



Naftali (Tuli) Herscovici
AnTeg
52 Agnes Drive
Framingham, MA 01901 USA
Tel: +1 (508) 788-5152
Fax: +1 (508) 788-6226
E-mail: tuli@ieee.org



Christos Christodoulou
Department of Electrical and
Computer Engineering
University of New Mexico
Albuquerque, NM 87131-1356 USA
Tel: +1 (505) 277-6580
Fax: +1 (505) 277-1439
E-mail: christos@ece.unm.edu

Calibration-Free RF-Based Localization Algorithm for Sensor Actuator Networks Using Particle Filters

Aly I. El-Osery¹, Wael Abd-Elmageed², and Moustafa Youssef²

¹Electrical Engineering Department, New Mexico Tech
801 Leroy Place, Socorro, NM 87801 USA
E-mail: elosery@ee.nmt.edu

²Institute for Advanced Computer Studies, University of Maryland
College Park, MD 20742 USA
E-mail: wamageed@umiacs.umd.edu, moustafa@cs.umd.edu

Abstract

A calibration-free localization algorithm for sensor actuator networks is presented in this paper. The algorithm uses sequential importance sampling to estimate the location of a number of limited-capability sensors, using the strength of the signals transmitted from the sensor node to GPS-enabled mobile actuator nodes that work as anchor nodes. Simulation results show that the algorithm converges to an error range of 1 m in approximately 10 time steps, for very small Rice factors and four anchor nodes. The algorithm was verified using the Network Simulator NS-2.28.

Keywords: Intelligent sensors; multisensor systems; sensor actuator networks; localization, received signal strength, Monte Carlo methods; position measurement; particle filters

1. Introduction

Sensor actuator networks have gained a high level of attention recently, due to technology advances in wireless communication, sensor design, and robotics. Sensor actuator networks provide a useful tool for monitoring, surveying, and acting in large areas, and, consequently, require the deployment of a large number of sensors. Such a large number of sensors precludes manual deployment,

configuration, and calibration. Due to these limitations, self localization is critically important.

The received-signal-strength indicator (RSSI) has attracted a lot of attention, as it is a standard feature in most radios. RSSI eliminates the need for additional hardware, and, hence, exhibits favorable properties with respect to power consumption and size. Despite the obvious advantages of using RSSI, there is a general skepticism in the wireless sensor network community. This skepticism is

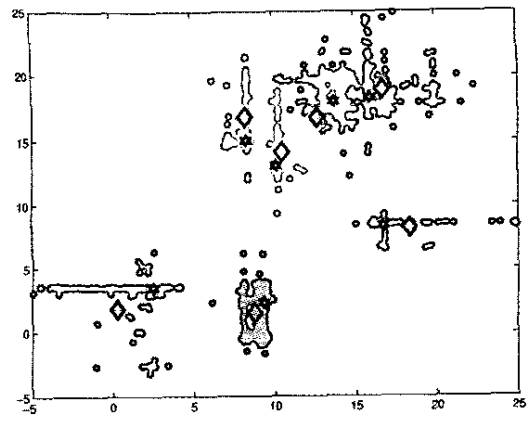
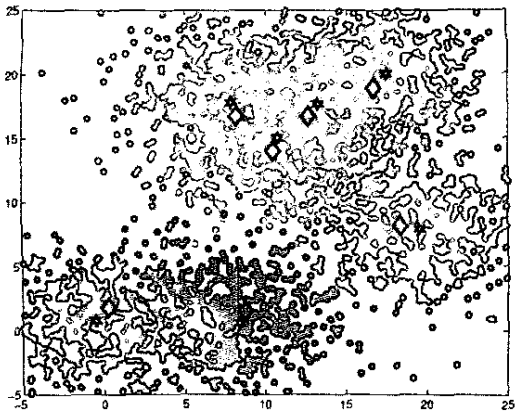


Figure 2d

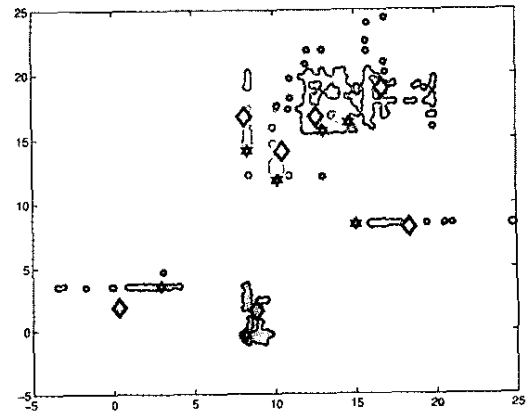
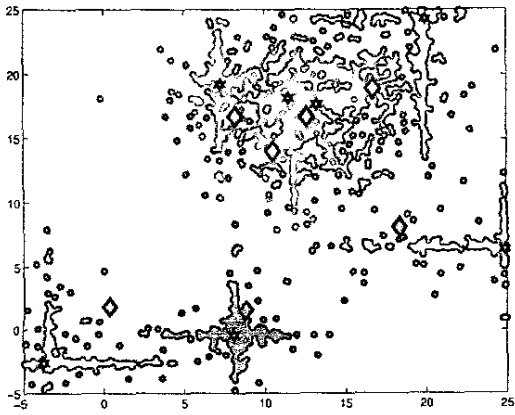


Figure 2b

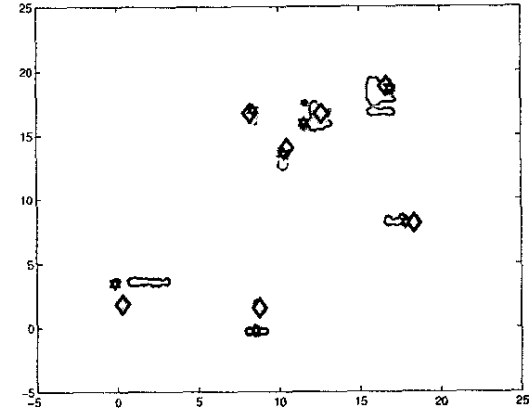
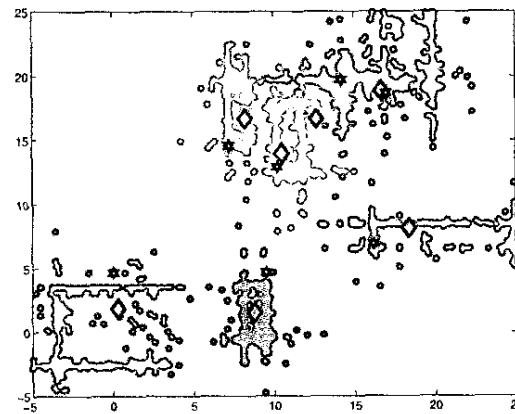


Figure 2c

Figure 2. Snapshots of the simulation using $K=3$. The actual locations of the sensor is represented by diamonds with different colors, and the corresponding particles are represented by dots with corresponding colors. The particle with the best objective-function value is represented by a star. From left to right and top to bottom (a-f), the plots show iterations 1, 2, 3, 4, 5, and 10, respectively.

primarily due to asymmetry in the radio links in older radios. In the past decade, several short-range wireless technologies have been developed to answer the demand for portable wireless devices. Among these is IEEE 802.15.4, which was developed to address the low-power and low-bit-rate needs of small and embedded devices. IEEE 802.15.4 is based on direct-sequence spread-spectrum, allowing the radio to reject multipath and interference [1]. In [2, 3], the authors showed that for radios utilizing IEEE 802.15.4, RSSI was a promising indicator, and overcame hardware mis-calibration issues in older radios.

In this paper, we present a calibration-free localization algorithm using sequential importance sampling (particle filters). Our system is unique in studying the problem of location determination in the context of sensor actuator networks, where the number of sensors is much larger than the number of actuators, where actuators are more capable and mobile, and where having extra hardware is cost prohibitive. Our system also uses particle filters to handle the noisy wireless channel measurements, where the distribution of the received signal strength might be non-Gaussian and multi-modal, and to detect outliers.

This paper is organized as follows. Related research is presented in Section 2. The propagation model is covered in Section 3. An introduction to particle filters, followed by our proposed algorithm, are introduced in Section 4. Simulation and results are presented in Section 5, and, finally, conclusions are given in Section 6.

2. Related Work

Many location-determination systems have been proposed for sensor networks. These systems can be categorized as either anchor-based or anchor-free. Anchor-based techniques, such as [4-6], depend on using reference points (also known as anchors or landmarks) the locations of which are known to estimate the location of other nodes. On the other hand, anchor-free techniques estimate the locations of nodes relative to each other, without using any reference points. Generally, anchor-based techniques are more accurate than anchor-free techniques. Since we propose an anchor-based algorithm, the rest of this section focuses on anchor-based techniques.

In [4], Brooks et al. presented a system that used computer-vision techniques to estimate the locations of mobile robots, and use the robots as anchor points. Two major limitations of the system were that it can only be deployed in an area where an infrastructure exists, and, secondly, it prohibits large deployments due to cost limitations.

Time-of-flight was used in [5] to estimate the range between a mobile robot and a number of stationary tags. The authors used a Kalman filter to handle the noisy data. Their approach, however, required a large number of tags to track a single robot, and was not rapidly deployable.

Wi-Fi deployments have been used by a number of groups to solve the localization problem. In [6-9], Wi-Fi deployments were used to estimate the sensor locations based on the signal strength. All of these systems required prior calibration of the area of interest, and were therefore not rapidly deployable.

3. Propagation Model

Typically, a non-line-of-sight radio-propagation path exists between a transmitter and a receiver. Consequently, radio waves must propagate via reflections, diffraction, and scattering, arriving at the receiver from many different directions and with different delays. Such a phenomenon, known as multipath, results in various levels of distortions to the received signal. Assuming flat fading, i.e., all frequency components of the received signal are subject to the same fading, the received band-pass signal can be expressed in quadrature form as

$$r(t) = g_I(t) \cos 2\pi f_c t - g_Q(t) \sin 2\pi f_c t, \quad (1)$$

where f_c is the center frequency, and $g_I(t)$ and $g_Q(t)$ are the in-phase and quadrature components of the received bandpass signal. As the number of incident waves increases, $g_I(t)$ and $g_Q(t)$ can be treated as Gaussian random processes. If the scattering environment has a specular or line-of-sight component, $g_I(t)$ and $g_Q(t)$ become nonzero-mean Gaussian random processes, with means $m_I(t)$ and $m_Q(t)$, respectively. Therefore, the received complex envelope,

$$g(t) = g_I(t) + jg_Q(t) \quad (2)$$

with magnitude $\alpha(t)$ has a Ricean distribution, given by

$$p_\alpha(\zeta) = \frac{\zeta}{\sigma^2} \exp\left\{-\frac{\zeta^2 + z^2}{2\sigma^2}\right\} I_0\left(\frac{\zeta z}{\sigma^2}\right), \quad \zeta \geq 0, \quad (3)$$

where

$$z^2 = m_I^2(t) + m_Q^2(t), \quad (4)$$

σ is the standard deviation of $g_I(t)$ and $g_Q(t)$, and $I_0(\cdot)$ is the modified Bessel function of order zero. Given this model, the Rice factor [10], which represents the ratio of the specular power to the scattered power, is given by

$$K = \frac{z^2}{2\sigma^2}. \quad (5)$$

When $K = 0$, then there is no specular component, and the channel exhibits Rayleigh fading. On the other hand, when $K = \infty$, then the channel does not exhibit any fading.

In addition to the small-scale fading caused by multipath propagation, the signal also decays inversely proportionally to the path length. In this paper, we used the two-ray ground-reflection model, which considers both the direct path and a ground-reflection path. Given the two-ray reflection model, the path loss is predicted by

$$L_{p(dB)} = -10 \log_{10} \left(\frac{h_t^2 h_r^2}{d^4} \right), \quad (6)$$

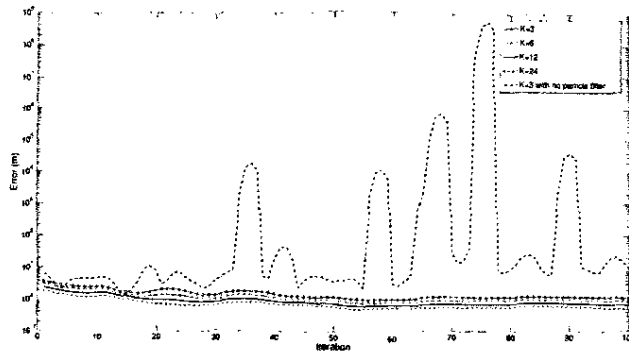


Figure 3. The average error using three anchor nodes for various values of K , as well as the error using the large-scale propagation model without particle filters. $K = 3$: blue; $K = 6$: red; $K = 12$: black; $K = 24$: green; the dash-dotted line is for $K = 3$ with no particle filter. The vertical axis is the error in meters, from 10^{-1} to 10^0 ; the horizontal axis is the iteration number, from 0 to 100.

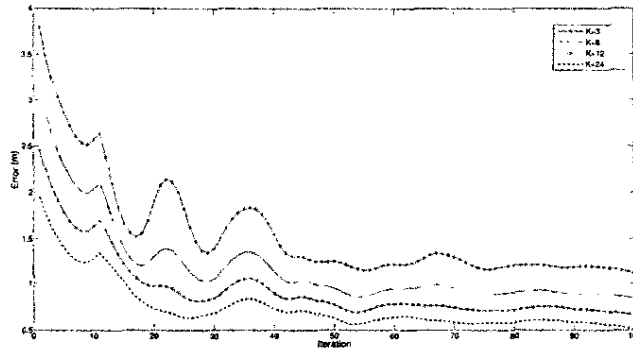


Figure 4. The average error using three anchor nodes for various values of K . $K = 3$: blue; $K = 6$: red; $K = 12$: black; $K = 24$: green. The vertical axis is the error in meters, from 0.5 to 4; the horizontal axis is the iteration number, from 0 to 100.

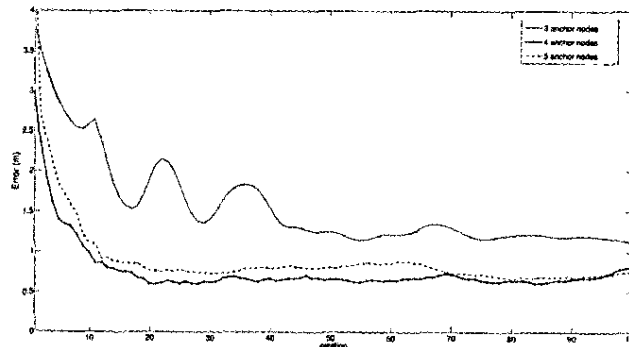


Figure 5. The average error for three (purple), four (black), and five (blue) anchor nodes, using a Ricean factor of 3. The vertical axis is the error in meters, from 0.5 to 4; the horizontal axis is the iteration number, from 0 to 100.

where d is the distance between the transmitter and the receiver, and h_t and h_r are the antenna heights of the transmitter and the receiver, respectively. In small areas, the large-scale path loss is just inversely proportional to the distance to the fourth.

4. The Localization Algorithm

The system may be modeled using a discrete-time state-space representation. \mathbf{x}_j^t will be used as the system state at time t , which represents a column vector of the relative distances between sensor node j and each of the anchor nodes. The measurement, \mathbf{y}_j^t , represents the received signal strengths at each of the anchor nodes from sensor node j . The system model may be written as

$$\mathbf{x}_j^t \sim p_j(\mathbf{x}_j^t | \mathbf{x}_j^{t-1}, \mathbf{n}_j^{t-1}) \quad (7)$$

and

$$\mathbf{y}_j^t \sim p_j(\mathbf{y}_j^t | \mathbf{x}_j^t, \mathbf{v}_j^t), \quad (8)$$

where $p_j(\mathbf{x}_j^t | \mathbf{x}_j^{t-1}, \mathbf{n}_j^{t-1})$ is the probabilistic state-transition density function, \mathbf{n}_j^{t-1} is the covariance of zero-mean process noise, $p_j(\mathbf{y}_j^t | \mathbf{x}_j^t, \mathbf{v}_j^t)$ is a probabilistic measurement function representing the propagation channel, and \mathbf{v}_j^t is the measurement noise. Our aim is to recursively estimate, in time, the posterior distribution $p(\mathbf{x}_j^{0:t} | \mathbf{y}_j^{1:t})$, using particle filters.

Similar to Kalman filtering [11], the idea behind particle filtering [12, 13] is to estimate the state of a given system and to track this state as it evolves over time. However, in contrast to Kalman filtering, the state of the system might follow a multi-modal, non-Gaussian probability density function (pdf).

In particle filters, each sample (also known as a particle) may be described by

$$\{\mathbf{s}_i^t(j), \pi_i^t(j); i=1, \dots, N^t; t=1, \dots, \infty\}, \quad (9)$$

where $\mathbf{s}_i^t(j)$ and $\pi_i^t(j)$ represent particle i at time t and its weight, respectively, such that

$$\sum_i^{N^t} \pi_i^t(j) = 1, \quad (10)$$

where N^t represents the number of particles, and the superscript t indicates that the number of particles may vary over time. Each particle $\mathbf{s}_i^t(j)$ represents the state of sensor j at time t with probability $\pi_i^t(j)$. In our localization algorithm, the state of each sensor node is defined as a vector representing the relative distances between sensor node j and each of the actuator (anchor) nodes.

The propagation of the particles over time follows the state-transition model of Equation (7), and, hence, the particles are generated according to

$$\mathbf{s}_i^t(j) \sim \mathcal{N}(\mathbf{x}_j^{t-1}, \mathbf{n}_j^{t-1}), \quad (11)$$

where $p_j(\mathbf{x}_j^t | \mathbf{x}_j^{t-1}, \mathbf{n}_j^{t-1}) = \mathcal{N}(\mathbf{x}_j^{t-1}, \mathbf{n}_j^{t-1})$.

The initialization of the algorithm to localize node j is as follows:

1. An estimate of the sensor location, \mathbf{y}_j^0 , is obtained by using the received signal strength and a trilateration algorithm, described briefly in the appendix, assuming that the channel exhibits only large-scale fading.
2. The initial set of particles, $\mathbf{s}_i^0(j)$, is generated using a two-dimensional normal distribution, centered at the estimated node location, with a diagonal covariance matrix the values of which are proportional to the deployment area. The covariance matrix is assumed diagonal since there is no correlation between the x and y coordinates of the node.
3. The inverse of the error between this estimate and the particle location is used as the importance weight to estimate the initial density $\hat{p}[\mathbf{s}_i^0(j)]$.

The main loop of the algorithm works in a similar fashion. At each time step, an estimate of the node location, \mathbf{y}_j^t , is computed using the trilateration algorithm. A new set of particles, $\mathbf{s}_i^t(j)$, is generated according to Equation (11), where \mathbf{n}_j^{t-1} is the adaptive noise covariance. This selection step eliminates the particles having low importance weights, and multiplies particles having high importance weights [12]. At each time step, t , the node location is updated as shown in Equation (12):

$$\mathbf{x}_j^t = \sum_{i=1}^{N^t} \pi_i^t(j) \mathbf{s}_i^t(j). \quad (12)$$

To summarize, the algorithm simply iteratively eliminates particles that exhibit high error with respect to the observed position, \mathbf{y}_j^t , and samples more of the particles that consistently exhibit low error. The following is a pseudo-code listing of the localization algorithm.

Algorithm: Localize Sensor (j)

1. Initialization

- Computer the initial node locations using the received signal strength and trilateration.
- Sample $\mathbf{s}_i^0 \sim \hat{p}(\mathbf{s}_i^0)$; $i=1, \dots, N_0$, where $\hat{p}(\mathbf{s}_i^0)$ is a Gaussian probability density function.

2. Importance sampling step

- For $i = 1, \dots, N_t$, sample $s_i^t(j) \sim \mathcal{N}(\mathbf{x}_j^{t-1}, \mathbf{n}_j^{t-1})$,
- For $i = 1, \dots, N_t$, evaluate the importance weights.
- Normalize the importance weights.

3. Selection step

- Resample according to importance weights.
- Set $t \leftarrow t + 1$ and go to Step 2.

5. Simulation and Results

While the system is scalable to large numbers, in order to test the proposed algorithm three scenarios, consisting of only seven stationary nodes in combination with three, four, and five mobile anchor nodes, were designed. The mobile nodes acted as the anchor nodes and moved randomly at a speed of 1 m/s. Stationary nodes transmitted a message, and the received signal strength was measured at each anchor node. The propagation channel was based on small-scale fading, with Ricean distribution modulating a two-ray reflection propagation model [14]. The discrete-event network-simulation program, NS version 2.28 [15], was used to test the designed scenario.

The simulation parameters were as follows: the number of particles was 500, the simulation area was $20 \text{ m} \times 20 \text{ m}$, the simulation duration was 100 s, the noise variance was 0.02 for both directions, and the location update rate was 2 Hz.

The received signal at the anchor nodes, assuming a Ricean channel with $K = 6$, is shown in Figure 1. Snapshots of the locations of the particles are shown in Figure 2, where the particles are represented by dots and the actual location of the node is marked by a diamond. As shown in Figure 3, the proposed algorithm with three anchor nodes converged, and the resulting error in position reached 0.5 m for $K = 24$, and 1.1 m for $K = 3$, which represented a very noisy environment. As a comparison, the raw location estimation based purely on the large-scale propagation model is also plotted. Figure 4 shows the performance of only our system. It is clear from the figure that there is a trend in the error among the different values of K . This trend is due to the anchor-node locations over time. The motion of the anchor nodes was the same for all simulations and, depending on their location, the trilateration error would be similar, hence the trend in the error curves.

In Figure 5, the system performance using three, four, and five anchor nodes is demonstrated. As shown, there was a significant improvement by using four anchor nodes as opposed to three. On the other hand, there was no significant difference between using four or five anchor nodes. This lack of difference is attributed to the uncertainty in the measurements, which resulted in an increase in the error if we increased the number of anchor nodes.

6. Conclusions

An algorithm for localizing limited-capability sensor nodes was presented. The strength of the signals transmitted from the

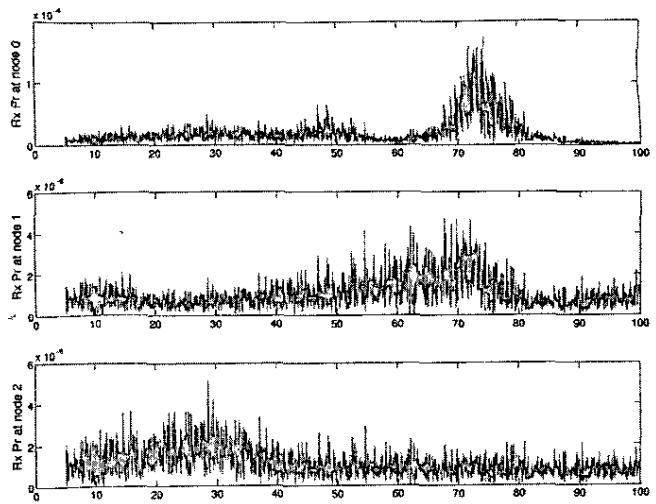


Figure 1. An example of the received signal strength from one of the sensor nodes using $K = 6$, where the anchor nodes are moving at 1 m/s.

nodes to GPS-enabled, mobile anchor nodes was used in a particle-filter framework to perform the localization. In contrast to other algorithms, such as [5], that localize a small number of mobile robots using a larger number of static nodes, the proposed algorithm attempts to solve the more challenging problem of localizing a large number of limited-capability sensor nodes using a smaller number of mobile robots.

Simulations were designed assuming a Ricean propagation model, with very small specular-to-scattered power ratios. Results showed that our system achieved a sub-meter accuracy, which is in the same range as the best-known RF-based system [9], which requires calibration of the area of interest.

We plan to implement the localization algorithm on three man-portable, tracked robots, named MATILDA, made by Mesa Robotics, which will act as the actuator nodes [16]. The electronics, including the sensor suite, were built around an X86 100 MHz single-board computer (SBC), running a distribution of *OpenEmbedded GNU/Linux*. Communication to the robots can occur in three ways: radio control that bypasses the single-board computer for failsafe local control, 802.11b wireless for command/data transfer, and cellular network service for command/data transfer. The sensor nodes are 2.4 GHz, IEEE 802.15.4-compliant wireless measurement systems, known as MICAz. These nodes run a small, energy-efficient software operating system known as *TinyOS*.

7. Appendix: Trilateration

Trilateration is the process of identifying a node location using distances. The distance may be computed in various ways, one of which uses received signal strength. The trilateration problem can be formulated as follows for the three-dimensional space. Given the positions of n anchor nodes, $\{(x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_n, y_n, z_n)\}$, and the estimated distances to these anchor nodes, $\{d_1, d_2, \dots, d_n\}$, we want to find the location of the node (x, y, z) that satisfies the following set of simultaneous nonlinear equations:

8. References

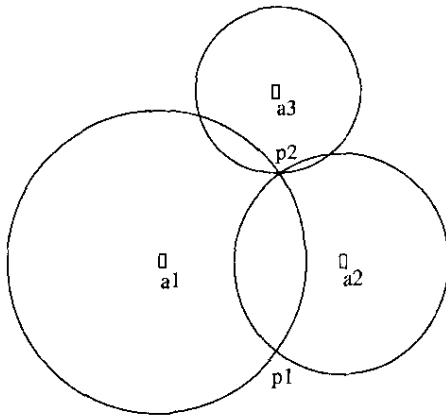


Figure 6a. Trilateration assuming perfect values and no interference, noise, or multipath, and, hence, a perfect circle intersection is possible.

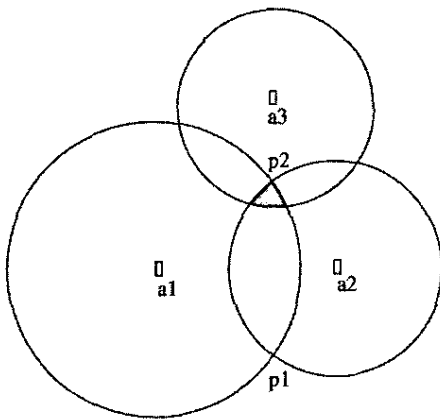


Figure 6b. Trilateration using received signal strength that is corrupted by noise, multipath, and interference, and, hence, exact circle intersections are not likely. Instead, a region, as shown by the shaded area, is obtained.

$$(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2 = d_i^2 \quad (13)$$

for $i = 1 \dots n$.

Note that for the three-dimensional space, only four anchor nodes are needed for locating the node, while in a two-dimensional space, three nodes are sufficient.

An example for the two-dimensional space is shown in Figure 6a. As shown in the figure, based on the received signal strength at anchor node a_1 , the transmitting node may be anywhere on a circle, providing infinite number of possibilities. By using the second anchor node, the choices are now limited to points p_1 or p_2 . Finally, using the third node, the position is finally determined. Hence, the localization problem has been reduced to finding the intersection of the circles. Unfortunately, in real life the received signal strength is highly distorted, and, therefore, perfect intersections of the circles are highly unlikely. In this case, finding the location that minimizes the distance error to the anchor nodes' position can be used. As shown in Figure 6b, in the case of imperfect distance estimation, instead of having a perfect intersection a region is obtained.

1. "IEEE Standard for Information Technology Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)," IEEE Std. 802.15.4-2003.

2. K. Srinivasan and P. Levis, "RSSI is Under Appreciated," Third Workshop on Embedded Networked Sensors (EmNets 2006), May 2006.

3. M. Petrova, J. Riihijarvi, P. Mahonen, and S. LaBellla, "Performance Study of IEEE 802.15.4 Using Measurements and Simulations," *Proceedings of IEEE WCNC*, Las Vegas, USA, April 2006.

4. A. Brooks, S. Williams, and A. Makarenko, "Automatic Online Localization of Nodes in an Active Sensor Network," IEEE 2004 International Conference on Robotics and Automation (ICRA'04), 2, 2004, pp. 1097-1102.

5. J. Djugash, S. Singh, and P. Ian Corke, "Further Results with Localization and Mapping Using Range from Radio," International Conference on Field & Service Robotics (FSR '05), 2005.

6. A. Howard, S. Siddiqi, and G. Sukhatme, "An Experimental Study of Localization Using Wireless Ethernet," International Conference on Field and Service Robotics, 2003.

7. P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF-based User Location and Tracking System," IEEE Infocom 2000, 2, March 2000, pp. 775-784.

8. J. Letchner, D. Fox, and A. LaMarca, "Large-Scale Localization from Wireless Signal Strength," AAAI, 2005, pp. 15-20.

9. M. Youssef and A. Agrawala, "The Horus WLAN Location Determination System," Third International Conference on Mobile Systems, Applications, and Services (MobiSys 2005), June 2005.

10. G. Stauber, *Principles of Mobile Communication, Second Edition*, Boston, Kluwer Academic Publishers, 2001.

11. R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME - Journal of Basic Engineering*, 82, 1960, pp. 35-45.

12. A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, New York, Springer, 2001, Chapter 1, pp. 3-14.

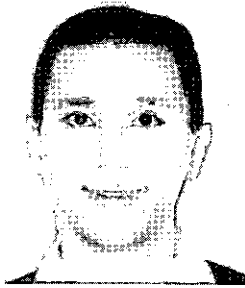
13. A. Doucet, C. Andrieu, and S. Godsill, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, 10, 3, 2000, pp. 197-208.

14. R. J. Punnoose, P. V. Nikitin, and D. D. Stancil, "Efficient Simulation of Ricean Fading within a Packet Simulator," IEEE Vehicular Technology Conference, 2, Boston, MA, September 2000, pages 764-767.

15. NS Simulator, <http://www.isi.edu/nsnam/ns/2.28>.

16. M. Briggs, D. Baird, W. Ogg, S. Aafloy, A. I. El-Osery, and Kevin Wedeward, "An Adaptable Outdoor Robotic Platform: Architecture, Communications, and Control," IEEE International Conference on Systems of Systems, Los Angeles, April 2006.

Introducing the Authors

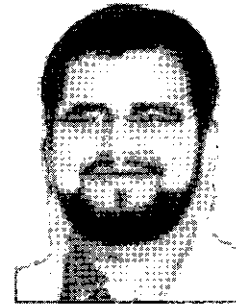


Dr. **Aly El-Osery** received his BS in 1997, MS in 1998, and PhD in 2002 in Electrical Engineering from the University of New Mexico. From 1997 to 2002, he was a research assistant at the Autonomous Control Engineering Center at the University of New Mexico. Dr. El-Osery has received many awards, among them Outstanding Junior (1996) and Outstanding Graduate Student (1998) from the Department of Electrical Engineering, and the School of Engineering award for Outstanding Graduate Student in Electrical and Computer Engineering (1998-1999).

In 2002, he joined the Electrical Engineering Department at New Mexico Institute of Mining and Technology, Socorro, New Mexico, as an Assistant Professor. His research interests are in the areas of multi-agent robotics, wireless communications, control systems, sensor networks, and soft computing. He has over 30 journal papers, book chapters, and conference publications in these areas. He is the cofounder of the Intelligent Systems and Robotics Group (ISRG) at New Mexico Tech. He is a member of the IEEE and several of its Societies, including the Vehicular Technology Society, Systems Man and Cybernetics Society, and Communication Society.



Wael Abd-Almageed is an Assistant Research Scientist at the University of Maryland at College Park. He received his BS. and MSc from Mansoura University, Egypt, in 1994 and 1997, respectively, and his PhD with Distinction from the University of New Mexico in 2003. Dr. Abd-Almageed was awarded the Outstanding Graduate Student award for his PhD research. His research interests include computer/robot vision, statistical pattern recognition, and ad-hoc sensor networks.



Moustafa Youssef is a research associate in the Department of Computer Science at the University of Maryland at College Park. He received his BSc and MSc in Computer Science from Alexandria University, Egypt, in 1997 and 1999, respectively, and the PhD degree in Computer Science from the University of Maryland in 2004. His research interests include location-determination technologies, pervasive computing, energy-aware computing, sensor networks, and protocol modeling. Dr. Moustafa is a Life Fellow for the Egyptian Society for Talented, and an elected member of the honor society Phi Kappa Phi, among others. He is a member of various professional societies, such as the IEEE, IEEE Computer Society, IEEE Communication Society, and ACM Sigmoble. Dr. Moustafa was the recipient of the 2003 University of Maryland Invention of the Year award for his Horus work. (15)