

Low Overhead Assignment of Symbolic Coordinates in Sensor Networks

Matthias Gauger^{1,2}, Pedro José Marrón², Daniel Kauker¹, and Kurt Rothermel¹

¹ IPVS, Universität Stuttgart, Germany

{gauger, kaukerdl, rothermel}@ipvs.uni-stuttgart.de

² Universität Bonn, Germany

{gauger, pjmarron}@cs.uni-bonn.de

Abstract. Approximate information on the location of nodes in a sensor network is essential to many types of sensor network applications and algorithms. In many cases, using symbolic coordinates is an attractive alternative to the use of geographic coordinates due to lower costs and lower requirements on the available location information during coordinate assignment. In this paper, we investigate different possible methods of assigning symbolic coordinates to sensor nodes. We present a method based on broadcasting coordinate messaging and filtering using sensor events. We show in the evaluation that this method allows a reliable assignment of symbolic coordinates while only generating a low overhead.

1 Introduction

Information on the context of nodes in a sensor network is essential to many types of sensor network applications. Examples of such context knowledge are the positions of nodes, their neighborhood or the external conditions of a node's surroundings. Among the different types of context, location information plays an especially important role in sensor networks as it is required if sensor readings are to be associated with the area they were recorded in.

Acquiring position information of sensor nodes in the form of geographic coordinates with an acceptable precision is a very difficult and often costly operation. This is especially true for indoor scenarios where localization technologies like GPS receivers do not work well. One possible alternative to determining geographic coordinates is to assign symbolic coordinates to nodes. Instead of describing positions in the form of a coordinate tuple, a symbolic coordinate represents areas of different shapes and sizes in the form of an abstract symbol. All sensor nodes in an area have the same symbolic coordinate. Examples of such symbolic coordinates are room numbers in a building or street addresses.

In different types of sensor network applications it is possible to use symbolic coordinates instead of geographic coordinates. One exemplary field is the retrieval of sensor data from specific areas of a sensor network. In many cases, symbolic coordinates directly represent the semantics of a location, for example, when a symbolic coordinate is associated with each room of a building. This

allows to implement data retrieval operations very easily without having to map from coordinates to areas first. Another possible application of symbolic coordinates is using them for cost-effective many-to-many routing of messages in sensor networks. We specify symbolic source routes from the sender node to the destination area and later translate these routes on the node level into specific routing decisions. The applications of the methods presented in this paper do not have to be limited to the assignment of symbolic coordinates. It is also possible to use them for node clustering (i.e., all nodes in a room form a cluster). In such room-level clusters the nodes typically generate related information which can be aggregated with less information loss than in arbitrary clusters.

We argue that assigning symbolic coordinates in sensor networks is usually much easier than assigning geographic coordinates and is possible with reasonable effort. In this paper, we discuss different approaches to this assignment for indoor scenarios and present one solution that achieves a very low error rate while only generating a small overhead and requiring no prior knowledge on the sensor network topology. The basic idea is to let an administrator broadcast symbolic coordinates in the different rooms a sensor network is deployed in and let the nodes use sensor information to filter out broadcast messages wrongly received from neighboring rooms.

The rest of this paper is organized as follows. The following section briefly reviews important related work. We present our three approaches to the assignment of symbolic coordinates in Section 3. In Section 4 we describe relevant details of our implementation before Section 5 provides an in-depth evaluation of our approaches. Section 6 concludes the paper and discusses some future work.

2 Related Work

Our work is related to the large set of approaches in the area of node localization which is one of the fundamental research problems in wireless sensor networks. Most localization approaches have in common that they require a set of anchor nodes with known positions. Then different techniques and algorithms are used to determine the positions of the other nodes (see for example [1]). One important factor is how distances between nodes are determined. There are range-free solutions that only consider the hop-count, distance estimations based on the received signal strength and solutions requiring special ranging devices (e.g., [2]). Elnahrawy, Li and Martin [3] discuss the fundamental limits of localization techniques based on signal strength when used in indoor scenarios.

The Spotlight localization system [4] is particularly related to our approach in that it also uses sensor events for the localization of nodes. A helicopter (the Spotlight device) which knows its own position flies over the sensor network and generates light events at certain points of time. The sensor nodes report when they detect events back to the helicopter, which is then able to compute the geographic coordinates of the nodes. However, as the authors aim to calculate geographic coordinates the required calculations are rather complex and a precise time synchronization of nodes is required.

Similar to our assignment of symbolic coordinates, Corke, Peterson and Rus [5] assign geographic coordinates to sensor nodes using radio communication. In their scenario, a robot helicopter equipped with a GPS receiver flies over the network area and periodically broadcasts beacon messages containing its current geographic coordinate. The sensor nodes on the ground typically receive multiple such beacons and need to estimate their position based on this information. The authors propose different methods for this calculation, including taking the mean of the positions or the signal strength weighted mean of the positions. A constraint-based method performed best in their experiments.

There has been active work on symbolic coordinates, their applications and underlying location models mostly in the area of pervasive computing (e.g., [6], [7]). Becker and Dürr [8] give a comprehensive overview of different geometric and symbolic location models from the perspective of pervasive computing and compare their suitability for different types of queries. In our work we assume that such a symbolic location model is available so that an administrator can assign symbolic coordinates to individual rooms.

3 Assignment of Symbolic Coordinates

In this section we introduce our three different approaches to assign symbolic coordinates in sensor networks. We start by describing the properties of our target scenarios in more detail.

Our target system consists of a set of sensor nodes that are distributed to different rooms in a building. Each room is uniquely identified by a symbolic coordinate, which can, for example, correspond to the respective room number. In the following discussion, we assume for simplicity that a symbolic coordinate can be represented by an integer value.

We assume that each sensor node is preprogrammed with a node identifier (node ID) and that this node ID is unique in the sensor network. However, we do not assume that there is any relation between this node ID and the location of the node in the network. In the beginning, the sensor nodes do not have any information about the symbolic coordinate of the room they are located in.

After the deployment of the sensor nodes there is one person – which we call administrator – responsible for assigning the correct symbolic coordinates to the nodes in the network. The administrator has a mobile client device that is able to directly communicate with the nodes of the sensor network. The client device can be used to send so-called coordinate messages to the nodes.

3.1 Individual Assignment of Symbolic Coordinates

The most basic way of assigning symbolic coordinates to nodes is to assign the respective coordinate to each node individually. In some scenarios it might be possible to directly encode the symbolic coordinate in the program code of the node like it is typically done with the node ID. However, this already limits the flexibility for the placement of nodes. Moreover, we generally expect the sensor

nodes to be delivered to users preprogrammed with an application software when the application field lies outside of typical research settings.

To assign symbolic coordinates to sensor nodes after the deployment of the network the administrator needs to send the correct symbolic coordinate in a coordinate message to each individual node using the node's wireless interface. Upon receiving such a message the node stores the new symbolic coordinate and uses it from this point on.

The clear advantage of an individual assignment of symbolic coordinates to nodes is that it avoids ambiguity: The administrator has complete control over which nodes receive which symbolic coordinate and is able to ensure the correctness of the assignment process. However, there are also a number of clear disadvantages. The assignment of coordinates requires individual communication with each sensor node in the network. Consequently, the required time and effort (and also the message complexity) grow with the number of nodes in the network. More important, it is necessary that the administrator has an up-to-date knowledge on the distribution of nodes to the individual rooms in the building.

3.2 Assignment of Symbolic Coordinates by Broadcast

Our second approach, the assignment of symbolic coordinates by broadcast, aims to address the disadvantages of having to separately assign a symbolic coordinate to each individual node in the network. Instead, the goal is to distribute the coordinate information to all nodes in the area in one step.

The network administrator needs to visit the different rooms covered by nodes of the network and has to send out a message containing the current symbolic coordinate information in each room. The message is sent by broadcast so that all nodes in the one-hop neighborhood of the client device receive the information. Upon receiving such a message, a sensor node overwrites its symbolic coordinate information with the newly received data.

One advantage of sending out coordinate information by broadcast is the lower overhead both for the administrator and in terms of messages as the configuration has to be done only once per room instead of once per sensor node. Moreover, no information is required on the position of individual sensor nodes since the coordinate information does not have to be addressed to specific nodes. The only information that must be available is the symbolic coordinate of the room the broadcast message is sent in.

It is typically desirable that sensor networks are connected across area boundaries like walls between rooms to provide for communication between different network parts. Therefore, the main challenge in the assignment of symbolic coordinates by broadcast is nodes that receive coordinate messages from neighboring rooms. Depending on the sequence of messages sent, these nodes might overwrite the correct coordinate information with data belonging to a neighboring room.

The basic approach to address this challenge is to control the signal strength of the messages sent by the client device. Ideally, the strength of the signal should still allow the message to reach all nodes inside the current room and none of the nodes in neighboring rooms. However, it is difficult to find this balance

especially if two nodes in different rooms are located very close to each other and the attenuation of the signal by the wall between them is small.

We have developed two extensions for the assignment of symbolic coordinates by broadcast. The first extension allows sensor nodes to store and manage multiple symbolic coordinates at the same time so that different coordinate messages do not overwrite each other's information. If a node has received multiple symbolic coordinates, then it lies in the border area of the rooms represented by these coordinates.

The second extension analyzes the signal strength of the different coordinate messages at the receiver, called Received Signal Strength Indication (RSSI) value, and uses this information to assign a coordinate to the node (i.e., the coordinate from the message with the highest RSSI value). However, this only works reliably if the signal attenuation of the walls between rooms is significant. This can be problematic because RSSI – despite its strong limitations in indoor scenarios [3] – is still rather an indicator for geographic distances among nodes than for the separation of nodes to different areas.

3.3 Assignment of Symbolic Coordinates by Assisted Broadcast

Both approaches to the assignment of symbolic coordinates presented so far have disadvantages. The first approach is quite intricate and requires detailed knowledge on the nodes' positions in the network whereas the second approach is susceptible to errors due to the propagation of coordinate messages across room boundaries. In the following we present a third approach which avoids these problems.

Like in the previous approach, the client device broadcasts coordinate messages. It is again the task of the administrator to set the symbolic coordinate sent out by the client in a way so that it corresponds to the current location of the administrator. The important difference lies in the handling of a received coordinate message by the sensor nodes. Instead of directly assigning a symbolic coordinate received in a coordinate message, a node first checks whether the new coordinate is confirmed by a sensor stimulus following the message. For example, the node checks whether the light level changes significantly after the receiving of the message.

Directly after sending out a coordinate message (within a specified time interval of a few seconds) the administrator has to trigger an event that can be detected by one of the sensors of the nodes. The goal is to distinguish nodes that are located in the same room as the administrator and that receive the coordinate message and detect the following sensor event from nodes in neighboring areas that only receive the coordinate message but are not affected by the sensor event triggered by the administrator.

Fig. 1 shows the pseudo code for checking whether a received symbolic coordinate should be assigned. Directly after having received a new coordinate message, the node records its current sensor value and starts a timer. When the timer fires, it records its sensor value again and calculates the absolute difference

```

int ownSymbolicCoordinate;
int candidateSymbolicCoordinate;

event receivedCoordinateMessage(int newSymbolicCoordinate) {
    candidateSymbolicCoordinate = newSymbolicCoordinate;
    initialSensorValue = getSensorValue();
    startTimer(eventDetectionTimerLength);
}

event timerFired() {
    int finalSensorValue = getSensorValue();
    if (abs(currentSensorValue - finalSensorValue) > eventThreshold) {
        ownSymbolicCoordinate = candidateSymbolicCoordinate;
    }
}

```

Fig. 1. Checking a new symbolic coordinate for applicability

of the two sensor values. If this value is above a specified event threshold, the sensor node detects an event and assigns the new symbolic coordinate.

The method described above relies on two important assumptions. First, it assumes that it is possible for the administrator to change the external conditions in a way that allows all sensor nodes in the room to detect these changes as an event. Second, it also assumes that similar changes to the external conditions in the neighboring rooms are unlikely to happen at the same time without explicit intervention by an external party.

The main advantage of the third approach is that the additional sensor stimulus triggered by the administrator prevents ambiguities in the assignment of the symbolic coordinates. This allows to assign the coordinate to a specific set of nodes without having to address each of the nodes in this set individually. However, this comes at the cost of the additional effort required for generating the external sensor stimulus. Moreover, it only works for actual sensor nodes that possess the sensor chip required for detecting the event. The first two approaches also work on other devices that are part of the sensor network, for example gateway nodes without any sensing functionality.

4 Implementation

We have implemented the three approaches described above for Tmote Sky sensor nodes based on the TinyOS 2.0 operating system. The Tmote Sky sensor nodes provide three different sensor chips: The SHT1x sensor chip from Sensirion for measuring temperature and humidity as well as the two light sensors S1087 and the S1087-01 from Hamamatsu. The first sensor captures the photo-synthetically active radiation (PAR), the other the total solar radiation (TSR). For the evaluation of our third approach we rely on the TSR light sensor as it is better suited for capturing artificial light than the PAR light sensor.

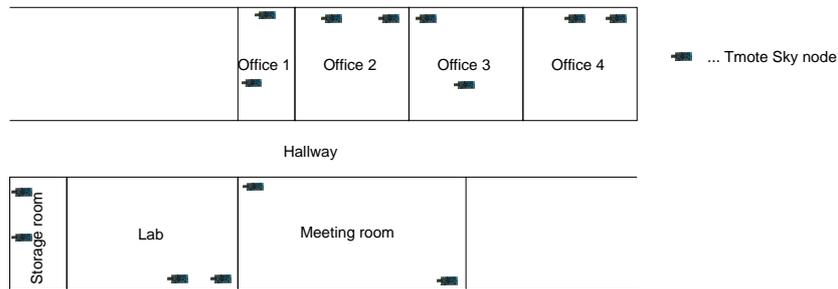


Fig. 2. Floor plan of the experiment area

The client application is implemented in C++ on the Qtopia application platform. It runs on Linux PDAs from Sharp (Sharp Zaurus SL-3200) that communicate with the sensor network using a Tmote Sky sensor node connected to the PDA over USB as a bridge node. In the future, we expect the availability of devices that are able to directly communicate with sensor nodes using a communication standard like IEEE 802.15.4. The client application supports all three approaches presented in this paper. Thus, symbolic coordinates can be sent to individual nodes or broadcasted with or without advertising a following sensor stimulus. The event detection threshold and the event detection timeout can be freely configured.

The output power of the CC2420 radio chips used by the Tmote Sky sensor nodes is programmable. The possible range of values starts with a minimum output power of -25 dBm and goes up to a maximum output power of 0 dBm. The client application allows to select any of the valid output power values.

5 Evaluation

In this section we present an evaluation of our different approaches to the assignment of symbolic coordinates in sensor networks. However, we do not discuss the first approach in detail since the manual assignment of coordinates to individual nodes should work in all cases as long as no packets are lost due to the unreliability of the wireless channel.

For evaluating our approach we deployed 14 sensor nodes in 7 different rooms of the computer science building at the Universität Stuttgart. Fig. 2 shows the floor plan and outlines the location of the nodes in the rooms. Note that we tried to create a somewhat irregular distribution of nodes with different distances among nodes in different rooms.

In a first set of experiments we investigated how reliable the assignment of symbolic coordinates using broadcast works. For this purpose, we sent out coordinate messages in each room with different output power levels and collected information on which nodes received the coordinate message with which RSSI value. We repeated each experiment five times for each signal strength level of the client device.

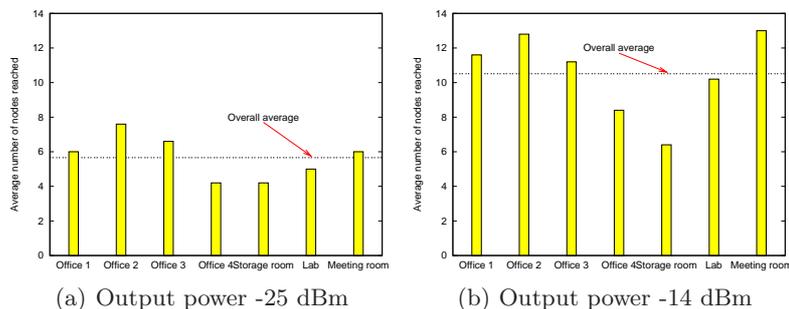


Fig. 3. Average number of nodes reached by coordinate message broadcasts

As a first result, these experiments showed that limiting the dissemination of coordinate messages to a single room is hardly possible even when using the minimum transmission power of the Tmote sensor nodes. To illustrate this, Fig. 3 (a) shows the average number of nodes reached from each room with the minimum output power of approximately -25 dBm and Fig. 3 (b) shows the same analysis for an output power of approximately -14 dBm. Even for the minimum output power, the overall average of 5.66 nodes reached from each area is much higher than the two nodes actually located in each area. On average, each node received coordinate messages from 2.83 different areas (5.26 for an output power of -14 dBm) with a maximum of 5 (7 for an output power of -14 dBm).

To deal with multiple coordinate messages received from different rooms we proposed to consider the RSSI values of the received messages and assign the symbolic coordinate from the message with the highest RSSI value. Using this extension results in a promising performance of the symbolic coordinate assignment using broadcast: Table 1 shows the maximum and the minimum number of nodes with correctly assigned symbolic coordinates for the experiments with signal output powers of -25 dBm and -14 dBm as well as the average percentage of nodes assigned correctly.

Table 1. Average performance for different sender signal strengths

| | Output power -25 dBm | Output power -14 dBm |
|------------------------------------|----------------------|----------------------|
| Max. # of nodes assigned correctly | 13 | 13 |
| Min. # of nodes assigned correctly | 11 | 10 |
| Avg. % of nodes assigned correctly | 88.57 | 77.14 |

Coordinate assignments with the smaller transmission power level clearly outperform the assignment with a higher transmission power level and only assign the wrong coordinate to between one and three nodes. An explanation for this can be found looking at the RSSI values: On average, the RSSI values of messages sent in the same room are only 6% (8% for -25 dBm) larger than

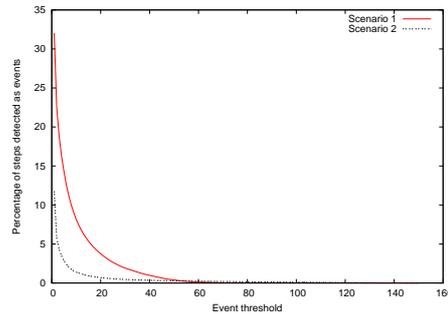


Fig. 4. Percentage of events detected for different event thresholds

the RSSI values of messages sent from different rooms. While the ratio between inside and outside RSSI values is a little higher for -14 dBm than for -25 dBm this cannot compensate the much higher number of coordinate messages each node receives. Due to the inherent variations of the RSSI values of received messages a larger number of coordinate messages received from neighboring rooms increases the probability that the RSSI value of one of these messages is larger than the RSSI value of the message received from the own room.

Overall, while the results of our second approach together with an analysis of the RSSI values produced good results in our experiments our analysis also made clear that RSSI is a fragile criterion that is not able to produce 100% reliable results.

Based on the results of the described experiments we next investigated our third approach – the assignment of symbolic coordinates by assisted broadcast.

First, we wanted to investigate how often TSR light events typically occur when they are not explicitly triggered by the user. For this purpose, we collected sensor data in two indoor scenarios. We distributed 12 nodes to 4 different rooms in both scenarios and collected the value of the TSR light sensor every 10 seconds over multiple days with each sensor node. In the analysis, we evaluated how often an event is detected for two consecutive measurements when varying the event detection threshold. Fig. 4 shows the result of this analysis for both scenarios.

The results show that for very small event thresholds a considerable percentage of measurement pairs triggers events. However, with an event threshold of 10 only in 8.1% (1.4%) and with an event threshold of 20 only in 3.7% (0.7%) of the cases an event has been detected. This indicates a quite small probability of unintended events occurring during the coordinate assignment using these thresholds especially since the event detection time period (10 seconds) was selected quite large in this case. Since changes to the level of illumination of the sensor nodes during the daytime are the main sources of events, the considerable differences between scenario 1 and scenario 2 can be explained by the fact that more nodes in scenario 1 were exposed to sunlight than in scenario 2.

Besides unintentionally detected events a second potential issue is how to trigger light events when the room is already brightly lit by sunlight coming

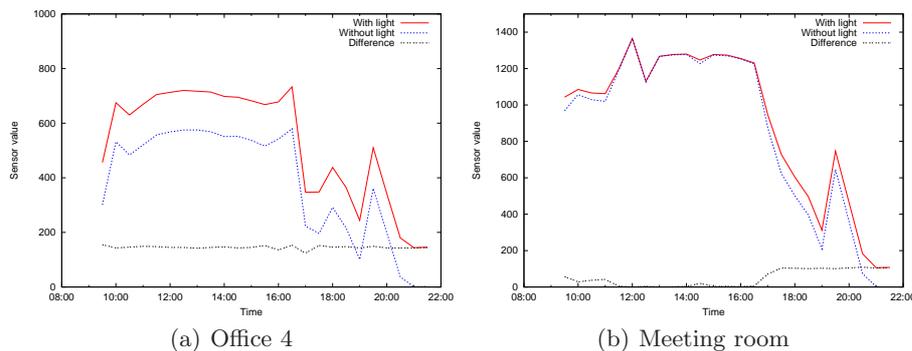


Fig. 5. Light sensor data with and without room light

through the window. To investigate how big of a problem this is we recorded TSR light sensor values in two of our rooms for one day. One recording was done in office 4 which has its windows to the north. To explore more extreme conditions we placed the sensor for the second recording directly behind one of the south-bound windows of the meeting room. Values were recorded every 30 minutes on a sunny day both with the light turned on and with the light turned off. Fig. 5 shows the resulting graphs for both rooms.

The difference between the sensor values recorded with and without light is relatively stable over the day in office 4 (Fig. 5 (a)). With difference values in the range of 140, a reliable detection of light events is possible irrespective of the time of day the coordinate assignment is done. The situation is different for the sensor at the window of the meeting room (Fig. 5 (b)). Here, the influence of the artificial light is considerably lower and the difference between the sensor values shrinks down to values around 0 at noon. Obviously, the artificial light in the room cannot add to the recorded sensor value anymore once a certain light level is reached in the environment. Therefore, the time period of the coordinate assignment must be selected carefully if nodes are placed at especially exposed locations. However, our experiences also show that for normal conditions event detection is possible all day if the event threshold is set to a reasonable value.

The last step of our evaluation is to investigate the results of using the third approach with different event detection thresholds. We performed a set of experiments choosing different event thresholds at different times of the day. In all of these experiments the client application sent the coordinate messages with the minimum possible transmission power and we used an event detection timer length of 4 seconds. Table 2 shows an overview of the results. Note that we deliberately ignored the RSSI values for this evaluation to emphasize the benefit of using events. However, the RSSI value could be used as an additional criterion.

If the event threshold is set to a too small value, then events can occur without being explicitly triggered by a user simply due to the variations of the sensor values over time. The consequence of this are so-called false positives

Table 2. Average success rates for different event thresholds

| Event detection threshold | Avg. % of nodes assigned correctly |
|---------------------------|------------------------------------|
| 5 | 53.57 |
| 10 | 100.0 |
| 20 | 100.0 |
| 100 | 78.57 |

during the assignment of symbolic coordinates – nodes that assign a coordinate in reaction to a coordinate message without lying in the room where the event is actually triggered. In our experiments we could observe this for an event threshold of 5 (see Table 2). The results of experiments performed in the evening or at night with this threshold lay above the average but even then the artificial light oscillated enough to generate some false positives.

To explore the other end of the spectrum, i.e., a high event detection threshold, we performed experiments with a threshold value of 100. As expected, the high event detection threshold reliably prevented the occurrence of any false positives. However, some of the intended recipients also did not detect an event and consequently did not assign the symbolic coordinate resulting in a success rate well below 100%.

With an event threshold of 10 or 20 all of our experiments assigned the correct symbolic coordinates to all nodes in the network irrespective of the time of day the experiment was performed. Therefore, selecting an event threshold in this range provides for a reliable assignment of symbolic coordinates to sensor nodes.

Summarizing the results of our evaluation, assigning symbolic coordinates by broadcast has shown a good performance when used together with RSSI filtering. If, however, a high accuracy is required, then assigning symbolic coordinates by assisted broadcast is able to provide a reliable solution as has been shown by our experiments.

6 Conclusions and Future Work

In this paper we have motivated the benefit of using symbolic coordinates in wireless sensor networks and have discussed the advantages and disadvantages of two canonical approaches for the assignment of such coordinates – the individual assignment and the assignment by broadcast. We have then presented a third solution that combines the advantages of both approaches with only minor additional effort for an externally triggered sensor event. The evaluation shows that this approach allows a simple yet reliable assignment of symbolic coordinates to sensor nodes in indoor scenarios. This way, the manual configuration of symbolic coordinates after the deployment of a sensor network is a viable alternative to more sophisticated node localization approaches.

A reliable indoor localization system for sensor networks that is able to determine the position of nodes without requiring user interaction or expensive

hardware while only generating a low message and computational overhead is definitely desirable. However, while this is not foreseeable, our solution provides a reliable assignment of symbolic coordinate information to sensor nodes that only generates a low overhead and only requires a reasonable amount of support by the user.

As part of future work we plan to investigate other types of sensor stimuli that can be used to disambiguate the assignment of symbolic coordinates to nodes. Particularly, we are interested in looking at combinations of sounders and microphones.

We are also interested in a combination of sensor networks and building automation systems, which automate the control of different mechanical and electrical systems in buildings. This would make it possible to turn on and turn off the lights in the rooms of a building automatically. This way, we could completely automate the assignment of symbolic coordinates by sending out symbolic coordinate messages and then triggering the light sensors in the respective rooms of the building.

Another aspect that we are actively working on is the grouping of nodes based on sensor data. Instead of relying on sensor stimuli deliberately triggered by the user we are interested in analyzing the sensor data collected by sensor nodes as part of their normal operation. Based on a similarity analysis of the sensor data from different nodes we want to decide which nodes reside together in the same room and group these nodes together.

References

1. Langendoen, K., Reijers, N.: Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks* **43**(4) (2003) 499–518
2. Priyantha, N.B., Chakraborty, A., Balakrishnan, H.: The cricket location-support system. In: *Proceedings of the 6th Int. Conf. on Mobile computing and networking.* (2000)
3. Elnahrawy, E., Li, X., Martin, R.P.: The limits of localization using signal strength: A comparative study. In: *Proceedings of The First IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON 2004).* (2004)
4. Stoleru, R., He, T., Stankovic, J.A., Luebke, D.: A high-accuracy, low-cost localization system for wireless sensor networks. In: *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems.* (2005)
5. Corke, P., Peterson, R., Rus, D.: Networked robots: Flying robot navigation using a sensor net. In: *Proceedings of the Eleventh International Symposium of Robotics Research (ISRR).* (2003)
6. Brumitt, B., Shafer, S.: Topological world modeling using semantic spaces. In: *Workshop Proc. of Ubicomp: Location Modeling for Ubiquitous Computing.* (2001)
7. Jiang, C., Steenkiste, P.: A hybrid location model with a computable location identifier for ubiquitous computing. In: *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing, London, UK, Springer-Verlag* (2002) 246–263
8. Becker, C., Dürr, F.: On location models for ubiquitous computing. *Personal Ubiquitous Computing* **9**(1) (2005) 20–31