

Error Characteristics of Ad Hoc Positioning Systems (APS)

Dragoş Niculescu and Badri Nath
Rutgers University
DATAMAN Lab
{dnicules,badri}@cs.rutgers.edu

ABSTRACT

APS algorithms use the basic idea of distance vector routing to find positions in an ad hoc network using only a fraction of landmarks, for example GPS enabled nodes. All the nodes in the network are assumed to have the possibility of measuring: range, angle of arrival (AOA), orientation, or a combination of them. We give a lower bound for positioning error in a multihop network for a range/angle free algorithm, and examine the error characteristics of four classes of multihop APS algorithms under various conditions, using theoretical analysis and simulations. Analysis of range/angle free, range based, angle based, and multimodal algorithms shows a complex tradeoff between the capabilities used, the density of the network, ratio of landmarks, and the quality of the positions obtained.

Categories and Subject Descriptors

C.2.2 [Network protocols]: Applications; C.2.3 [Network operations]: Network Management

Keywords

Ad hoc networks, multihop positioning, ad hoc positioning system, APS, Cramér-Rao lower bound, error analysis

General Terms

Performance, Algorithms

1. INTRODUCTION

Position and orientation information of individual nodes in ad hoc networks is useful for both service and application implementation. Services that can be enabled by availability of position include routing and querying. At application level, position is required in order to label the reported data in a sensor network, whereas position and orientation enable tracking. Nodes in an ad hoc network may have local capabilities such as the possibility of measuring ranges to neighbors, angle of arrival (AOA), or global capabilities, such as

GPS and digital compasses (a compass shows orientation with respect to north).

One scenario involving sensor networks frequently mentioned in literature is that of aircraft deployment of sensors followed by in flight collection of data by simply cruising the sensor field. This and other meteorological applications are implicitly assuming that the data provided by the sensor is accompanied by the sensor's position. It is thus possible to attach the sensed information to a geographical map of the monitored region. If this is an absolute necessity for making sense of the observed data, accurate position might also be useful for routing and coordination purposes.

For some ad hoc networks, algorithms such as Cartesian routing [1], geocast [2], or GoAFR [3] enable routing with reduced or no routing tables at all, and are appropriate for devices like the Rene mote [4], with only half a kilobyte of RAM. Location Aided Routing [5] is an improvement that can be applied to some ad hoc routing schemes when position is available, by limiting the search for a new route to a smaller destination zone. Positioning and orientation algorithms are appropriate for indoor location aware applications, when the network's main feature is not the unpredictable, highly mobile topology, but rather temporary and ad hoc deployment. These networks would not justify the cost of setting up an infrastructure to support positioning, like proposed in [6], [7], or [8].

When deploying an ad hoc network that uses positions at application or service levels, the important decisions to be made are with respect to the node density, availability of landmarks, and node capabilities versus the desired quality of the positions. "One hop" solutions, such as GPS, in which each node is within communication range with the landmarks, are desirable for their simplicity, but may not be applicable in many setups because of form factor, power, cost, or line of sight conditions. Multihop solutions, which make the object of this study, while being more appropriate with respect to the mentioned conditions, face problems of error propagation, and provisioning.

Error buildup is inherent in a multihop environment because angle and distance measurements are affected by error at each hop. Range measuring methods include signal strength, TDOA (time difference of arrival) based on the six order of magnitude difference between ultrasound and RF signals, and UWB. Angle measurements are provided by antenna arrays, or by TDOA methods, as shown in the prototype of the Cricket compass [9].

Provisioning of large ad hoc networks includes not only the networking related aspects, but the management of the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobiHoc'04, May 24–26, 2004, Roppongi, Japan.

Copyright 2004 ACM 1-58113-849-0/04/0005 ...\$5.00.

error, when positioning is needed. As we show in this paper, there is a tradeoff between capabilities of the nodes (angle and range measuring hardware), the density of the network, the placement of landmarks, and the quality of the positions obtained. By analyzing four multihop algorithms that make use of different capabilities - range/angle free, angle, range, angle+range - we examine the relationship between position error and the way the network is provisioned.

This paper makes two contributions: First, we derive a Cramér-Rao lower bound for positioning error of a multi-hop range/angle free positioning algorithm (APS/*DV-hop*) - equation (8). Its importance is in bounding the position error achievable by this multihop algorithm for a given network setup. The simulated algorithm closely follows the trend of the theoretical bound. Second, we analyze a multimodal algorithm that uses both angle and range measurements. Although it has higher hardware requirements, its behavior is predictable through analysis, and its performance is bounded (literally) in the parameter space by three other algorithms: range/angle free, range based, and angle based. Its complex relationship with the other analyzed algorithms provides insights with respect to the quality of hardware needed to obtain high quality positions in a multi-hop environment (Figure 10). An important insight is that for the analyzed algorithms, if the statistical measurement error is beyond a certain limit, then it is no better than not measuring at all. When provisioning an ad hoc network in terms of density or measurement hardware used, this result helps in choosing a solution that gives the lowest positioning error for the given conditions.

The remainder of the paper is organized as follows: next section has a short review of APS family of algorithms, which are the subject of this analysis. Section 3 introduces error models for trilateration (3.2), *DV-hop* (3.4), and *DV-position* (3.5) algorithms. In section 4 we comparatively analyze four algorithms with different hardware requirements. For the better flow of the presentation, most mathematical derivations are detailed in four appendices A-D, at the end of the paper.

2. APS ALGORITHMS

The assumptions made by most multihop solutions are that some measuring hardware gives some reading with respect to a node's neighbor. A node may measure *distance* or *range* to its neighbors, with a given accuracy. Alternatively, it may measure *bearing* with respect to the node's axis, or *angle* from which the signal is received, some times called AOA (angle of arrival). In extreme cases, a node has neither of the mentioned capabilities and has to rely on simple connectivity only. In the case it has them both, it can make more precise estimations, and we refer to this as "multimodal".

The positioning problem to be solved is: *given imprecise bearing / range measurements to neighbors in a connected ad hoc network where a small fraction of the nodes have self positioning capability, find positions for all nodes in the network*. The difficulty of the problem stems from the fact that the capable nodes (landmarks) comprise only a small fraction of the network, and most regular nodes are nodes are not in direct contact with enough landmarks.

Ah Hoc Positioning System (APS) [10, 11] is a hybrid between two major concepts: distance vector (DV) routing, and beacon based positioning (GPS). What makes it similar

to DV routing is the fact that information is forwarded in a hop by hop fashion, independently with respect to each landmark. What makes it similar to GPS is that eventually each node estimates its own position, based on the landmark readings it gets. The APS concept has been shown to work using simple connectivity, range measurements [10], angle measurements [11], and a combination of them [12].

All propagations work very much like a mathematical induction proof. The fixed point: nodes immediately adjacent to a landmark get their bearings / ranges directly from the landmark. The induction step: assuming that a node has some neighbors with bearing / range for a landmark, it will be able to compute its own bearing / range with respect to that landmark, and forward it further into the network. What remains to be found is a method to compute this induction step, for any combination of local capabilities: connectivity only, ranging, angle measurements, and multimodal = angle + range.

If for some reason a node does not get enough ranges / bearings in order to triangulate / trilaterate for a position, it could wait for the neighbors to successfully position themselves and either use local measurements in order to get a position, or simply use a weighted average with the positions of those neighbors. Even if position is available at a node, smoothing with the positions of the neighbors has been reported beneficial in certain cases [13, 14, 15].

2.1 DV-hop propagation method

This is the most basic scheme, which doesn't require any hardware to measure ranges or angles to neighbors. It only relies on the connectivity of the underlying graph and it comprises of three non-overlapping stages.

(I) First, it employs a distance vector exchange so that all nodes in the network get shortest paths, in hops, to the landmarks. Each node maintains a table $\{X_i, Y_i, h_i\}$ and exchanges updates only with its neighbors. X_i, Y_i are the coordinates of landmark i , and function here as a name for the landmark as well. h_i is the shortest distance in hops to that landmark. This phase is the classical Bellman Ford distributed shortest path algorithm that is found in internet routing (RIP and BGP).

(II) In the second stage, after it cumulates distances to other landmarks, a landmark estimates an average size for one hop, which is then deployed as a correction to the nodes in its neighborhood.

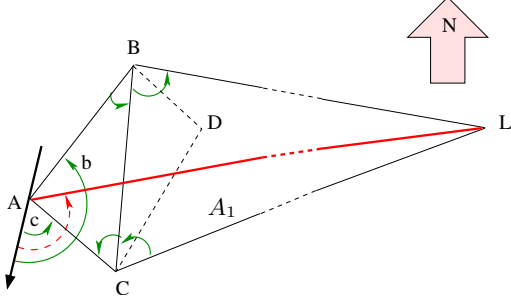
(III) After receiving the correction, an arbitrary node may estimate distances to landmarks, in meters, which can be used to perform the trilateration. The correction a landmark (X_i, Y_i) computes is

$$c_i = \frac{\sum \rho_j}{\sum h_j}, i \neq j, \text{ all landmarks } j \text{ heard by } i \quad (1)$$

where h_j is the shortest distance, in hops from node i to node j , and $\rho_j = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}$ is the straight line distance between i and j .

The advantages of *DV-hop* are that it is completely distributed and localized, and it only relies on connectivity. The drawback is that it will only work for isotropic networks, that is, when the properties of the graph are the same in all directions, so that the corrections that are deployed reasonably estimate the distances between hops. Also, shortest path to the landmark is assumed to be obtained by the distance vector procedure. This might not be the case in vari-

Figure 1: *Euclidean, DV-bearing propagation*



able duty cycle sensor networks, mobile networks, or when processing is slow. When this happens a landmark proceeds to stage (II), or a node to stage (III) with a less than optimal path. These problems can be alleviated with careful provisioning of the timeouts between the stages in order to accommodate proper operating ranges for the density and the size of the network.

2.2 Euclidean and DV-radial

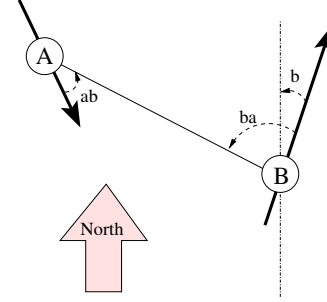
The range based scheme *Euclidean* works by propagating the estimated Euclidean distance to the landmark, so this method is the closest to the nature of GPS. *Euclidean* makes use of range measurements to neighbors. The angle based scheme *DV-bearing* propagates in a similar fashion, but each node computes and forwards a bearing - the angle under which it "sees" a landmark. It therefore needs angle measurement hardware, and can benefit from the presence of a compass in each node of the network.

(I) An arbitrary node A needs to have at least two neighbors B and C which have estimates - angles or ranges - for the landmark L (Figure 1). For *Euclidean*, A also has measured estimates of distances for AB , AC , and BC , so there is the condition that: either B and C , besides being neighbors of A , are neighbors of each other, or A knows distance BC , from being able to map all its neighbors in a local coordinate system. For the quadrilateral $ABCL$, all the sides are known and one of the diagonals, BC is also known. This allows node A to compute the second diagonal AL , which in fact is the Euclidean distance from A to the landmark L . For *DV-bearing*, A knows all the angles in triangles $\triangle ABC$ and $\triangle BCL$, indicated with continuous arrows. This allows for the computation of the bearing of A with respect to L , indicated with a dashed arrow. This step is the induction step, which propagates the capability from nodes B and C to node A which is one hop further from landmark L .

(II) Once a node has ranges or angles to three landmarks, it may, by itself, estimate its position, by applying a trilateration procedure for the range based, or a triangulation procedure for the angle based method. The trilateration procedure is described in section 3.1.

The advantage of the range / angle based methods is that they provide better accuracy under certain conditions, and there is only one communication stage (no corrections). On the down side, they require additional hardware, when compared to *DV-hop*, which makes them susceptible to measurement error.

Figure 2: *DV-position propagation method*



2.3 DV-position

If ranging and angle measurements (AOA) are both available at all nodes, *DV-position* can be used as a method that makes simultaneous use of both capabilities for increased accuracy.

If compasses are available at landmarks, neighbors of landmarks can position themselves in one step using *DV-position*. In Figure 2, if landmark B has a compass and can measure angle \hat{b} , and also AOA capability to measure the angle \hat{ba} at which A is seen, then it is possible to compute the equation line on which A is placed. If the range AB is known, node A can find its own position. The absolute orientation of A can also be found, as $2\pi - (\hat{ba} + \pi - \hat{ab}) + \hat{b}$. Using further propagation, node A can then behave as a less accurate landmark for further away nodes.

DV-position is a much simpler scheme than the previous ones, but requires more hardware - both angle and range measurements. Nevertheless, its analysis is meaningful since it provides some insights to the behavior of the other algorithms and the quality of the hardware that is needed to produce good positions. Prototypes of small form factor nodes that feature the AOA capability (such as the Cricket Compass [9]), and good ranging capability (such as Medusa node [14]), make multimodal positioning a feasible solution.

3. POSITIONING ERROR ANALYSIS

3.1 Trilateration review

In Global Positioning System (GPS [16]), **trilateration** (in reality multilateration) uses range estimates to at least three known satellites to find the coordinates of the receiver, and four satellites to also find the clock bias of the receiver. For our ad hoc positioning purposes, we are using a simplified version of the trilateration, as we only deal with distances, and there is no need for clock synchronization. The trilateration procedure starts with an a priori estimated position that is later corrected towards the true position. Let $\hat{\mathbf{x}}$ be the estimated position, $\mathbf{x} = [x \ y]^T$ the real position, $\mathbf{x}_i = [x_i \ y_i]^T$ the known coordinates of the satellites, $\rho = \|\mathbf{x}_i - \mathbf{x}\|$ the true range, and $\hat{\rho}_i = \|\mathbf{x}_i - \hat{\mathbf{x}}\|$ the measured range to satellite i . The problem is to find \mathbf{x} given ρ_i and \mathbf{x}_i . We start with an initial guess $\hat{\mathbf{x}}$, which provides associated $\hat{\rho}_i$. The distance equation to each satellite is $\rho_i = \sqrt{(x_i - x)^2 + (y_i - y)^2}$. The correction of the range, $\Delta\rho$ is approximated linearly using Taylor expansion. If \hat{J}_i is the unit vector of $\hat{\rho}_i$, $\hat{J}_i = \frac{\mathbf{x}_i - \hat{\mathbf{x}}}{\hat{\rho}_i}$ and $\Delta\mathbf{x} = \hat{\mathbf{x}} - \mathbf{x}$, then the approximate of the correction in the range is: $\Delta\rho =$

$\hat{\rho}_i - \rho_i \simeq \hat{J}_i \cdot \Delta \mathbf{x}$. Performing the above approximation for each satellite independently leads to a linear system in which the unknown is the position correction $\Delta \mathbf{x} = [\Delta x \ \Delta y]$:

$$\Delta \rho = J \Delta \mathbf{x}$$

$$\begin{bmatrix} \Delta \rho_1 \\ \Delta \rho_2 \\ \dots \\ \Delta \rho_n \end{bmatrix} = \begin{bmatrix} \hat{J}_{1x} & \hat{J}_{1y} \\ \hat{J}_{2x} & \hat{J}_{2y} \\ \dots & \dots \\ \hat{J}_{nx} & \hat{J}_{ny} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (2)$$

When all measurements have the same uncertainty, the system is solved as with OLS: $\Delta \rho = J \Delta \mathbf{x} \Rightarrow \Delta \mathbf{x} = J^+ \Delta \rho$, where $J^+ = (J^T J)^{-1} J^T$ is the pseudoinverse of J . If independent uncertainties σ_i are available for range estimates, the above system is solved using WLS with weights $W = \text{diag}\{\frac{1}{\sigma_i^2}\} = \text{Cov}[\Delta \rho]^{-1}$, J^+ becomes

$$J^+ = (J^T W J)^{-1} J^T W \quad (3)$$

After each iteration, the corrections Δx and Δy are applied to the current position estimate. The iteration process stops when the correction in position is below a chosen threshold. The uncertainties W are obtained as the estimated errors of the ranges computed by *DV-hop* and *Euclidean*. If W is known, it is actually possible to give a lower bound for the covariance of \mathbf{x} , or position error, that can be achieved by trilateration.

3.2 CRLB for trilateration

The Cramér-Rao lower bound is method that sets a lower bound on the variance of *any* unbiased estimator. Its main merit is that it provides a benchmark against which to evaluate the performance of an estimator. In our case the trilateration problem is cast as an estimation problem by considering the true position \mathbf{x} as a parameter to be estimated. The distribution of errors of distances to landmarks $\hat{\rho}$ is assumed to be known at this point in the presentation.

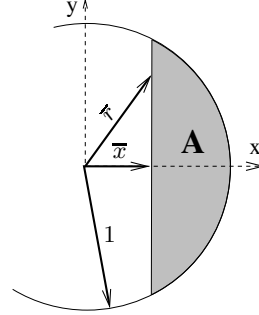
Using notations introduced in the previous section, we now estimate the error in the obtained position given the range to landmark estimation error. This approach is applicable to all the algorithms that use trilateration as a final phase (*DV-hop* and *Euclidean*). The hop by hop nature of multihop algorithms always produces normal errors for a sufficiently large number of hops – we confirmed this by simulation for all algorithms mentioned here.

In appendix B, we show that the CRLB of the variance in the estimated parameter x is:

$$\text{CRLB}(\text{Cov}[\mathbf{x}]) = (J_0^T W J_0)^{-1} \quad (4)$$

where $J_0 = \frac{\mathbf{x}_i - \mathbf{x}}{\rho_i}$, using the true positions and ranges. This means that for a known setup, when landmark constellation is given as \mathbf{x}_i , true position as \mathbf{x} , and errors in ranges as W , the best estimation error will be limited by $\text{Var}[x] \geq \text{CRLB}(\text{Cov}[\mathbf{x}])_{11}$ and $\text{Var}[y] \geq \text{CRLB}(\text{Cov}[\mathbf{x}])_{22}$. While we don't have an analytical model for *Euclidean's* uncertainties W , they are propagated through the network and used in the triangulation phase (details can be found in [10]).

Figure 3: Progress per hop for MFR policy



3.3 DV-hop range error

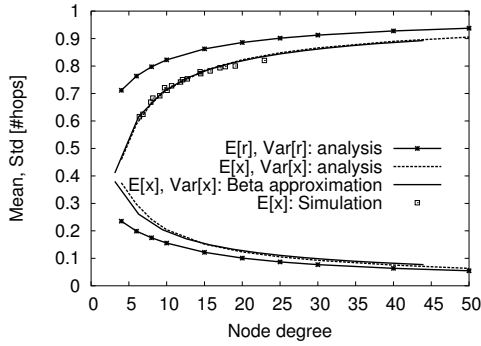
In *DV-hop*, the main source of range error stems from the fact that a node translates the length of shortest path to a landmark into a Euclidean distance by assuming that every hop produces progress c_i - equation (1). Greedy geographic forwarding has been shown in [17] to produce paths with low dilation, that is, good shortest path approximates. This means that if we apply a geographic forwarding policy such as MFR [18] in a **uniform, dense** network, the progress made by each hop is a good approximation for c_i . If the network is not dense enough, areas with low connectivity result in either dropping the packet by the greedy forwarding scheme, or requiring the use of an obstacle avoiding algorithm. The analysis provided here only covers the cases when the network is dense enough so that simple greedy forwarding succeeds. High density however, is a requirement for a network to be connected in the case nodes are distributed randomly in a large network [19]. Kleinrock and Silvester [18] show that for greedy forwarding, the progress only depends on the node density. For *DV-hop*, this implies that in uniform networks, nodes may locally estimate the correction based on their local perception of density, and thus eliminate the need for the second stage of the algorithm.

Assuming a Poisson spatial distribution of nodes with rate λ , and a wireless range of radius 1, the expected number of neighbors of a node will be $\pi \lambda$. Throughout the paper we will use both the average number of neighbors, or the rate λ when referring to density. A node chooses a next hop that produces most progress \bar{x} towards a destination on the horizontal axis (Figure 3). There are no nodes in area A , which allows for the following derivation of the distribution of \bar{x} .

$$\begin{aligned} A(\bar{x}) &= 2 \int_{\bar{x}}^1 \sqrt{1-t^2} dt = \arccos(\bar{x}) - \bar{x} \sqrt{1-\bar{x}^2} \\ F_{\bar{X}}(\bar{x}) &= \{\bar{X} \leq \bar{x}\} = e^{-\lambda A(\bar{x})} \\ f_{\bar{X}}(\bar{x}) &= \{\bar{X} = \bar{x}\} = -\lambda A'(\bar{x}) e^{-\lambda A(\bar{x})} \\ &= 2\sqrt{1-\bar{x}^2} e^{-\lambda(\arccos(\bar{x}) - \bar{x}\sqrt{1-\bar{x}^2})} \end{aligned} \quad (5)$$

Unfortunately, there are no closed forms for the first moments $E[\bar{x}]$ and $V[\bar{x}]$ of this distribution, but it can be approximated as a Beta distribution. In Figure 4, numerically evaluated variance $V[\bar{x}]$ and expectation $E[\bar{x}]$ as functions of density λ are shown together with the approximation to beta distribution. Also in Figure 4 are shown values of c_i

Figure 4: Mean and standard deviation of \bar{x}, \bar{r}



obtained from simulation¹ in a network with 1000 nodes with increasing densities, which match very closely values obtained from the numerical evaluation for the respective densities. Since these values only depend on the density of the network, we precomputed a table with all the values necessary for the experiments in this paper, namely for densities up to 20 neighbors.

In *DV-hop*, range estimates are obtained as $\rho_i = h_i c_i$, where h_i is the number of hops to landmark i , and c_i is approximated by $E[\bar{x}]$. If geographic forwarding model is used to approximate shortest path to the landmark, after jumping h_i times (assumed independent), the obtained range is $E[\rho_i] = h_i E[\bar{x}]$ and its variance

$$V[\rho_i] = h_i V[\bar{x}] \quad (6)$$

Referring to Figure 3, we now explore the behavior of \bar{r} , the actual distance traveled in one hop by the MFR policy. Its distribution is actually needed for the analysis of *DV-position*, but its characterization is based on the pdf of \bar{x} . The pdf of \bar{r} is obtained as a sum of probabilities of a next hop, summed over the entire circle of radius \bar{x} .

$$f_{\bar{r}}(\bar{r}) = \int_0^{2\pi} \frac{f_{\bar{x}}(\bar{r} \cos t)}{2\sqrt{1 - \bar{r}^2 \cos^2 t}} dt \quad (7)$$

\bar{r} is not available in closed form, but in Figure 5 we can see its behavior through numerical evaluation, for two different densities. As expected, larger jumps are much more likely than short jumps, which means larger errors if \bar{r} is to be used as a measure of distance. In Figure 4, the mean and standard deviation of \bar{r} are shown as a function of density.

In the remaining sections the model for range error is assumed to be linear in the number of hops. We also assume that the model extends to continuous distances, that is, the variance in range error is linear with the distance

¹We implemented our own null MAC, unit disk graph, event based simulator with minimal support for positioning. The simulator uses an error model for angle and range measurements, and a broadcast primitive with perfectly circular coverage. The decision not to use an existing simulator such as *ns-2* was based on: 1. the need to use large topologies (10000 nodes), 2. the paper's focus on the quality of the positions rather than on the communication aspects of the positioning protocols, 3. the reduced number of required primitives: node broadcast and DV shortest paths.

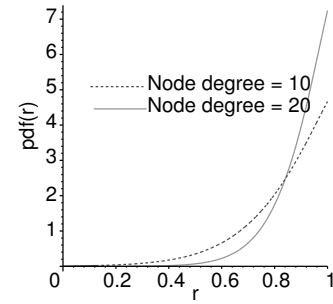


Figure 5: Probability distribution of \bar{r} for two node densities

$V[\rho_i] = \frac{\rho_i}{E[\bar{x}]} V[\bar{x}]$. This allows us to estimate the range uncertainties W that can be merged in trilateration CRLB (4).

3.4 CRLB for DV-hop positioning

The last step of the analysis is to evaluate how range measurements are assembled into a position estimate, and what the covariance of this estimate is. We assume that *DV-hop* uses a TTL value of h for all the aspects of its protocol, and that the fraction of nodes that are landmarks is f . This entails that the landmarks are assumed a spatial Poisson process with the rate $f\lambda$. A node is then able to contact on average $n = \pi(hE[\bar{x}])^2 f\lambda$ landmarks situated in a circle of radius $hE[\bar{x}]$. After applying the trilateration procedure (section 3.1), the covariance of the position estimate is bounded using (4). The actual derivation is detailed in appendix C, and the obtained covariance is:

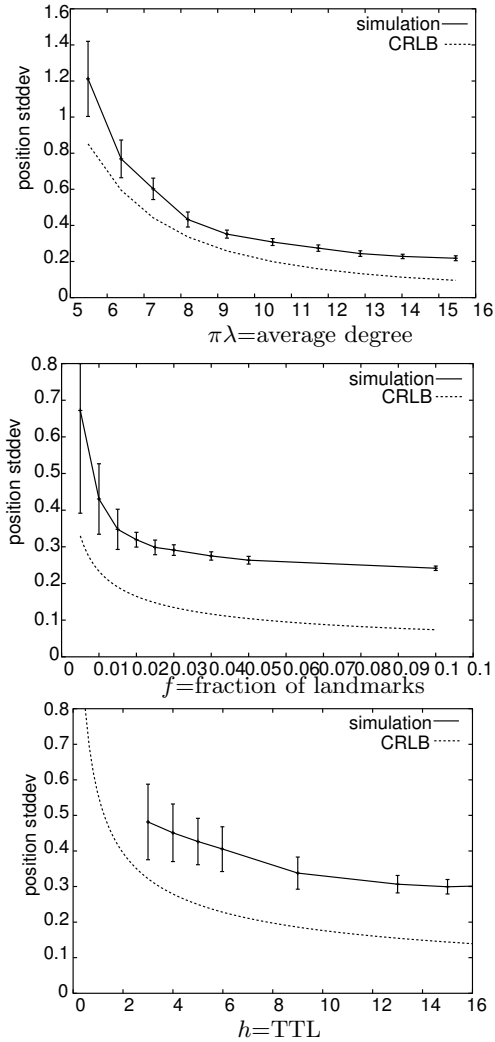
$$CRLB(Cov[r_u]) = \frac{1}{f\pi h \lambda E^2[\bar{x}]} I_2 \quad (8)$$

The CRLB of the positioning error covariance is inversely proportional to the fraction of landmarks deployed and to the TTL used by the method, but also depends on the density of deployment λ (second fraction). Even if $V[\bar{x}]$ and $E[\bar{x}]$ cannot be shown in closed form as functions of λ (5), their approximations using the Beta distribution can provide an estimate of the error for provisioning purposes.

In order to verify the behavior of the algorithm when compared to the lower bound, we traced the three parameters f , h , and λ using simulation. Figure 6 shows the values of the x coordinate of position standard deviation ($Cov[r_u]_{11}$ in hops) obtained by varying each parameter and keeping the other two constant, both in simulation, and using equation (8). The values used were $\lambda = \frac{10}{\pi}$, $h = 15$, $f = 0.025$ and the simulation was run in a network with 1000 nodes. Although the bound is not tight, it properly describes the trend of positioning performance with respect to the three parameters. The difference is a constant factor, some of the which is due to our simplifying approximations. One such approximation is the independence assumption between paths to different landmarks, which may not always be disjoint. This however, is only degrading the bound, not increasing it above the tight lower bound, which means *DV-hop* is actually closer to the bound than shown in our estimation.

A closed form of the lower bound can be determined if we don't use weighting in the multilateration process by setting

Figure 6: *DV-hop* positioning error as a function of density, landmark ratio, and TTL - simulation vs. lower bound analysis: $\sqrt{CRLB(\lambda)}$ $\sqrt{CRLB(f)}$ $\sqrt{CRLB(h)}$



$W = I$. In this case, equation (19) becomes

$$J_0^T W J_0 = \begin{bmatrix} \sum \frac{x_i^2}{\rho_i^2} & \sum \frac{x_i y_i}{\rho_i^2} \\ \sum \frac{x_i y_i}{\rho_i^2} & \sum \frac{y_i^2}{\rho_i^2} \end{bmatrix} \quad (9)$$

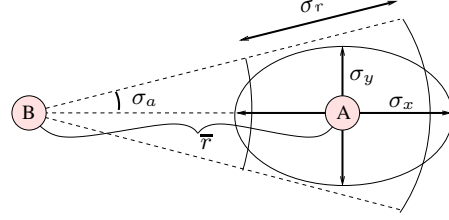
$$CRLB(Cov[r_u]) = \frac{2}{n} I_2$$

where n is the number of landmarks contacted by a node. It is of course a weaker bound (lower than (8)), but it gives an easier interpretation of the achievable accuracy, by confirming the intuition that position estimates are improved when more landmarks are provided.

3.5 DV-position error

In order to propagate the uncertainty at each step, we assume that each node's position has an associated uncertainty covariance matrix. In order to assume normal spatial

Figure 7: Normal approximation of uncertainty



errors, angle of arrival (AOA) errors and range errors are approximated, assuming their independence. In Figure 7, assume the uncertainty at node B is U_B , and node A , using an AOA reading with uncertainty $\sigma_a^2 = V[a]$ and a range reading with uncertainty $\sigma_r^2 = V[r]$ infers its position with uncertainty U_A .

$$U_A = U_B + \begin{bmatrix} \frac{\bar{r}_x}{\bar{r}_y} \\ \frac{\bar{r}_x}{\bar{r}_y} \end{bmatrix}^T \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \begin{bmatrix} \frac{\bar{r}_x}{\bar{r}_y} \\ \frac{\bar{r}_x}{\bar{r}_y} \end{bmatrix} \quad (10)$$

When angular errors are small, the eigenvalues of the covariance matrix are given by: $\sigma_x^2 = \sigma_r^2$ and $\sigma_y^2 = r^2 \sin^2(\sigma_a)$, where r is the actual range measured. The unit eigenvectors \bar{r}_x and \bar{r}_y are given by the direction of the vector \bar{r} , respectively its normal, or they can be seen as the columns of a rotation matrix. One advantage of expressing the uncertainty using a covariance matrix is that it can easily be cumulated as the position gets propagated. A second advantage is that positions obtained from different landmarks can be merged in order to get the best position estimate, as well as an estimate of its uncertainty. If $[x_i \ y_i]$ is the estimate from landmark i , and U_i is the uncertainty accumulated along the path from landmark i , the final uncertainty is:

$$U = \left(\sum_{i=1}^n U_i^{-1} \right)^{-1} \quad (11)$$

and the position estimate:

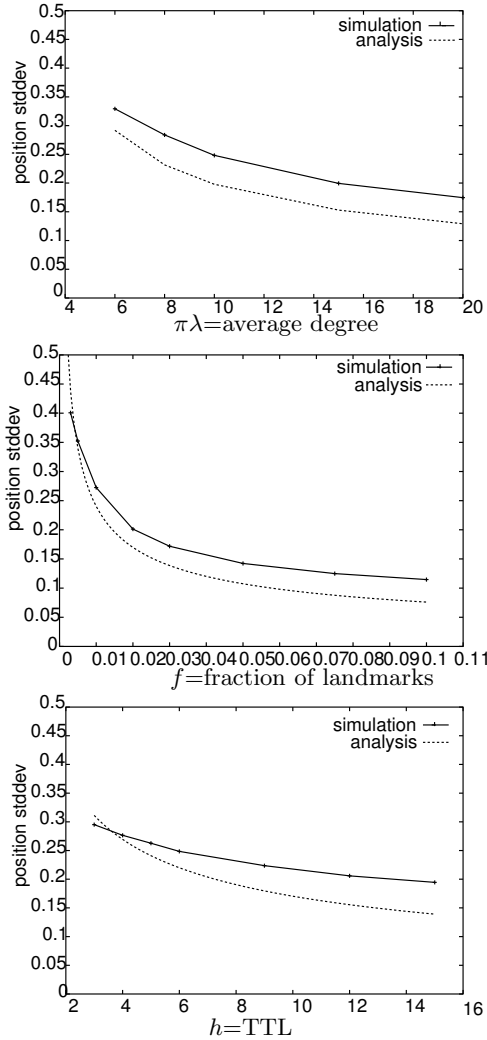
$$[x \ y] = \left(\sum_{i=1}^n [x_i \ y_i] U_i^{-1} \right) U \quad (12)$$

This is a simplified form of a Kalman filter that combines several "readings" with associated uncertainties, in order to produce the most likely estimate and its uncertainty. Intuitively, the uncertainty grows with each error affected measurement (10), but can be reduced if several landmarks are used (11). The actual position estimate is in fact a weighted estimate considering all the independent estimates and their uncertainties (12).

Since *DV-position* uses shortest path in propagate position estimates, the range measured between two successive nodes may be approximated as a function of deployment density using the random variable for range advancement \bar{r} in the MFR algorithm (equation (7) and Figure 4).

In order to get a closed form of the final uncertainty U , we make the simplifying assumptions that angular errors and range errors produce a circular error ellipse. This avoids the need to apply rotation matrices at each step of the analysis when moving from a landmark to a node, and also to

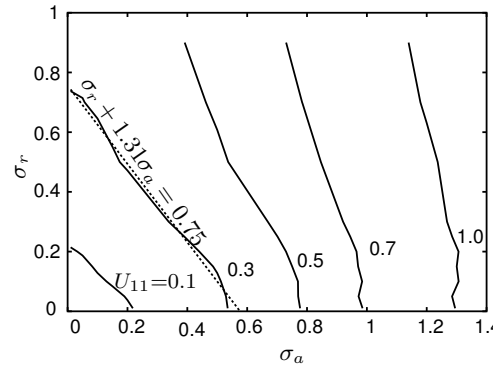
Figure 8: *DV-position* positioning error as a function of density, landmark ratio, and TTL for a circular error covariance ($\sigma_r = \sigma_a = 0.32$) - simulation vs. analysis: $\sqrt{U_{11}(\lambda)}$ $\sqrt{U_{11}(f)}$ $\sqrt{U_{11}(h)}$



apply rotation matrices to combine estimates from different landmarks. Taking $\sigma_r = \sigma a$ assumes that σ is a relative deviation, meaning that uncertainty in measured range increases with the actual range, which is consistent with behavior of most radio hardware. Taking $\sin(\sigma_a) = \sigma$ just looks at particular combination of errors, for example, for a standard angular deviation of $5.72^\circ \simeq 0.1$ a range error with a standard deviation of 10% of the measured range would be used. Again, this circular covariance is assumed just for the purposes of obtaining a closed form position error, as the actual algorithm operates with arbitrary covariances. Using the same assumptions of Poisson distribution of nodes, and TTL limited operation of APS, the derivation of U is detailed in appendix D and shown to be:

$$U = \frac{1}{2f\pi h} \sigma^2 \frac{E[\bar{r}^2]}{\lambda E^2[\bar{x}]} I_2 \quad (13)$$

Figure 9: *DV-position* position error $\sqrt{U_{11}}$: the error surface is projected onto the plane of parameters. The 0.3 isoline indicates performance of DV-hop in the same network ($h = 15, f = 0.01, \lambda = \frac{10}{\pi}$).



Taking into account that σ is relative to the range measured, and $E[\bar{r}^2] = V[\bar{r}] + E^2[\bar{r}]$, relation (13) is similar to the CRLB derived for *DV-hop* (8), exhibiting the same type of dependence on the TTL h , fraction of landmarks f , and density λ .

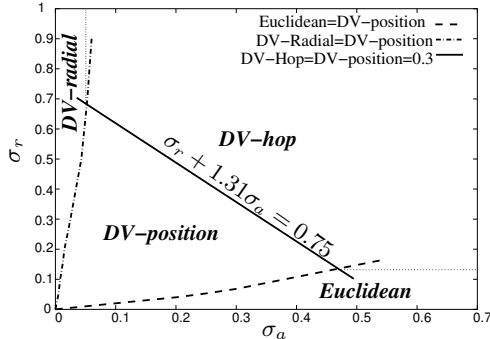
We simulated *DV-position* in a large network with 10000 nodes in order to verify the validity of (13). In Figure 8 one parameter is varied, while keeping the other two constant at $\sigma = 0.32, h = 5, f = 0.01, \lambda = \frac{20}{\pi}$ - the vertical axis shows the standard deviation of the x coordinate of the achieved position, in hops. The small difference between the analysis and the simulation is justified by the possible lack of independence assumed by (11) and (12). Even when a high degree is used, 20 in this case, shortest paths from a node to faraway landmarks may share one or more edges. We confirmed this by tracking the values of U_i (20), or by using a single landmark, both of which were closely predicted by simulation.

We haven't yet developed an error model for *Euclidean*, although through simulation we found that ranges to landmarks are normally distributed. This means that the same CRLB used for the trilateration (4) can be used, which would likely provide uncertainty estimates similar to (8) and (13). *DV-bearing* however, is using triangulation as the final phase, and while its analysis is not included in this study, simulation based results are used in the following discussion, as a means to cover all options for the hardware capabilities.

4. DISCUSSION

A closed form for the general case *DV-position* uncertainty U (11) would be more laborious to obtain, therefore we evaluated the behavior of the algorithm by simulation, also to accommodate comparison to the other two algorithms for which we don't have closed form uncertainties (*Euclidean* and *DV-bearing*). We tracked the error produced by the algorithm for all combinations of AoA error and range error for values of $\sigma_r = 0..0.9$ and $\sigma_a = 0.. \frac{\pi}{2}$ in the same network of 10000 nodes. We now fix the three parameters used in the previous section at $f = 0.01, h = 15, \lambda = \frac{10}{\pi}$, and look at the algorithm's behavior with respect to range error σ_r and angle error σ_a . The error surface corresponding

Figure 10: Partitioning of parameter space between 4 algorithms: *DV-hop* (range/angle free), *DV-radial* (angle based), *Euclidean* (range based), and *DV-position* (multimodal: range and angle based).



to this combinations is projected into the parameter space with a few error levels indicated in Figure 9. For example, the curve labeled $U_{11} = 0.1$ indicates combinations of standard deviations for angle and range measurements which produce a positioning error standard deviation of 0.1 hops. An interesting isoline is the one for positioning error of 0.3 because that is the performance achieved by *DV-hop* in the same network. This curve almost fits a straight line that partitions the parameter space between the two algorithms. There is no point in using *DV-position* to get worse errors than *DV-hop* gets, the latter doing so without any measurement hardware. For this particular conditions, precision of the hardware must satisfy the inequality $\sigma_r + 1.3\sigma_a < 0.75$ in order to use anything else than a range free algorithm like *DV-hop*. Of course, this demarcation line will be different for different conditions.

In order to further partition the parameter space, we ran *DV-radial* (an improved version of *DV-Bearing*) and *Euclidean* in the same network, in order to ascertain in which situations these two algorithms perform better than *DV-position* and *DV-hop*. The same surface used to generate the isolines in Figure 9 is now intersected with the curves produced by *DV-radial* and *Euclidean*. The resulted intersection lines span different error levels, but their projection into parameter space is indicated in Figure 10. The dashed line marks points (σ_a, σ_r) in which *DV-position* using angle errors of σ_a and range errors of σ_r performs as well as *Euclidean*, which only makes use of range measurements with the same error σ_r . Due to the increased sensitivity to propagated errors, *DV-radial* and *Euclidean* use parameters $f = 0.05$, $h = 8$, $\lambda = \frac{10}{\pi}$.

The common line of *DV-hop* and *DV-position* is the same from Figure 9, an isoline indicating positioning error of 0.3. This partitioning of the parameter space shows that in absence of highly performant measuring hardware for **both** angles and ranges, range free algorithms such as *DV-hop*, provide the best performance tradeoff. On the other hand, existing prototype hardware, such as the Cricket compass [9] provide AoA with an estimated $\sigma_a = 0.1 \simeq 5^\circ$ for certain conditions. This is not enough to choose the angle only algorithm *DV-radial*, but may be enough for *DV-position*, given a satisfactory range measurement performance (Medusa nodes [14] achieve centimeter accuracy).

This study confirms the intuition that multimodal sensing has the potential of higher performance than single measurement approaches (angle only, or range only), but can compete with range / angle free positioning only when high quality measurements are available. Single measurement solutions (range only, or angle only) require even higher accuracy measurements, but their hardware requirements are available today in prototypes used by the research community. Even if measurement error inherently builds up in a multihop estimation environment, careful provisioning of the network density and landmark ratios can provide low error positions for most combinations of capabilities.

5. RELATED WORK

Positioning in ad hoc networks has recently received a lot of attention mostly with the advent of sensor networks. Doherty [20] used convex optimization to determine positions based exclusively on connectivity. Also using centralized methods, but accepting both mere connectivity and range measurements, MDS-MAP [21] finds positions using multidimensional scaling. The centralized solutions may not be desirable in certain situations, but they can be used as performance benchmarks since they make use of the knowledge of the entire topology.

One hop solutions are those in which nodes can directly contact a landmark, for example GPS. Bulusu [6] proposed the use of a grid of strong landmarks, so that any node can contact some landmarks. This solution is distributed, but landmarks here play more the role of an infrastructure, having to completely cover the entire network.

In the class of multihop algorithms, AhLOS [14] is a method that divides the network into groups of nodes containing enough landmarks to solve a nonlinear system positioning all the nodes in the group. The method is distributed and localized, and is closest in spirit to APS.

Analytic characterization of performance of a positioning algorithm is achieved for a simplified model of a square neighborhood in [22]. An idea similar with APS/*DV-hop* has been independently explored in the context of amorphous computing by Nagpal [13], who has also given an upper bound on the accuracy as $\frac{\pi r}{4N_{avg}}$, where r is the wireless radius, and N_{avg} the average number of neighbors of a node. For the AhLOS method, [23] gives the Cramér-Rao lower bound of the covariance of achieved positions. Moses [24] investigates the use of a calibration phase before the actual positioning, and gives the CRLB for calibration uncertainty. While most approaches address the average case for uniform deployment of nodes, Bischoff et al. [25] show that hop based algorithms are not competitive in the worst case deployment. We shortly analyzed another case of nonuniform deployment in a previous paper [10]. For a particular case of nonuniform density, *DV-hop* is shown to be less predictable in performance than *Euclidean*.

Although Taylor series is the most used method for position estimation, including in GPS [16], closed form estimators that perform close to CRLB are available, one of which is presented in [26].

Although this paper only considers distributed/multihop algorithms, a more comprehensive study would include the centralized methods as well, which is part of our future work. An especially interesting direction is a comparison of the position error lower bounds achievable for a given network

setup in a centralized versus a distributed approach. Another aspect that is neglected so far is that of communication cost of obtaining positions. Since communication has the dominating cost in the functioning of most ad hoc network, it is interesting to evaluate the relationship between positioning error and energy spent by these algorithms.

6. CONCLUSION

We analyze the error characteristics of range / angle free, range based, angle based, and multimodal positioning algorithms that use distance vector as their main propagation method. Using a simplified network model, we assume a uniform deployment of the nodes, without obstacles, and gaussian error measurements, in order to provide error bounds for APS algorithms. Range free and multimodal methods are analyzed and shown to produce errors that are inversely proportional with the fraction of landmarks and with the TTL associated with the methods. The dependence of the covariance error is characterized in terms of deployment density, even if no closed form is available. Partitioning of the parameter space between the four analyzed algorithms provides insights with respect to the hardware performance needed to achieve good positioning in a multihop environment.

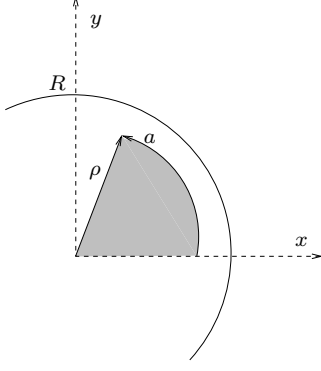
7. ACKNOWLEDGMENTS

This research work was supported in part by DARPA under contract number N-666001-00-1-8953 and NSF grant ANI-0240383.

8. REFERENCES

- [1] G. Finn. Routing and addressing problems in large metropolitan-scale internetworks. Technical Report ISI Research Report ISI/RR-87-180, University of Southern California, March 1987.
- [2] J. C. Navas and Tomasz Imielinski. Geographic addressing and routing. In *ACM MOBICOM*, Budapest, Hungary, September 26-30 1997.
- [3] Fabian Kuhn, Roger Wattenhofer, Yan Zhang, and Aaron Zollinger. Geometric ad-hoc routing: Of theory and practice. In *22nd ACM Symposium on the Principles of Distributed Computing (PODC)*, Boston, MA, USA, July 2003.
- [4] Jason Hill, Robert Szcwcyk, Alec Woo, Seth Hollar, David Culler, and Kristofer Pister. System architecture directions for networked sensors. In *ASPLOS-IX*, Cambridge, MA, November 2000.
- [5] Y.-B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. In *ACM MOBICOM*, October 1998.
- [6] Nirupama Bulusu, John Heidemann, and Deborah Estrin. GPS-less low cost outdoor localization for very small devices. In *IEEE Personal Communications Magazine*, Special Issue on Smart Spaces and Environments. October 2000.
- [7] Paramvir Bahl and Venkata N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *INFOCOM*, Tel Aviv, Israel, March 2000.
- [8] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *ACM MOBICOM*, Boston, MA, August 2000.
- [9] N.B. Priyantha, A. Miu, H. Balakrishnan, and S. Teller. The cricket compass for context-aware mobile applications. In *ACM MOBICOM*, Rome, Italy, July 2001.
- [10] Dragoş Niculescu and Badri Nath. DV based positioning in ad hoc networks. *Telecommunication Systems, Kluwer*, 22(1-4):267–280, January-April 2003.
- [11] Dragoş Niculescu and Badri Nath. Ad hoc positioning system (APS) using AoA. In *INFOCOM*, San Francisco, CA, April 2003.
- [12] Dragoş Niculescu and Badri Nath. Position and orientation in ad hoc networks. *Elsevier Ad Hoc Networks*, 2003. to appear.
- [13] Radhika Nagpal. Organizing a global coordinate system from local information on an amorphous computer. Technical Report 1666, MIT AI Lab, 1999.
- [14] A. Savvides, C.-C. Han, and M. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *ACM MOBICOM*, Rome, Italy, 2001.
- [15] Koen Langendoen and Niels Reijers. Distributed localization in wireless sensor networks a quantitative comparison. Technical Report PDS-2002-3, Delft University of Technology, The Netherlands, 2002.
- [16] B.W. Parkinson and J.J. Spilker. *Global Positioning System: Theory and Application*. American Institute of Astronautics and Aeronautics, 1996.
- [17] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. In *3rd International Workshop on Discrete Algorithms and methods for mobile computing and communications*, Seattle, WA, August 1999.
- [18] Leonard Kleinrock and John Silvester. Optimum transmission radii for packet radio networks or why six is a magic number. In *IEEE National Telecommunications Conference*, pages 4.3.1–4.3.5, Birmingham, Alabama, 1978.
- [19] T.K. Philips, S.S. Panwar, and A.N. Tantawi. Connectivity properties of a packet radio network model. In *IEEE Transactions on Information Theory*, volume 35, pages 1044–1047, September 1989.
- [20] L. Doherty, L. E. Ghaoui, and K. S. J. Pister. Convex position estimation in wireless sensor networks. In *IEEE INFOCOM*, Anchorage, AK, April 2001.
- [21] Yi Shang, Wheeler Ruml, Ying Zhang, and Markus Fromherz. Localization from mere connectivity. In *ACM MOBIHOC*, Annapolis, MD, June 1-3 2003.
- [22] Slobodan Simić and Shankar Sastry. A distributed algorithm for localization in random wireless networks. Technical report, UC Berkeley, EECS, 2002.
- [23] A. Savvides, W. Garber, S. Adlakha, R. Moses, and M. B. Srivastava. On the error characteristics of multihop node localization in ad-hoc sensor networks. In *IPSN03, International Workshop on Information Processing in Sensor Networks*, PARC, Palo Alto, CA, April 22-23 2003.
- [24] R. Moses, D. Krishnamurthy, and R. Patterson. A self-localization method for wireless sensor networks. *Eurasip Journal on Applied Signal Processing, Special Issue on Sensor Networks*, 2002.
- [25] Regina Bischoff and Roger Wattenhofer. Analyzing connectivity-based multi-hop ad-hoc positioning. In

Figure 11: Joint distribution of ρ and α .



IEEE Percom, Orlando, Florida, USA, March 2004.
 [26] Y. T. Chan and K. C. Ho. A simple and efficient estimator for hyperbolic location. *IEEE Transactions on Signal Processing*, 42(8):1905–1915, August 1994.

APPENDIX

A. EXPECTATIONS

Given a circle of radius R , with points that are Poisson distributed, we wish to find the expectations of $\frac{\cos^2(\alpha)}{\rho}$, $\frac{\sin^2(\alpha)}{\rho}$, and $\frac{\sin(2\alpha)}{2\rho}$, where ρ and α are the polar coordinates. Referring to Figure 11, the joint distribution function for the polar coordinates is:

$$\begin{aligned} F_{\rho\alpha} &= P\{a < \alpha, r < \rho\} = \frac{\alpha\rho^2}{2} \frac{1}{\pi R^2} \\ f_{\rho\alpha} &= \frac{\partial^2 F_{\rho\alpha}}{\partial\rho\partial\alpha} = \frac{\rho}{\pi R^2} \\ f_{\rho} &= \frac{2\rho}{R^2} \\ f_{\alpha} &= \frac{1}{2\pi} \end{aligned}$$

meaning that the polar coordinates are independent with a Poisson deployment. Let $E = \frac{1}{\rho}$, with $E \in [\frac{1}{R}, \infty)$.

$$\begin{aligned} F_{\frac{1}{\rho}}(e) &= P\{E \leq e\} = P\{\frac{1}{\rho} \leq e\} = \\ &= P\{\rho \geq \frac{1}{e}\} = 1 - P\{\rho \leq \frac{1}{e}\} \\ &= 1 - \frac{1}{R^2 e^2}. \\ f_{\frac{1}{\rho}}(e) &= \frac{2}{R^2 e^3} \\ E[\frac{1}{\rho}] &= \int_{\frac{1}{R}}^{\infty} \frac{2}{R^2 e^2} de = \frac{2}{R} \end{aligned} \quad (14)$$

Let $C = \cos(\alpha)$, with $C \in [-1, 1]$.

$$\begin{aligned} F_C &= P\{\cos(\alpha) < c\} = 1 - \frac{\arccos(c)}{\pi} \\ f_C &= \frac{1}{\pi\sqrt{1-c^2}} \\ f_{C^2}(x) &= \frac{1}{\pi\sqrt{x(1-x)}} \\ E[\cos^2(\alpha)] &= \frac{1}{2} \end{aligned} \quad (15)$$

In a similar fashion, distributions for $\sin(\alpha)$ and $\sin^2(\alpha)$ are derived and shown to be identical with $\cos(\alpha)$ and $\cos^2(\alpha)$. Let $S = \sin(2\alpha)$, with $S \in [-1, 1]$.

$$\begin{aligned} P_s &= \{ \sin(2\alpha) < s \} = \frac{1}{2} + \frac{\arcsin(|s|)}{\pi} \\ f_s &= \frac{1}{\pi\sqrt{1-s^2}} \\ E[\sin(2\alpha)] &= 0 \end{aligned} \quad (17)$$

Using (14), (15), and (17) we conclude that

$$\begin{aligned} E[\frac{\cos^2(\alpha)}{\rho}] &= E[\frac{\sin^2(\alpha)}{\rho}] = \frac{1}{R} \\ E[\frac{\sin(2\alpha)}{\rho}] &= 0 \end{aligned} \quad (18)$$

B. TRILATERATION CRLB

Since the Fisher information matrix uses derivatives of the log-likelihood, we will need the following partial differentiations:

$$\begin{aligned} \frac{\partial\rho_i}{\partial x} &= \frac{x - x_i}{\rho_i} \\ \frac{\partial^2\rho_i}{\partial x\partial y} &= -\frac{(x - x_i)(y - y_i)}{\rho_i^3} \end{aligned}$$

\mathbf{x} is the position to be estimated, seen as a parameter here, whereas $\hat{\rho}$ are the normally distributed ranges to a landmark. Their expectations are the true ranges ρ , and their covariance is given by the diagonal matrix W .

$$\begin{aligned} p(\hat{\rho}; \mathbf{x}) &= \sqrt{\frac{\det(W)}{(2\pi)^n}} \exp\left\{-\frac{1}{2}[\hat{\rho} - \rho]^\top W[\hat{\rho} - \rho]\right\} \\ L(\mathbf{x}; \hat{\rho}) &= -\frac{1}{2}\left\{\ln\left(\frac{(2\pi)^n}{\det(W)}\right) + [\hat{\rho} - \rho]^\top W[\hat{\rho} - \rho]\right\} \\ &= -\frac{1}{2}\left[\ln\left(\frac{(2\pi)^n}{\det(W)}\right) + \sum \frac{(\hat{\rho}_i - \rho_i)^2}{\sigma_i^2}\right] \\ \frac{\partial L}{\partial x} &= \sum \frac{(\hat{\rho}_i - \rho_i)}{\sigma_i^2} \frac{\partial\rho_i}{\partial x} \\ \frac{\partial^2 L}{\partial x^2} &= \sum \frac{1}{\sigma_i^2} \left[-\left(\frac{\partial\rho_i}{\partial x}\right)^2 + (\hat{\rho}_i - \rho_i) \frac{\partial^2\rho_i}{\partial x^2}\right] \\ \frac{\partial^2 L}{\partial x\partial y} &= \sum \frac{1}{\sigma_i^2} \left[-\frac{\partial\rho_i}{\partial x} \frac{\partial\rho_i}{\partial y} + (\hat{\rho}_i - \rho_i) \frac{\partial^2\rho_i}{\partial x\partial y}\right] \end{aligned}$$

Fisher information matrix $I(\mathbf{x})$ is defined as

$$\begin{aligned}
I(\mathbf{x}) &= - \int_{-\infty}^{\infty} \begin{bmatrix} \frac{\partial^2 L}{\partial x^2} & \frac{\partial^2 L}{\partial x \partial y} \\ \frac{\partial^2 L}{\partial x \partial y} & \frac{\partial^2 L}{\partial y^2} \end{bmatrix} p(\hat{\rho}; \mathbf{x}) d\hat{\rho} \\
&\quad - \int_{-\infty}^{\infty} \frac{\partial^2 L}{\partial x^2} p(\hat{\rho}; \mathbf{x}) d\hat{\rho} = \\
&\quad \int_{-\infty}^{\infty} p(\hat{\rho}; \mathbf{x}) \sum \frac{1}{\sigma_i^2} \frac{(x-x_i)^2}{\rho_i^2} d\hat{\rho} + \\
&\quad \int_{-\infty}^{\infty} p(\hat{\rho}; \mathbf{x}) \sum \frac{(\hat{\rho}_i - \rho_i)(x-x_i)(y-y_i)}{\sigma_i^2 \rho_i^3} d\hat{\rho} = \\
&\quad \sum \frac{1}{\sigma_i^2} \frac{(x-x_i)^2}{\rho_i^2}
\end{aligned}$$

$$\text{because } \int_{-\infty}^{\infty} p(\hat{\rho}; \mathbf{x}) d\hat{\rho} = 1; \int_{-\infty}^{\infty} p(\hat{\rho}; \mathbf{x}) \hat{\rho} d\hat{\rho} = \rho$$

In the same manner it is shown that

$$- \int_{-\infty}^{\infty} \frac{\partial^2 L}{\partial x \partial y} p(\hat{\rho}; \mathbf{x}) d\hat{\rho} = \sum \frac{1}{\sigma_i^2} \frac{(x-x_i)(y-y_i)}{\rho_i^2}$$

which produces

$$\begin{aligned}
I(\mathbf{x}) &= \begin{bmatrix} \sum \frac{1}{\sigma_i^2} \frac{(x-x_i)^2}{\rho_i^2} & \sum \frac{1}{\sigma_i^2} \frac{(x-x_i)(y-y_i)}{\rho_i^2} \\ \sum \frac{1}{\sigma_i^2} \frac{(x-x_i)(y-y_i)}{\rho_i^2} & \sum \frac{1}{\sigma_i^2} \frac{(y-y_i)^2}{\rho_i^2} \end{bmatrix} \\
&= \begin{bmatrix} \frac{x-x_1}{\rho_1} & \frac{y-y_1}{\rho_1} \\ \frac{x-x_2}{\rho_2} & \frac{y-y_2}{\rho_2} \\ \dots & \dots \end{bmatrix}^\top W \begin{bmatrix} \frac{x-x_1}{\rho_1} & \frac{y-y_1}{\rho_1} \\ \frac{x-x_2}{\rho_2} & \frac{y-y_2}{\rho_2} \\ \dots & \dots \end{bmatrix} \\
&= J_0^\top W J_0
\end{aligned}$$

The matrix J_0 is in fact the Jacobian of the true ranges with respect to the true position.

$$CRLB(Cov[\mathbf{x}]) = (J_0^\top W J_0)^{-1}$$

C. DV-HOP CRLB

W.l.o.g., assume the true position to be in the origin $r_u = 0$. The variances in ranges are known to be inversely proportional with the distance in hops.

$$\begin{aligned}
J_0 &= \begin{bmatrix} \frac{x_1}{\rho_1} & \frac{y_1}{\rho_1} \\ \frac{x_2}{\rho_2} & \frac{y_2}{\rho_2} \\ \dots & \dots \end{bmatrix} \\
W &= \begin{bmatrix} \frac{E[\bar{x}]}{\rho_i V[\bar{x}]} & 0 & 0 \\ 0 & \frac{E[\bar{x}]}{\rho_2 V[\bar{x}]} & 0 \\ 0 & 0 & \dots \end{bmatrix}
\end{aligned}$$

$$\begin{aligned}
J_0^\top W J_0 &= \frac{E[\bar{x}]}{V[\bar{x}]} \begin{bmatrix} \sum \frac{x_i^2}{\rho_i^3} & \sum \frac{x_i y_i}{\rho_i^3} \\ \sum \frac{x_i y_i}{\rho_i^3} & \sum \frac{y_i^2}{\rho_i^3} \end{bmatrix} \\
&= \frac{E[\bar{x}]}{V[\bar{x}]} \begin{bmatrix} \sum \frac{\cos^2(\alpha_i)}{\rho_i} & \sum \frac{\sin(2\alpha_i)}{2\rho_i} \\ \sum \frac{\sin(2\alpha_i)}{2\rho_i} & \sum \frac{\sin^2(\alpha_i)}{\rho_i} \end{bmatrix} \\
&= \frac{E[\bar{x}]}{V[\bar{x}]} n \begin{bmatrix} E[\frac{\cos^2(\alpha)}{\rho}] & E[\frac{\sin(2\alpha)}{2\rho}] \\ E[\frac{\sin(2\alpha)}{2\rho}] & E[\frac{\sin^2(\alpha)}{\rho}] \end{bmatrix}
\end{aligned} \tag{19}$$

where the sums in (19) are over all the landmarks inside the TTL circle of radius $hE[\bar{x}]$, and α_i are polar angle corresponding to (x_i, y_i) . Using the results in appendix A it is shown (18) that $E[\frac{\cos^2(\alpha_i)}{\rho_i}] = E[\frac{\sin^2(\alpha_i)}{\rho_i}] = \frac{1}{hE[\bar{x}]}$, and $E[\frac{\sin(2\alpha_i)}{2\rho_i}] = 0$.

$$\begin{aligned}
CRLB(Cov[r_u]) &= \frac{V[\bar{x}]}{E[\bar{x}]n} \begin{bmatrix} \frac{1}{hE[\bar{x}]} & 0 \\ 0 & \frac{1}{hE[\bar{x}]} \end{bmatrix}^{-1} \\
&= \frac{hV[\bar{x}]}{n} I_2 \\
&= \frac{1}{f\pi h} \frac{V[\bar{x}]}{\lambda E^2[\bar{x}]} I_2
\end{aligned}$$

D. UNCERTAINTY FOR DV-POSITION

Assuming there are n_i hops to landmark i , and a circular error covariance:

$$\begin{aligned}
U_i &= \sum_{j=1}^{n_i} \begin{bmatrix} r_j^2 \sigma^2 & 0 \\ 0 & r_j^2 \sin^2(\sigma_a) \end{bmatrix} \\
&\simeq \sigma^2 \sum_{j=1}^{n_i} r_j^2 I_2 \\
&= \sigma^2 E[\bar{r}^2] n_i I_2 \\
&= \sigma^2 E[\bar{r}^2] \frac{\rho_i}{E[\bar{x}]} I_2
\end{aligned} \tag{20}$$

Using derivations in appendix A,

$$\begin{aligned}
\sum_{i=1}^n U_i^{-1} &= \frac{E[\bar{x}]}{\sigma^2 E[\bar{r}^2]} \sum \frac{1}{\rho_i} I_2 \\
&= \frac{E[\bar{x}]}{\sigma^2 E[\bar{r}^2]} \frac{2n}{hE[\bar{x}]} I_2 \\
(\sum_{i=1}^n U_i^{-1})^{-1} &= \frac{h\sigma^2 E[\bar{r}^2]}{2n} I_2 \\
U &= \frac{1}{2f\pi h} \sigma^2 \frac{E[\bar{r}^2]}{\lambda E^2[\bar{x}]} I_2
\end{aligned}$$