Sensor Location through Linear Programming with Arrival Angle Constraints

Camillo Gentile and John Shiu
National Institute of Standards and Technology
Wireless Communication Technologies Group
Email: camillo.gentile@nist.gov

***Abstract*— In previous work, we established a linear programming framework to determine sensor location from measured link distances between neighboring nodes in a network. Besides providing greater accuracy compared to other techniques, linear programs in particular suit large networks since they can be solved efficiently through distributed computing over the nodes without compromising the optimality of the objective function. This work extends our framework to determine sensor location from measured arrival angles instead. An extensive simulation suite substantiates the performance of the algorithm according to several network parameters, including noise up to 15% the maximum error; the proposed algorithm reduces the error up to 84% depending on the noise level.**

***Index Terms*— Convex optimization, Network localization**

I. Introduction

The falling price and reduced size of sensors in recent years have fueled the deployability of dense networks to monitor and relay environmental properties such as temperature, moisture, and light [1]. The ability to self-organize and find their locations autonomously and with high accuracy proves particularly useful in military and public safety operations. In dense networks, multilateration can render good location accuracy despite significant errors in range estimates between sensors. This has launched a research area known as sensor location which seeks to process potentially enormous quantities of data collectively to achieve optimal results. Most practical systems require local distributed processing to cope with dynamic links or nodes in motion to maintain a network updated; alternatively relaying information across a large network sanctions the centralized processing of obsolete data, limiting scalability.

The pioneering work in sensor location only used the link distances between nodes measured through received signal strength or arrival time [2], [3], [4], [5], [6], [7], however received signal strength suffers from inaccuracies due to multipath fading and arrival time often requires expensive equipment or large overhead for synchronization between nodes. Arrival angle offers an alternative measurement system suited in particular to sensor nodes equipped with smart antennas [8]. In time, many algorithms have incorporated arrival angle to complement link distances [9], [10], [11], [12], [13], [14], but to our knowledge only two consider angle measurements alone: the first concentrates on sensors with limited computational power, emphasizing rapid organization rather than location performance [7]; the second formulates an optimization program with requisite geometrical constraints on the sensor locations [15]. In order to employ efficient convex optimization techniques, the authors *relax* the original quadratic constraints to semi-definite constraints. Furthermore the approach requires a heuristic parameter whose value depends on the size and geometry of the network, which if not chosen carefully leads to an infeasible solution [16].

Along the lines of our previous work [17], this paper also formulates an optimization program, however with convex geometrical constraints on the arrival angles, necessitating no relaxation of the original constraints and so furnishes a tighter solution. Besides the exact solution, linear programming in particular suits large networks since it can be solved through distributed computing over the nodes without compromising the optimality of the objective function [18].

The paper reads as follows: Section II formally states the sensor location problem and the subsequent section proposes an optimization program to solve it. In Section IV we show how to *reconstruct* the sensor locations from the solution to the program, followed by a section detailing an approach to refine this solution. Section VI presents the results from an extensive simulation suite to substantiate the robustness of our algorithm to varying levels of noise and other network parameters, and the last section provides conclusions and directions for further research.

II. Problem statement

Let a network contain two types of nodes: n_A anchor nodes (or anchors) with known location and n_S sensor nodes (or sensors) with unknown location, for a total of $n = n_A + n_S$ nodes. For simplicity, let the nodes lie on a plane such that node i has location $\mathbf{x}_i \in \mathcal{R}^2$ indexed through $i,\ i = 1 \dots n_A$ for the anchors and $i = n_A + 1 \dots n$ for the sensors. The set N contains all pairs of nodes between which the link distance d_{ij} is less than a network parameter R known as the *radio range*, i.e. $(i,j),\ i < j;\ ||\mathbf{x}_i - \mathbf{x}_j||_2 < R$. The set M contains all triplets of nodes which form a triangle in the network: $(i,j,k),\ (i,j) \in N;\ (j,k) \in N;\ (i,k) \in N$.

Equipped with an omnidirectional antenna array or a directional rotational antenna [19], according Fig. 1 node i measures the arrival angle θ_j^i from a neighboring node j in the clockwise (or counter-clockwise) sense relative to its *bearing* θ^i. Given these *measured angles* and the locations

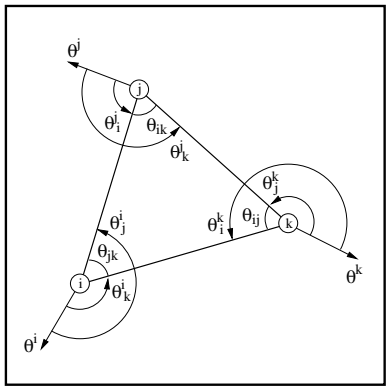


Fig. 1. The arrival angles, the interior angles, and the relative bearing of a node.

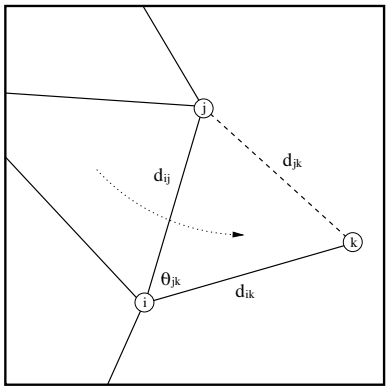


Fig. 2. Node k has a single connection to the network.

of the anchor nodes in the network, this paper proposes a solution to the problem first posed in [20] to find the locations of the sensor nodes and their bearings in a reference coordinate system.

III. Arrival angle constraints

The most effective approach to sensor location employs convex optimization by placing geometrical constraints on the link distances between neighboring nodes of the network [2], [15], [17]. In the place of link distances, here we constrain the arrival angles between neighbors instead. Denote the *interior angle* $\theta_{ij} = \theta_{ji} = |\theta_i^k - \theta_j^k|$ as the absolute difference between the arrival angles from neighboring nodes i and j relative to θ^k (see Fig. 1). Exploiting the triangular structure of the network, the sum of the three interior angles of each triangle must equal π. The optimization program we solve can be stated as follows:

$$\begin{aligned} \min \sum_{(i,j)\in N} |\delta_j^i| + |\delta_i^j| \\ \text{s.t.} \quad \left.\begin{aligned} \overbrace{\theta_j^i - \theta_k^i}^{\theta_{jk}} + \overbrace{\theta_k^j - \theta_i^j}^{\theta_{ik}} + \overbrace{\theta_i^k - \theta_j^k}^{\theta_{ij}} &= \pi \\ \theta_j^i - \theta_k^i &\geq 0 \\ \theta_k^j - \theta_i^j &\geq 0 \\ \theta_i^k - \theta_j^k &\geq 0 \end{aligned}\right\}, \ \forall (i,j,k) \in M \end{aligned} \tag{1}$$

where $\theta_j^i = \hat{\theta}_j^i + \delta_j^i$. The objective function minimizes the sum of the absolute *residuals* α_{ij} between the measured angles $\hat{\theta}_j^i$ and the *estimated angles* θ_j^i while the constraints ensure that the latter conform to the requisite geometry of the physical world.

The optimization program above can be converted to a linear program in canonical form by removing the absolute signs and introducing four bounding constraints:

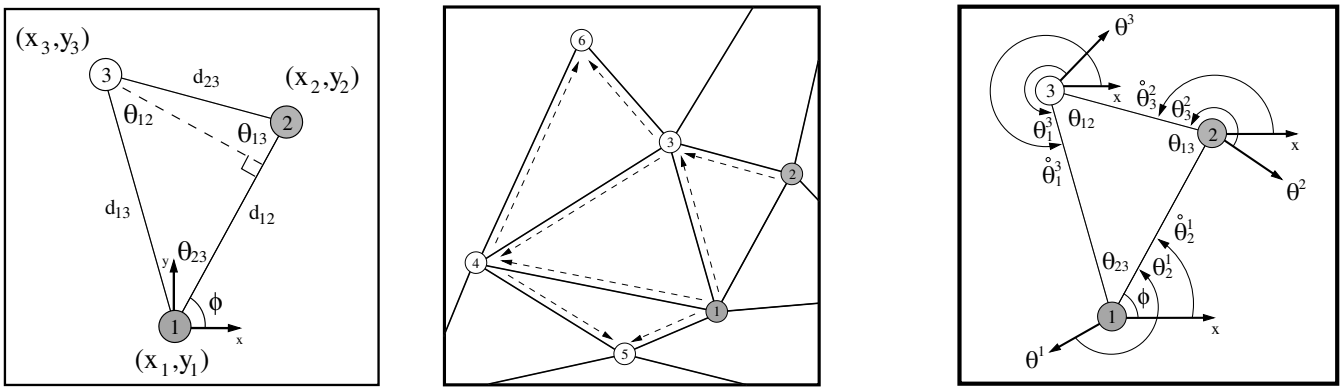
$$\begin{aligned} \min \sum_{(i,j)\in N} \delta_j^{i+} + \delta_j^{i-} + \delta_i^{j+} + \delta_i^{j-} \\ \text{s.t.} \quad \left.\begin{aligned} \theta_j^i - \theta_k^i + \theta_k^j - \theta_i^j + \theta_i^k - \theta_j^k &= \pi \\ \theta_j^i - \theta_k^i &\geq 0 \\ \theta_k^j - \theta_i^j &\geq 0 \\ \theta_i^k - \theta_j^k &\geq 0 \end{aligned}\right\}, \ \forall (i,j,k) \in M \\ \left.\begin{aligned} \delta_j^{i+} &\geq 0 \\ \delta_j^{i-} &\geq 0 \\ \delta_i^{j+} &\geq 0 \\ \delta_i^{j-} &\geq 0 \end{aligned}\right\}, \ \forall (i,j) \in N \end{aligned} \tag{2}$$

where $\theta_j^i = \hat{\theta}_j^i + \delta_j^{i+} - \delta_j^{i-}$. Let $\tilde{\delta}_j^i$ be in the optimal solution to (1) and let $\{\tilde{\delta}_j^{i+}, \tilde{\delta}_j^{i-}\}$ be in any feasible solution to (2) such that $\{\tilde{\delta}_j^{i+}, \tilde{\delta}_j^{i-}\} \geq 0$ and $\tilde{\delta}_j^i = \tilde{\delta}_j^{i+} - \tilde{\delta}_j^{i-}$. Further let $\hat{\delta}_j^{i+} = \tilde{\delta}_j^{i+} - \min\{\tilde{\delta}_j^{i+}, \tilde{\delta}_j^{i-}\}$ and $\hat{\delta}_j^{i-} = \tilde{\delta}_j^{i-} - \min\{\tilde{\delta}_j^{i+}, \tilde{\delta}_j^{i-}\}$; now $\{\hat{\delta}_j^{i+}, \hat{\delta}_j^{i-}\}$ are not only in a feasible solution to (2) since $\{\hat{\delta}_j^{i+}, \hat{\delta}_j^{i-}\} \geq 0$ and $\tilde{\delta}_j^i = \hat{\delta}_j^{i+} - \hat{\delta}_j^{i-}$, but are also optimal since $\hat{\delta}_j^{i+} + \hat{\delta}_j^{i-} \leq \tilde{\delta}_j^{i+} + \tilde{\delta}_j^{i-}$ in its objective function. Moreover since $\hat{\delta}_j^{i+} \cdot \hat{\delta}_j^{i-} = 0$, then $\hat{\delta}_j^{i+} + \hat{\delta}_j^{i-} = |\hat{\delta}_j^{i+} - \hat{\delta}_j^{i-}| = |\tilde{\delta}_j^i|$, proving the equivalence of (1) and (2) at optimality.

The advantage of our approach lies in the linear constraints which ensure the convexity of the problem without relaxing the original geometrical constraints. The linear program above does not directly yield the sensor locations $\mathbf{x}_i,\ i = n_A+1...n$, but only the estimated arrival angles $\theta_j^i,\ (i,j) \in N$. In fact the complete algorithm requires an a posteriori *location reconstruction* stage described in the following section to furnish the locations of the sensors from these angles. Note that (2) can be applied to the triangles formed in three-dimensional networks as well; this paper does not treat the reconstruction stage for such networks for the sake of brevity.

A. Additional arrival angle constraints

Case I: Consider node i in Fig. 2 with two neighboring nodes j and k which themselves are not neighbors, i.e.



(a) The geometry. (b) Location propagation throughout the network. (c) Reconstructed reference arrival angles and node bearings.

Fig. 3. Location reconstruction.

$$\left.\begin{array}{l}(i,j)\in N\\(i,k)\in N\\(j,k)\notin N\end{array}\right\}\Rightarrow\left.\begin{array}{c}d_{ij}<R\\d_{ik}<R\\d_{jk}\geq R\end{array}\right\}\Rightarrow\left.\begin{array}{l}d_{jk}>d_{ij}\\d_{jk}>d_{ik}\end{array}\right\}.$$

While node i can estimate $\theta_{jk} = |\theta^i_j - \theta^i_k|$, nodes j and k cannot estimate θ_{ik} and θ_{jk} respectively due to lack of communication between each other, forfeiting the constraint on the sum of the interior angles of $\triangle_{ijk}$. Alternatively we seek a constraint on θ_{jk} through the Law of Sines which states that $\frac{d_{jk}}{d_{ij}} = \frac{\sin\theta_{jk}}{\sin\theta_{ij}}$, so $d_{jk} > d_{ij} \Rightarrow \sin\theta_{jk} > \sin\theta_{ij}$. Since $\sin\theta$ increases monotonically in the interval $0 \leq \theta \leq \pi$ for which the interior angle θ is valid, then $\theta_{jk} > \theta_{ij}$. It follows that $\theta_{jk} > \theta_{ik}$ through the same reasoning. Now

$$\left.\begin{array}{l}\theta_{jk}>\theta_{ij}\\\theta_{jk}>\theta_{ik}\end{array}\right\}\Rightarrow 2\cdot\theta_{jk}>\theta_{ij}+\theta_{ik},$$

and substituing the inequality above into $2\cdot\theta_{jk}+\theta_{jk}$ we have $3\cdot\theta_{jk} > \theta_{ij}+\theta_{ik}+\theta_{jk}$, but $\theta_{ij}+\theta_{ik}+\theta_{jk} = \pi$, so $\theta_{jk} > \frac{\pi}{3}$.

Case II: The triangle is the smallest polygon with non-trivial interior angles and so provides the tightest constraints on each by reducing the terms in the sum of its interior angles. However if there exists a simple polygon of order $m > 3$ which cannot be decomposed into smaller triangles of the network, include an additional constraint in the linear program such that its interior angles sum to $(m-2)\cdot\pi$.

IV. Location reconstruction

The reconstruction stage yields the locations of the sensors in the network from the arrival angles estimated through the linear program (2). The stage originates at any two anchor nodes (shaded) sharing a neighboring sensor node (unshaded), as in Fig. 3(a). Given $\mathbf{x}_1 = (x_1, y_1)$ and $\mathbf{x}_2 = (x_2, y_2)$ and θ_{12}, θ_{13}, and θ_{23}, the following set of equations furnishes the unknown location $\mathbf{x}_3 = (x_3, y_3)$ with respect to the reference coordinate system centered at $\mathbf{x}_1$ [6]:

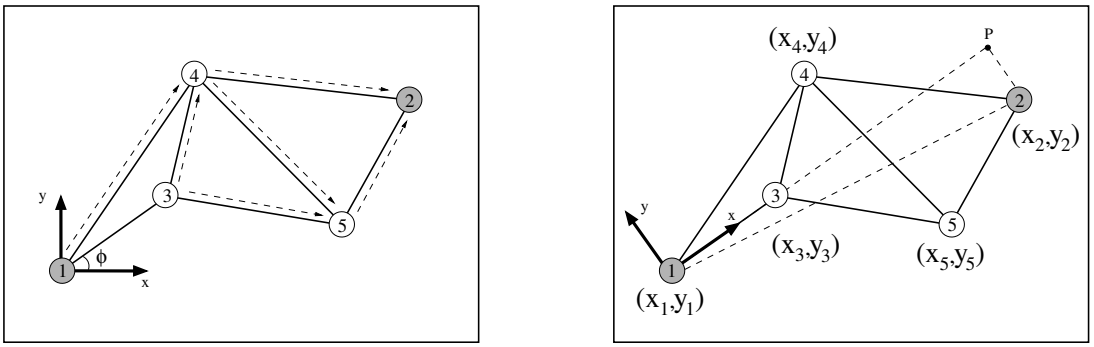
$$\begin{array}{ll}(a) & d_{12} = \sqrt{(x_1-x_2)^2+(y_1-y_2)^2}\\(b) & d_{13} = \sin\theta_{13}\cdot\left(\dfrac{d_{12}}{\sin\theta_{12}}\right)\\(c) & \left[\begin{array}{c}x'_3\\y'_3\end{array}\right] = \left[\begin{array}{c}x_1\\y_1\end{array}\right] + \left[\begin{array}{c}d_{13}\cdot\cos\theta_{23}\\d_{13}\cdot\sin\theta_{23}\end{array}\right]\\(d) & \phi = \arctan\left(\dfrac{y_2-y_1}{x_2-x_1}\right)\\(e) & \left[\begin{array}{c}x_3\\y_3\end{array}\right] = \left[\begin{array}{c}x_1\\y_1\end{array}\right] + \left[\begin{array}{cc}\cos\phi & -\sin\phi\\\sin\phi & \cos\phi\end{array}\right]\left[\begin{array}{c}x'_3-x_1\\y'_3-y_1\end{array}\right]\end{array} \quad (3)$$

We say that the two anchor nodes *propagate* their locations to the sensor node. Once the location of the sensor node is known, it serves with another *known* sensor (or anchor) to determine the location of an *unknown* sensor neighoring the two. Following the propagation path from the originating anchor pair in Fig. 3(b), n_1 and n_2 propagate their locations to n_3, n_1 and n_3 to n_4, n_1 and n_4 to n_5, and n_3 and n_4 to n_6.

Denote the *reference arrival angle* $\mathring{\theta}^i_j$ as the arrival angle at node i from a neighboring node j in the clockwise sense relative to the reference coordinate system, as in Fig. 3(c). Note that $\mathring{\theta}^i_j = \mathring{\theta}^j_i + \pi$. The reference arrival angles can be easily found during the reconstruction stage by tracing their values through the interior angles from the origin of the reference coordinate system. In the figure, $\mathring{\theta}^1_2 = \phi$, $\mathring{\theta}^1_3 = \phi + \theta_{23} + \pi$, and $\mathring{\theta}^2_3 = \phi - \theta_{13} + \pi$. Now to compute the bearing of node i, simply subtract the estimated arrival angle from the reference arrival angle as $\theta^i = \mathring{\theta}^i_j - \theta^i_j$.

A. Network topologies

Case I: In general network topologies, especially those with low anchor density, no single sensor node neighbors any two anchors, as in Fig. 4(a). Consider as an alternative the



(a) The path $n_1 \rightarrow n_3 \rightarrow n_4 \rightarrow n_5 \rightarrow n_2$ connects the two anchors. (b) Location propagation from $\mathbf{x}_1$ and $\mathbf{x}_3$ to $\mathbf{x}_4$, $\mathbf{x}_5$, and $\mathbf{x}_2$.

Fig. 4. Location propagation between anchors.

connection between the two anchors through the minimum-hop path $n_1 \rightarrow n_3 \rightarrow n_4 \rightarrow n_5 \rightarrow n_2$: here propagation originates from one known anchor n_1 and one unknown sensor n_3 rather than from an anchor pair. Since $\mathbf{x}_3$ is unknown, the value of ϕ is also unknown; instead orient a relative coordinate system centered at $\mathbf{x}_1$ along the line between n_1 and n_3, as in Fig. 4(b). Now propagate $\mathbf{x}_1$ and $\mathbf{x}_3$ to $\mathbf{x}_4$, $\mathbf{x}_5$, and $\mathbf{x}_2$ as described before, however rather than in terms of real values and with respect to the reference coordinate system, here in terms of the variables (x_3, y_3) and with respect to the relative coordinate system. These locations assume the following form with individual real-valued terms (d^x, d^y) computed pairwise and stepwise through (3): $(x_4, y_4) = (x_1 + d^x_{14}, y_1 + d^y_{14})$, $(x_5, y_5) = (x_4 + d^x_{45}, y_4 + d^y_{45})$, and $(x_2, y_2) = (x_5 + d^x_{25}, y_5 + d^y_{25})$, or compactly

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 + d_{3P} \\ y_1 + d_{2P} \end{bmatrix}, \tag{4}$$

where $d_{3P} = d^x_{14} + d^x_{45} + d^x_{25}$ and $d_{2P} = d^y_{14} + d^y_{45} + d^y_{25}$ represent the distances from n_1 and n_2 respectively to the imaginary point P in the relative coordinate system. Once the propagation reaches n_2, backward substitute (x_2, y_2) in (4) to solve for the unknown values (x_1, y_1), which in turn yield the locations of the other sensor nodes along the minimum-hop path. Now that all nodes along the path have known neighbors with which to propagate their locations to other unknown sensors, propagation can continue as described previously.

Case II: To enable location propagation to it, a node must bear connections from at least two nodes with known locations. For node k in Fig. 2 with a single connection to the network, the reconstruction problem is ill-posed given only the estimated angle θ_{jk} and the reconstructed distance d_{ij} on the shown propagation path to it. Assume $d_{jk} = R$ to provide the third component which completely specifies the triangle $\triangle_{ijk}$ and allows reconstruction of $\mathbf{x}_k$.

B. Multiple location solutions

Scale remains the inherent drawback of using measured angles for sensor location as opposed to measured distances: while knowing the three link distances of a triangle identifies it uniquely, knowing the three angles of a triangle merely generates a family of similar triangles. Hence the constraints in (2) are necessary for geometrical consistency, but not sufficient to ensure a unique solution for the sensor locations since they do not account for scale. In fact we shall see that the reconstructed location depends on the propagation path chosen from the originating anchor pair, where the scale of each triangle is determined by the reconstructed distance common to the preceding triangle on the path.

Consider the network in Fig. 5 as an example. There exists buts one propagation path to sensor n_3 from the anchor pair (n_1, n_2): reconstruct $\mathbf{x}_3$ through the interior angles θ_{12}, θ_{13}, and θ_{23} together with $\mathbf{x}_1$ and $\mathbf{x}_2$; the known distance d_{12} determines the scale of $\triangle_{123}$ and in the process furnishes d_{13} and d_{23}. In sequence, there exist two paths labeled a and b to sensor n_4: in reconstructing $\mathbf{x}_4$ through path a, d_{13} determines the scale of $\triangle_{134}$ and in turn renders location $\mathbf{x}^a_4$; likewise, in reconstructing $\mathbf{x}_4$ through path b, d_{23} determines the scale of $\triangle_{234}$ and in turn renders $\mathbf{x}^b_4$. Hence the angles estimated from (2) may yield multiple solutions. Since each propagation from the originating anchor pair accumulates location error due to the sequential application of more and more erroneous angle measurements, choose the minimum-hop path to the sensor to reconstruct its location; if there exist multiple candidates with minimum hops as in the figure, average the candidate reconstructed locations and any associated reconstructed distances into single values respectively.

V. Link distance constraints

This section proposes a method to synthesize the multiple location solutions inherent to measured angles into a unique solution which ultimately proves more robust to noise. The method employs the reconstructed distances given from the previous section in the linear program taken from [17]:

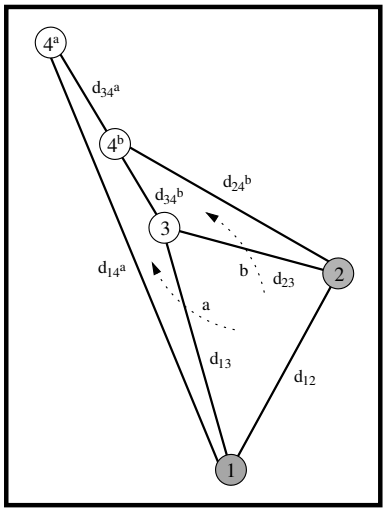


Fig. 5. Node k has a single connection to the network.

$$\begin{aligned} &\min \sum_{(i,j)\in N} |\alpha_{ij}| \\ &\text{s.t.} \quad \left.\begin{array}{l} d_{ij} + d_{jk} \geq d_{ik} \\ d_{ij} + d_{ik} \geq d_{jk} \\ d_{jk} + d_{ik} \geq d_{ij} \end{array}\right\}, \ \forall (i,j,k) \in M \end{aligned} \tag{5}$$

where $d_{ij} = \hat{d}_{ij} + \alpha_{ij}$. The problem minimizes the sum of the absolute *residuals* α_{ij} between the reconstructed distances $\hat{d}_{ij}$ and the *estimated distances* d_{ij}, provided that the latter conform to requisite geometrical constraints. The approach follows the one described in Section III, except that, as explained in Section IV-B, the distance constraints are tighter than the angle contraints and in turn yield a unique location solution. Like (2), the linear program in (5) yields only the estimated distances from which the final sensor locations are reconstructed. The recontruction stage in [17] amounts to converting the three estimated distances of each network triangle to three estimated angles through the Law of Cosines and proceeding as in Section IV, however now the reconstruction is independent of the path chosen.

Hence the combined algorithm to estimate the arrival angles in Section III and reconstruct the sensor locations in Section IV effectively serves as an initialization step to this section. If given additionally the measured distances as input to the algorithm, then making $\hat{d}_{ij}$ a weighted sum of the reconstructed and measured distances offers a method to integrate both measured arrival angles and measured link distances.

VI. Results

In order to quantify the performance of our algorithm, we conduct experiments on a network with the same structure as in [15], [17], [18]. The network contains 50 sensor nodes uniformly distributed throughout a 1×1 unit area. The three parameters of the experiments vary as the number of anchor nodes, the radio range, and the noise level in the measurements. The measured angles are generated from the ground-truth angles $\bar{\theta}_j^i$ by adding Gaussian noise as in [20]. So the algorithm accepts as input $\theta_j^i = \bar{\theta}_j^i + \mathcal{N}(0, \sigma)$, where the standard deviation $\sigma = noise \cdot \pi$ is a fraction of the maximum error π. Figure 6(a) illustrates an example network with $\#anchors = 3$, $R = 0.25$, and $noise = 0.05$. The anchors and sensors appear as blue and cyan asterisks respectively, and the links as blue lines between neighboring nodes. The network contains 211 links for an average node connectivity of 7.9623.

The two performance measures for an experiment are the *average relative angle error* over all the measured angles

$$e_\theta = \frac{1}{2N} \sum_{(i,j)\in N} \frac{|\bar{\theta}_j^i - \theta_j^i|}{\bar{\theta}_j^i} + \frac{|\bar{\theta}_i^j - \theta_i^j|}{\bar{\theta}_i^j} \tag{6}$$

and the *average location error* over all the sensor nodes

$$e_{\mathbf{x}} = \frac{1}{n_S} \sum_{i=n_A+1}^{n} ||\bar{\mathbf{x}}_i - \mathbf{x}_i||_2, \tag{7}$$

where $\bar{\mathbf{x}}_i$ and $\mathbf{x}_i$ denote the ground-truth and estimated locations. We present results from two algorithms for each experiment: algorithm 1 runs (2) on the measured angles and subsequently reconstructs the sensor locations. The angle error is $e_\theta^1 = 0.031$ and the location error is $e_{\mathbf{x}}^1 = 0.093$; algorithm 2 runs (5) on the reconstructed distances from algorithm 1 and subsequently reconstructs the final sensor locations, reducing the angle error to $e_\theta^2 = 0.019$ and the location error to $e_{\mathbf{x}}^2 = 0.066$. The true locations appear in Figure 6(b) as blue astericks and the estimated locations from algorithms 1 and 2 appear as cyan and green astericks respectively, connected to the true location by error lines.

Table I contains the results for 36 experiments as the cross product of $\#anchor = \{3, 5, 7\}$, $R = \{0.20, 0.25, 0.30\}$, and $noise = \{0, 0.05, 0.1, 0.15\}$. The result for each experiment is reported as the average over ten trials of randomly distributed nodes in the network. The average connectivity of the networks for three anchors is 5.4372 for $R = 0.20$, 7.7238 for $R = 0.25$, and 10.2477 for $R = 0.30$. For each slot in the table, there are four results shown according to the legend for algorithms 1 and 2: $\boxed{\begin{array}{cc} e_\theta^1 & e_{\mathbf{x}}^1 \\ e_\theta^2 & e_{\mathbf{x}}^2 \end{array}}$.

Even with perfect angle measurements, the location problem may be ill-posed as explained in Section IV-A: due to lack of connectivity, the algorithms must resort to heuristics to reconstruct some of the locations. This explains the nonzero location error in some slots of the first row of Table I despite the zero angle error; for some experiments, the heuristics actually add more error to the measured angles as seen from the results for algorithm 2. For higher levels of noise, algorithm 1 reduces the angle error up to 52% ($e_\theta^1 = 0.024$, $e_{\mathbf{x}}^1 = 0.087$) for $noise = 0.05$, but only up to 27% ($e_\theta^1 = 0.109$, $e_{\mathbf{x}}^1 = 0.176$) for $noise = 0.15$; algorithm 2 performs much better, delivering values of 84% ($e_\theta^1 = 0.008$, $e_{\mathbf{x}}^1 = 0.047$) and 47% ($e_\theta^1 = 0.079$, $e_{\mathbf{x}}^1 = 0.095$) respectively. Boosting network

connectivity through R increases the number of triangles in the network, but since the interior angles of individual triangles are estimated independently for each triangle, algorithm 1 witnesses no significant reduction in error; increasing connectivity however does improve algorithm 2 since there are more triangles incident on a particular link to use in estimating its value. Raising the number of anchors in the network also enhances the performance of both algorithms.

The computational complexity of the simplex algorithm used to solve linear programs typically varies as $\mathcal{O}(\#constraints)$ [21]. The sparsity of our constraint matrices allows for more efficient algorithms than the simplex, and so the expected complexity should not exceed $\mathcal{O}(\#constraints)$. The upper bound on the number of constraints coincides with a fully-connected network, where the number of triangles is $\binom{n}{3} = \frac{n(n-1)(n-2)}{2}$ and each one introduces one and three constraints in (2) and (5) respectively, resulting in a complexity of $\mathcal{O}(n^3)$. The observed complexity for up to $R = 0.30$ is typically much less: $MATLAB$ solved the larger linear program (5) with 1692 contraints using an interior-point algorithm in less than one second on a 1GHz Pentium IV processor.

VII. Conclusions and further work

In previous work, we established a framework through linear programming to determine sensor location from measured link distances between neighboring nodes in a network. This work extends the framework to determine sensor location from measured arrival angles instead. An extensive simulation suite substantiates the performance of the algorithm according to several network parameters, including noise up to 15% the maximum error; the proposed algorithm reduces the error up to 84% depending on the noise level.

Formerly we showed that the linear program with link distance constraints can be solved efficiently through distributed computing over the nodes without compromising the optimality of the objective function; our current work shows analogous results for the linear programs with arrival angle constraints. In addition, our current work involves integrating both measured link distances and measured arrival angles.

References

[1] P.F. Gorder, "Sizing up smart dust," *IEEE Journal on Computing in Sciences and Engineering*, vol. 5, no. 6, pp. 6-9, Nov. 2003.

[2] L. Doherty, K.S.J. Pister, and L. El Ghaoui, "Convex Position Estimation in Wireless Sensor Networks," *IEEE INFOCOM*, pp. 1655-1663, April 2001.

[3] A. Savvides, H. Park, and M.B. Srivastava, "The Bits and Flops of the N-hop Multilateration Primitive For Node Localization Problems," *ACM Conf. on Wireless Sensor Networks and Applications*, pp. 112-121, Sept. 2002.

[4] Y. Shang, W. Rumi, Y. Zhang, and M.P.J. Fromherz, "Localization from Mere Connectivity," *ACM Conf. on Mobile Ad Hoc Networking and Computing*, pp. 201-212, June 2003.

[5] C. Savarese, J.M. Rabaey, and J. Beutel, "Location in Distributed Ad-Hoc Wireless Networks," *IEEE Conf. on Acoustics, Speech, and Signal Processing*, pp. 2037-2040, May 2001.

[6] S. Capkun, M. Hamdi, and J.-P. Hubaux, "GPS-free positioning in mobile Ad-Hoc networks," *IEEE Hawaii Conf. on System Sciences*, pp. 255-264, Jan. 2001.

[7] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS)," *IEEE Conf. on Global Communications*, pp. 2926-2931, Nov. 2001.

[8] T. Dimitriou and A. Kalls, "Efficient Delivery of Information in Sensor Networks using Smart Antennas," *Algorithmic Aspects of Wireless Sensor Networks, Springer*, vol. 3121, pp. 109-122, June 2004.

[9] A. Savvides, W.L. Garber, R.L. Moses, and M.B. Srivastava, "An Analysis of Error Inducing Parameters in Multihop Sensor Node Localization," *IEEE Transactions on Mobile Computing* vol. 4, No. 6, pp. 567-577 Nov./Dec. 2005.

[10] R.L. Moses and R. Patterson, "Self-Calibration of Sensor Networks," *Proceedings of the SPIE*, vol. 4743, pp. 108-119, April 2002.

[11] C.-L. Chen and K.-T. Feng, "An Efficient Geometry-Constrained Location Estimation Algorithm for NLOS Environments," *IEEE Conf. on Wireless Networks, Commmunications, and Mobile Computing*, pp. 244-249, April 2006.

[12] C.-L. Chen and K.-T. Feng, "Hybrid Location Estimation and Tracking System for Mobile Devices," *IEEE Vehicular Technology Conf., Spring*, pp. 2648-2652, May 2005.

[13] S. Ventkatraman and J. Caffrey, Jr., "Hybrid TOA/AOA Techniques for Mobile Location in Non-Line-of-Sight Environments," *IEEE Conf. on Wireless Networks, Communications, and Mobile Computing*, pp. 274-278, March 2004.

[14] L. Cong and W. Zhuang, "Hybrid TDOA/AOA Mobile User Location for Wideband CDMA Cellular Systems," *IEEE Trans. on Wireless Communications*, vol. 1, no. 3, pp. 439-447, July 2002.

[15] P. Biswas and Y. Ye "Semidefinite Programming for Ad Hoc Wireless Sensor Network Localization," *IEEE Conf. on Information Processing in Sensor Networks* pp. 46-54, April 2004.

[16] P. Biswas, T.-C. Liang, K.-C. Toh, T.-C. Wang, and Y. Ye, "Semidefinite Programming Approaches for Sensor Network Localization with Noisy Distance Measurements," *http://www.stanford.edu/~yyye/ieee_tase5.pdf*.

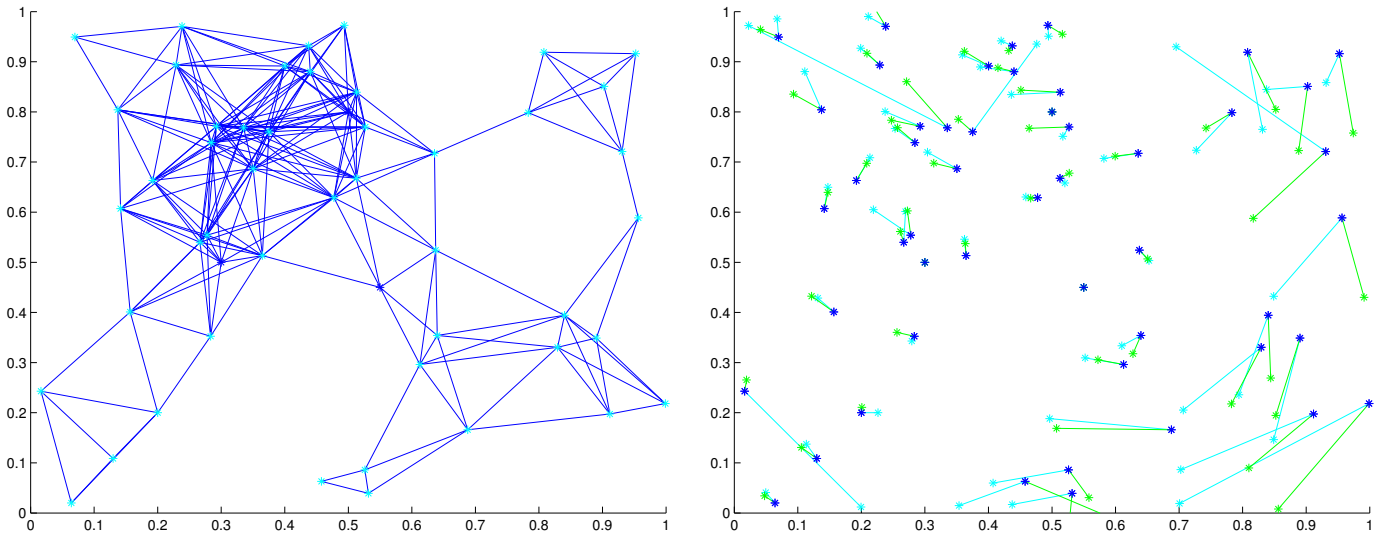
[17] C. Gentile, "Sensor Location through Linear Programming with Triangle Inequality Constraints," *IEEE Conf. on Communications*, pp. 3192-3196, May 2005.

[18] C. Gentile, "Distributed Sensor Location through Linear Programming with Triangle Inequality Constraints," *IEEE Conf. on Communications*, pp. 4020-4027, June 2006.

[19] G.D. Durgin, V. Kukshya, and T.S. Rappaport, "Wideband Measurements of Angle and Delay Dispersion for Outdoor and Indoor Peer-to-Peer Radio Channels at 1920 MHz," *IEEE Trans. on Antennas and Propagation*, vol. 51, no. 5, pp. 936-944, May 2003.

[20] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) Using AOA," *IEEE INFOCOM*, pp. 1734-1743, March 2003.

[21] M.S. Bazaraa, J.J. Jarvis, and H.D. Sherali, "Linear Programming and Network Flows," *John Wiley & Sons, Inc.*, 1990.



(a) The triangular connectivity of a network.

(b) The reconstructed sensor locations.

Fig. 6. Example network with three anchor nodes, $R = 0.25$, and $noise = 0.05$.

noise	R=0.20						R=0.25						R=0.30					
	3		5		7		3		5		7		3		5		7	
0.00	0.000	0.079	0.000	0.064	0.000	0.049	0.000	0.008	0.000	0.005	0.000	0.003	0.000	0.000	0.000	0.000	0.000	0.000
	0.003	0.050	0.003	0.042	0.002	0.029	0.001	0.004	0.001	0.002	0.000	0.001	0.000	0.000	0.000	0.000	0.000	0.000
0.05	0.035	0.097	0.034	0.094	0.034	0.091	0.031	0.094	0.030	0.092	0.030	0.090	0.027	0.092	0.025	0.089	0.024	0.087
	0.026	0.078	0.023	0.074	0.021	0.071	0.019	0.065	0.018	0.061	0.017	0.059	0.012	0.052	0.009	0.049	0.008	0.047
0.10	0.077	0.150	0.077	0.145	0.073	0.137	0.076	0.143	0.073	0.138	0.072	0.133	0.071	0.141	0.069	0.138	0.068	0.129
	0.061	0.123	0.057	0.115	0.055	0.114	0.054	0.095	0.052	0.093	0.050	0.089	0.047	0.839	0.047	0.082	0.045	0.081
0.15	0.128	0.196	0.127	0.184	0.126	0.178	0.119	0.193	0.117	0.187	0.115	0.182	0.113	0.188	0.111	0.181	0.109	0.176
	0.104	0.139	0.102	0.098	0.099	0.099	0.097	0.119	0.095	0.106	0.092	0.102	0.085	0.104	0.081	0.098	0.079	0.095

TABLE I

NUMERICAL RESULTS FOR EXPERIMENTS