On the Impact of Node Placement and Profile Point Selection on Indoor Localization

Israat Tanzeena Haque, Ioanis Nikolaidis, and Pawel Gburzynski

Computing Science Dept.,
University of Alberta,
Edmonton AB T6G 2E8, Canada
{israat,yannis,pawel}@cs.ualberta.ca

Abstract. We present an indoor localization technique based on RF profiling using the received signal strength (RSS) measurements from a set of pre–selected reference points. We do not attach any interpretative significance to the measurements other than use them to calculate their difference from the measurements of the reference points. We study the performance of our technique in an environment with multiple adjacent rooms and find that it gives better results compared to the application of the k-Nearest Neighbor algorithm that has been used in the literature for the same task. We also study the proposed scheme and two other well–known localization schemes, with respect to the sensitivity of the localization on the number and layout of the reference points, as well as on the number and layout of the deployed fixed points (pegs) from where the measurements are collected. We find that one can achieve good localization performance with either fewer reference points or with fewer pegs as long as their layout is chosen carefully.

1 Introduction

Location-based services are a family of, predominantly wireless, network services that, in order to operate effectively, have to rely on some form of location information [4]. While the problem of determining location information has been well supported by the Global Positioning System (GPS) [7], location tracking inside buildings poses specific challenges. An alternative is to determine the location of a node without relying on GPS, but instead based on some ground-deployed permanent (or semi-permanent) wireless infrastructure, i.e., via indoor *radiolocation* techniques. The vast majority of the literature in this area considers WiFi wireless access points as a convenient (semi-) permanent infrastructure. The coordinates of the (semi-) permanent infrastructure nodes are assumed to be known and act as the reference by which the location of other nodes will be determined. Access points are already deployed to extend Local Area Networks (LANs) to wireless users, and localization is a secondary task carried out via the same infrastructure. A relevant, usually unstated, assumption is that the placement of access points is restricted and/or performed for completely different reasons than localization, i.e., to ensure adequate coverage for roaming users. Yet another assumption is that the number of access points cannot be arbitrary and is usually

J. Wozniak et al. (Eds.): WMNC 2009, IFIP AICT 308, pp. 220–231, 2009.

restricted by cost and connectivity considerations. We would rather not have to deal with restrictions on the number and placement of the access points used for radiolocation.

One heavily explored research direction is that of determining which access point–based localization algorithm is best and under what circumstances, e.g. [8]. Instead, in this paper we examine the impact on localization from decisions about the number and placement of the (semi-) permanent nodes (called *pegs* in the sequel). To this end, we use a localization scheme, dubbed LEMON[1] which we previously proposed in [3] and which was found to provide good localization accuracy. In essence, our research is motivated by cost considerations. Namely, a large number of simple low-cost wireless transceivers, like those usually assumed to be part of a wireless sensor node, can be purchased for the cost of a single access point. For example, for the cost of a single access point implementing the most advanced version of the IEEE 802.11 standard (currently approximately USD $200.00) one can purchase a dozen or so boards equipped with the a low power RF transceiver and microcontroller (like, respectively, the Texas Instruments CC1100 and MSP430). With low power battery–based operation and fixed placement of the pegs, e.g., in furniture and walls, we could conceivably have a finer granularity of localization information than what is provided by a handful of WiFi access points that can be acquired with the same expense. The devices to be localized, which we subsequently call *tags*, are essentially disposable, which compares favorably with the cost of a (raw) GPS module (which cannot be used indoor, anyway).

While the cost–benefit relation appears to be self–evident, a shortcoming of populating a space with low cost wireless transceivers is that their readings are not reliable, or even consistent. The low production cost, usually implies poor accuracy when it comes to taking the measurements from such inexpensive nodes “seriously.” Nevertheless, the hope is that, collectively, many poor quality measurements will be better than few, somewhat more accurate, ones. Additionally, the attainable bit rates of the low cost transceivers are usually low. For these reasons we forego localization techniques that are known to be very sensitive to the noise measurement (like lateration and angulation) or that demand extensive data transfers between pegs and localized tags. In fact in environments that exhibit strong influence by multipath propagation effects, even the accuracy of angulation and lateration techniques is questionable. We therefore consider only signal strength pattern matching techniques, i.e., techniques that do not attempt to link the signal strength measurements to any model–based estimate for the actual distance between pegs and localized tag. Specifically, we restrict our attention to location fingerprinting/profiling techniques based on the readings collected at a number of locations, called the *reference points*.

We note that the simplest form of information that can be reasonably extracted from a measurement is the received signal strength (RSS) [1]. RSS measurements do not require specialized additional hardware, as they are routinely available already from most RF transceivers. The commonly acknowledged

[1] Location Estimation by Mining Oversampled Neighborhoods.

disadvantage of RSS is its poor and unpredictable correlation with distance resulting from the multi-path effect, which is typically quite serious inside a building. On the other hand, RSS-based measurements are available from almost any transceiver design used today, regardless of which physical layer protocol it implements. In location fingerprinting, the perceived attributes of the tag's signal are compared against a pre-collected set of samples from known (profiled) reference points [1]. By resorting to profiling, one can hope to compensate for the intrinsic characteristics of the environment which render direct transformations of the perceived attributes of RF signals into distances or angles highly unreliable. This hope underlies our work. It appears that we would need to measure a large number of reference points to counter the imprecise nature of the measurements. This brings up the second issue discussed in this paper, which is determining the smallest number of reference points that are sufficient for a given accuracy of localization. Each reference point measurement could be considered an overhead, since a user (profiler) would have to perform it and associate it with actual coordinates. Whenever the environment changes drastically, e.g., when furniture are rearranged or changed, re-measurement would be necessary. Therefore, the smaller the number of required reference points, the better.

In the next section we present LEMON in detail and summarize the results from [3]. In Section 3 we detail the experiments that were carried out for the present study and the localization errors observed. Finally, Section 4 summarizes our findings and outlines future research directions.

2 LEMON

LEMON can be viewed as a combination of a range-free approach with "traditional" profiling: the scheme is driven by a (somewhat fuzzy) concept of neighborhood, while its objective is to produce an "educated" estimation of the actual location from the coordinates of selected reference point readings. Technically, in LEMON, the location of the static nodes (pegs) need not be known. A tag (the node to be localized) can be essentially a node of the same type as a peg. During the profiling stage, the network collects and stores in a database (maintained on a central server) *samples* acquired from tag devices placed at pre-selected points (the *reference points*) within the monitored area.

A single sample stored in the database can be viewed as a triplet $< C, \Omega, \tau >$, where C stands for the known coordinates of the sampled point, Ω is the so-called association set, and τ, called the sample's *class*, identifies the (settable) RF parameters of the transmitter (typically transmission power, bit rate, and channel number). τ's role is to discern samples collected under different "options" of the tag's transmitter, such that they will only be matched to (future) readings acquired under the same options. For now, we only consider a planar version of the problem, i.e., $C = (x, y)$. The association set Ω consists of pairs $< p, r >$, where p identifies a peg, and r is the RSS value measured by that peg. A tracked tag periodically emits RF packets that include a sequence number used to uniquely identify them. A peg receiving such a packet will forward to the central server a report consisting of its own identifier, the tag identifier, the packet number and class. Having received a number of such reports referring to the same packet, the server will built an association set Φ representing the combined momentary measurements of the tag's RSS by all the pegs that can hear the tag.

The initial step of the localization algorithm is to select from the database a subset of samples representing the best match to the association list Φ. First of all, only the samples of the same class as the received reports are subject to selection. To further narrow down the search, the server finds in Φ the pair $< p_m, r_m >$, such that r_m is the highest among all pairs. Then, it only considers those samples form the database whose association lists include p_m as one of the pegs. This pre-selection boils down to the postulate that the peg p_m appearing to be very close to the tracked tag be a member of all samples that will be used for estimating the tag's location.

Let $\Omega = \{\omega_1, \ldots, \omega_k\}$ and $\Psi = \{\psi_1, \ldots, \psi_m\}$ be two associations sets. By the distance between these sets, we understand

$$D(\Omega, \Psi) = \sqrt{\sum_{j=1}^{N} (R_\Omega(j) - R_\Psi(j))^2} \tag{1}$$

where N is the total number of pegs in the network and $R_\Omega(j)$ is defined as r_j, if the pair $< p_j, r_j >$ occurs in Ω, and 0 otherwise.

In the second step, the server evaluates the distance of each pre-selected sample (its association list) from the current association list Φ. Then, it selects K samples with the smallest distance.

In the last step, the coordinates of the selected samples are averaged to produce the estimated coordinates of the tag. The averaging formula biases the samples in such a way that the ones with a smaller distance from Φ contribute proportionally more. Let D_{max} be the maximum distance from Φ among the best K selected samples and $S_D = \sum_i^K D_i$ be the sum of all those distances. The tracked coordinates are estimated as:

$$x_{est} = \frac{\sum_{i=1}^{K} x_i \times (D_{max} - D_i)}{K \times D_{max} - S_d} \quad \text{and} \quad y_{est} = \frac{\sum_{i=1}^{K} y_i \times (D_{max} - D_i)}{K \times D_{max} - S_d} \tag{2}$$

where (x_i, y_i) are the coordinates associated with sample i.

Note that the above approach does not link RSS to any distance metric but treats it as a purely numerical attribute of a sample whose value should be close to the observed value. The averaging formula factors in the magnitude of discrepancy between RSS values (in terms of distance between points in Euclidean space), but this is a purely numerical interpolation and not an application of any RF propagation model. LEMON does not impose any restriction on the number of pegs or reference points. It relies on the matching rules to locate those samples that best apply to a particular "instance" of localization. If the number of samples is large, the role of the last-step interpolation becomes secondary: we do not purport to expect that the specific RSS values encode useful information about the distance. In particular, it may make sense to oversample the area, e.g., collecting multiple samples from the same point. For example, those multiple samples may correspond to the different orientations of the tag, as in [1]. In the rest of this paper we also study how localization is affected when multiple orientations (at the same point) are identified as such vs. what happens when they are ignored.

3 Experiments

A prototype LEMON system was implemented and tested at our university campus. Early results have been reported in [3]. The devices for both tags and pegs are the EMSPCC11 from Olsonet Communications,[2] which is a low-cost low-power mote for wireless sensor networking, programmable in PicOS [2]. The node employs the CC1100 RF module from Texas Instruments operating within the 916 MHz band. The RF module of EMSPCC11 offers several settings. The transmission power can vary from −15 dBm to 10 dBm (in 8 discrete steps), the bit rate options are 5 kbps, 10 kbps, 38 kpbs, and 200 kbps, and there are 256 different channels (numbered 0 to 255) with 200 kHz spacing. All combinations are possible and, in principle, sensible. The experiments reported on in the rest of this paper were carried out at the lowest power setting with 5 kps transmission rate using channel 0.

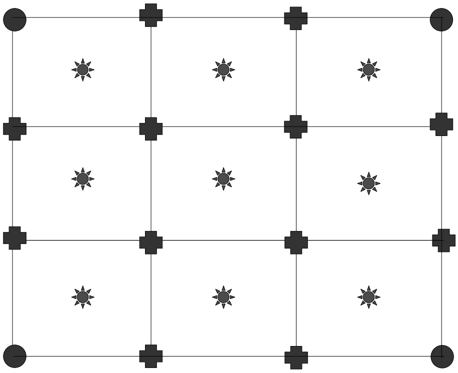


Fig. 1. Layout of pegs (circles), reference points (crosses), and localized points (stars)

In our earlier work [3] we demonstrated how localization in three distinctly different rooms resulted in a localization error less than one meter. Even though LEMON's accuracy has not been yet maximized, it already challenges the accuracy of location estimation reported in [5] which was about 2 m using an even finer grid than ours. Also, our analysis of data revealed that most of the problems with localization resulted from assigning too much relevance to low RSS

[2] See http://www.olsonet.com/Documents/emspcc11.pdf

values, i.e., corresponding to weak reception, which would exhibit large statistical fluctuations. The numerical RSS readings presented by the RF module of EMSPCC11 are positive numbers, roughly between 80 and 150, representing a shifted dB signal level of the received packet. Assuming that those readings are always between *MIN* and *MAX*, we applied the following scaling formula:

$$R_s = \left(\frac{r - MIN}{MAX - MIN} \right)^{\alpha} \tag{3}$$

where $\alpha > 1$. Note that for $\alpha = 1$ we effectively obtain the original (not rescaled) case, as a linear transformation of all RSS readings, does not change the outcome of our algorithm. The best results have been observed for $\alpha = 3$ which allowed us to estimate the tag location with an error of less than 1 m in 90% of the cases.

In this paper we restrict our attention to the problem of localization in multiple (adjacent) rooms focusing on the layout of pegs and the number and location of reference points, subject to certain regular patterns.

3.1 Multiple Room Localization

We conducted an experiment using three adjacent rooms. The dimension of each room is about 3 m × 5 m. Each room includes four wooden tables and chairs, a metal file cabinet and two desktop computers. We placed four pegs at the corners of each room and generated a grid with the edge size around 1 m. In particular, in each room twelve readings were taken at the reference points (indicated by crosses in Figure 1) and nine locations were estimated: by placing tags at the nine points positioned at the center of each grid cell (indicated by stars in Figure 1).

It was noted previously [1] that orientation impacts localization. To verify that observation, we collected tag readings for four different directions (North, South, East, and West) at each profiled or localized point. At the localization stage, to take the orientation aspect into account, we would only use for estimation the profiled data collected from tags oriented in the same way as the

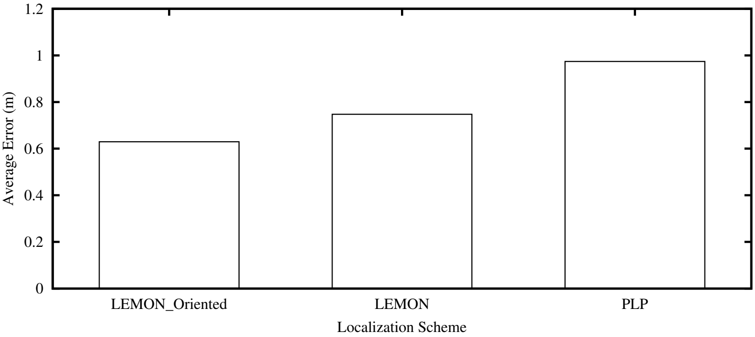


Fig. 2. LEMON (with & without orientation) vs. PLP's k-NN

one being localized. Even though the line of sight (LOS) was not interrupted regardless of the orientation of the tag (suggesting that the orientation should have no impact on location estimation), the observed error was larger when the orientation aspect was ignored (i.e., all profiled data were used regardless of the orientation of the localized tag). The observed difference was about 12 cm on the average (Figure 2), in favor of the localization based solely on the reference points collected with the same orientation. While the difference is not huge, it crisply illustrates how an even minor change of tag placement (namely rotation on the same spot) influences localization. Hence, baring information about orientation, we should not expect localization accuracy to be any tighter than the difference found between oriented vs. non-oriented localization.

A second element of this study is a comparison against a localization scheme based on the k–Nearest Neighbor (k–NN) algorithm. Specifically, the PowerLine Positioning (PLP) proposed in [6], employs k–NN for multi–room localization. It is one of the few papers in the bibliography where localization is designed specifically with multiple room environments in mind. PLP is also a profiling based approach, but it uses powerlines as the means to distribute RF signals throughout a building. Apart from the unconventional use of the powerline as an "antenna," PLP's approach is to apply a sufficiently large k for the nearest neighbors to localize and identify the room where the tag is located. The reference points from the identified room are then used with a smaller k to estimate the tag's location. We tested the same logic as PLP using our experimental data with k equal to six and four for, respectively, selecting the room and then estimating the location within the room. The results show that the estimation accuracy of PLP degrades as it subdivides the signal space according to the physical layout, and then it localizes within a selected space. There is apparent loss of information when we first identify the room and then localize strictly within that room; hence, the large error distance shown in Figure 2, even against LEMON without orientation information. LEMON outperforms PLP when we consider the three rooms as one signal space and use all reference points to perform the estimation. What this demonstrates is that the intuition of restricting the localization to a subset of reference points that might appear "closest" results in loss of information that could be crucial to localization, i.e., reference points in adjacent rooms *do matter*.

3.2 Effect of Reference Points

In the rest of this study, we compare LEMON against two well known general–purpose localization schemes, LANDMARC [5] and RADAR [1]. Specifically, we investigate the effect of the number of reference points and their arrangement on localization error. For this purpose, we gathered 49 reference points, i.e., from *every* grid point of a 7×7 grid, where the pegs were positioned at 16 locations (depicted as circles in Figure 3). When a reference point measurement was at the same location as a peg, the actual measurement was taken 6 cm away from the peg. We then removed reference points from the database in a regular fashion such that the density of reference points remained roughly the same across the

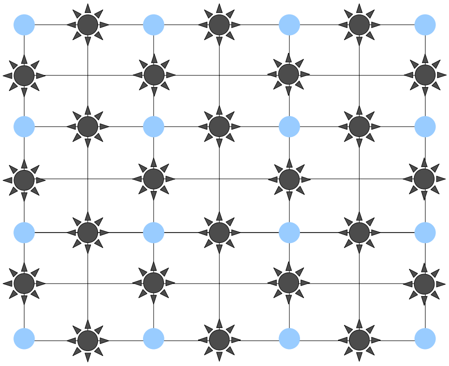
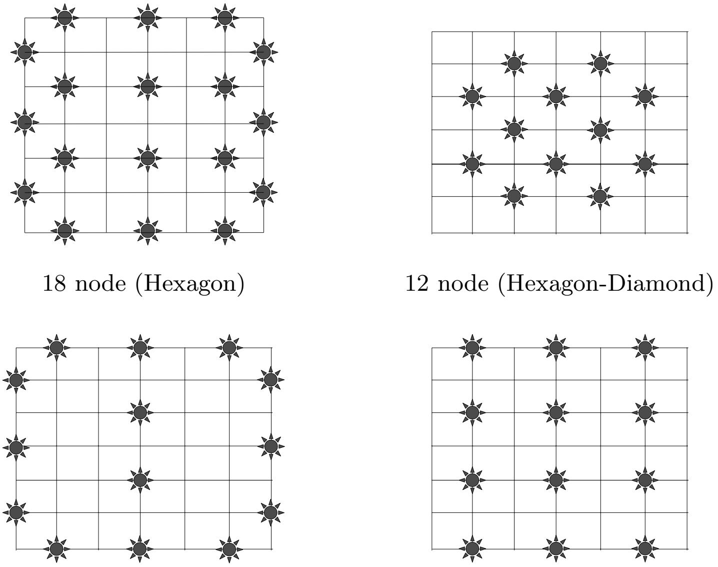


Fig. 3. Locations of pegs and of 24 reference points (Diamond)

grid. Figure 3 shows the layout of reference points after removing 25 points; the resulting setup is dubbed *Diamond* and consists of measurements from 24 reference points. By further removing points in the same fashion, i.e., one at a time, we arrive at two new layouts (see Figure 4) called, respectively, *Hexagon* (18 points) and *Nested-Hexagon-Diamond* (12 points). Starting with the Hexagon layout and eliminating more points, we produce two additional layouts dubbed *DoubleD* (14 points) and *Square* (12 points), respectively.



18 node (Hexagon)

12 node (Hexagon-Diamond)

14 node (DoubleD)

12 node (Square)

Fig. 4. Reference point location configurations

The average error in location estimation for LEMON, LANDMARC, and RADAR in each of the above layouts is shown in Figure 5. The average error for LEMON with all the initial 49 reference points was 0.43 m, but we could remove points in a regular fashion and end up with 12 points while the error distance was still less than 1 m. This indicates that it is possible to have a less complex deployment of LEMON without degrading the performance significantly (see, e.g., the case of Dia(24)). Overall, we have observed that the more the reference points the better the localization. Yet, the number of reference points alone is not sufficient, as their placement matters as well (see for example the case of DD(14) vs. Square(12)). Moreover, LEMON is less sensitive to the number of reference points and their layout compared to LANDMARC and RADAR. In fact, of all three, RADAR appears to be the one scheme that deteriorates the most when the number of reference points is reduced.

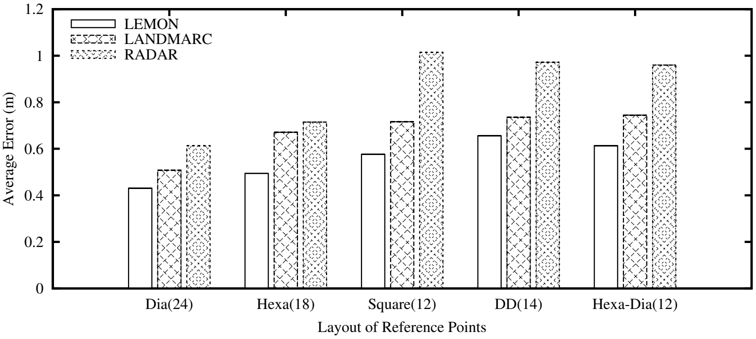


Fig. 5. Localization error for different reference point configurations

3.3 Effect of Peg Placement

The unique feature of LEMON is its flexibility of using as many pegs as needed without imposing any restriction on their number. However, there might exist better or worse ways of placing those pegs that may offer better or worse estimation. Finding the best placement of pegs for a target localization accuracy may pose a challenge. We may not need, say, 16 Pegs for 7×7 grid to keep the error distance below 1 m. Thus we conducted another experiment to try to produce configurations with fewer pegs that still provide a localization error less than 1 m. Figure 6 shows two such layouts with 8 Pegs and the corresponding average localization error is shown in Figure 7. As we can see, it is possible to eliminate half of the Pegs and still maintain error distance well below 1 m. We also found that removing just four pegs (the “center square” of circles in Figure 3) would leave 12 pegs (called “Square” in Figure 7) but somewhat higher localization error than the configurations with the fewer but “better placed” pegs. This may suggest that a clever design of placing pegs on the target area may help us reduce the error distance. Our observations indicate that removing pegs from inside the

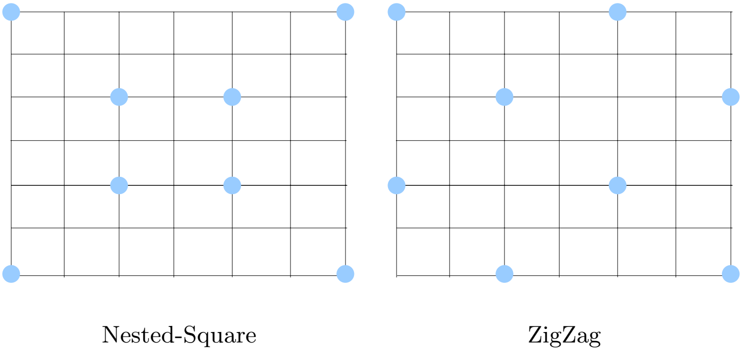


Fig. 6. Peg layouts for eight pegs

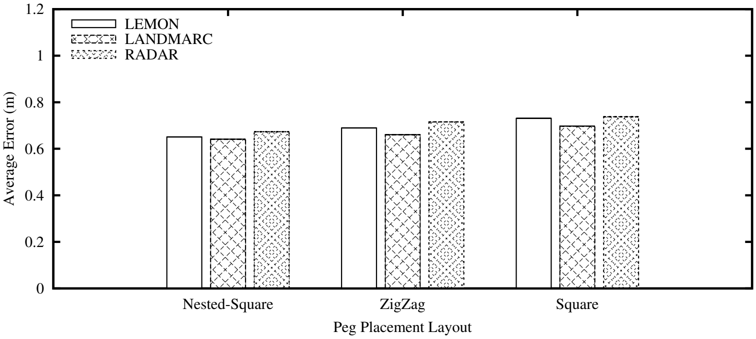


Fig. 7. Localization error for different peg layouts

grid may not be a good idea. Yet, what is even more interesting to note is that all three schemes (LEMON, LANDMARC, and RADAR) are relatively insensitive to the peg layout, compared to the impact that the layout (and number) of reference points could have. Additionally, LANDMARC appears to have a slightly better performance than the other two, but the statistical significance of the differences is very small. The lesson is that peg placement is not as critical an issue as the dense "sampling" of the space by many reference points. This is good news, as the placement of pegs is likely to be constrained or even dictated by external factors, e.g., placement of furniture, walls, etc.

4 Conclusions

Our proposed scheme, LEMON, can easily be applied to in-building localization using inexpensive devices and can provide an average localization error well below 1 m. Furthermore, we have found evidence that attempts at restricting the population of reference points to those in the suspected proximity of the target (e.g., for complexity reasons) tend to backfire. Put differently, the final accuracy of estimation depends on information collected from distant reference points in a surprisingly significant sort of way.

LEMON's use of a rich set of reference points, collected by pegs positioned at fixed locations, was put under the microscope and we found out that a smaller set of reference points or a smaller set of fixed points (peg stations) can provide good localization, in particular if their layout has been carefully designed. This is particularly true for the layout and number of reference points. In fact, faced with a choice between introducing more reference points or more pegs, the answer appears to be in favor of more reference points.

We can confidently say that the limits of LEMON have not been yet reached. LEMON achieved a localization error below 1 m compared to the accuracy of location estimation reported in [5], which was about 2 m using a *finer* grid than in our case. Yet, LEMON did not apply any of the additional techniques outlined in [5] which could conceivably improve its performance of LEMON even further. For example, one idea of LANDMARC and its derivatives [5], is to assign a fixed set of reference tags constantly providing profile samples. This approach can help by updating the reference point measurements almost in real–time to factor in the possible changes in the monitored space, e.g., due to furniture moving, the number of inhabitants etc. Thus one can think of adapting LEMON on-line, such that a subset of reference points in the database is being perpetually replaced with new readings provided by pre-installed "fixed tags." Finally, the uniformity and low cost of the equipment makes LEMON a highly viable and very practical solution.

The selection of reference points and/or peg locations appears to be an extremely important facet of the problem, of which we have just scratched the surface in this paper. We plan to define the placement problem formally as an optimization problem. Its solution may guide real-life deployments of pegs and the ways of collecting reference points (profiling) as to minimize the system's complexity (cost) for the required accuracy of localization. The analytical results can be verified with our experimental data.

References

1. Bahl, P., Padmanabhan, V.N.: RADAR: an in-building RF-based user location and tracking system. In: Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000), Israel (March 2000)
2. Gburzyński, P., Olesinski, W.: On a practical approach to low-cost ad hoc wireless networking. Journal of Telecommunications and Information Technology (1), 29–42 (January 2008)
3. Haque, I., Nikolaidis, I., Gburzyński, P.: A scheme for indoor localization through RF profiling. In: Proceedings of the Intl. Workshop on Synergies in Communications and Localization (SyCoLo 2009), Dresden, Germany (to appear) (June 2009)
4. Küpper, A.: Location–Based Services: Fundamentals and Operation. Wiley, Chichester (2005)
5. Ni, L.M., Liu, Y., Lau, Y.C., Patil, A.P.: LANDMARC: indoor location sensing using active RFID. Wirel. Netw. 10(6), 701–710 (2004)

6. Patel, S.N., Truong, K.N., Abowd, G.D.: PowerLine Positioning: A practical sub-room-level indoor location system for domestic use. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 441–458. Springer, Heidelberg (2006)
7. United States Coast Guard Navigation Center. Global positioning system standard positioning service specification (June 1995)
8. Wallbaum, M., Diepolder, S.: Benchmarking wireless lan location systems. In: Proceedings of the 2005 Second IEEE International Workshop on Mobile Commerce and Services (WMCS 2005), Munich, Germany, July 2005, pp. 42–51 (2005)