Topology and Routing in Sensor Networks*

Sándor P. Fekete and Alexander Kröller

Department of Computer Science
Braunschweig University of Technology
D-38106 Braunschweig, Germany
{s.fekete,a.kroeller}@tu-bs.de

Abstract. At ALGOSENSORS 2004 we presented a first algorithm to detect the boundary in a dense sensor network. This has started a new field of research: how to establish topology awareness in sensor networks without using localization. Three years later, at ALGOSENSORS 2007, we present a number of further results.

After discussing issues of distance estimation and computation of coordinates, we give an overview over the boundary recognition problem and show a new approach to solving it. Then we show how to use the boundaries for higher-order topology knowledge; the outcome is a graph that describes the network topology on a very high level, while being small enough to be distributed to all nodes. This allows every node in the network to obtain knowledge about the global topology. Finally, we show how to use these structures for efficient routing.

1 Introduction

In recent time, the study of wireless sensor networks (WSN) has become a rapidly developing research area. Typical scenarios involve a large swarm of small and inexpensive processor nodes, each with limited computing and communication resources, that are distributed in some geometric region; communication is performed by wireless radio with limited range. Upon start-up, the swarm forms a decentralized and self-organizing network that surveys the region.

From an algorithmic point of view, these characteristics imply absence of a central control unit, limited capabilities of nodes, and limited communication between nodes. This requires developing new algorithmic ideas that combine methods of distributed computing and network protocols with traditional centralized network algorithms. In other words: how can we use a limited amount of strictly local information in order to achieve distributed knowledge of global network properties? As it turns out, making use of the underlying geometry is essential.

In this paper, we give an overview of topics and results discussed and presented during an invited presentation at ALGOSENSORS 2007. Section 2 deals with location awareness; we start by describing distance estimation without the use of special hardware; this is followed by a discussion of the limitations of the

* This invited survey article is based in parts on excerpts from our articles [6,7,8,15].

M. Kutyłowski et al. (Eds.): ALGOSENSORS 2007, LNCS 4837, pp. 6–15, 2008.

computation of node coordinates. Section 3 gives a description of our approach to topology recognition, consisting of boundary recognition and topological clustering, which has been turned into a video, based on large-scale simulation. Section 4 describes some new approaches to routing.

2 Location Awareness

One of the key problems in sensor networks is to let nodes know their location, for example, by storing coordinates w.r.t. a global coordinate system. Unless all nodes are equipped with special localization devices (e.g., GPS/Galileo), there needs to be an algorithm that computes positions based on information available to the network.

2.1 Distance Estimation

Practical localization algorithms often use connectivity information enriched with distance estimates for adjacent nodes [17]. Note that the corresponding decision problem is NP-hard [1].

Various ways to measure distance exist. Examples include the transmission time-of-flight over a wireless channel, the latency of infrared communication, or the strength of a wireless signal that decreases with distance. Good approaches have an average error of about 10–20% of the maximal communication range.

Our approach does not rely on special hardware or node capabilities. Assuming the probability of successful communication decreases with increasing distance, the expected fraction of a node's neighbors that it shares with an adjacent node defines a monotonically decreasing function that can be inverted, resulting in a distance estimator based on this fraction. All that is required is the ability to exchange neighbor lists and a model of communication characteristics.

We assume that nodes are uniformly distributed over the plane, with density δ. That is, the expected number of nodes in a region $A \subset \mathbb{R}^2$ of area $\lambda(A)$ equals $\delta\lambda(A)$. The neighborhood N_i of a node i depends on communication characteristics, which are modelled by an appropriate communication model. We focus on symmetric models only, i.e., $i \in N_j$ iff $j \in N_i$. The model is a probability function $p(d)$ that defines the probability that two nodes i and j with distance $d = \|i - j\|$ are connected. Hence, the expected size of a neighborhood is $\mathbb{E}[|N_i|] = \delta \int_{\mathbb{R}^2} p(\|x\|)dx$ for all nodes i.

We want to estimate the distance of i and j by counting how many of i's neighbors are shared with j. The expected size of this fraction is

$$\begin{aligned} f_p(d) &:= \mathbb{E}[|N_i \cap N_j|/|N_i \setminus \{j\}|] \qquad (1) \\ &= \frac{\int_{\mathbb{R}^2} p(\|x\|)p(\|x - (d,0)^{\mathsf{T}}\|)dx}{\int_{\mathbb{R}^2} p(\|x\|)dx}, \end{aligned}$$

where $d = \|i - j\|$. If f_p^{-1} exists, two nodes i and j can exchange their neighbor lists, compute the shared fraction $\varphi_{i,j} = |N_i \cap N_j|/|N_i \setminus \{j\}|$ and estimate their

Table 1. Average estimation errors for different densities

Scaled density $\pi\delta$	5	8	10	15	20	40	80
Error (inner nodes)	.225	.183	.165	.137	.120	.087	.062
Error (boundary nodes)	.257	.201	.182	.154	.135	.101	.077

distance as $f_p^{-1}(\varphi_{i,j})$. Note that $\varphi_{i,j}$ and $\varphi_{j,i}$ may be different, so some additional tie breaking or averaging scheme must be used.

There is an elegant way to implement this approach for practical purposes, as proposed by Buschmann et al. [3]: Instead of f_p^{-1}, a small discrete value table of f_p is stored in the nodes, and the estimate is done by reverse table lookup. This even works for p or f_p obtained by numerical or field experiments, and it can be implemented using only integer arithmetic.

A widely used model for radio networks is the Unit Disk Graph (UDG), where two nodes i and j are connected by a link iff $\|i-j\| \leqslant 1$.

For UDGs, the estimated neighborhood fraction (1) is $f : [0,1] \to [0,1]$ with

$$f(d) = \frac{2}{\pi}\left(\arctan(\tfrac{d}{2}) - \tfrac{d}{2}\sqrt{1-(\tfrac{d}{2})^2}\right) . \tag{2}$$

f^{-1} exists, but unfortunately we lack a closed formula for it. Instead, we approximate f^{-1} by its Taylor series about $f(0) = 1$. Here, we use the 7th-order Taylor polynomial

$$\begin{aligned} t_7(\varphi) = &-\frac{\pi}{1!\,2}(\varphi-1) - \frac{\pi^3}{3!\,2^5}(\varphi-1)^3 \\ &-\frac{13\pi^5}{5!\,2^9}(\varphi-1)^5 - \frac{491\pi^7}{7!\,2^{13}}(\varphi-1)^7. \end{aligned} \tag{3}$$

We do not use a higher order because evaluating the polynomial on practical embedded systems would become numerically unstable.

To evaluate the UDG estimator's performance, we ran some simulations. Table 1 shows their results. The first row contains the expected size of a neighborhood, without boundary effects. For UDGs, this is $\pi\delta$. Furthermore, the average errors are reported. The error is relative to the communication range, which is the common measure for distance estimators. The average is taken separately for two classes of links: for "inner" links, the communication ranges of both end-nodes are fully contained in the network region. For "boundary" links, both end-nodes lie at most 1 from a straight boundary. This separation has two benefits: First, the estimate in its current form focuses on the inner links only, and second, it removes the dependency on the network region's shape from the evaluation.

One can see how our approach already reaches the desired accuracy of 20% for an average neighborhood size of just eight, and gets even better for larger neighborhood size.

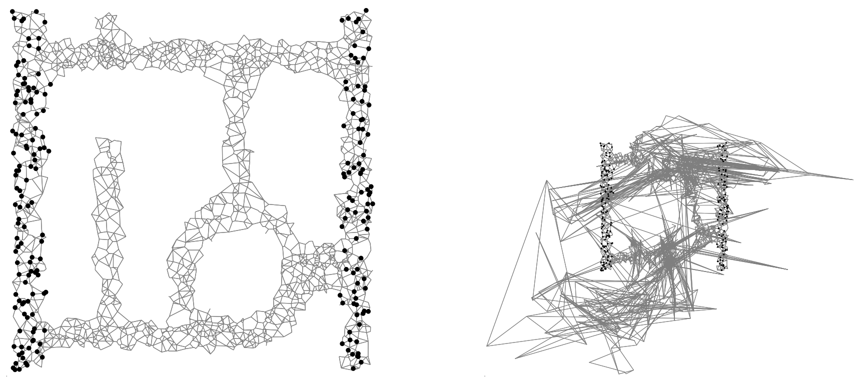

Fig. 1. Left: Example network with marked anchor nodes. Right: Result of Ad-Hoc Positioning [18].

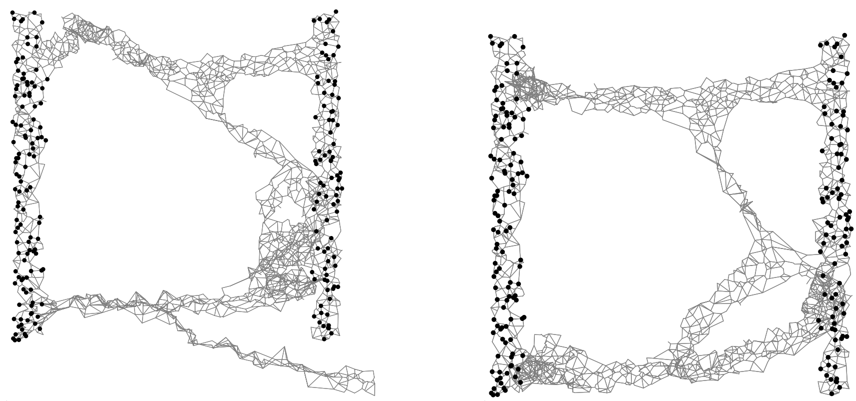

Fig. 2. Left: Result of Robust Positioning [20]. Right: Result of N-Hop Multilateration [21].

2.2 Coordinates

When trying to understand the network structure, a seemingly natural approach is to determine the underlying node coordinates, based on trigonometric computations that use node distances. A variety of methods have been proposed, including [18, 20, 21, 19]. Unfortunately, these methods show a number of problems in the presence of errors and large numbers of nodes: see our Figures 1, 2, 3 for an example with 2200 nodes, with a subset of "anchor" nodes that know their precise coordinates marked in black; node distances have an a random error with a standard deviation of 1%.

Quite clearly, the resulting embeddings do not only produce imprecise node coordinates; more importantly, they do not accurately reflect the network topology. This means that in the context of exploiting the network topology for

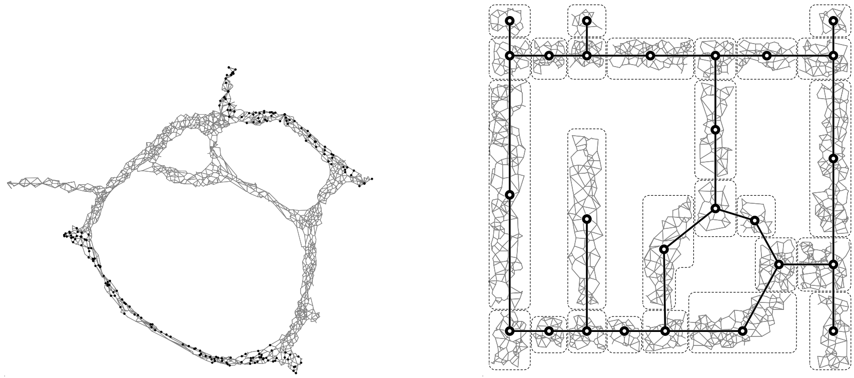

Fig. 3. Left: Result of Anchor-Free Localization [19]. Right: Our alternative: clusters and cluster graph.

purposes such as routing, computing more or less accurate coordinates is indeed a red herring; as will show in the following, a suitable alternative is to consider topological clustering, as motivated in Figure 3. See our article [15] for more technical details.

3 Topology Awareness

3.1 Boundary Recognition

Recognizing the network boundary is vital for detecting objects entering or leaving the monitored area or events that affect the network structure. Boundary detection is also a stepping stone towards organizing the network.

In the setting described above, we first proposed the problem in [9], using a probabilistic setting; see [5] for a refined approach. A different approach for our problem was suggested by Funke [11], and requires a particular boundary structure and sufficient density; see [12] for details. Another approach was proposed by Wang et al. [22]. Our method described in [14] yields deterministically provable results for any kind of boundary structure. We assume that the communication graph is a $\sqrt{2}/2$-quasi unit disk graph, a generalization of unit disk graphs first introduced by [2].

See Figure 4 for a geometric representation of a graph called a *flower*. Non-neighboring nodes have a minimum distance, so an independent node set requires a certain amount of space in the embedding. On the other hand, the space surrounded by a cycle is limited. This packing argument allows conclusions about the relative embedding of nodes. Applying a similar argument repeatedly, the central nodes can deduce that they lie inside of the outer cycle.

Because flowers are strictly local structures, they can be easily identified by local algorithms. In the shown example network of 60,000 sparsely connected nodes, our procedure identified 138 disjoint flowers; a single suffices for the second stage of our algorithm.

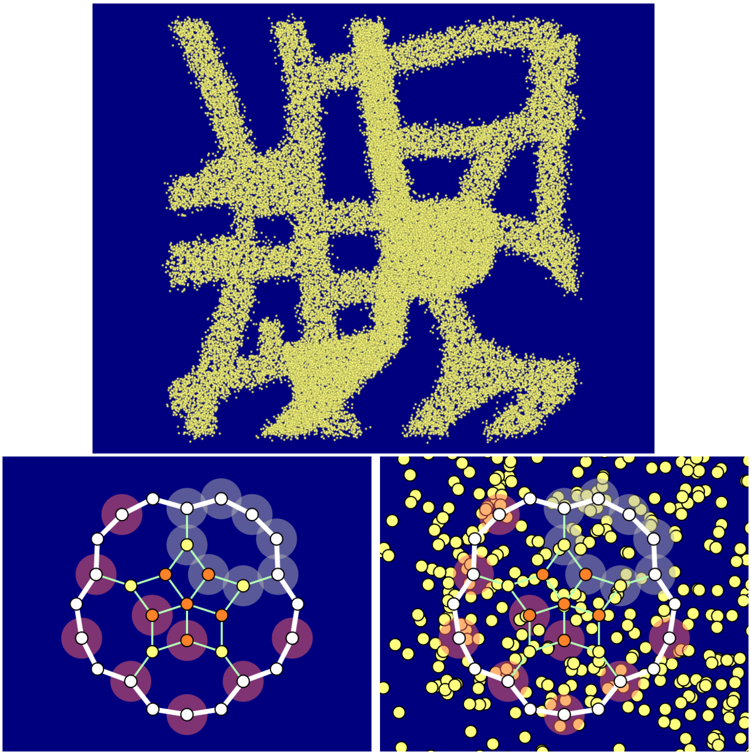

Fig. 4. Top: A sensor network, obtained by scattering 60,000 nodes in a of street network. Bottom left: A flower subgraph, used for reasoning about boundary and interior of the network. Right: A flower in the context of the network shown above.

In this second stage (cf. Figure 5), we augment flowers by adding extensions to their outer cycles, such that insideness can still be proven for all contained nodes. By repeating a local search procedure, the flowers grow to enclose more and more nodes and merge together, eventually leading to a single structure that contains most of the network.

3.2 Topological Clustering

We use the identified boundaries to construct a topological clustering; see Figure 6. By considering the hop count from the boundary, we get a shortest-path forest. Nodes that have almost the same distance to several pieces of the boundary form the medial band of the region. Nodes close to three different boundary portions form vertices of the medial band, or medial vertices. After identifying a set of medial vertices, we also know their distance from the boundary. This makes it easy to grow the corresponding intersection cluster to just the right

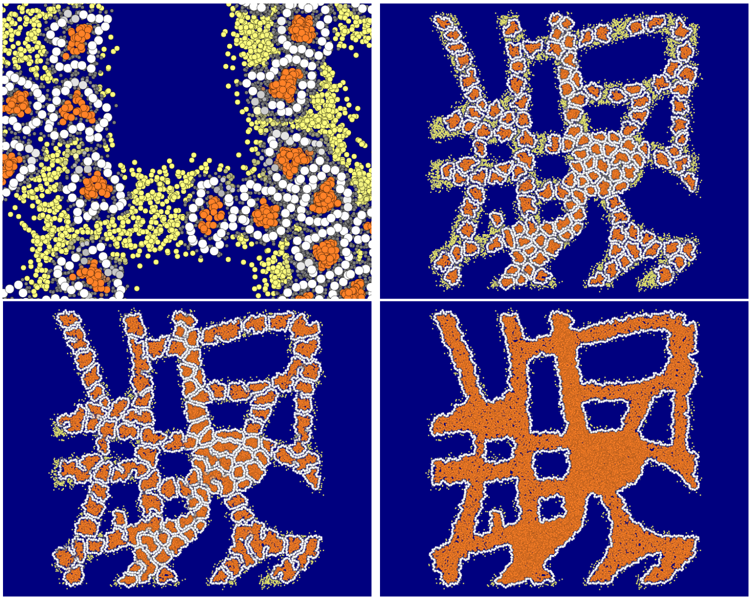

Fig. 5. Top left: A set of expanding flowers in a network. Top right: A later stage of the flower expansion. Bottom left: Flowers merging together. Bottom right: The result of boundary recognition.

size. Thus, we can identify all intersection clusters in the network. Finally, parts of the network adjacent to intersection clusters give rise to street clusters.

In the end, we have structured the network into a natural set of clusters that reflect its topology. This makes it is possible to perform complex tasks, such as tracking and guiding, based on purely local operations.

3.3 Our Video

Using our toolbox Shawn [16] for the simulation of large and complex networks, we have produced a video that illustrates two procedures for dealing with the above algorithmic challenges: one identifies the boundaries of the network; the other constructs a clustering that describes the network topology. For more technical details of the underlying algorithmic side, see our paper [14]. Our software is freely available at `www.sourceforge.net/projects/shawn`.

The video starts by describing sensor nodes and their deployment. Following an introduction of the algorithmic challenge, the next scene illustrates the problem of boundary recognition. Next is the concept of flowers and the way they allow local, deterministic reasoning about nodes lying in the interior of the network. Their extension by augmenting cycles is demonstrated in the following scene, leading to full-scale boundary recognition. The next part introduces the medial band and the recognition of medial vertices. In the final sequence, this leads to the construction of clusters.

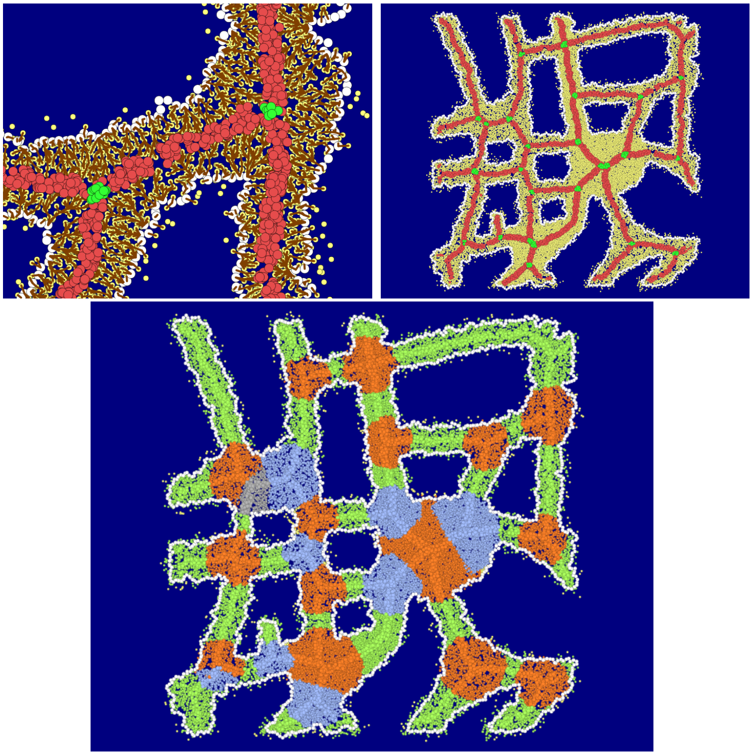

Fig. 6. Top left: The shortest-path forest defines the medial band. Top right: Medial band and medial vertices. Bottom: The resulting intersection and street clusters.

4 Routing

Once the above cluster structure has been extracted, it is straightforward to use it for network tasks such as routing and tracking: on the large scale, use the cluster graph G, either on a global scale (if G is small enough to be available all over the network), or for local handover between clusters (if G is not available); within each cluster, use virtual coordinates that arise (a) by hop distance from the cluster boundary (b) by hop distance along the cluster boundary. Using more refined geometric structures and properties, it is possible to achieve good routing results. See the forthcoming Ph.D. thesis [13] of the second author for details.

A related problem is to find approximately shortest paths in a clustered sensor network, without having access to node coordinates or a detailed routing table. For the case of a subdivision into convex clusters, our group has developed a memory-efficient, local algorithm that guarantees a 5-approximation of the shortest routing path. See the masters thesis by Förster [10], and our related paper [4] for details.

Acknowledgments

Parts of this work presented in this survey were based on a long-standing, very fruitful collaboration with Stefan Fischer, Dennis Pfisterer and Carsten Buschmann from University of Lübeck. We thank our colleagues for making it possible. The work by Alexander Kröller was part of project "SwarmNet" within the DFG priority programme SPP 1126, "Algorithms for large and complex networks", grant numbers FE 407/9-1 and FE 407/9-2.

References

1. Aspnes, J., Goldenberg, D., Yang, Y.R.: On the computational complexity of sensor network localization. In: Nikoletseas, S.E., Rolim, J.D.P. (eds.) ALGOSENSORS 2004. LNCS, vol. 3121, pp. 32–44. Springer, Heidelberg (2004)
2. Barrière, L., Fraigniaud, P., Narayanan, L.: Robust position-based routing in wireless ad hoc networks with unstable transmission ranges. In: DIALM 2001, Rome, Italy, pp. 19–27. ACM Press, New York, NY, USA (2001)
3. Buschmann, C., Hellbrück, H., Fischer, S., Kröller, A., Fekete, S.P.: Radio propagation-aware distance estimation based on neighborhood comparison. In: Langendoen, K., Voigt, T. (eds.) EWSN 2007. LNCS, vol. 4373, pp. 325–340. Springer, Heidelberg (2007)
4. Fekete, S.P., Förster, K.-T., Kröller, A.: Using locally optimal paths for memory-efficient routing in sensor networks, manuscript (2007)
5. Fekete, S.P., Kaufmann, M., Kröller, A., Lehmann, N.: A new approach for boundary recognition in geometric sensor networks. In: Proceedings 17th Canadian Conference on Computational Geometry, pp. 82–85 (2005)
6. Fekete, S.P., Kröller, A.: Geometry-based reasoning for a large sensor network. In: Proc. 22nd Annual ACM Symposium on Computational Geometry, pp. 475–476 (2006), `http://compgeom.poly.edu/acmvideos/socg06video/index.html`
7. Fekete, S.P., Kröller, A., Buschmann, C., Fischer, S.: Geometric distance estimation for sensor networks and unit disk graphs. In: Proceedings 16th International Fall Workshop on Computational Geometry (2006)
8. Fekete, S.P., Kröller, A., Pfisterer, D., Fischer, S.: Algorithmic aspects of large sensor networks. In: Proceedings Mobility and Scalability in Wireless Sensor Networks, pp. 141–152. CTI Press (2006)
9. Fekete, S.P., Kröller, A., Pfisterer, D., Fischer, S., Buschmann, C.: Neighborhood-based topology recognition in sensor networks. In: Nikoletseas, S.E., Rolim, J.D.P. (eds.) ALGOSENSORS 2004. LNCS, vol. 3121, pp. 123–136. Springer, Heidelberg (2004)
10. Förster, K.-T.: Clusterbasierte Objektüberwachung in drahtlosen Sensornetzwerken: Lokal optimale Wege in der Ebene und ihre Anwendung in Sensornetzwerken. Master's thesis, Braunschweig University of Technology, Braunschweig (2007)
11. Funke, S.: Topological hole detection in wireless sensor networks and its applications. In: Proc. DIALM-POMC, pp. 44–53 (2005)
12. Funke, S., Klein, C.: Hole detection or: how much geometry hides in connectivity? In: Proc. 22nd ACM Symp. Comput. Geom., pp. 377–385 (2006)
13. Kröller, A.: Algorithms for topology-aware sensor networks. PhD thesis, Braunschweig University of Technology, Braunschweig (2007)

14. Kröller, A., Fekete, S.P., Pfisterer, D., Fischer, S.: Deterministic boundary recognition and topology extraction for large sensor networks. In: Proc. 17th ACM-SIAM Sympos. Discrete Algorithms, pp. 1000–1009 (2006)
15. Kröller, A., Fekete, S.P., Pfisterer, D., Fischer, S., Buschmann, C.: Koordinatenfreies Lokationsbewusstsein. Information Technology 47, 70–78 (2005)
16. Kröller, A., Pfisterer, D., Buschmann, C., Fekete, S.P., Fischer, S.: Shawn: A new approach to simulating wireless sensor networks. In: DASD 2005. Proceedings Design, Analysis, and Simulation of Distributed Systems, pp. 117–124 (2005)
17. Langendoen, K., Reijers, N.: Distributed localization in wireless sensor networks: A quantitative comparison. Computer Networks 43(4), 499–518 (2003)
18. Niculescu, D., Nath, B.: Ad-hoc positioning system (APS). In: Proceedings of the 44th IEEE Global Telecommunications Conference (GLOBECOM 2001), pp. 2926–2931 (November 2001)
19. Priyantha, N.B., Balakrishnan, H., Demaine, E., Teller, S.: Anchor-free distributed localization in sensor networks. In: Proc. 1st Intl. Conf. on Embedded Networked Sensor Systems (SenSys), Los Angeles, CA, pp. 340–341. ACM Press, New York (2003)
20. Savarese, C., Rabay, J., Langendoen, K.: Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In: Proc. USENIX Technical Ann. Conf., Monterey, CA (2002)
21. Savvides, A., Park, H., Srivastava, M.B.: The bits and flops of the n-hop multilateration primitive for node localization problems. In: Proc. 1st ACM Intl. Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA (2002)
22. Wang, Y., Gao, J., Mi