybe temporary, network has been formed by a procedure such as ZigBee network forming. Nodes are reachable via their low-level network or MAC address. The wireless channel can be measured as a whole, or the problem can be subdivided rudimentarily, for example, by only powering nodes on a particular floor or section of the building.

A procedure and protocol has been devised to gather the measured connectivity graph. It is formulated in terms of ZigBee, but it is not directly dependent on particular ZigBee features. Support for many-to-one routing, however, is desirable for fast and efficient data collection. The procedure works as follows.

- (1) The network is put into graph-assembly mode, temporarily suspending routing and other network-layer mechanism.
- (2) Participating nodes visit a set of channels.
- (3) In each channel participating nodes listen for incoming graph-assembly beacons. Receivers are in promiscuous mode; that is, MAC filtering is suspended, as well as power cycling. Received beacons are stored in a graph-assembly table, also recording the beacon's RSSI.
- (4) Every node also sends a small number of graph-assembly beacons at random points of time in each channel, containing mainly its low-level address and device type. Stay time in each channel is chosen such that beacon collisions are sufficiently improbable.
- (5) After the last visited channel, normal network operation is restored and the graph-assembly tables are downloaded to the data sink.

Based on the downloaded graph tables (cf. Table 1), assembly of the measured connectivity graph \mathbf{G}_M is straightforward.

- (i) Each vertex is represented by one table; vertices are attributed with low-level address and node type descriptor.
- (ii) Each edge is represented by a pair of rows in the tables of the respective vertices. Edge weights are given by

$$\begin{aligned} \mathcal{G}_{M,AB} &= \mathcal{G}_{M,BA} \\ &= \max(\max(\text{RSSI}_{A \rightarrow B}) \\ &\quad - \text{TXpwr}_A, \max(\text{RSSI}_{B \rightarrow A}) - \text{TXpwr}_B). \end{aligned} \quad (2)$$

TABLE I: Graph-assembly table.

| Low-level address | Node type descriptor | TX-power | RSSI channel 1 | RSSI channel n |
|-------------------|----------------------|----------|----------------|------------------|
|-------------------|----------------------|----------|----------------|------------------|

RSSI (in dB m) is the received signal strength indicator as reported by the transceiver when receiving the beacons. TXpwr (in dB m) is the nominal transmit power of the node sending the beacon. Edge weights $g_{M,AB}$ (in dB) are estimates for the interference-corrected channel attenuation between nodes A and B . Antenna, matching, and insertion losses at transmitter and receiver need not be accounted for but are dealt with during preprocessing. A small number of missing graph-assembly tables can be tolerated because all relevant information is redundantly available in the assembly tables of neighbouring nodes.

An alternative edge weight is given by $g'_{M,AB} = \text{avg}(\dots)$. A large deviation between $g_{M,AB}$ and $g'_{M,AB}$ indicates the presence of an asymmetrical link. This can happen for a variety of causes, but if the majority of links of a particular node are asymmetric, it is a strong indication for a defect in the node's transmission or reception path. It does not matter for the purpose of address assignment because the max function masks the problem. Problems concerning the antenna are usually symmetrical and cannot be detected by this mechanism.

3.3. Preprocessing the Graphs/Noise Cancellation. This section discusses the third step (matching algorithm) of the three-step process outlined in Section 1.1. Specifically, the mechanics of preprocessing measured and simulated connectivity graphs are considered. Preprocessing involves the following steps:

- (i) optionally thinning out edges with very small weights (e.g., -95 dB and below),
- (ii) estimation of the global offset,
- (iii) normalization, dynamic range compression,
- (iv) cancellation of vertex noise.

There is a strong correlation between distance and path loss (cf. Figure 2). While in the indoor NLOS environment it is not directly useful without the ray tracer, the fact remains and it leaves its imprint on the data. One important consequence of this correlation is the fact that in the typical graph-assembly table, the majority of neighbours have a low or very low RSSI. This is because the volume of a sphere increases with the cube of the radius; that is, in a building with evenly distributed nodes, more neighbours are near the limit of radio coverage than there are neighbours which are reliable communication partners. This also means that a disproportionately high number of edge weights in the measured connectivity graph are near the noise floor or the limit of receiver sensitivity—the weight distribution is heavy at the low end. Figure 4 illustrates the point by means of the test data set.

In the simulation, receivers are not limited with respect to sensitivity or noise. The low end of the weight distribution in measured and simulated connectivity graphs therefore

tends to look markedly dissimilar. There are more issues complicating things.

- (i) Graph-assembly tables may be limited in size, which results in edges missing in the measured but not the simulated connectivity graph, also predominately at the low end.
- (ii) Edge noise also tends to be larger at the low end because inaccuracies with respect to material parameters and geometry tend to accumulate in transmission paths that are subject to several transmissions and reflections.

There is a tradeoff with respect to choosing which edges to keep and which to discard. Discarding many edges at the low end reduces noise and thereby increases the accuracy of the match and simultaneously lowers computation time of graph matching. The problem lies with choosing corresponding edges to discard in both graphs, especially considering that the graphs are permuted and the data is noisy. Eagerly discarding many edges means potentially discarding noncorresponding edges in measured and simulated connectivity graphs, and thereby amplifying a small amount of noise to a much bigger problem by having many unmatched edges.

The approach taken by the reference implementation is therefore to keep all edges. An alternative, but ultimately abandoned, approach is to try to minimize the number of unmatched edges by forcing the number of edges in both graphs to be equal. This can be achieved by discarding edges from the bottom of the simulated weight distribution until the number becomes equal. The strategy fails whenever there are outliers with respect to vertex noise. Vertices with high positive vertex noise end up almost without any edges, which reduces the quality of the match.

Choosing the global offset is nonobvious. Aligning the weight distributions by its means is suboptimal for later graph matching because the distributions are heavy at their low end which is also noisy. The median is better. Still, better results are obtained by aligning at the 9th 10-quantile, that is, choosing *off* such that the weight distributions of \mathbf{G}_M and $(\mathbf{G}_S + \text{off } \mathbf{J}_{SP})$ share a point where 90% of the weights of each distribution lie below and 10% above. This strategy relies mostly on the magnitude of LOS weights while at the same time being reasonably immune to outliers.

Dynamic range compression serves to emphasise the relative importance of edges weights with low noise at the top of the distribution over graph edges with high relative noise. It is combined with normalization of the graph edges between 0 and 1:

$$\mathbf{G}_M^{(C)} = (c_1 \mathbf{G}_M + c_2 \mathbf{J}_M)^\alpha,$$

$$\mathbf{G}_S^{(C)} = (c_1 (\mathbf{G}_S + \text{off } \mathbf{J}_S) + c_2 \mathbf{J}_S)^\alpha,$$

$$\begin{aligned}
c_1 &= (\max(\max(\mathbf{G}_M), \max(\mathbf{G}_S + \text{off } \mathbf{J}_S)) \\
&\quad - \min(\min(\mathbf{G}_M), \min(\mathbf{G}_S + \text{off } \mathbf{J}_S)))^{-1}, \\
c_2 &= -c_1 \min(\min(\mathbf{G}_M), \min(\mathbf{G}_S + \text{off } \mathbf{J}_S)).
\end{aligned} \tag{3}$$

α is the parameter that governs the amount of compression applied. The reference implementation chooses $\alpha = 2.5$ which is empirically determined. $\mathbf{G}_M^{(C)}$ and $\mathbf{G}_S^{(C)}$ are the normalized and compressed adjacency matrices. \mathbf{J}_M and \mathbf{J}_S are logical matrices indicating the nonzero elements of \mathbf{G}_M and \mathbf{G}_S . c_1 and c_2 are derived from minimal and maximal edge weights in order to normalize the graphs.

Vertex Noise Elimination: the underlying idea is to mathematically transform the wireless network's connectivity graph into an analogue representation where vertex noise is baseless. In (1) each wireless node transmits a fixed amount of RF energy (normalized to 0 dB) into the channel of which a small amount is received at its neighbours, represented by the edge weights. Vertex noise is a systematic deviation between simulation and measurement where the energy balance between a node and its neighbours is biased. In the analogue representation, the energy balance is not normalized to the transmitted energy but to the received energy. Find a multiplier for each vertex such that the sum of the received energy is 0 dB (in fact 1 because of normalization) in the transformed adjacency matrices $\mathbf{G}_M^{(S)}$ and $\mathbf{G}_S^{(S)}$:

$$\begin{aligned}
\mathbf{G}_M^{(S)} &= \mathbf{D}_M \mathbf{G}_M^{(C)} \mathbf{D}_M, \\
\mathbf{G}_S^{(S)} &= \mathbf{D}_S \mathbf{G}_S^{(C)} \mathbf{D}_S.
\end{aligned} \tag{4}$$

\mathbf{D}_M and \mathbf{D}_S are diagonal matrices with the vertex multipliers. $\mathbf{G}_M^{(S)}$ and $\mathbf{G}_S^{(S)}$ are doubly stochastic, symmetric matrices (columns and rows sum to 1). The elements are fractions between 0 and 1 and express how much a particular edge contributes to the hypothetical 0 dB received energy figure.

Finding the multipliers and stochastic matrices is achieved by applying the Sinkhorn-Knopp (SK) algorithm [23] on the normalized matrices $\mathbf{G}_M^{(C)}$ and $\mathbf{G}_S^{(C)}$. This works as advertised as long as they are fully indecomposable. This seems to be the case as long as the underlying graphs are connected; that is, the previous thinning out of edges has not led to any unconnected subgraphs.

The transformed graphs $\mathbf{G}_M^{(S)}$ and $\mathbf{G}_S^{(S)}$ allow us to reformulate the original problem (1) as

$$\mathbf{G}_M^{(S)} = \mathbf{P}^T \mathbf{G}_S^{(S)} \mathbf{P} + \mathbf{N}^{(S)}. \tag{5}$$

Since $\mathbf{G}_M^{(S)}$ and $\mathbf{G}_S^{(S)}$ are doubly stochastic, $\mathbf{N}^{(S)}$ is zero mean and therefore per definition free of vertex noise. Moreover, the original vertex noise does not otherwise significantly

magnify $\mathbf{N}^{(S)}$. Temporarily setting $\alpha = 1$, we can express $\mathbf{N}^{(S)}$ as

$$\begin{aligned}
\mathbf{N}^{(S)} &= \mathbf{D}_M (c_1 \mathbf{P}^T \mathbf{G}_S \mathbf{P} + c_1 \text{off } \mathbf{J}_{SP} + c_2 \mathbf{J}_M) \mathbf{D}_M \\
&\quad - \mathbf{D}_{SP} (c_1 \mathbf{G}_M + c_2 \mathbf{J}_{SP}) \mathbf{D}_{SP} \\
&\quad + c_1 \mathbf{D}_M (\mathbf{N}_V \mathbf{J}_{SP} + \mathbf{J}_{SP} \mathbf{N}_V + \mathbf{N}_E) \mathbf{D}_M \\
&\quad + c_1 \mathbf{D}_{SP} (\mathbf{N}_V \mathbf{J}_{SP} + \mathbf{J}_{SP} \mathbf{N}_V + \mathbf{N}_E) \mathbf{D}_{SP}.
\end{aligned} \tag{6}$$

The first part of (6) contains two terms that apply the vertex multipliers \mathbf{D}_M and $\mathbf{D}_{SP} = \mathbf{P} \mathbf{D}_S \mathbf{P}^T$ to the respective other adjacency matrix, but as defined it does so by observing the right permutational order. In the second part, the original noise is downscaled using the vertex multipliers. This equation can be further analysed in two important special cases.

Case 1 ($\alpha = 1$, $\mathbf{P} = \mathbf{I}$, $\mathbf{N}_V = \mathbf{0}$ (edge noise only)). The inputs into the SK algorithm (4) differ only by \mathbf{N}_E which is zero mean (1). By the rules of the SK algorithm, this leads to identical vertex multipliers; that is, $\mathbf{D}_S = \mathbf{D}_M$. The first part of (6) can therefore be rewritten to $\mathbf{P}^T \mathbf{G}_S^{(S)} \mathbf{P} - \mathbf{G}_M^{(S)}$, which is $-\mathbf{N}^{(S)}$. The whole equation simplifies to $\mathbf{N}^{(S)} = c_1 \mathbf{D} \mathbf{N}_E \mathbf{D}$. In the absence of vertex noise, the preprocessed noise $\mathbf{N}^{(S)}$ is no more than correctly scaled edge noise.

Case 2 ($\alpha = 1$, $\text{off} = 0$, $\mathbf{P} = \mathbf{I}$, $\mathbf{N}_E = \mathbf{0}$ (vertex noise only)). This case is studied by first introducing modified vertex noise \mathbf{N}'_V and modified edge noise \mathbf{N}'_E : $\mathbf{G}_M = \mathbf{N}'_V (\mathbf{P}^T \mathbf{G}_S \mathbf{P} + \text{off } \mathbf{J}_{SP}) \mathbf{N}'_V + \mathbf{N}'_E$. The modified definition is largely equivalent to (1), with the difference being that vertex specific deviations between \mathbf{G}_M and \mathbf{G}_S are represented by a multiplicative factor \mathbf{N}'_V , not a summand. From $\mathbf{N}'_E = 0$, as well as the other preconditions, follow $\mathbf{G}_S^{(S)} = \mathbf{G}_M^{(S)}$ because the multiplicative \mathbf{N}'_V is drawn into the vertex multiplier \mathbf{D}_S by the SK algorithm. From there follows $\mathbf{N}^{(S)} = 0$. Under the preconditions of Case 2, preprocessing perfectly eliminates modified vertex noise. Given the more useful original definition of vertex noise instead, but also given $\mathbf{N}_V < \mathbf{G}_M, \mathbf{G}_S$, it can be shown that there remains some residual $\mathbf{N}^{(S)}$ which is however zero mean and therefore no longer behaves like vertex noise.

In general, \mathbf{P} is of course unequal \mathbf{I} . Finding \mathbf{P} is after all the point of address assignment. But the behaviour demonstrated by the special cases transfer well to the general case. Vertex noise is to a certain degree suppressed and to a certain degree mutated into edge noise and the latter transforms proportionally with the adjacency matrices.

The downside of vertex noise elimination is that the edge weight distribution of $\mathbf{G}_M^{(S)}$ and $\mathbf{G}_S^{(S)}$ compares unfavourably with the situation before vertex noise elimination—almost all edges very close to zero. Still, it is worth the effort in case of the demonstration data set of Section 4. Whether this is true in all cases remains to be the subject of future research.

3.4. Weighted Graph Matching. This section discusses the third step (matching algorithm) of the three-step process outlined in Section 1.1. The weighted graph matching problem (WGM) is well known in the field of combinatorial

optimization. It is equivalent to the quadratic assignment problem, which is NP-complete. There exist no efficient algorithms for finding optimal solutions. Several heuristic approaches have been surveyed for suitability in address assignment [24–29]. We chose to implement the algorithm of Gold and Rangarajan [28] because it compares favourably along several dimensions important to address assignment:

- (i) performance—in the sense of low-order computational complexity; Zaslavskiy lists it amongst the best performing [29],
- (ii) converges well on noisy data,
- (iii) deals with missing and spurious vertices and edges,
- (iv) takes advantage of optional vertex attributes,
- (v) supports sparse problems,
- (vi) relatively simple and parallelizable algorithm, which makes it suitable for implementation with GPGPU (general purpose computation on graphics processing unit).

The graduated assignment approach of Gold and Rangarajan is related to deterministic annealing. It relaxes or convexifies the problem by first allowing values other than “0” and “1” in the vertex assignments. Instead of a permutation matrix, the algorithm computes a doubly stochastic matrix, indicating the similarity of vertices given no particular assignment. As the control parameter increases, the problem becomes increasingly nonconvex; the doubly stochastic matrix is pushed more and more towards a permutation matrix. The algorithm thereby avoids local minima because it started off in the global minimum of the convex problem and tracks the evolution of this global minimum from the maximally convex version of the problem to the original nonconvex version.

The algorithm is implemented using a combination of Matlab and CUDA. It performs well (minutes) for graphs up to 2000 vertices and 80000 edges, at which point the necessary data structures become too large to fit into GPU (Nvidia GeForce GTX 560 Ti, 384 CUDA cores, 2 GB RAM) memory.

Anchors are nodes for which the address assignment is known a priori. Typically, this is the network’s coordinator, nodes specially joined to the network for the sole purpose of commissioning, or any other node that the commissioning technician visits manually during the procedure. In terms of the matching algorithm, anchors are not unlike additional vertex attributes (like device type) in that they amend the objective function and thereby bias or restrict the state space. Indeed, anchors could be implemented within the formalism of Gold and Rangarajan in terms of vertex attributes, whereas each anchor corresponds to one extra global binary vertex attribute which is zero for all vertices except the anchor.

On the other hand, anchors are a column and row of the permutation matrix \mathbf{P} that is known a priori. A simpler approach is therefore to slightly change the *softassign* step of the algorithm and replace it by a definitive assignment for just the affected columns and rows. This can be useful for noisy problems, where a small number of anchors can direct the algorithm towards convergence.

Weighted graph matching is a least square-error optimization. This leads to ambiguity in the interpretation of matching errors, that is, misassigned nodes. An error can mean one of two things.

- (i) The weighted graph-matching algorithm converged to a suboptimal solution.
- (ii) The weighted graph matching algorithm converged to the optimal solution and noise was biased in a way that made the wrong solution optimal.

For all but for toy-sized problems (less than approximately 11 vertices), it is essentially impossible to compare the result with the real optimal solution, so in the majority of cases the question cannot be answered.

The number of vertices in measured and simulated graphs may be unequal. Failure to transmit the graph-assembly table should not normally be the cause of missing vertices because the existence of the underlying node is evident in the graph-assembly tables of neighbouring nodes. Not working, not installed, or spuriously installed nodes, however, result in graphs of unequal sizes. The weighted graph-matching algorithm deals with that by introducing a suitable number of slacks. They should be introduced only in this stage because zero-sum rows and columns cause problems during vertex-noise elimination.

4. Experimental Evaluation

In order to demonstrate the viability of the approach, we leverage the data set originally collected by King et al. [1]. It consists of 12 IEEE 802.11 “WiFi” access points and 130 fingerprinting positions of RSSI measurements. Out of the 12 access points, 2 are excluded because one has no data associated and the other is redundant. Of the fingerprinting positions, 24 in regular 4.5 m grid have been selected for the demonstration, leading to a total size 34 vertices in the data set. Fading effects are countered using a slightly different strategy than the one described in Section 3.1. Instead of measuring at different frequencies, King et al. repeated each measurement several times with different orientations and slight displacements of the reception antenna.

4.1. Modelling of the Demonstration Site. The site is modelled with WinProp (cf. Figure 3), using the floor plan in the publication as basis and filling in the details about the materials and mounting heights after a personal visit at the site. The walls throughout the surveyed corridors and offices are metal-plated, so are the doors. This makes the site somewhat unusual if not especially challenging. Wave propagation between rooms therefore happens primarily above the suspended ceiling. The high reflectivity of the environment requires a higher than usual number of samples (rays) to adequately approximate the RF-energy distribution.

Not included in the model are furniture, conduits above the suspended ceiling, columns of the building’s reinforced frame, details of the façade, and the characteristics of the reception antenna. Also unknown is the state of the metal doors (fully open/half open/closed) at the time of the measurements. In the simulation all doors are assumed closed.

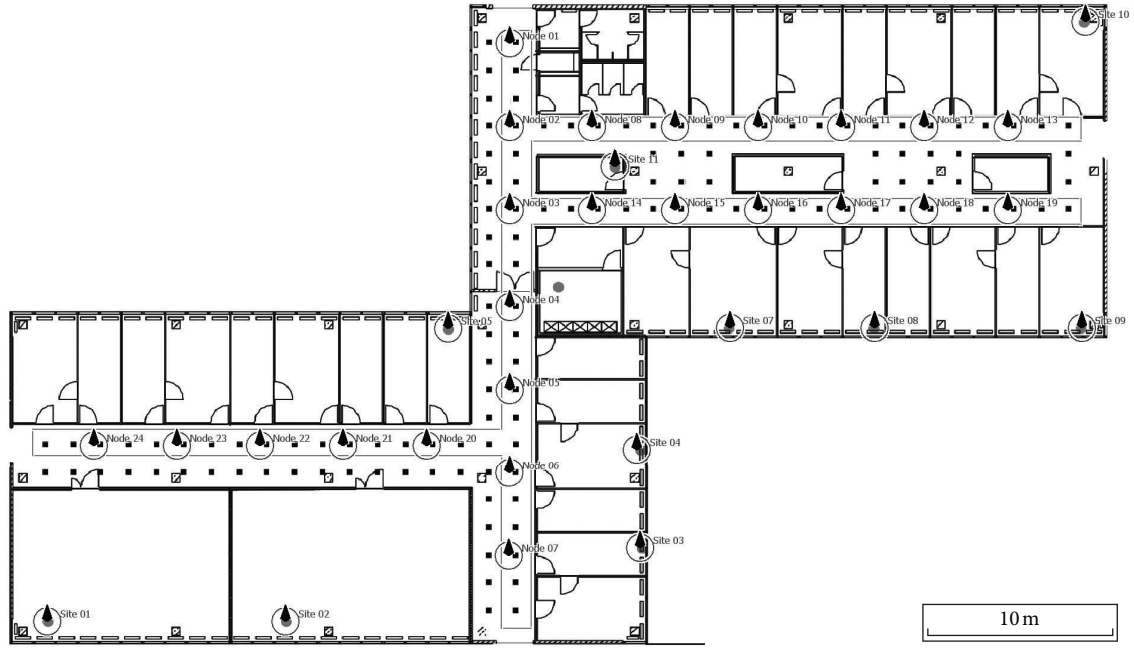


FIGURE 3: Floor plan of the Mannheim site as viewed in the WinProp simulation tool. “Site x” are WiFi access points and “Node x” are the selected points of measurement and together form the simulated connectivity graph.

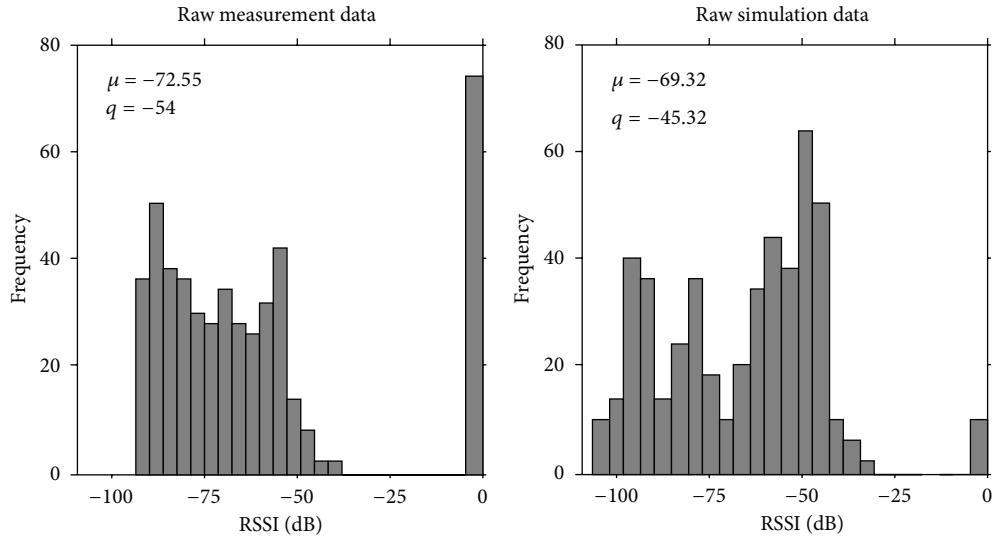


FIGURE 4: Raw measurement data versus simulation data. Measurements are cut off at -95 dBm due to interference and receiver sensitivity. “ q ” is the 9th 10-quantile used for estimating the global offset. Zeros mean nonexistent edges; they are not counted towards μ and q .

This may be one of the major contributors to noise in the test data set. Table 2 details the parametrization of the Winprop ray-tracing simulation.

Figure 4 is a comparison between the measurement of King et al. and the simulation. The large deviation of the 9th 10-quantile (global offset off) can be attributed to unknowns with regard to the exact details of the RSSI measurement (antenna, cables, and WiFi equipment). It is not relevant for the purpose of address assignment and is compensated in the preprocessing stage. The deviation between simulation

and measurement—here treated as vertex noise and edge noise—is depicted in Figure 5. Relative edge noise ($\mathbf{N}_{ER} = \mathbf{N}_E \oslash (0.5\mathbf{G}_M + 0.5\mathbf{P}^T \mathbf{G}_S \mathbf{P})$) fits a Gaussian model well with $\sigma = 0.08$ dB/dB (\oslash means elementwise division on nonzero elements). As explained in Section 3, this is to be expected because noise is the consequence of a multiplicative product of uncertain quantities. Furthermore, edge noise is not constant over the range of edge weights, with edges at the low end being more noisy than the rest. Extraction of vertex noise from the test data set is done by directly solving (1) and

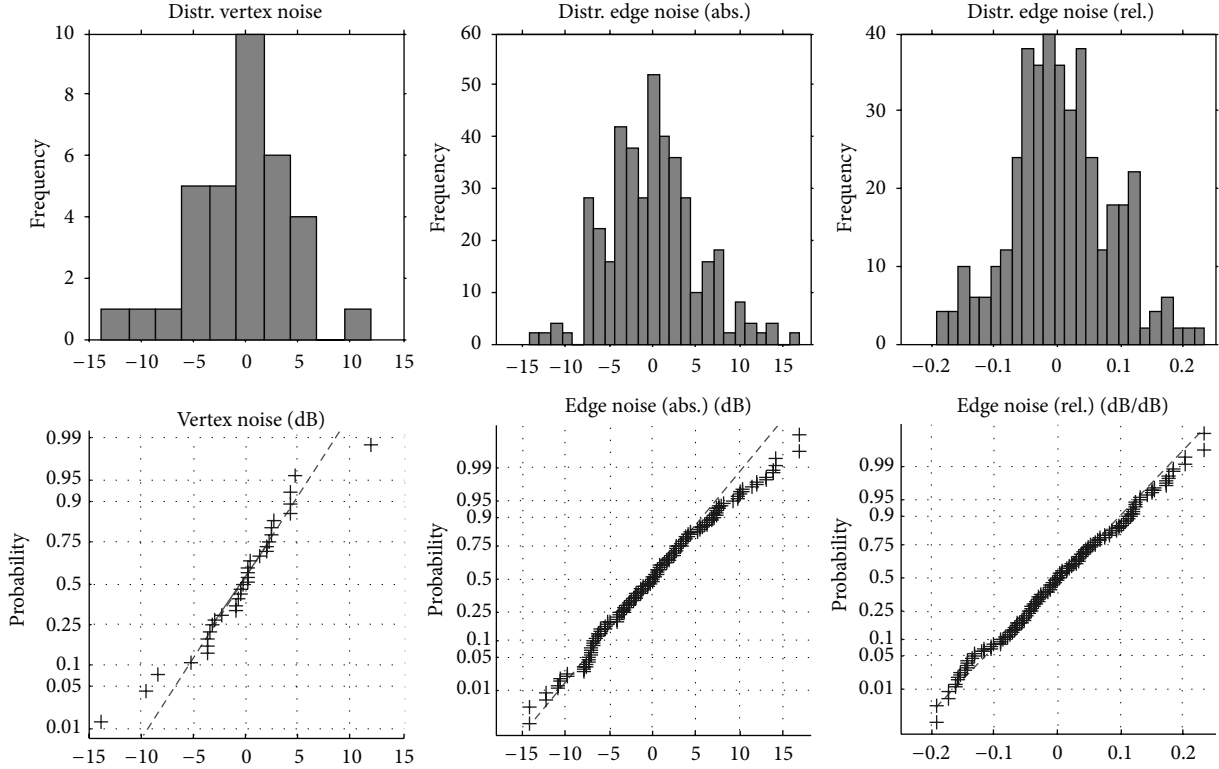


FIGURE 5: Measured versus simulated data, noise distribution, and no preprocessing except elimination of global offset. Outliers in vertex noise are due to modelling inaccuracies. As expected, edge noise fits the Gaussian model well, especially when expressed relatively to edge weight.

TABLE 2: Most important configuration parameters of the ray-tracing simulation with WinProp.

| | |
|---|---------|
| Max transmissions/reflections/diffractions/scattering of each ray | 4/4/2/1 |
| Total max interactions (reflections + diffractions + scattering) | 4 |
| Uncorrelated simulation (not coherent, no phase) | |
| Empirical loss model (not based on Fresnel equation) | |

optimizing for minimal edge noise. (This is possible with the test data set because only in this case the permutation \mathbf{P} is known a priori.)

4.2. Completing the Test Data Set. The data set of King et al. was recorded for the purpose of fingerprinting the channel rather than address assignment. Instead of wireless node-to-node measurement of RSSI, it contains measurements from access point to fingerprinting position. What is missing are RSSI measurements from access point to access point and from fingerprinting position to fingerprinting position.

The missing edges of the measured connectivity graph are estimated by taking values from the simulation and overlaying them with the noise profile extracted from comparing the available measurements with the corresponding data from the simulation; confer Figure 5. Vertex noise is applied unaltered, and edge noise is sampled from a random

number generator. The measured data set thus consists of 203 real measured edges and 276 estimates with identical noise properties. We consider the resulting synthetic measurement graph of 34 vertices and 479 edges (measurements and estimates) to be a fair and realistic approximation of what could be expected by true node-to-node measurements of RSSI.

The experiment is set up as follows.

- (1) Analyse noise, fill in missing edges in \mathbf{G}_M , and optionally enhance noise artificially.
- (2) Apply random permutation \mathbf{P} in order to get simulated connectivity graph \mathbf{G}_S .
- (3) Preprocess; use weighted graph matching to recover address assignment \mathbf{P}' .
- (4) Compare \mathbf{P} with \mathbf{P}' and count matching errors.

No anchors or additional vertex attributes are added to the data set.

4.3. Results. There are two sources of randomness in the test data set and graph-matching procedure: the random Gaussian noise used to complement the measured connectivity graph and the random permutation. The results of ten test runs are therefore averaged.

The presented method of address assignment is able to perfectly match (34 of 34 vertices) the test data set. Figure 6 details the matching performance if the noise is artificially

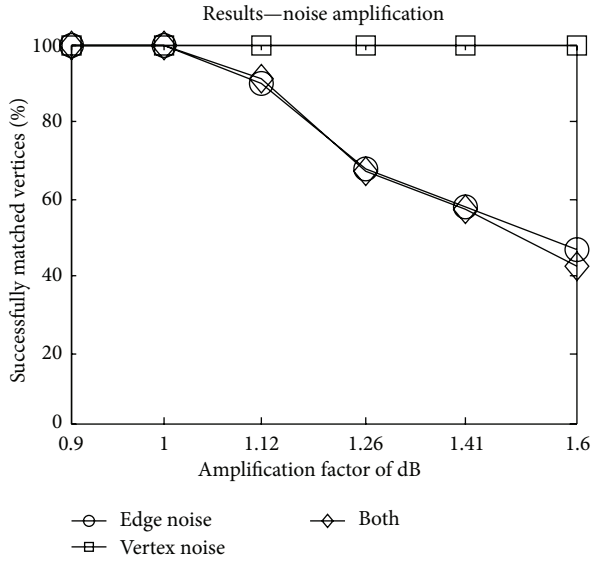


FIGURE 6: Artificial amplification of noise in the test data set. The factor in the abscissa scales a logarithmic quantity. Vertex noise is well suppressed and edge noise should not be any bigger than $\sigma = 5.15$ dB. 5% more edge noise and sporadic matching errors manifest. 10% more edge noise and performance degrades to 90%.

inflated—vertex noise, edge noise, and both. Performance degradation first starts with a single vertex, “Site 01,” in the lower left of Figure 3; it is also the positive outlier visible in the vertex-noise histogram in Figure 5. This vertex is responsible for the misordering of neighbouring vertices “Node 23” and “Node 24.”

Figure 6 suggests that the algorithm is stressed to its limit by the test data set. On the other hand, the noise being scaled is a logarithmic quantity. In the tails of the edge noise distribution (Figure 5, tails at 15 dB, $\sigma = 5.15$ dB), the factor of dB of 1.26 means linearly 2.45 times more deviation between measurement and simulation.

A second experiment is conducted with fully random graphs of 500 nodes. The graph G_M is constructed by choosing 500 uniformly distributed random points in a $50 \times 50 \times 100$ m rectangular area and calculating RSSI using the inverse square law with an empirical “inverse square” of -3.5 and dipole antennas. Then G_S is derived by applying vertex and edge noises. Results are depicted in Figure 7. The noise envelop is not directly comparable with Figure 6, as in the former case the variance is constant over the dynamic range and in the latter case it scales linearly with 0.08 dB/dB (cf. Figure 5). The apparent inferior performance at 5 dB edge noise (100% versus 85%) is attributable to this fact and not to the increased graph size.

5. Conclusion

Section 1.1 introduces a novel method for automatic address assignment, based on a three-step process of channel estimation, channel simulation, and graph matching. The individual steps are discussed in Section 3.

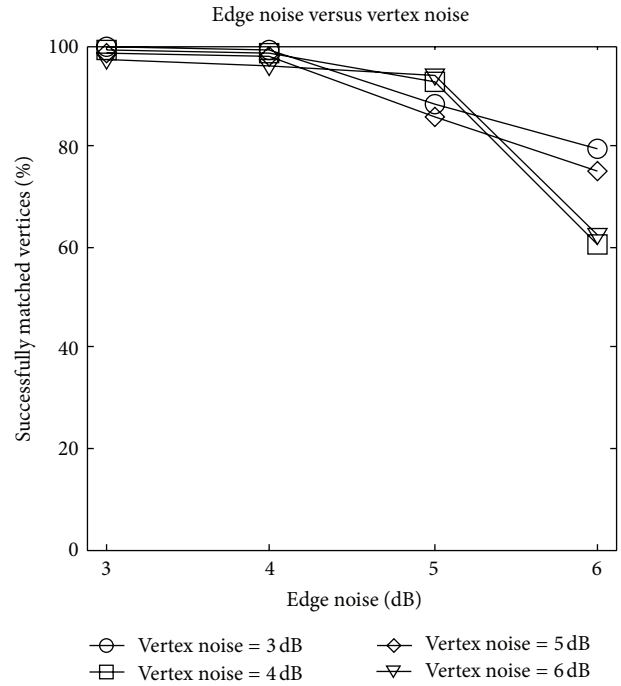


FIGURE 7: Edge noise versus vertex noise on random graphs of 500 vertices. Contrary to Figure 6, the (Gaussian) edge noise is uniform, not scaled, over the dynamic range; the result therefore appears slightly inferior.

In order to assess the potential general applicability of the method, it is necessary to put the experimental results of Section 4 into perspective. The test data is believed to be more challenging than the average application because of the following aspects.

- (i) The site is difficult to handle because of metal-plated walls.
- (ii) The situation during which the measurement was performed by King et al. was more challenging and less controlled than what can be expected during typical application of the address assignment procedure (furniture, unknown state of metal-plated doors).
- (iii) Additional vertex attributes (especially device type) are not present in the data set but would be in typical applications.

Yet, the proposed address assignment procedure is able to solve the problem perfectly and reliably. This is achieved without low-level optimization of the material properties used in the ray tracer. The existence of outliers (vertex noise) in the data set and the fact that the proposed method is able to cope with it suggest that it will also cope with real-world challenges such as misplaced and inappropriately mounted nodes.

The results show that the proposed method of address assignment is able to significantly lower the cost of commissioning large-scale indoor wireless automation networks. This is achieved by changing a task that previously requires manual intervention on a per-node basis to one that is performed once, for the whole network.

Easy address assignment is an enabler for the Internet of Things because it is an enabler for ubiquitous positioning. The address assignment procedure provides as its byproduct a dense network of anchors (wireless nodes with known ID and known location) and a detailed channel map. This is precisely what indoor positioning systems as surveyed in Section 2 need in order to provide reliable and high-quality service. The second enabling effect is bringing down the costs of installing low-power indoor wireless networks.

For industrial application of the proposed method, the different pieces of the solution have to be more closely integrated: building automation planning tool, 3D modelling package, ray tracer, and, last but not least, the matching procedure. This has to be done in a way that integrates with established industry procedures and with respect to roles and responsibilities. The architect or planner has to deliver 3D models of the structure, the planner of the automation network the deployment plan. Both have to be imported and refined in the ray tracer. Using the commissioning tool, the commissioning technician has to perform the channel measurement and do the graph matching. The resulting address assignment needs to be committed to the network, again using the commissioning tool.

Furthermore, the commissioning technician may decide on site to perform the procedure on the whole network or on smaller parts. The technician may have last minute changes or may define a small number of anchors, for example, in parts of the building where the ray-tracing model is lacking for some reason. The proposed procedure supports all these scenarios, but the tools need to be created.

Topics of future research include the study of applicability of the procedure to outdoor environments, for example, in environmental monitoring or street lighting scenarios. Some low-power wireless platforms also come with the ability to perform time-of-flight measurements. While the graph-matching stays the same, preprocessing the time-of-flight data poses different challenges.

Disclosure

The authors declare that there is no conflict of interests or competing financial interest related to the work described.

Acknowledgment

The work presented in this paper has been carried out in the SmartCoDe project, cofunded by the European Commission within the 7th Framework Programme (FP7/2007–2013) under Grant no. 247473.

References

- [1] T. King, T. Haenselmann, and W. Effelsberg, "On-demand fingerprint selection for 802.11-based positioning systems," in *Proceedings of the 9th IEEE International Symposium on Wireless, Mobile and Multimedia Networks (WoWMoM '08)*, Newport Beach, Calif, USA, June 2008.
- [2] S. Knauth, R. Kistler, D. Käslin, and A. Klapproth, "SARBAU - Towards highly self-configuring IP-fieldbus based building automation networks," in *Proceedings of the 22nd International Conference on Advanced Information Networking and Applications (AINA '08)*, pp. 713–717, Okinawa, Japan, March 2008.
- [3] B. D. Westphal, J. R. Hall, D. P. Bohlmann, and G. A. Zuercher, "BACnet protocol MS/TP automatic MAC addressing," USPTO 12/326, 852, 2008.
- [4] G. Kiwimagi, C. McJilton, and M. Gookin, "Configuration application for building automation," USPTO US, 2005/0119767 A1, 2004.
- [5] R. R. Lingemann, "Building automation system," USPTO 10/608, 828, 2003.
- [6] L. Feri, G. M. P. J. Linnartz, J. W. Rietman, W. C. T. Schenk, C. J. Talstra, and H. Yang, "Efficient address assignment in coded lighting systems," EPO EP, 2417834 A1, 2012.
- [7] A. Savvides, M. Srivastava, L. Girod, and D. Estrin, "Localization in sensor networks," in *Wireless Sensor Networks*, Springer, New York, NY, USA.
- [8] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer*, vol. 34, no. 8, pp. 57–66, 2001.
- [9] R. Stoleru, J. A. Stankovic, and S. H. Son, "Robust node localization for wireless sensor networks," in *Proceedings of the 4th Workshop on Embedded Networked Sensors (EmNets '07)*, pp. 48–52, June 2007.
- [10] A. Papapostolou and H. Chaouchi, "WIFE: wireless indoor positioning based on fingerprint evaluation," in *Networking*, vol. 5550 of *Lecture Notes in Computer Science*, 2009, Proceedings of the 8th International IFIP-TC 6 Networking Conference, Aachen, Germany, 2009.
- [11] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '00)*, pp. 775–784, Tel Aviv, Israel, 2000.
- [12] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, "COMPASS: a probabilistic indoor positioning system based on 802.11 and digital compasses," in *Proceedings of the 1st ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH '06)*, pp. 34–40, usa, September 2006.
- [13] 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 3rd Workshop on Real-World Wireless Sensor Networks (REALWSN '08)*, pp. 1–5, April 2008.
- [14] N. Patwari, R. J. O'Dea, and Y. Wang, "Relative location in wireless networks," in *Proceedings of the 53rd Vehicular Technology Conference (VTS '01)*, pp. 1149–1153, May 2001.
- [15] C. Liu, T. Scott, K. Wu, and D. Hoffman, "Range-free sensor localisation with ring overlapping based on comparison of received signal strength indicator," *International Journal of Sensor Networks*, vol. 2, no. 5-6, pp. 399–413, 2007.
- [16] C. Wan, A. Mita, and S. Xue, "Non-line-of-sight beacon identification for sensor localization," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 459590, 6 pages, 2012.
- [17] K. Sinha and A. D. Chowdhury, "A beacon selection algorithm for bounded error location estimation in ad hoc networks," in *Proceedings of the International Conference on Computing: Theory and Applications (ICCTA '07)*, pp. 87–92, Kolkata, India, March 2007.
- [18] R. A. Valenzuela, "Estimating Local Mean Signal Strength of Indoor Multipath Propagation," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 1, pp. 203–212, 1997.

- [19] G. J. M. Janssen, P. A. Stigter, and R. Prasad, "Wideband indoor channel measurements and BER analysis of frequency selective multipath channels at 2.4, 4.75, and 11.5 GHz," *IEEE Transactions on Communications*, vol. 44, no. 9, pp. 1272–1288, 1996.
- [20] F. M. Landstorfer, "Wave Propagation Models for the Planning of Mobile Communication Networks," in *Proceedings of the 29th European Microwave Conference*, pp. 1–6, Munich, Germany.
- [21] R. A. Valenzuela, S. Fortune, and J. Ling, "Indoor propagation prediction accuracy and speed versus number of reflections in image-based 3-D ray-tracing," in *Proceedings of the 48th IEEE Vehicular Technology Conference (VTC '98)*, pp. 539–543, May 1998.
- [22] Y. Rahmat-Samii, "GTD, UTD, UAT and STD: a historical revisit," in *Proceedings of the IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communication*, pp. 1145–1148, Cape Town, South Africa, 2012.
- [23] R. Sinkhorn and P. Knopp, "Concerning nonnegative matrices and doubly stochastic matrices," *Pacific Journal of Mathematics*, vol. 21, no. 2, pp. 343–348, 1967.
- [24] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 3, pp. 265–298, 2004.
- [25] H. A. Almohamad and S. O. Duffuaa, "Linear programming approach for the weighted graph matching problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 5, pp. 522–525, 1993.
- [26] M. M. Zavlanos and G. J. Pappas, "A dynamical systems approach to weighted graph matching," in *Proceedings of the 45th IEEE Conference on Decision and Control (CDC '06)*, pp. 3492–3497, December 2006.
- [27] S. Umeyama, "Eigendecomposition approach to weighted graph matching problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 5, pp. 695–703, 1988.
- [28] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 377–388, 1996.
- [29] M. Zaslavskiy, F. Bach, and J.-P. Vert, "A path following algorithm for the graph matching problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2227–2242, 2009.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

