Link Longevity Kalman-Estimator for Ad Hoc Networks

Rui Huang, Gergely V. Záruba, and Manfred Huber
Computer Science and Engineering Department
The University of Texas at Arlington
416 Yates, 300NH, Arlington, TX 76019
Email: rxh1725@omega.uta.edu, {zaruba,huber}@cse.uta.edu

***Abstract*— In ad hoc networks, links are established on the fly as mobile nodes move in and out of each other's transmission range. Due to this mobility, routes can be disrupted while in service. Availability of a good estimation of link longevity between neighboring nodes could permit the selection of a more stable route, thus enabling a better enforcement of quality of service (QoS) contracts. In this paper, we propose three link longevity estimators that could be embedded into mobile nodes. In our approach, link longevity estimates do not require knowledge of the mobility patterns/models of nodes. The foundation of all three designs lies in extended Kalman filtering, where a linear process model is implemented to represent the state transition, while a non-linear measurement model is included to account for the received signal strength indication measurements. We provide with the mathematical background for our estimators and their input/output vectors, and show various performance metrics using extensive simulations. We conclude that our filters provide good estimates for the remaining up-time of wireless links.**

I. INTRODUCTION

In ad hoc networks, links are established on the fly as mobile nodes are moving in and out of the transmission ranges of each other. This node mobility results in a constantly changing network connectivity graph. Due to the distributed nature of ad hoc networks, a route between two arbitrary nodes is likely a combination of multiple links over several intermediate nodes. The selection of the sequence of links between nodes is the task of the ad hoc routing protocol. Due to link failures caused by node movement, routes can be disrupted while in service. Loss of links can invalidate routing entries, which in turn causes undesired latencies in packet delivery. Availability of a good estimation of link longevity between neighboring nodes could permit the selection of a more stable route, thus enabling a better enforcement of quality of service (QoS) contracts.

There are a number of previous works addressing the link longevity problem based on probability models [1], [2]. Their contribution lies in defining node mobility models and performing a mathematical analysis on these models to quantify statistical properties of the longevity of links. In [1], the link longevity is measured as the probability of the link remaining available for a time t. The probabilistic model is then used as the basis to form and maintain clusters within ad hoc networks to maximize the cluster stability. The probability calculation assumes that all nodes move according to an epoch based movement pattern. For each node, the movement history is divided into a sequence of epochs. Within each epoch, the node moves at a randomly selected (but constant) direction and velocity. The pure probability based model is extended in [2] by incorporating a measurement model. First, the link longevity, in term of remaining time (t) in which the link remains available, is measured under the assumption that the nodes maintain their constant velocities. A probability model is then applied to calculate the probability of the link availability at time t by considering the cases of varying velocities within t.

A non-probabilistic solution is proposed in [3], in which the link longevity estimation is used to measure the stability of the entire route so that a handoff can be triggered in anticipation. The longevity estimation relies on a Global Positioning System (GPS) receiver at each node to provide the location and velocity data. The remaining connection time between two nodes is calculated from the GPS data assuming that the nodes maintain their headings and speeds. If GPS data is not available other measurement-based models could be used. In [4], the location, velocity and acceleration of a mobile node are estimated by measuring the received signal strength indication (RSSI) from multiple base stations in a cellular network. The measured power levels are feed into a Kalman filter. Since base station locations are assumed to be well-known in a cellular network, mobile nodes can use them as reference points.

RSSI measurements can also be used to estimate the location of mobile nodes in ad hoc networks. In [5], the authors propose a method of propagating the location data from nodes that are GPS-equipped. Other non-GPS nodes can then deduce their distances to the GPS nodes by measuring the RSSI from neighboring nodes. The actual location can then be calculated using triangulation methods from the distance information. The method is further improved in [6] by assuming non-GPS nodes are equipped with devices that measure the incoming signal directions. The directional information allows the receivers to obtain the angle of arrival (AoA) of the signal thus allowing more accurate location estimates.

In this paper we present a measurement-based approach to the link longevity problem. Unlike the probability-based solutions that rely on a particular mobility model [1], [2], we propose an estimator to be embedded into the mobile

0-7803-7954-3/03/$17.00 ©2003 IEEE.

nodes that operates regardless of the mobility model. The estimator's basis is a Kalman filter [9] used to estimate the relative location of two nodes based solely on simple signal properties like RSSI. The information obtained from the filter is then used to derive the expected time the link will remain available. A major advantage of Kalman filters is that they can quickly and efficiently compute estimates. Therefore, they are particularly suited for ad hoc networks due to potentially limited computing power of mobile nodes. Our solution is similar to the estimator in [4], but it is designed to work in the distributed ad hoc environment, where all nodes are mobile and the relative location needs to be determined. Furthermore, unlike the measurement-based location estimators [5], [6], our solution is geared towards estimating the link longevity in time t instead of the exact node locations. This means that a node will not only need to determine the locations of other nodes but also a change in their movement patterns.

The remainder of the paper is organized as follows. Section II describes three different the link longevity estimator designs including detailed process models used by the Kalman filters. Section III presents the simulation results and compares the three estimators under different movement and noise conditions. Section IV concludes this paper.

II. KALMAN ESTIMATOR DESIGNS

This section outlines three different designs of an ad hoc link longevity estimator. Each of the designs is based on extended Kalman filters [10]. In all three designs, a linear process model is implemented to represent the state transition. The process model assumes that nodes maintain their current direction and velocity between each state update. Corrections to the errors in the process model are made via the measurement model of the filter. The need for the extended Kalman filter arises due to the measurement model's inherent non-linearity (radio signal power levels are not linear with the propagation distance). Each filter design is unique: they rely on input from different types of sensors and/or keep their state information in different variables in the process model.

A. Design I

Our first filter design assumes that both the incoming signal strength and its direction are observable at mobile nodes. Using a signal propagation model, the distance and direction to a transmitting node can be estimated from the received signal strength (RSSI sensor) and the direction of the received signal (direction sensor). Given the measurements the estimator is able to track the relative location and velocity in both x and y directions. Therefore, the state variable of the filter at time t is $(x_t,\, y_t,\, v_x,\, v_y)$, where x_t and y_t represent the relative displacement at time t, while v_x and v_y represent the relative velocity. By further assuming that nodes move at a constant velocity, a new state can be derived from the previous state using a linear process update function $x_{t+1} = x_t + v_x \Delta t$ and $y_{t+1} = y_t + v_y \Delta t$, where Δt is the observation interval. The state transition matrix is therefore

$$\begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Given a pre-determined transmission range, the expected time of link termination can be calculated from the state variables.

Hardware-wise, sensors that measure RSSI are widely available for mobile devices. Indeed most off-the-shelf technologies implicitly provide such information (e.g., most Wi/Fi cards provide with RSSI). However, sensors that measure the signal direction require much more sophistication to antenna design (which cannot be easily justified for location estimation only). Nevertheless, our first model provides with a simple yet precise estimator design such that subsequent designs can be referenced and compared.

B. Design II

Our second design relaxes some of the previous assumptions by requiring only the availability of an RSSI sensor, thus only relying on easy-to-measure properties. In this case, it is not possible for the receiver to estimate the relative position and velocity of the sender. Yet, the relative distance to the sender and the rate of the distance change over time Δt can be estimated. Figure 1 explains how the rate of distance change is related to the location and velocity of the nodes. Let us consider two nodes, n_0 and n_1 that are moving with absolute velocity $(v_{x_0},\, v_{y_0})$ and $(v_{x_1},\, v_{y_1})$ in some Descartes coordinate system. Let D_t be the distance between n_0 and n_1 at time t, and R_t be the rate of the distance change at t. Here, we define R_t to be positive if the nodes are moving away from each other and negative otherwise. In general, let θ_x and θ_y be the angles from the displacement line (*away* from the other node) moving counter-clockwise to the x and y components of the absolute velocity. R_t can be calculated as the sum of the portions from all four velocity segments as $R_t = v_{x_0} cos(\theta_{x_0}) + v_{y_0} cos(\theta_{y_0}) + v_{x_1} cos(\theta_{x_1}) + v_{y_1} cos(\theta_{y_1})$.

For the Kalman filter design, we denote the state variables to be $(D_t,\, R_t)$. Note that in reality R_t is not constant over time (even at a constant velocity) since both nodes move simultaneously. Nevertheless, the process model maintains that R is a constant, resulting in the following process update function: $D_{t+1} = D_t + R_t \Delta t$. The state transition matrix is therefore

$$\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

To assess the error that this assumption incurs, it is important to note that as the change to R_t decreases D_t increases. Because a link longevity estimate is more useful when the nodes are further away (i.e., when D_t is large), this assumption, though incorrect, should have minimal impact on the overall result (as it was verified by our simulation studies).

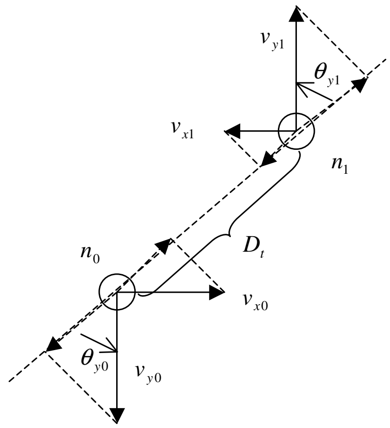

Fig. 1. Calculating R_t from velocity segments.

C. Design III

The third design - similarly to the second - requires only the availability of RSSI sensors at nodes. However, in this case we derive the new distance estimates based on a history of previous estimates. Since our underlying assumption is that the nodes move at constant speeds and headings, it is possible to derive the new distance estimate based on previous estimates. For the filter based on k previous states, where $k > 1$, the state space consists of the following variables: (D^2_{t-k}, …, D^2_{t-1}, D^2_t , S^2). The variables $D_{t-k}, \ldots, D_{t-1}, D_t$ are the distance estimates at time $t-k, \ldots, t-1,$ and t, respectively, while S is the *relative speed* between the two nodes. Given the relative velocity as v_x and v_y, S is simply $\sqrt{v_x^2 + v_y^2}$. The new state is derived under the assumption that v_x and v_y are constant.

The new state is calculated from the previous state as follows. A new estimate $\hat{D}^2_{t+1}$ can be calculated from each historical reading $D^2_{t-i}(1 \le i \le k)$ as well as the latest reading D^2_t. Since relative velocities are assumed to be constant, we can envision the receiver as stationary while the sender is moving at S to a fixed direction. From Figure 2, let A be the stationary location of the receiver, and B, C and D be the relative location of the sender at the time of $t-i$, t and $t+1$ respectively. Thus, $D_{t-i} = AB$, $D_t = AC$, and $\hat{D}_{t+1} = AD$. BC and CD can be derived from the assumption that S is constant and that the filter runs with a period of Δt. By solving the triangulation in Figure 2, a new estimate of $\hat{D}^2_{t+1}$ can be found as

$$\hat{D}^2_{t+1} = -\frac{D^2_{t-i}}{i} + \frac{i+1}{i} D^2_t + (i+i) S^2 \Delta t^2$$

Note that the above estimate $\hat{D}^2_{t+1}$ is obtained from D^2_t and a single previous reading of D^2_{t-i}. We can then repeat the above calculation for all $D^2_{t-i}(1 \le i \le k)$ and average a total number of k estimates. The new distance estimate D^2_{t+1} is therefore the following:

$$D^2_{t+1} = \sum_{i=0}^{k-1} \frac{-D^2_{t-k+i}}{k(k-i)} + \frac{D^2_t}{k} \cdot \sum_{i=0}^{k-1} \frac{k+1-i}{k-i} + \frac{S^2 \Delta t^2}{k} \cdot \sum_{i=0}^{k-1} (k+1-i)$$

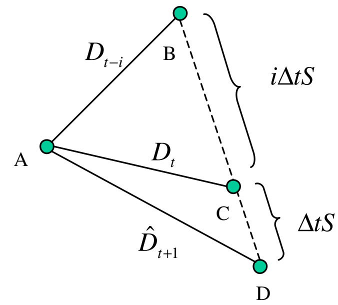

Fig. 2. Calculating D^2_{t+1} based on previous calculation of D^2_t and D^2_{t-i}.

The above equation translates to following state transition matrix:

$$\begin{bmatrix} 0 & 1 & 0 & \ldots\ldots & 0 \\ \vdots & 0 & 1 & \ldots\ldots & \vdots \\ \vdots & \vdots & & & \vdots \\ T_{k-1} & \ldots. & T_0 & V & W \\ 0 & \ldots\ldots\ldots & & 0 & 1 \end{bmatrix}$$

Here, $T_i = \frac{-1}{k(k-i)}$, $V = \frac{1}{k} \cdot \sum_{i=0}^{k-1} \frac{k+1-i}{k-i}$, and $W = \frac{\Delta t^2}{k} \cdot \sum_{i=0}^{k-1} (k+1-i)$.

By averaging the estimates from all k previous values, we expect the estimator be less prone to abnormal sensor readings. Meanwhile, it will takes longer to adapt to the change of movement pattern. Variable k provides this design with an additional parameter (besides the Kalman filter variances) to adjust the filter to the environment. As verified by simulations, the third design is expected to show better performance with a larger k when the node movement pattern is more predictable and the sensor reading is noisier.

III. Simulation

To evaluate the three link longevity estimators, we have implemented them around our own C++ based discrete event simulation engine. All our simulations involve two mobile nodes moving within a 2000m side-length square. The node movement model is based on the epoch model used in [1] with the following properties:

1) The entire movement path of the node is defined by a sequence of "epoch," i.e., $(e_1, e_2, \cdots, e_n)$.
2) The duration of each epoch is I.I.D. exponentially distributed with a mean of $1/\lambda$.
3) Within each epoch, the node moves at a constant velocity.
4) At the end of each epoch, nodes randomly select a new velocity vector. The direction of the movement is I.I.D. randomly chosen from a uniform distribution between 0 and 2π. The absolute value of the velocity is I.I.D. normally distributed with a mean μ of and a variance of σ^2.

Note that since immobile nodes do not cause any difficulties for our system, we eliminate the idle time between epochs from the original model, and thus the nodes are always on the move. For our simulations, we use $\mu = 10m/s$ and $\sigma^2 = 10$.

When a node hits the border of the square, its bounces back with the same angle much like a ping-pong ball. Of the two mobile nodes simulated, one is designated as the sender and the other as the receiver. The sender continuously transmits signal, and the receiver continuously monitors the incoming signal. The signal propagation model is given by $p = c \cdot d^{-2}$, in which the power of the received signal p is inversely proportional to the second power of the distance d. Here, c is an arbitrary constant. When the received signal power p is below a threshold p_{min}, it is considered too weak to be captured by the receiver thus the link breaks. For our simulations, we select $c = 10^6$ and $p_{min} = 1$. Note the c and p_{min} selection does not affect the overall simulation results, as long as the same values are used in the observation model of the filters. In fact, the same can be said about all other signal propagation models - all we require is a model that represents the receiving power as a function of distance. Given our signal model, the threshold $p_{min} = 1$ translates to a transmission range of 1000m when noise is not considered. To estimate the time when the link will be cut the receiver processes the sensor data every 0.1 second. The sensor data is then feed into all three estimators simultaneously to obtain their estimations for comparison.

Noise is incorporated based on the noise model in [8]. The model considers the fact that radio signals hardly ever propagate in line-of-sight path. Instead, they tend to bounce off from nearby structures along the way due to multipath fading and far field scattering. The actual distance of signal propagation at time t is given by $d_t = d'_t + m_t$, where d'_t is the geometric distance between the two nodes and m_t the extra distance covered due to signal reflection. m_t is recursively defined as

$$m_t = m_{t-1} + P_0 N(0, {\sigma_0}^2) + P_1 N(0, {\sigma_1}^2),$$

where P_0 and P_1 are the probability of the whether or not the signal bounces off a different structure, and N is a Gaussian distribution with a zero mean. Since the distance can change more drastically when the signal bounces off a different (than before) structure, it can be assumed that ${\sigma_0}^2 >> {\sigma_1}^2$. Furthermore, m_t should be non-negative for all t. For our simulations, we use $P_0 = 0.1$ and $P_1 = 0.9$.

A. Estimator Convergence

Figure 3 shows how the link longevity estimator of the three different designs converges in a typical scenario when the sender and receiver are pulling away from each other at constant velocities. The figure shows the error of the estimations as the filters analyze the incoming signals. Clearly, the first design converges the quickest due to its extra signal direction sensor. Of the estimators relying solely on the RSSI sensor, the estimators based on the third design take longer to converge than the estimator of the second design but they provide better estimations (due to the availability of previous estimates). Of the two different versions of the third design, the

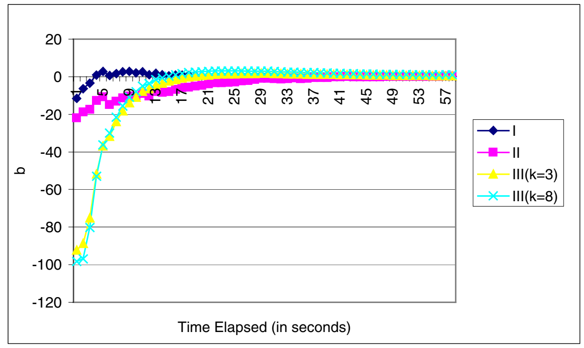

Fig. 3. Convergence of estimation error.

one that relies more on historical estimates ($k = 8$) converge a little quicker than the other which relies on less available previous estimates ($k = 3$).

B. Effect of Node Movement

Figure 4 and 5 show the effect of node mobility on the estimators by varying the mean of the epoch duration ($1/\lambda$). For this simulation, we keep the variances of the movement model constant at ${\sigma_0}^2 = 50$ and ${\sigma_1}^2 = 5$. We let the sender and receiver run for a duration of 10^6 seconds of simulation time. For each simulation run, we obtain the results by varying $1/\lambda$ from 20 to 200. As the mean ($1/\lambda$) epoch length increases, the node movement becomes more predictable, and thus the link longevity becomes easier to predict. To qualify the results, we define an estimation to be *acceptable* if it is within the range of $+/-10$ seconds when the link actually breaks. Furthermore, we denote $T_{success}$ to be the time *before* the actual link breakage when the different estimators converge to the acceptable range. There are also cases that the estimators never manage to produce acceptable estimations before link breakages. As such, we let $P_{success}$ be the probability of an acceptable estimation can be obtained in average. Essentially, $T_{success}$ indicates how *good* the estimations are, and $P_{success}$ indicates how *fast* they are obtained.

Figure 4 and 5 show the effect of node movement on $T_{success}$ and $P_{success}$. The figures indicate that the extra direction sensor in the first design greatly contributes to its superior performance. The second is not far behind from the first in terms of $P_{success}$. The two versions of the third design do not have a great performance in terms of $P_{success}$. However, they outperform the first design in $T_{success}$, indicating that in general the estimators based on the third design take longer to adjust to the movement updates. The figures also indicate that the more sensor data is processed the better the precision of the estimate will be. Furthermore, the gap in term of $P_{success}$ between $k = 8$ and $k = 3$ of the third increases slightly indicating that a larger k is better suited when node movement is more predictable.

C. Effect of Signal Noise

The effect of sensor noise on $T_{success}$ and $P_{success}$ is captured in Figures 6 and 7. For these simulations we have

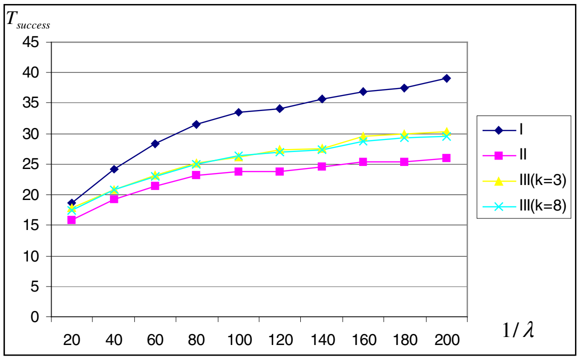

Fig. 4. The effect of node movement on $T_{success}$.

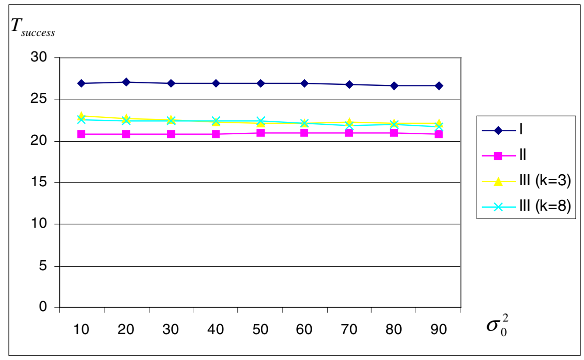

Fig. 6. The effect of noise on $T_{success}$.

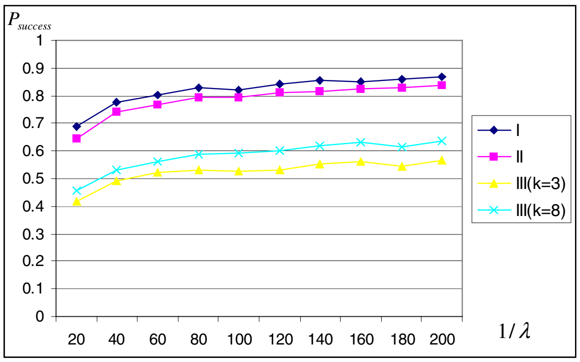

Fig. 5. The effect of node movement on $P_{success}$.

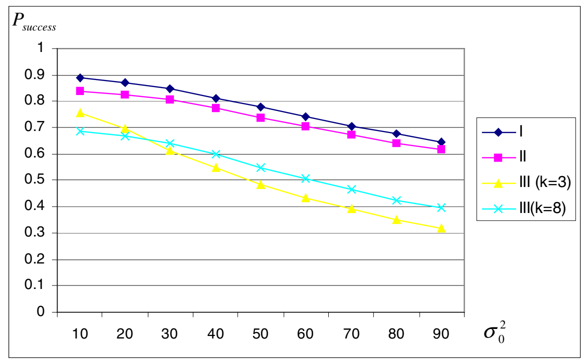

Fig. 7. The effect of noise on $P_{success}$.

set the mean of the epoch duration to $1/\lambda = 50$. We then vary the variance ${\sigma_0}^2$ of the noise model from 10 to 90. The other variance ${\sigma_1}^2$ is set to one tenth of the current ${\sigma_0}^2$ value.

Figure 6 shows that the quality of the estimations varies little as the noise increases. Meanwhile, Figure 7 indicates that it takes longer for the estimators to converge to the acceptable range with increasing noise. Since all of the estimator designs are based on Kalman filters, it is not surprising that they are rather resilient to noise, even though our noise model is not Gaussian. However, noise does have an effect on the estimators in that it takes longer to obtain acceptable estimations in a noisier environment. Comparing the two cases of the third design (where $k = 3$ and $k = 8$) we can observe that a larger k is better suited for a noisier environment.

IV. Conclusion

This paper has presented three different estimators that predict the link longevity in ad hoc mobile networks. These extended Kalman filter based estimators obtain their estimation by tracking the node movement by employing RSSI measurements. Since Kalman filters are known to be light-weighted, these estimators are especially suitable for mobile nodes with strict resource constraints. Our simulation demonstrates that all estimators are capable of producing useable estimations, even though their performance is subjected to the underlying predictability of the node movement and the sensor noise. The simulations also indicate that estimators fed with a radio signal direction sensor provide only slightly better estimates than estimators based solely on RSSI readings.

References

[1] A. B. McDonald and T. Znati, "A mobility-based framework for adaptive clustering in wireless ad-hoc networks," *IEEE Journal on Selected Areas in Communication* Special Issue on Wireless Ad-Hoc Networks, vol. 17, no. 8, August 1999.

[2] S. Jiang, D. He, and J. Rao, "A prediction-based link availability estimation for mobile ad hoc networks," *Proceedings of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, pp. 1745-1752, 2001.

[3] W. Su and M. Gerla, "IPv6 flow handoff in ad hoc wireless networks using mobility prediction," *Proceedings of the IEEE GLOBECOM'99*, pp. 271-275, Rio De Janeiro, Brazil, 1999.

[4] T. Liu, P. Bahl, and I. Chlamtac, "A hierarchical position-prediction algorithm for efficient management of resources in cellular networks," *Proceedings of the IEEE GLOBECOM'97*, Phoenix, Arizona, November 1997.

[5] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," *Proceedings of the IEEE GLOBECOM'01*, San Antonio, 2001.

[6] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AoA," *Proceedings of the IEEE INFOCOM*, San Francisco, 2003.

[7] N.B. Priyantha, A. Miu, H. Balakrishnam, and S. Teller, "The cricket compass for context-aware mobile applications," *Proceedings of the 6th ACM MOBICOM*, Rome, Italy, July 2001.

[8] P.J. Nordlund, F. Gunnarsson, and F. Gustafsson, "Particle filters for positioning in wireless networks," *Proceedings of the XI. European Signal Processing Conference (EUSIPCO)*, 2001.

[9] R.E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME – Journal of Basic Engineering*, pp. 35-45, March 1960.

[10] H.W. Sorenson, "Least-square esimation: from Gauss to Kalman," *IEEE Spectrum*, vol. 7, pp. 63-68, July 1970.

0-7803-7954-3/03/$17.00 ©2003 IEEE.