Hop-Count based Node-to-Anchor Distance Estimation in Wireless Sensor Networks

Di Ma, Bang Wang, Hock Beng Lim, Meng Joo Er
Intelligent Systems Center, Nanyang Technological University, Singapore.
{madi0002, wangbang, limhb, emjer}@ntu.edu.sg

***Abstract*—Localization is one of the most important research issues in *Wireless Sensor Networks* (WSNs). Hop-count based localization has been proposed as a cost-effective alternative to the expensive hardware-based localization schemes. In this paper, we propose a new method to estimate distances between anchors and nodes based on hop-count information. Localization is then achieved using the estimated distances. Simulation results show that the performance of our proposed algorithm is much better than that of the DV-hop algorithm [1] in terms of the node-to-anchor distance estimation and localization accuracy, and the improvement is proportional to the node density.**

***Index Terms*—localization, Wireless Sensor Networks, hop-count based localization**

I. Introduction and related work

Localization in WSNs is an active research area and many localization techniques and algorithms have been proposed in the literature. Determining the node-to-anchor distance is normally the first step for most of these localization algorithms [2]. Some techniques make use of special hardware to obtain the node-to-anchor distance. For example, in the time difference of arrival (TDOA) localization algorithm, two types of transceivers, namely radio and acoustic, are needed. There are also localization algorithms that use only the connectivity information to estimate the node-to-anchor distance. Although the latter algorithms may not achieve very high localization accuracy, their cost-effectiveness is advantageous for applications in large-scale and resource-constrained sensor networks. This paper proposes a new method to estimate distances between anchors and nodes based only on hop-count information.

Hop-count based localization algorithms can be broadly classified as being centralized or distributed. A centralized algorithm relies on a central node to compute the locations of all nodes, whereas a distributed algorithm relies on mutual communication between nodes. We briefly review some of the existing methods in the literatures.

The DV-hop algorithm [1] might be the first distributed hop-count based localization algorithm. The basic idea of the DV-hop algorithm is to use the hop-counts between anchors (nodes with known positions) to estimate a *correction factor* (meter/hop), and then use the correction factor to estimate the distances between anchors and unknown position nodes. Several extensions of DV-hop have been proposed to improve its performance. Savarese et al. proposed a modified DV-hop algorithm [3] consisting of two phases; namely, start-up and refinement phase. For the start-up phase, they use Hop-Terrain to generate a rough initial estimation of the nodes' locations. In the refinement phase, a refinement algorithm runs iteratively to continuously improve the position estimation online.

In [4], Dulman et al. improved the DV-hop algorithm by applying the statistics of the hop-distance in a uniformly distributed network in the correction factor computation. On the other hand, the DHL (density-aware hop-count localization) algorithm by Wong et al. [5] considers the case of non-uniform node distributions and adjusts the correction factor according to local node densities. The HCRL (hop-counts-ratio based localization) algorithm [6] uses only the ratio of node-to-anchor hop-counts to estimate the distances in order to save the message overhead used in the DV-hop algorithm for disseminating anchors' information.

One of the popular centralized methods is known as the MDS [7] (multi-dimensional scaling) technique. This method has its origins in psychometrics and psychophysics. It can be seen as a set of data analysis techniques that display the structure of distance-like data as a geometrical picture [8]. The convex constraint satisfaction approach of [9] formulates the localization problem as a feasibility problem with radial constraints. Nodes which can hear each other are constrained to lie within a certain distance of each other. The convex constraint problem is in turn solved by efficient semi-definite programming to find a global optimal solution.

This paper proposes a new method, named HCNP (*Hop-Count based Neighbor Partition*), to partition neighbor nodes into disjoint sets by exploiting the hop-count information, and then apply geometric characteristics to estimate the distances between anchors and nodes. Our algorithm is first described in Section II and then evaluated via simulations in Section III. Section IV concludes the paper by pointing out some future research directions.

II. Hop Count based Neighbor Partition for Node-to-Anchor Distance Estimation

A. System model

As that in the DV-hop algorithm and its extensions, we also adopt a unit disk communication model. A node can communicate with all nodes within a disk centered at itself with radius the communication range r, and cannot communicate with the nodes beyond the disk. We assume that nodes and anchors are randomly deployed according to a uniform distribution and can form a connected network. Let N and M denote the number of nodes and anchors, respectively

978-1-4244-2309-5/09/$25.00 ©2009 IEEE

($M \ll N$ usually). There are two reasons we adopt an uniform nodes distribution model: Firstly, uniform distribution of nodes has been assumed in many well-known localization algorithms (such as APIT [10], convex position estimation [9], etc.). The uniform nodes distribution model might not capture all sensor deployment scenarios, however, it is certainly the most appropriate placement model for a large-scale WSN (Eg. thousands of nodes been dropped from an airplane). Secondly, many of the theoretical work in relating hop-count and distance are also based on the assumption of uniform distribution of nodes (see [11] and [12]).

B. Hop count based neighbor partition

In the hop-count based node-to-anchor distance estimation, an anchor needs to set up the hop-counts relative to itself for all other nodes by a controlled flooding (see e.g., [13]). Let h denote the hop-count of a node with respect to an anchor, and $h = 0$ for the anchor and $h > 0$ for other nodes. Actually, the hop-count information provides a natural geometric division of the network field. We can use concentric circles to particiipation the field into different concentric rings all centered at the anchor. A ring labeled as h contains all nodes with hop-count h. Let d^h_{min} and d^h_{max} denote, respectively, the minimum and maximum node-to-anchor distance among all the nodes with hop-count h. The ring h is determined by two concentric circles, the inner one with radius d^h_{min} and the outer one with radius d^h_{max}. In a network with finite nodes, the radius of the outer circle of the ring h normally is not equal to the radius of the inner circle of the ring $h+1$, i.e., $d^h_{max} \neq d^{h+1}_{min}$. Furthermore, the hop boundary distance C normally is not equal to $d^h_{max} - d^h_{min}$, i.e., $C \neq d^h_{max} - d^h_{min}$ and not the same across different rings.

Fig. 1 illustrates the hop count of 599 randomly distributed nodes in a $[0, 60] \times [0, 60]$ square field to the anchor located at the center of the field. It is observed that in such a network the maximum hop count is 5. Furthermore, it is clearly seen that in such a network, $d^h_{max} \neq d^{h+1}_{min}$ and $d^h_{max} - d^h_{min} \neq d^{h+1}_{max} - d^{h+1}_{min}$.

The following *perfect hopping* definition gives a scenario where the statements $d^h_{max} = d^{h+1}_{min}$ and $C = d^h_{max} - d^h_{min}$ for all h become true almost surely:

There exists a distance C^j ($C^j \leq r$) for each anchor j, such that the distance of a node to the anchor j is bounded in-between $[C^j \times (h^j - 1), C^j \times h^j]$, if and only if this node has a hop-count h^j ($h^j \geq 1$).

In such a perfect hopping scenario, the hop 1 nodes are uniformly distributed in $[0, C^j]$; the hop 2 nodes are uniformly distributed in $[C^j, 2C^j]$; and so on. We call C^j as the *hop boundary distance* in the following discussions. It is important to note that the probability of a network having perfect hopping tends to 1 only when the node density is close to infinite and in such a case, $C = r$. The DV-hop algorithm also applies the perfect hopping scenario and computes the correction factor as the hop boundary distance. In what follows, we design our algorithm based on the perfect hopping scenario and evaluate its performance in network with finite nodes.

For a node with hop-count $h \geq 1$ to an anchor, we can partition the node's neighbors into three disjoint sets based on their hop-counts to the same anchor, namely, those neighbors with hop-count $h-1$, h, and $h+1$. Let S_i denotes the set of neighbors for the node i and let S^{h-1}_i, S^h_i and S^{h+1}_i denote the sets of its neighbors with hop-count $h-1$, h and $h+1$, respectively. The following relationships hold for all i,

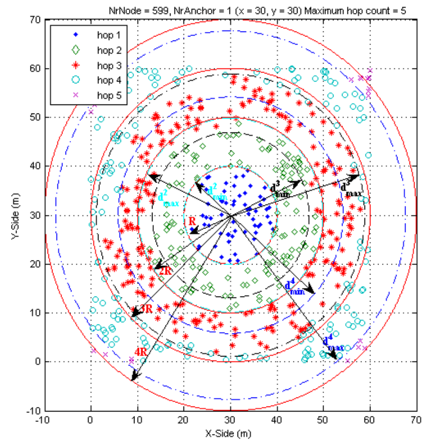

Fig. 1. Distribution of nodes in different hops shows that $d^h_{max} \neq d^{h+1}_{min}$ and $C \neq d^h_{max} - d^h_{min}$

$$S_i = S^{h-1}_i \bigcup S^h_i \bigcup S^{h+1}_i$$

$$S^{h-1}_i \bigcap S^h_i = \varnothing,\ S^h_i \bigcap S^{h+1}_i = \varnothing,\ S^{h-1}_i \bigcap S^{h+1}_i = \varnothing$$

Note that such neighbor partition is a by-product of the controlled flooding for hop count setup and does not require additional message overhead.

C. Intersection area estimation

For a node with hop-count h, its communication disk intersects with three rings, namely the ring $h-1$, h and $h+1$, and creates 3 disjoint regions, as illustrated in Fig. 2 where the anchor is represented by the triangle and the hop boundary distance is C^j. Let a^{h-1}_i, a^h_i and a^{h+1}_i denote the intersection regions, respectively. Obviously, all nodes within a^{h-1}_i (a^h_i/a^{h+1}_i) are the $h-1$ neighbors of the node i and hence form the set S^{h-1}_i (S^h_i/S^{h+1}_i). Furthermore, it is clearly seen from Fig. 2 that the areas of the 3 regions is dependent on the distance between the node and the anchor. In the figure, the node i is closer to the anchor than the node k and has a larger intersection area with the ring $h-1$ but a smaller intersection area with the ring $h+1$. If the node-to-anchor distance is known, we can compute the areas of the 3 intersection regions. On the other hand, if we find their areas, we then can compute the node-to-anchor distance d easily using geometric equations.

We approximate the area of an intersection region by the number of nodes falling into that region. This resembles the Hit-or-Miss Monte-Carlo integration method given the uniform distribution assumption, and the approximation becomes more accurate when the node density increases [14]. Let n_i denote the number of neighbors of node i, and n^{h-1}_i, n^h_i and n^{h+1}_i be the number of nodes that fall in the intersection region a^{h-1}_i,

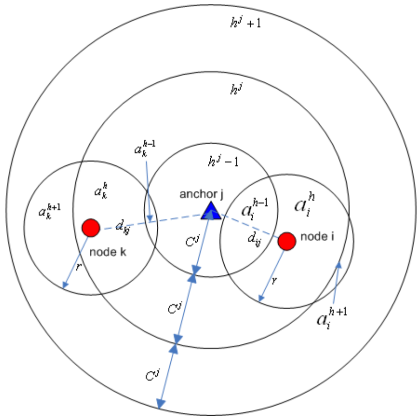

Fig. 2. Intersection area estimation.

a_i^h and a_i^{h+1}. Since all nodes are uniformly distributed in a $l \times l$ (m^2) field, the areas of the 3 intersection regions are estimated as follows:

$$Area(a_i^{h-1}) = \widetilde{A}_i^{h-1} = \frac{n_i^{h-1}}{N} \times l^2$$
$$Area(a_i^{h}) = \widetilde{A}_i^{h} = \frac{n_i^{h}+1}{N} \times l^2$$
$$Area(a_i^{h+1}) = \widetilde{A}_i^{h+1} = \frac{n_i^{h+1}}{N} \times l^2$$

D. Node-to-Anchor distance estimation

The analytical form of the intersection area is given by

$$\begin{aligned} A_i^{h-1} &= r^2\cos^{-1}\left(\frac{d^2+r^2+(hC)^2}{2dr}\right) \\ &+ (hC)^2\cos^{-1}\left(\frac{d^2+(hC)^2-r^2}{2dCh}\right) \\ &- \frac{1}{2}\sqrt{4d^2(hC)^2-(d^2-r^2+(hC)^2)^2} \end{aligned}$$

$$\begin{aligned} A_i^{h} &= r^2\cos^{-1}\left(\frac{d^2+r^2+((h+1)C)^2}{2dr}\right) \\ &+ ((h+1)C)^2\cos^{-1}\left(\frac{d^2+((h+1)C)^2-r^2}{2d((h+1)C)}\right) \\ &- \frac{1}{2}\sqrt{4d^2((h+1)C)^2-(d^2-r^2+((h+1)C)^2)^2} - A_i^{h-1} \end{aligned}$$

$$\begin{aligned} A_i^{h+1} &= \pi r^2 - r^2\cos^{-1}\left(\frac{d^2+r^2+((h+1)C)^2}{2dr}\right) \\ &+ ((h+1)C)^2\cos^{-1}\left(\frac{d^2+((h+1)C)^2-r^2}{2d((h+1)C)}\right) \\ &- \frac{1}{2}\sqrt{4d^2((h+1)C)^2-(d^2-r^2+((h+1)C)^2)^2} \end{aligned}$$

We use the estimated areas of the intersection regions to compute the node-to-anchor distance d. Let f denote the nonlinear function in the above equations, we use the secant method [15] to solve d numerically as follows:

$$d_{n+1} = d_n - \frac{d_n - d_{n-1}}{f(d_n) - f(d_{n-1})} f(d_n),$$

where the first iteration d_0 is set to a reasonable value(eg: $0.5r$). If $\widetilde{A}_i^{h-1}$, $\widetilde{A}_i^{h}$ and $\widetilde{A}_i^{h+1}$ are the perfect estimations of A_i^{h-1}, A_i^{h} and A_i^{h+1}, respectively, then the solutions of d from these equations should be the same. That is $d^{h-1} = d^h = d^{h+1}$. However, since the area estimation is not perfect, we use the mean of them as the estimation of the node-to-anchor distance.

$$d = \frac{d^{h-1} + d^h + d^{h+1}}{3}$$

After obtaining the node-to-anchor distance, the location then is computed by the normal multilateration method.

E. Implementation issue

We have presented the HCNP algorithm under the perfect hopping assumption. In order to implement the HCNP algorithm, the hop boundary distance must be known. However, the perfect hop boundary does not exist in an imperfect hopping network. This can be seen in Fig. 3, the distance distribution for adjacent hops overlap with each other, indicating an imperfect hopping scenario. To address this issue, we introduce a pseudo-hop boundary distance to replace the perfect hop boundary in the algorithm implementation. As shown in Fig. 4, we use attenuated Gaussian distribution [16] to model the conditional pdf $f(d|h)$, where d represents the distance and h represents the hop-count number.

$$f(d|h_k) = \alpha^k N(m_k, \sigma_k),$$

Where α is the attenuation coefficient less than 1, m_k and σ_k are the mean and standard deviation for each Gaussian, k is the index of hop-count. The solution of d for $P(d|h_k) = P(d|h_{k+1})$ is thus the pseudo-hop boundary distance (dotted line separating adjacent Gaussian distribution in Fig. 4. We could always obtain the pseudo-hop boundary using Monte-Carlo simulation if we known the statistic distribution of nodes. In case Monte-Carlo simulation is not feasible or distribution is unknown, we could still use a heuristic method to compute pseudo-hop boundary distance $\hat{C}^j$ to approximate the perfect hop boundary distance.

Let $\bar{d}_1$, $\bar{d}_2$,...,$\bar{d}_h$ represent the mean node-to-anchor distance for hop 1, 2,..., h nodes. The pseudo-hop boundaries for hop 1, 2,..., h are constructed as $[0, D_1]$, $[D_1, D_2]$,..., $[D_{h-1}, D_h]$ where $D_2 - D_1 = 2\bar{d}_2$, $D_3 - D_2 = 2\bar{d}_3$,..., $D_h - D_{h-1} = 2\bar{d}_h$.

In short, we seek a pseudo-hop boundary that bounds as much as possible nodes from the same hop into the same gap by reconstructing the region of uniform distribution using mean node-to-anchor distance. We use the DV-hop algorithm to estimate the mean node-to-anchor distance in the implementation. The reason we call it "pseudo" is because hop nodes are not necessarily all bounded in $[D_{h-1}, D_h]$. Rather, uniform distribution of nodes in this bound will give back the same mean node-to-anchor distance.

III. Simulation studies

A. Simulation settings

The area of the square field is set to be 100×100 m^2. The communication range r is set to $10m$. We vary the

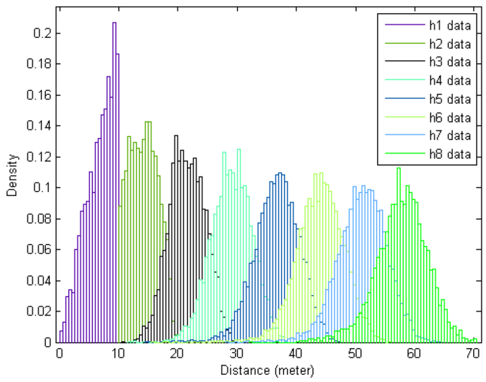

Fig. 3. Distribution of distance in each hop in a 450 nodes network

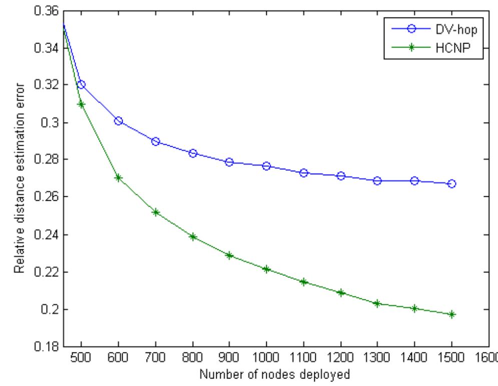

Fig. 5. Distance estimation error with varying parameter: number of nodes.

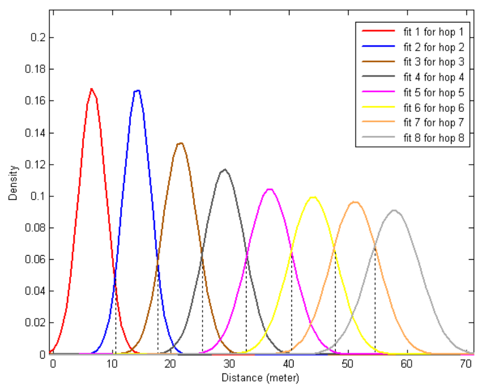

Fig. 4. Attenuated Gaussian modelling of $f(d|h)$.

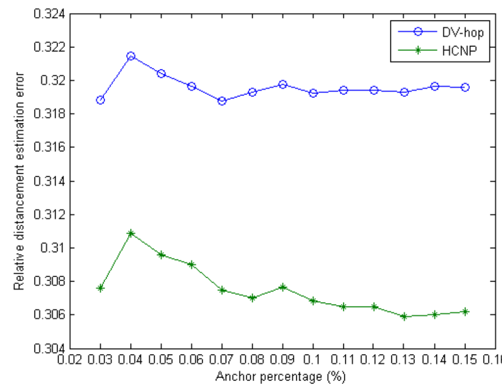

Fig. 6. Distance estimation error with varying parameter: anchor%.

number of nodes and the percentage of anchors deployed in the simulation studies. The distance estimation error and localization error of the DV-hop and the HCNP algorithms are compared. To ensure a fully connected network can be formed, we start with 450 nodes in the first simulation and fix the node number to be 500 in the second one.

B. Simulation results

1) Distance estimation: First, we compare the node-to-anchor distance estimation of the DV-hop and the HCNP algorithms by carrying out 50 independent simulations. The distance estimation error is defined by

$$e_{dist} = \frac{\frac{1}{N \times M} \sum_{i=1}^{N} \sum_{j=1}^{M} |\tilde{d}_{ij} - d_{ij}|}{r},$$

where $\tilde{d}_{ij}$ and d_{ij} is the estimated distance and true distance of node i to anchor j, respectively. The distance estimation results are plotted in Fig. 5 and 6 respectively. The x axis of the plots are the number of nodes and anchor%, the y axis of the plots is the relative distance estimation error - ratio of distance estimation error and communication range. It is seen in Fig. 5 that the distance estimation has been greatly improved when compared with the DV-Hop algorithm. In particular, when there are 1500 nodes, the distance estimation error improvement compared with the DV-Hop is 26.3%. In Fig. 6, the distance estimation of the HCNP is again better than the DV-hop, however, the performance difference of the two algorithms seems to be independent to the anchor%.

2) Localization: We also compare the performance of localization using the two methods based on the same simulation settings. Let (x_i, y_i) and $(\hat{x}_i, \hat{y}_i)$ denote the true and estimated sensor location respectively. The localization error is defined as follows:

$$e_{loc} = \frac{\sum_{i=1}^{N} \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}}{N \times r}.$$

Fig. 7 and 8 show that the localization performance of the HCNP is better than the DV-hop. We can see in Fig. 7 that the localization error by using our method is lower than that of the DV-Hop. In particular, when there are 1500 nodes, the localization error improvement compared with the DV-Hop is 19.0%. In Fig. 8, though the localization error of the HCNP is lower than the DV-hop, we notice that the performance difference between the two algorithms seems to be independent to the anchor%. This phenomenon is similar to what we noticed in the distance estimation simulations. Based on the simulation results, we thus conclude that the

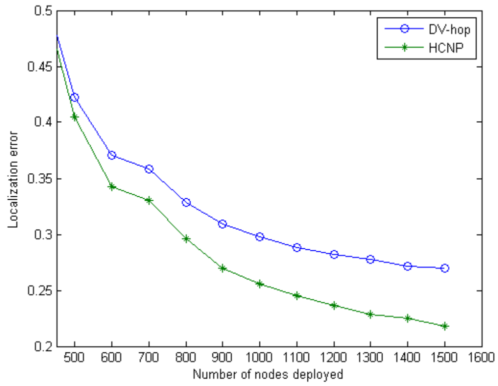

Fig. 7. Localization error with varying parameter: number of nodes.

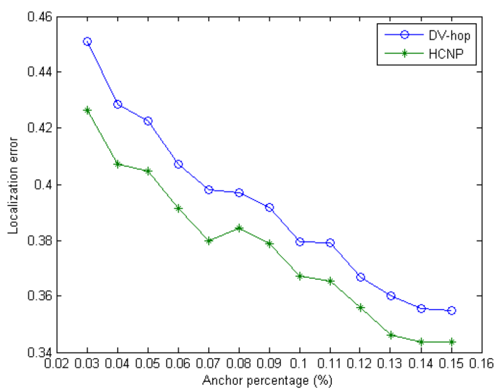

Fig. 8. Localization error with varying parameter: anchor%.

performance of the HCNP only depends on the node density. Intuitively, this conclusion is not surprising as the distance estimation(which affects the localization) gets more accurate when there are more nodes deployed.

To illustrate how node density affect the performance of the HCNP algorithm, we plot the percentage of distance estimation and localization improvement in Figure 9 with respect to the number of nodes deployed. It can be seen that the percentage of improvement is proportional to the node density for both cases.

IV. Conclusion

In this paper, we have presented a new hop-count based WSNs localization algorithm - HCNP. The algorithm divides a node's neighbors based on their hop count and then derives the node-to-anchor distance based on its neighbor sets and their geometric relations. The simulation results show that the performance of the HCNP in terms of distance estimation and localization is better than those of DV-Hop and the improvement increases with the increase of the node density. In future research, we will further study the detailed relationship between the node density and the performance the proposed algorithm.

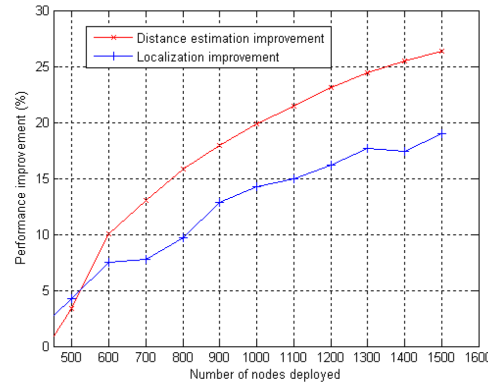

Fig. 9. Percentage of improvement.

References

[1] D. Niculescu and B. Nath, "Ad hoc positioning system (aps)," *Global Telecommunications Conference*, vol. 5, pp. 2926 – 2931, 2001.

[2] K. Langendoen and N. Reijers, "Distributed localization in wireless sensor networks: a quantitative comparison," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 43, pp. 499 – 518, November 2003.

[3] C. Savarese, J. Rabay and K. Langendoen, "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," *USENIX Technical Annual Conference*, pp. 317–327, June 2002.

[4] S.Dulman and P. Havinga, "Statistically enhanced localization schemes for randomly deployed wireless sensor networks," *Intelligent Sensors, Sensor Networks and Information Processing Conference*, pp. 403–410, December 2004.

[5] S. Y. Wong, J. G. Lim, S. Rao and W. K. Seah, "Density-aware hop-count localization (dhl) in wireless sensor networks with variable density," *Wireless Communications and Networking Conference, IEEE*, vol. 3, pp. 1848 – 1853, March 2005.

[6] S. Yang, J. Yi and H. Cha, "Hcrl: A hop-count-ratio based localization in wireless sensor networks," *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 31–40, June 2007.

[7] Y. Shang, W. Ruml, Y. Zhang and M. Fromherz, "Localization from mere connectivity," *international Symposium on Mobile Ad Hoc Networking and Computing*, pp. 201 – 212, 2003.

[8] I. Borg and P. Groenen, *Modern Multidimensional Scaling: Theory and Applications*, Springer-Verlag, 1997.

[9] L. Doherty, K. p. and L. El Ghaoui, "Convex position estimation in wireless sensor networks," *INFOCOM*, vol. 3, pp. 1655–1663, April 2001.

[10] T. He, C. Huang, B. M. Blum, J. A. Stankovic and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," *MobiCom*, 2003.

[11] L. Kleinrock and J. Silvester, "Optimum transmission radii for packet radio networks or why six is a magic number," *IEEE National Telecommunications Conference*, 1978.

[12] J. Kuo and W. Liao, "Hop count distribution of multihop paths in wireless networks with arbitrary node density modeling and its applications," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 4, 2007.

[13] B. Wang, C. Fu and H. B. Lim, "Layered diffusion based coverage control in wireless sensor networks," in *IEEE the 32rd Conference on Local Computer Networks (LCN)*, 2007, pp. 504–511.

[14] R. Rubinstein and D. Kroese, *Simulation and the Monte Carlo Method*, Wiley, 2 edition, 2007.

[15] M. Heath, *Scientific Computing - An Introductory Survey*, McGraw Hill, 2001.

[16] L. Zhao and Q. Liang, "Hop-distance estimation in wireless sensor networks with applications to resources allocation," *EURASIP*, 2007.