# Tab WoNS: Calibration Approach for WSN based Ultrasound Localization Systems

## Extended Abstract

Clemens Mühlberger
Chair of Computer Science V
Am Hubland
Würzburg, Germany
muehlberger@informatik.uni-wuerzburg.de

Marcel Baunach
Chair of Computer Science V
Am Hubland
Würzburg, Germany
baunach@informatik.uni-wuerzburg.de

## ABSTRACT

Accurate localization of objects is often very important, thus various methods and systems exist. Unfortunately, some support easy deployment but are less accurate while others are more accurate but require a complex and time-consuming deployment stage. When the localization system meets some demands, a widely self-organizing deployment becomes possible. This paper describes which pre-conditions and abilities are required, how self-configuration could be realized for such a system and which minimal number of calibration steps are required during deployment.

## 1. INTRODUCTION

Lots of application scenarios for wireless sensor networks (WSNs) are in need of accurate knowledge about current positions of several objects. Because this can be such an important basic requirement for the system's successful operation, not only different approaches were proposed, but also based on these solutions various localization systems were built and successfully proven (c.f. [1, 4, 5, 6, 7, 10, 11, 12]). Indeed, the sometimes costly deployment process of such a WSN was hardly analyzed so far. Yet, deployment can be very time-consuming and cost-intensive, especially when numerous beacons have to be mounted and calibrated quite precisely. That's why this paper presents a self-deployment strategy for beacon-based localization systems to save time and costs. Therefore section 2 defines some basic requirements for the self-deployment procedure, whereas section 3 introduces the localization system SNoW Bat [4] to form the basis of our further research. Section 4 specifies our deployment approach Tab WoNS, which is quite similar to the localization stage of SNoW Bat. Our thoughts and results for the calibration during deployment stage are presented in section 5 and finally a conclusion in section 6 closes this paper.

## 2. SYSTEM REQUIREMENTS

The self-deployment strategy presented in this paper requires the desired localization system to comply with the following requirements. First of all, the sensor nodes of the localization system are classified as *beacons* (i.e. anchor nodes with known and constant position) and *clients* (i.e. mobile nodes with unknown and variable position). To derive basic conditions for efficient calibration during the self-deployment stage, our approach presumes that the beacons are aligned in a regular pattern. For this paper a roughly grid-like[1] layout is analyzed.

The underlying localization system uses ultrasound signals (so called *chirps*) for distance measurement. But it is sufficient – even for deployment stage – if beacons are just able to receive ultrasound signals whereas only clients are capable of transmitting ultrasound for distance measurement. There will be no need for the anchors to generate ultrasound signals, which saves energy and reduces environmental pollution by ultrasound emitted from the quite large number of beacons in such systems. Besides we are just interested in three-dimensional localization, the two-dimensional equivalent would be much easier but less realistic. However, for simplification we will assume that the anchors are fixed at the common ceiling plane and a mobile client moves freely thereunder within a certain distance range (cf. Fig. 1). Additionally the distance between ceiling and mobile client is not necessarily known.

## 3. LOCALIZATION SYSTEM

This section introduces the underlying localization system SNoW Bat. It operates based on the thunderstorm principle, where radio packets correspond to lightning and ultrasound signals to thunder. According to the requirements from section 2, all sensor nodes are divided into beacons and clients. After deployment each beacon knows its position and is able to detect ultrasound signals, whereas no client knows any coordinates a priori. However, each client is able to generate ultrasound signals and to initiate the following localization procedure:

1. If a client wants to know its current position, it broadcasts a radio packet containing its ID and the time difference between this radio message and the subsequent ultrasound chirp.
2. Each anchor receiving this synchronizing radio packet activates its ultrasound receiver and thus expects the client's ultrasound signal after the specified waiting time.

---

[1] If the anchors are adjusted to an exact grid, each anchor can already determine its position if only one single anchor knows its position and if the anchors have additional knowledge about their placement, orientation, etc.

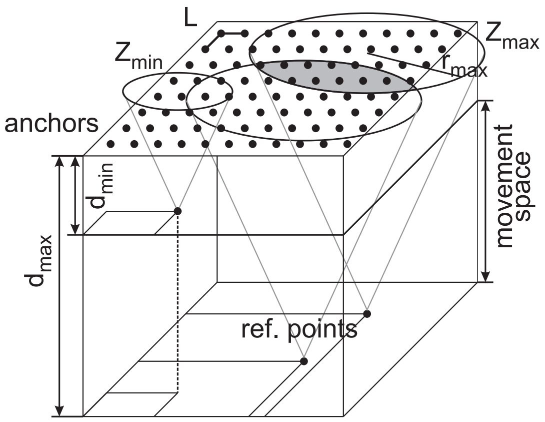

**Figure 1: Schematic system design: anchors are at the ceiling along a grid-pattern with grid constant $L$, clients are thereunder. The ultrasound signal is emitted as spherical sector, which defines a coverage zone $Z$ with radius $r$.**

3. The mobile client transmits the ultrasound chirp as denoted in the previous radio broadcast.
4. Each beacon which detects this ultrasound signal calculates its distance towards the mobile client from the time difference of arrival between the radio and ultrasound signal. If an anchor just receives the initiating radio message but no chirp, it aborts measurement and waits again for further clients to be localized.
5. Each beacon, which successfully completed its distance measurement, passes this result together with its known position to the requesting client as unicast. This is done by the self-organizing communication protocol HashSlot [2, 3].
6. By using for example multilateration, the client can estimate its three-dimensional position autonomously from at least four such distance measurements.

## 4. DEPLOYMENT STRATEGY

Inspired by the localization procedure, we head for the deployment stage. The main idea of our deployment approach is reversing the roles of anchors and clients during deployment. That is, just during deployment a *reference client* below the anchor plane knows its position, whereas every beacon doesn't yet know its coordinates. The remaining requirements from section 2 are retained: only the (reference) client generates ultrasound signals, which can be detected by the grid-aligned beacons at the anchor plane. The deployment procedure will than be as follows:

1. The reference client is situated at a known position (so-called *reference point*) and broadcasts a radio packet containing its ID, its current position and the time difference between this radio message and the subsequent ultrasound chirp.
2. Each beacon with yet unknown position receiving this synchronizing radio packet activates its ultrasound receiver and thus expects the client's ultrasound signal after the specified waiting time.
3. The reference client transmits the ultrasound chirp as denoted in the previous radio broadcast.
4. Each beacon which detects this ultrasound signal calculates its distance towards the mobile client from the time difference of arrival between the radio and ultrasound signals. If an anchor just receives the initiating radio message but no chirp, it aborts measurement and waits again for radio packets of the reference client.
5. If a beacon successfully completed at least four distance measurements to non-collinear reference points, it estimates its fixed position, e.g. by using multilateration. From now on, this beacon knows its position and is available for locating mobile clients with unknown position as described in section 3.

Obviously, deployment (this section) and regular operation (section 3) are very similar. Their only differences are the knowledge of the nodes' position and the handling of the distance information. In comparison to the procedure during the localization stage, no replies from beacons to clients are required anymore. This way, the hardware for clients and beacons remains the same for deployment and localization. Still, this deployment procedure implies two open issues which need a closer examination:

1. First, the reference client needs exact knowledge about its own position now. For example, this can be done manually, but an inaccuracy of a couple of millimeters already produces drastic errors during localization.
2. Next, the number of required known positions of the reference client during deployment should be kept as small as possible to save energy, time and costs, mainly manpower.

Because the first item depends on the craftsmanship of the calibrating persons as well as the accuracy of the used measuring instruments, we will initially focus on the second item in the next section.

## 5. CALIBRATION

In this section we address the localization system calibration after installation of the static beacons at the anchor plane. The goal is to determine accurate anchor positions while achieving a short duration for this exceptionally important process. As already described in section 4, we use certain carefully selected reference points at well known positions to allow each anchor-to-be to measure at least four distances in a three-dimensional space. Of course, increasing this number yields higher precision but also consumes more time and power. Thus, the number of reference points shall be minimized but must still allow proper and complete operation. Therefore we define the ratio

$$Q := \frac{|\text{anchors-to-be}|}{|\text{reference points}|}$$

as central metric for the solution's quality and try to maximize $Q$.

Since we consider a grid-like pattern for the anchors, we also expect a regular pattern for the optimum reference point

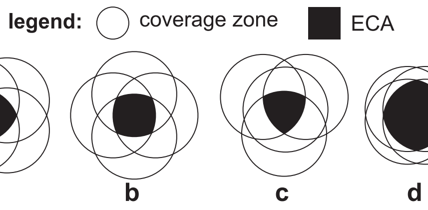

**Figure 2: Some basic layouts, each consists of four overlapping circles of equal radii forming an explicit coverage area (solid black).**

alignment. SNoW Bat uses a regular grid for several reasons. It is easy to install, contributes to the performance of the self-organizing HashSlot [2, 3] wireless communication protocol, and allows easy adaption to the environmental conditions. The last point allows to easily provide complete localization service coverage throughout the entire operational area. As figure 1 shows, the selected grid constant $L$ depends on the minimum distance $d_{min}$ of the mobile objects from the anchor plane and the resulting minimum ultrasound coverage zone $Z_{min}$. According to [3], $L$ is computed to always guarantee a minimum number of four nodes within an ultrasound coverage zone $Z$ independent from its overlay position within the grid. Thus $L$ stays fixed after system installation and obeys to the spatial geometry and hardware constraints during system design. On the other hand, $d_{max}$ limits the maximum number of anchors within the largest possible ultrasound coverage zone $Z_{max}$. Unfortunately, no analytical solution for the exact number of grid points within a circle is yet known. However, approximations for the so called *lattice point problem* are presented in e.g. [8, 9]. Anyway, the reference points must obviously be placed at distance $d_{max}$ from the anchor plane. Then, their final number decreases along with increasing difference $d_{max} - d_{min}$. Still, the optimal alignment remains to be found.

The idea so far is to initially place four non-collinear reference points in a way to maximize the intersection area of all resulting coverage zones, Fig. 2d shows an example. This intersection produces the so called *explicit coverage area* (ECA). Then, the just created *basic layout* is repeated with a certain distance to implicitly cover the anchors in-between the explicit areas (*implicit coverage area*, ICA), too. Fig. 3d shows a corresponding alignment of reference points. Indeed, this special basic layout is rather adverse, since several reference points are close to each other and the implicitly covered areas are very small. If we increase the space between the basic layouts to enlarge the implicit coverage areas, we might require some additional reference points to keep fourfold coverage (e.g. the bold circle in Fig. 3a).

During our analysis we found that the proportion between explicitly and implicitly covered areas should be roughly balanced to omit extra reference points. Then, implicitly and explicitly covered anchor sets with about the same size alternate. Fig. 3a-c shows some more basic patterns with balanced proportion.

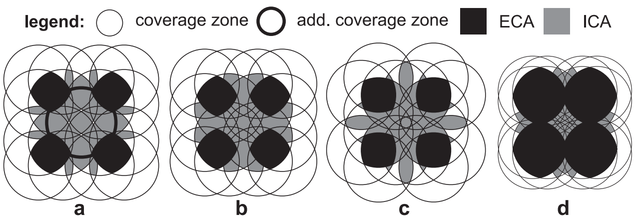

**Figure 3: Slightly different alignment strategies when using basic layouts a, b and d from Fig. 2.**

**Table 1: Quality $Q$ of various alignment strategies for $r_{max} = 9L$**

| Strategy | $Q$ |
|---|---|
| Fig. 3a | 42.44 |
| Fig. 3b | 41.17 |
| Fig. 3c | 39.06 |
| Fig. 3d | 39.59 |

Regarding the metric describe above, table 1 shows that the quality $Q$ is exceptionally good for these balanced alignments since few reference points are required for calibrating many anchors. Hence, pattern 3a offers the best quality of the analyzed strategies, although it requires additionally reference points.

Still, the achievable precision for each anchor's position estimation needs careful consideration. Indeed, the calibration accounts significantly to the accuracy of the localization system's regular operation. Thus, we recommend the following method for checking the anchors and to perform partial recalibration of the anchors by additional measurements where necessary:

1. Calibrate the system as described above.
2. Operate the system to localize objects at some additional well-known test points by using each anchor at least once.
3. Compare the correct, known position to the estimated one and recalibrate these anchors which were involved in deviations larger than an acceptable threshold.

## 6. CONCLUSION AND FUTURE WORK

In this paper we presented an efficient strategy for the deployment of an ultrasound based localization system. Therefore some basic system requirements were defined. E.g. the static anchor nodes must be mounted along a grid-like pattern above the observed mobile clients. One benefit of our strategy is the usage of identical hardware for calibration during deployment as well as for localization during regular operation. Just the roles of anchors and mobile clients will be reversed.

Since the anchors initially do not know their coordinates, the basic idea of our approach is to use a reference node at well known positions for measuring some distances to these anchors, which then calculate their static position. The problem is to find a minimal set of reference points and to still provide a sufficient number of distances for complete anchor coverage. We showed, that this problem is not trivial and does not only depend on the system accuracy and parameters (grid constant, etc.) but also on the reference node's positions during the calibration process.

So far, we just analyzed a few different reference point layouts and strategies. The next step is to integrate the deployment procedure as described above into the existing localization system SNoW Bat. Analyzing systems with arbitrarily aligned anchors is another interesting issue. Hence, we want to proof some upper or lower bounds for the number of required reference points within such systems.

## 7. REFERENCES

[1] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In *INFOCOM (2)*, pages 775–784, 2000.

[2] M. Baunach. Speed, Reliability and Energy Efficiency of HashSlot Communication in WSN Based Localization Systems. In R. Verdone, editor, *EWSN*, volume 4913 of *Lecture Notes in Computer Science*, pages 74–89. Springer, 2008.

[3] M. Baunach, R. Kolla, and C. Mühlberger. A Method for Self-Organizing Communication in WSN Based Localization Systems: HashSlot. In *LCN*, pages 825–832. IEEE Computer Society, 2007.

[4] M. Baunach, R. Kolla, and C. Mühlberger. SNoW Bat: A high precise WSN based location system. Technical Report 424, Institut für Informatik, Universität Würzburg, May 2007.

[5] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low-cost outdoor localization for very small devices. *Personal Communications, IEEE [see also IEEE Wireless Communications]*, 7(5):28–34, 2000.

[6] Y. Fukuju, M. Minami, H. Morikawa, and T. Aoyama. DOLPHIN: An Autonomous Indoor Positioning System in Ubiquitous Computing Environment. In *WSTFES '03: Proceedings of the IEEE Workshop on Software Technologies for Future Embedded Systems*, page 53. IEEE Computer Society, 2003.

[7] A. Günther and C. Hoene. Measuring Round Trip Times to Determine the Distance Between WLAN Nodes. In R. Boutaba, K. C. Almeroth, R. Puigjaner, S. X. Shen, and J. P. Black, editors, *NETWORKING*, volume 3462 of *Lecture Notes in Computer Science*, pages 768–779. Springer, 2005.

[8] M. N. Huxley. Corrigenda: Exponential Sums and Lattice Points II. *Proc. London Math. Soc.*, s3-68(2):264–, 1994.

[9] H. Keller. Numerical Studies of the Gauss Lattice Problem. Technical Report CRPC-TR97699, Center for Research on Parallel Computation, Houston, Jan. 1997.

[10] R. Prasad and M. Ruggieri, editors. *Applied Satellite Navigation Using GPS, GALILEO, and AugmentationSystems*. Artech House, 2005.

[11] N. B. Priyantha. *The Cricket Indoor Location System*. PhD Thesis, Massachusetts Institute of Technology, June 2005.

[12] A. Savvides, C.-C. Han, and M. B. Strivastava. Dynamic fine-grained localization in Ad-Hoc networks of sensors. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 166–179, New York, NY, USA, 2001. ACM Press.