

Distributed Inference for Network Localization Using Radio Interferometric Ranging

Dennis Lucarelli¹, Anshu Saxena¹, Ryan Farrell^{1,2}, and I-Jeng Wang¹

¹ Applied Physics Laboratory, Johns Hopkins University, Laurel, MD

² Computer Science Department, University of Maryland, College Park, MD

Abstract. A localization algorithm using radio interferometric measurements is presented. A probabilistic model is constructed that accounts for general noise models and lends itself to distributed computation. A message passing algorithm is derived that exploits the geometry of radio interferometric measurements and can support sparse network topologies and noisy measurements. Simulations on real and simulated data show promising performance for 2D and 3D deployments.

1 Introduction

Self-localization is a fundamental, yet not completely solved, problem in the design and deployment of sensor networks. It is *fundamental* because sensor networks are envisioned to provide visibility and monitoring with inexpensive devices in GPS denied areas. Despite many localization algorithms and improvements in device hardware, one can argue that the problem is not completely solved since no single method has been widely adopted for such a fundamental problem. Most notably, there is a need for a localization system capable of handling the multipath effects encountered indoors and in dense urban areas. The main impediment to the creation of such a system is an effective means of obtaining range measurements in a multipath environment and the development of localization algorithms that can account for these effects.

Various technologies such as ultrasound/RF TDOA ranging [1], acoustic TOA (e.g. [2]), and received radio signal strength (e.g. [3]), have been proposed and demonstrated for acquiring pairwise distance estimates. Broadly speaking and despite the ingenuity of these approaches, these methods are plagued by short range, poor precision, or the requirement of an ancillary system devoted just to ranging. Given these limitations, researchers have proposed methods for network localization that do not rely on ranging at all. These so-called *range free* methods, see for example [4,5,6], use either a camera system or in the case of [5] a steerable laser to localize the nodes. Again, these solutions require additional hardware and calibration to solve the problem.

A recent breakthrough has changed the localization landscape considerably. Researchers at Vanderbilt University have proposed and demonstrated a surprisingly simple, yet powerful, method for ranging using only the radio that produces centimeter ranging accuracy at ranges up to 160 meters [7,8]. This technique,

known as radio interferometric ranging, exploits electromagnetic interference to obtain an observable that is a function of the locations of four nodes sensors (known here as a *quad*) involved in the measurement. As such, it does not directly produce pairwise distance measurements, but rather a linear combination of four of the possible six pairwise distances. This fact renders localization algorithms that rely on pairwise distances unable to capitalize on this new technique. Perhaps the most remarkable feature of this technique is that it can be implemented with coarse time synchronization on inexpensive radios found on widely available sensor network devices.

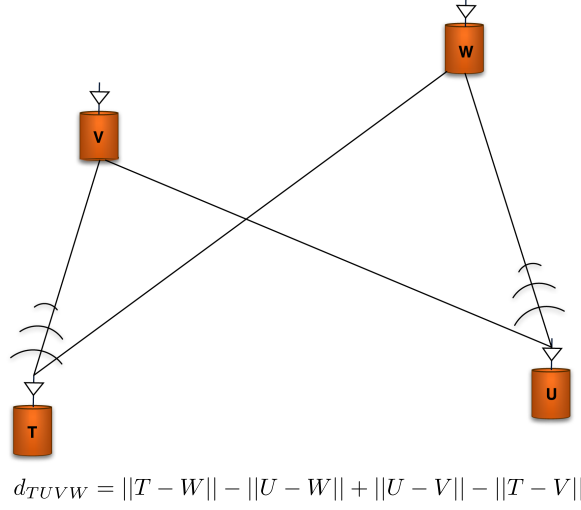
In this paper, we propose a distributed algorithm for network localization using radio interferometric ranging. We derive and exploit the geometry underlying the ranging technique that enables the algorithm. In the next section we show how the location of a node is constrained conditional on the location of the three other nodes involved in the ranging measurement. We show that, in the two dimensional case, knowing the location of three nodes constrains the fourth to a branch of a hyperbola. By taking two independent measurements on those four nodes, one can obtain another distinct hyperbola thus further constraining the nodes location to lie on the intersection of these conics. Given that the knowledge of three nodes and two independent interferometric range measurements (RIMs) reduces the uncertainty of the fourth node to just one or two intersection points, it seems plausible that a multilateration procedure can be derived akin to trilateration in systems with pairwise measurements. Indeed, assuming a 2D deployment with four anchor nodes¹ and a sensor within RIM's range of the four anchors, the unknown node can participate in up to three separate independent quads [8]. Further assuming that for each quad two independent measurements are obtained, a set of intersections points can be computed for the unknown sensor and this set of points could then potentially be used to determine the location of the unknown node.

In contrast, we adopt a probabilistic approach. Given the nonlinear relationships defined by the ranging procedure, their resulting uncertainty and our ultimate goal of developing a robust means of network localization in multi-path environments, a nonparametric probabilistic approach is preferred. We embed the underlying geometry in a flexible probabilistic model that lends itself to distributed computation. With the appropriate definition of the model, the distributed inference algorithm, known as *belief propagation*, in a sense, “comes for free.” In this regard, our approach is very much in the spirit of Ihler et al. [9], but adapted to the subtleties of dealing with radio interferometric ranging.

2 Conditional Geometry of Radio Interferometric Measurements

The functional form of the radio interferometric range measurement presents a unique challenge in designing a distributed localization algorithm. On a set of

¹ Three anchors will not suffice, since, in the general case, the uncertainty of the unlocalized node can only be reduced to two distinct intersection points.

**Fig. 1.** Radio interferometric measurement

four sensors labeled T, U, V, and W, after post processing the RIM is given by [7,8]

$$d_{TUVW} := ||T - W|| - ||U - W|| + ||U - V|| - ||T - V|| + \eta \quad (1)$$

where η represents an additive noise term. However, if three of the four nodes involved in the measurement are known, say for example, $\{T, V, W\}$, the measurement reduces to a quadratic equation in the coordinates of the unknown node,

$$d_{TUVW} = -||U - W|| + ||U - V|| + k' + \eta \quad (2)$$

where k' is the constant given by $||T - W|| - ||T - V||$. As is evident, in two dimensions the locus of points satisfying this equation is a conic section. Specifically, by setting $d_{TUVW} = k^*$ and neglecting the noise term for a moment, the equation

$$k = k^* - k' = -||U - W|| + ||U - V|| \quad (3)$$

describes the location of node U conditional on the measurement and the locations of the nodes $\{T, V, W\}$.

Equation 3 defines one branch of a hyperbola with foci at the points $\{V, W\}$. There are two independent RIMs on four nodes [8] corresponding to two distinct hyperbolae, thus the uncertainty of the unknown sensor location can be further reduced by taking an additional measurement. For example, the measurement $d_{TVUW} := ||T - W|| - ||V - W|| + ||U - V|| - ||T - U||$ defines a second hyperbola with the unknown node location at the intersection points of the hyperbolae defined by the measurements $\{d_{TUVW}, d_{TVUW}\}$. Figure 2 depicts the

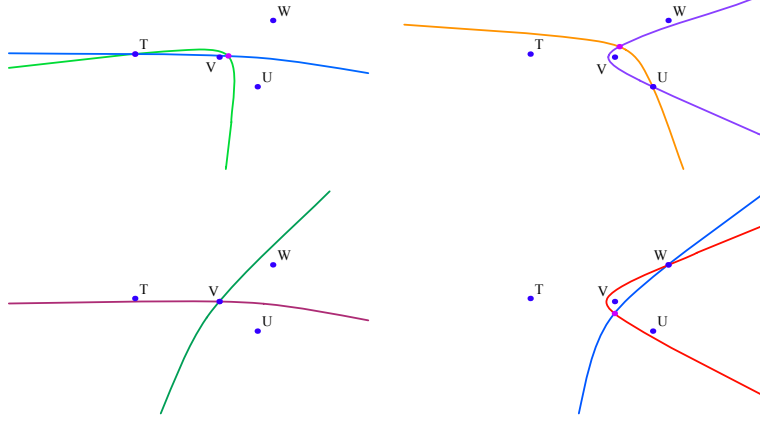


Fig. 2. Intersection points of hyperbolae defined by two independent RIMs

scenario with the unknown sensor cycling through all four possibilities with the two measurements fixed. From the figure we can get a feel for the sensitivity of the intersections to the input data. For example, the figure in the upper left quadrant (where T is the unknown node location) we see that not only two intersection points exist but also the hyperbolae almost coincide on the arc where T lies. Clearly this geometry is very susceptible to noise in the input data and hints to the difficulty in crafting a localization algorithm for ad-hoc networks with interferometric ranging. A favorable geometry is depicted in the lower left quadrant, where the hyperbolae intersect in a unique isolated point V .

Computing the intersection points of two conics, in general, requires solving a quartic equation that does not admit a convenient closed form solution. However, if the hyperbolae share a common focus, as is the case for the measurement set $\{d_{TUVW}, d_{TVUW}\}$, this system reduces to a quadratic equation and the intersection points can be computed exactly and efficiently. For example, the location of sensor V is a common focus of the hyperbolae defined by d_{TUVW} and d_{TVUW} , if sensor U is the unknown node.

Since a literature survey did not uncover the procedure for computing intersection points of conics sharing a focus, a derivation is presented here. If a translation and rotation are applied to bring the common focus to the origin and the other focus of one of the hyperbolae to the x -axis, then the equation in polar coordinates of the hyperbola with both foci on the x -axis is

$$r_1(\theta) = \frac{m_1}{e_1 \cos(\theta) - 1}, \quad (4)$$

where

$$m_1 = \|f_1\| \frac{e_1^2 - 1}{2e_1}, \quad (5)$$

e_1 is the eccentricity of the hyperbola and $\|f_1\|$ is the distance between the two foci. If the angle of elevation of the semimajor axis of the second hyperbola is ϕ , then the equation of that hyperbola is

$$r_2(\theta) = \frac{m_2}{e_2 \cos(\theta - \phi) - 1}, \quad (6)$$

where

$$m_2 = \|f_2\| \frac{e_2^2 - 1}{2e_2}, \quad (7)$$

e_2 is the eccentricity and $\|f_2\|$ is the distance between the foci. These expressions describe both branches of the hyperbolae. Once we find the intersections, we will eliminate those involving the extraneous branches. With the common focus at the origin, there is a range of values within $\arctan(\sqrt{e^2 - 1})$ of the positive direction of the semimajor axis (i.e., $\theta = 0$ for the first hyperbola, $\theta = \phi$ for the second) for which there are two points of the hyperbola – one corresponding to a positive value of r_i and lying on one branch of the hyperbola, and one corresponding to a negative value of r_i for θ halfway around the circle on the other branch. In order to find all intersections of the hyperbolae, we need to set $r_1(\theta) = r_2(\theta)$ to find intersections of portions of each hyperbola with the same sign of r_i and $-r_1(\theta + \pi) = r_2(\theta)$ to find intersections of the negative part of one hyperbola with the positive part of the other.

These equalities each yield a quadratic expression in $\cos(\theta)$ whose standard form coefficients are:

$$a_1 = \left(\frac{e_1}{m_1} - \frac{e_2}{m_2} \right)^2 + 2 \frac{e_1 e_2}{m_1 m_2} (1 - \cos(\phi)) \quad (8)$$

$$b_1 = 2 \left(\frac{e_1}{m_1} - \frac{e_2}{m_2} \cos(\phi) \right) \left(\frac{1}{m_2} - \frac{1}{m_1} \right) \quad (9)$$

$$c_1 = \left(\frac{1}{m_2} - \frac{1}{m_1} \right)^2 - \left(\frac{e_2}{m_2} \right)^2 (1 - \cos^2(\phi)) \quad (10)$$

for the first equality and

$$a_2 = \left(\frac{e_1}{m_1} - \frac{e_2}{m_2} \right)^2 + 2 \frac{e_1 e_2}{m_1 m_2} (1 - \cos(\phi)) \quad (11)$$

$$b_2 = 2 \left(\frac{e_1}{m_1} - \frac{e_2}{m_2} \cos(\phi) \right) \left(\frac{1}{m_1} + \frac{1}{m_2} \right) \quad (12)$$

$$c_2 = \left(\frac{1}{m_1} + \frac{1}{m_2} \right)^2 - \left(\frac{e_2}{m_2} \right)^2 (1 - \cos^2(\phi)) \quad (13)$$

for the second. If the discriminant $d_i = b_i^2 - 4a_i c_i$ is negative, zero, or positive, that equality has no solutions, one solution, or two solutions, respectively. Therefore, the total number of intersections of the hyperbolae can be anywhere between zero and four. If there are solutions, they are given by:

$$\cos(\theta_i) = \frac{-b_i \pm \sqrt{d_i}}{2a_i} \quad (14)$$

for $i = 1$ or 2 indicating which hyperbolic intersection equality we are evaluating. In order to solve for θ_i , we must calculate $\sin(\theta_i)$ as well to ensure we determine the correct quadrant.

$$\sin(\theta_1) = \frac{\cos(\theta_1)(e_1 m_2 - e_2 m_1 \cos(\phi)) - m_2 + m_1}{e_2 m_1 \sin(\phi)} \quad (15)$$

$$\sin(\theta_2) = \frac{\cos(\theta_2)(e_1 m_2 - e_2 m_1 \cos(\phi)) + m_1 + m_2}{e_2 m_1 \sin(\phi)} \quad (16)$$

The solutions for θ_i then can be calculated by taking the arctangent of the quotient of the sine and cosine, maintaining the correct quadrant. We eliminate those values that are outside the range of values for the branch of each hyperbola that correspond to the measurement constraint. After doing so we will have zero, one, or two values of θ_i remaining. Although for the most part we will have a nonzero number of intersections remaining since the nodes are embedded in the space and the measurements are derived from their positions, noise in the measurements can occasionally cause a situation where no intersections are possible. This indicates that the positions of at least one of the three foci is inconsistent with the measurements. When intersections remain, plugging their values back into Eq. (6) and rotating and translating back to the original coordinate system gives us the possible locations of the unknown sensor node.

The radio interferometric technique is not restricted to two dimensions. Moreover, the geometry associated with the measurements generalizes as well. In the 3D case, the conditional uncertainty of a node given the location of the three others is a hyperboloid. Two RIMs reduce the uncertainty to the intersection curve of the hyperboloids. Again if two hyperboloids share a common focus, solving a simple quadratic equation leads to an analytic parameterization of the intersection curve [10]. It turns out that this intersection curve is a hyperbola or an ellipse. Examples of the geometry of 3D RIMs is depicted in Figure 3 where a total of four measurements are taken, two generating a hyperbolic intersection and

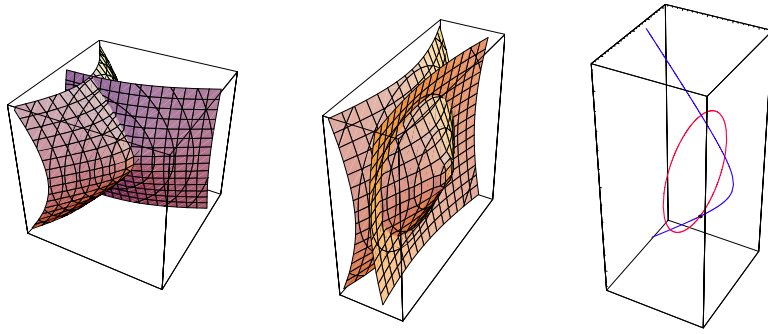


Fig. 3. Intersection curves of hyperboloids defined by RIMs in 3D

the other two generating an elliptical intersection. The location of the unknown sensor lies on the intersection of these two curves as depicted in the figure.

3 Problem Formulation

In the preceding section, the geometric intuition behind the proposed algorithm was outlined. These considerations, even in the 2D case, do not suffice to design a robust, scalable localization algorithm. Our approach embeds the multilateration primitive into a probabilistic model that allows for *soft assignments* of the sensor locations that are iteratively refined over time.

In addition to the nonlinearity introduced by the form of the interferometric measurement, noise and errors from a variety of sources affect the measurement value [7]. These sources of error, such as multipath effects, carrier frequency inaccuracy, time synchronization error and signal processing errors, are modeled by a noise distribution in our algorithm. These errors manifest in the post processing of the interferometric measurements as a possible error that is a multiple of the wavelength of the carrier frequency. This distribution can therefore be modeled as a small Gaussian mixture with components centered at integer multiples of the wavelength of the carrier frequency. It has been shown that iterative filtering techniques can be applied to the radio interferometric ranging procedure to produce high precision range estimates in a mild multipath environment that effectively remove the non-Gaussian nature of the errors [8]. Though it has not been explicitly demonstrated, an RSSI technique such as radio interferometry will likely suffer performance degradation in a high multi-path environment such as indoors or in a dense urban area. Our approach attempts to explicitly handle the effects of ranging error by associating a random variable to each sensor location and defining the joint distribution of the sensor locations that incorporates the intrinsic geometry of the radio interferometry technique.

The formalism used to capture the uncertainty and the measurement model is a probabilistic graphical model. Let $\mathcal{X} = \{x_1, x_2 \dots x_n\}$ be a collection of random variables. A graphical model is a factored representation of the joint distribution over \mathcal{X} defined by a set of *potential functions* $\psi(\cdot)$ that encode the coupling among the variables and an undirected graph that represents the notion of conditional independence among sets of random variables. Given a graph and set of potential functions, the joint distribution can be written as

$$p(x_1, x_2, \dots, x_n) \propto \prod_{c \in \mathcal{C}} \psi_c(x_c) \quad (17)$$

where \mathcal{C} is the set of fully connected subsets or *cliques* of the graph and $x_c = \{x_u : x_u \in c\}$. Graphical models are often employed for problems where the cliques can be restricted to pairs of nodes and the joint distribution given measurements is given by

$$p(x_1, x_2, \dots, x_n | \mathcal{D}) \propto \prod_{(u,v) \in \mathcal{E}} \psi_{uv}(x_u, x_v, d_{uv}) \prod_u \psi_u(x_u, d_u) \quad (18)$$

where \mathcal{E} is the set of edges in the graph and \mathcal{D} denotes the entire set of measurements, $d_{uv}, d_u \in \mathcal{D}$ and $\psi(x_u, d_u)$ is a local potential function that is used to capture any *a priori* knowledge for a node or to model a local measurement.

The graphical representation of the joint distribution has inspired many inference algorithms that exploit this graphical structure to achieve greater efficiency. An iterative message passing algorithm known as *belief propagation* is one of the best known of these methods for computing marginal distributions due to its simplicity and excellent empirical results on high dimensional problems. For problems involving spatial data in ad-hoc sensor networks, one can exploit the analogy of a *communications* graph, where an edge signifies a communication channel between sensors, and identify these two graphical representations to develop a probabilistic model for fusing measurement data that includes a simple message passing algorithm for performing inference on that model.

For continuous systems with pairwise couplings, the belief propagation update equations are given by an expression for computing a message from node u to node v , denoted $m_{uv}(x_v)$, and an expression for computing the belief at a node, β_v , which is an estimate of the marginal distribution of the random variable x_v . At iteration n , the message update is given by

$$m_{uv}^n(x_v) \propto \int \psi(x_u, x_v) \psi(x_u) \prod_{(w,u) \in \mathcal{E} \setminus (v,u)} m_{wu}^{n-1}(x_u) dx_u \quad (19)$$

Note that the message sent from node x_v in the previous iteration is excluded from the message product for consistency of the marginalization procedure.

Roughly speaking, Eq. (19) represents node u 's *belief* about the marginal of node v given its measurements and the messages from its neighbors in the graph from the previous iteration. Fusion of these messages to approximate its marginal is achieved by simply taking the product of received messages and the local potential function,

$$\beta_v^n(x_v) \propto \psi(x_v) \prod_{(w,v) \in \mathcal{E}} m_{wv}^n(x_v) \quad (20)$$

In the context of sensor networks, the attraction of belief propagation in a graphical model is evident since it is a simple way to perform *global* inference using *local* communications and *distributed* computation. However, the correctness of belief propagation is not guaranteed unless one restricts the graphical structure to a tree. Fortunately, the so called *loopy* version of belief propagation, where one carries out the message and fusion updates of Eqs. (19) and (20) without regard to the existence of loops in the graph, has shown excellent empirical performance in a variety of settings. There has been some progress in understanding the convergence of loopy belief propagation. The most useful characterization for this discussion is that if the loopy algorithm converges, it will converge to fixed points of the so called *Bethe free energy* [11]. Alternatively, one may always aggregate random variables into a *junction tree* and conduct message passing on that tree to perform inference exactly; however, the complexity of the algorithm

is exponential in a graphical property of the junction tree known as the *tree width*.

As may be intuitively obvious, a graphical model formulation of the network localization problem with interferometric ranging will not have pairwise couplings since the variables are coupled by the measurement that involves four nodes to obtain one observable. The joint distribution factors according to this coupling and is given by

$$p(x_1, \dots, x_n | \mathcal{D}) \propto \prod_t \psi(x_t) \prod_{(tuvw) \in \mathcal{Q}} \psi(x_t, x_u, x_v, x_w, d_{TUVW}) \quad (21)$$

where \mathcal{Q} is the set of quads in the graph.

There are a number of ways to define *higher order* belief propagation. One method, as mentioned above, is to aggregate variables into a junction tree. This technique has been extended for distributed inference problems in sensor networks with lossy communications in [12]. In this work, we follow a derivation obtained by minimizing the Bethe free energy in a manner analogous to the case with pairwise potential functions [13].

Let \mathcal{T}_w be the set of index triples that share an interferometric range measurement with node x_w . Formally, if $(tuv) \in \mathcal{T}_w$, we can express the message from the set $\{x_t, x_u, x_v\}$ to node x_w , denoted $m_{tuvw}^n(x_w)$, as

$$m_{tuvw}^n(x_w) \propto \int \psi(x_t) \psi(x_u) \psi(x_v) \psi(x_t, x_u, x_v, x_w) \prod_{(qrs) \in \mathcal{T}_v \setminus (tuv)} m_{qrs}^{n-1}(x_v) \prod_{(qrs) \in \mathcal{T}_u \setminus (tuv)} m_{qrs}^{n-1}(x_u) \prod_{(qrs) \in \mathcal{T}_t \setminus (uvw)} m_{qrs}^{n-1}(x_t) dx_t dx_u dx_v \quad (22)$$

Analogous to the pairwise case, the estimate of the marginal is simply the product of the incoming messages with the local potential function

$$\beta^n(x_w) \propto \psi(x_w) \prod_{(tuv) \in \mathcal{T}_w} m_{tuvw}^n(x_w) \quad (23)$$

While being functionally simple, the expression defining the messages suffers from two serious drawbacks. First, the integral is clearly intractable for densely connected networks of resource constrained devices such as sensor networks. We defer the discussion of this important consideration until the next section where a suitable approximation technique is presented. The second drawback of such an expression, which is clearly exacerbated in the case of four node couplings, is the complexity and coordination required among the sensors to implement the message passing. While being decentralized, a direct implementation of an algorithm utilizing (22) would be a poor *distributed* algorithm due to the complexity of the message passing. For example, as depicted in Figure 4, to pass a message from the triple $(ijk) \in \mathcal{T}_l$ to x_l , the sending triple would have to elect a leader node to receive the individual nodes' contributions and perform the product and integral in (22). Following the approach suggested in [14,9,15], we can simplify

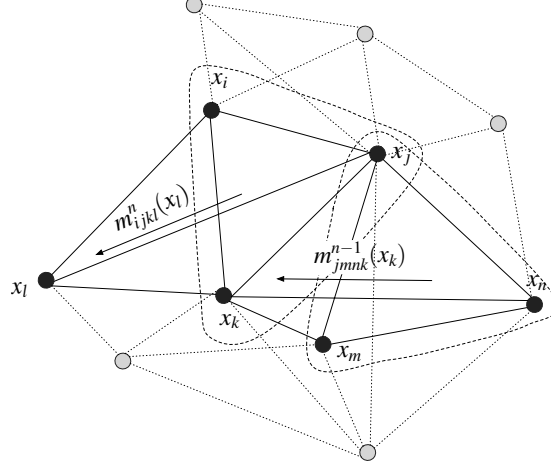


Fig. 4. Higher-order belief propagation message passing

the message passing to broadcast communications at the cost of local memory. To see this, note that the beliefs (23) contain much of the information required to form the message (22), except for the four variable potential function and the correction required for consistency. Thus, given the measurements, each node can reconstruct the incoming messages from the collection of beliefs broadcast by nodes who share measurements with it by forming,

$$m_{tuvw}^n(x_w) \propto \int \psi(x_t, x_u, x_v, x_w) \frac{\beta^{n-1}(x_t)\beta^{n-1}(x_u)\beta^{n-1}(x_v)}{m_{uvwt}^{n-1}(x_t)m_{tvwu}^{n-1}(x_u)m_{tuwv}^{n-1}(x_v)} dx_t dx_u dx_v. \quad (24)$$

In this formulation, a node updating its belief reconstructs the incoming messages by using neighboring nodes' beliefs and forming (24). These messages are then used in the node's belief update. It then calculates messages it would send to each of its neighbors using (22) and stores them locally to use in (24) in the next iteration while broadcasting its newly calculated belief instead of sending $\{m_{uvwt}(x_t), m_{tvwu}(x_u), m_{tuwv}(x_v)\}$.

4 Nonparametric Belief Propagation

In the previous section, formal expressions were presented for performing distributed inference over continuously valued random variables. A continuous model is favored over a discrete model, arising from a discretization of the sensor field, for a variety of reasons. First, the resolution of the localization solution is dictated by the size of the grid and thus the state space of the random variable grows quadratically with the precision of the solution. This would force an intractable summation in the discrete analogue of the BP equation (22). A discrete model also implies *a priori* knowledge of the size of the field which

cannot be guaranteed for ad-hoc deployments. A Gaussian model is inappropriate as well, since radio interferometric measurements exhibit non-Gaussian errors and, as shown previously, two measurements on four nodes restrict the localization solution to one or two intersection points of hyperbolae in the two dimensional case, which is a non-linear coupling of the random variables. Therefore a non-Gaussian, non-linear approach is preferred. Despite the advantages of the non-Gaussian continuous model, at this point we have only exchanged an intractable summation with an intractable integral. The key innovation of *nonparameric belief propagation* [16,17] that enables the integration of these methods to sensor networks is an efficient technique to *stochastically approximate* the message products and integral appearing in (22, 23) by sampling based methods.

To this end, we pursue a nonparametric approximation of the beliefs and messages as mixtures of Gaussians,

$$m_{tuvw}(x_w) = \sum_i^N \alpha_w^i \mathcal{N}(x_w, \mu_w^i, \Lambda) \quad (25)$$

where $\mathcal{N}(x, \mu, \Lambda)$ is a Gaussian random variable centered at the sample μ and covariance Λ , α_w^i is the weight of the i th Gaussian component of the mixture, and N is the number samples. Since the product of Gaussian mixtures is a Gaussian mixture and further assuming that the potential function can be modeled as a Gaussian mixture, the products appearing in (22, 23) are well defined and again Gaussian mixtures, albeit with $\mathcal{O}(N^q)$ components for products of q messages. If N is large enough to represent the distribution and for $q > 2$ messages, exact sampling of (22, 23) would be intractable. Several techniques for approximate sampling from Gaussian mixtures were presented in [18]. In our simulations we used an approach from [18] known as *mixture importance sampling*, though other methods performed similarly. Covariances are determined by the so-called *rule of thumb*, which is simply an estimated weighted variance of the samples.

5 Description of the Algorithm

Given the machinery of nonparametric belief propagation and analytic expressions for the intersection sets resulting from the underlying geometry, the distributed algorithm is relatively straightforward to define. To do so, we must define the probabilistic model specified by the graph and its single and four node potential functions. The graph describing the coupling of the random variables is given by the ad-hoc deployment of the sensor nodes and the radio interferometric measurements collected in the field. Thus for each measurement, we define a clique on the four participating nodes. In this work, we also assume that this graph is contained in the communications graph so that there is a communications link between all nodes sharing a measurement. In practice this may not be the case as it has been demonstrated that interferometric measurements can be obtained with relatively weak signals that are not of sufficient fidelity

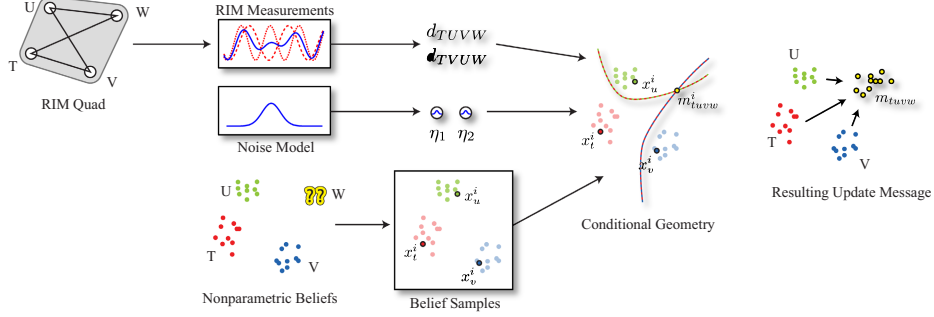


Fig. 5. Belief update

to support communications [8]. In this case, the probabilistic graphical model and the communications graph would not coincide. In our simulations we did not explicitly model the communication channel and therefore do not make this distinction here.

Figure 5 graphically depicts the belief update process for a single sample from a single quad. To update node W , the nodes $\{T, U, V, W\}$ perform the interferometric procedure to obtain the measurement set $\{d_{TUVW}, d_{TVUW}\}$. These measurements are perturbed by a sample from the noise distribution and with a sample from the belief of nodes T, U , and V the intersection set is formed to obtain one sample estimate of W 's location. The sampling procedure is repeated for all samples. As described above, the mechanics of belief propagation generalize this process for fusing the contributions from multiple quads to further refine the belief estimate and mitigate the impact of noise and multi-modality.

In the localization algorithm, the single node potentials, $\psi(x_u)$, serve as a way of incorporating a node's prior location information into the probabilistic model. As in most network localization algorithms, we ground the problem by employing *anchor nodes* which know their precise location. This information is given by some other procedure outside of the algorithm; for example, these nodes are localized using GPS. Thus for a d -dimensional anchor node the potential is given by

$$\psi(x_{anchor}) = \mathcal{N}([x_1^{gt}, \dots, x_d^{gt}], \Lambda_{anchor}) \quad (26)$$

where x_i^{gt} denotes *ground truth* and Λ_{anchor} is a diagonal covariance matrix encoding the uncertainty in the anchor's position. In our simulations we assume that the anchors have perfect knowledge of their location, thereby effectively assigning a delta function to the position covariance, though this assumption can be easily relaxed. Also, in our implementation, anchor nodes do not update their beliefs. Whereas the anchors have perfect location information, the non-anchor nodes have total ignorance. This can be modeled by specifying the single node potential as a uniform random variable over the entire sensor field. In addition to implying some *a priori* information about the size of the field, in our experiments we found that estimates were adversely affected by sampling

error of this uniform distribution. Therefore, we instantiate the single node potentials for non anchors as empty. Since the single node potentials serve as prior information, empty single node potentials achieve the desired uncertainty but this choice affects initialization and the scheduling of the message updates. We address this issue in the following section.

The four node potential functions define the coupling given by the interferometric measurement. Given a model of the noise distribution, p_η we can formally write the potential function as

$$\psi(x_t, x_u, x_v, x_w) = p_{\eta 1} \cdot p_{\eta 2} \quad \text{where} \quad (27)$$

$$p_{\eta 1} = p_\eta (d_{TUVW} - (\|x_t - x_w\| - \|x_v - x_w\| + \|x_u - x_v\| - \|x_t - x_v\|)) \quad (28)$$

$$p_{\eta 2} = p_\eta (d_{TVUW} - (\|x_t - x_w\| - \|x_u - x_w\| + \|x_u - x_v\| - \|x_t - x_u\|)) \quad (29)$$

Thus given an instantiation of the random variables and the potential functions defined as above, the joint distribution (21) expresses its likelihood. In our algorithm, these formal expressions are instantiated by the analytic expressions obtained by solving for the intersection set of the hyperbolae as described in the following.

The algorithm is initialized by performing the radio interferometric ranging procedure. Currently the coordination and estimation is executed at a base station [7,8,19]. Frequency and phase estimation is performed at the node level and from that information an estimate of the measurement d_{TUVW} can be computed. In this paper, we assume a situation where the range estimates can be obtained in the network and so that with the following algorithm, the entire localization procedure is distributed and performed in the network. In this scenario, local messages are sent so that all nodes involved in a measurement receive the range estimate. Also, as described previously, we only consider quads where two independent interferometric measurements have been taken. For simplicity, we assume that these are of the form d_{TUVW} and d_{TVUW} .

The expression defining the four node potential function is now made more concrete. The messages appearing in (24) are reconstructed at a node according to the following. Upon receiving a collection of weighted samples

$$\{\beta^n(x_t), \beta^n(x_u), \beta^n(x_v)\} = \{(x_t^i, \alpha_t^i), (x_u^i, \alpha_u^i), (x_v^i, \alpha_v^i)\}_{i=1}^N \quad (30)$$

representing a current set of beliefs for which measurements exist, node x_w propagates these samples through the potential function to construct the message m_{tuvw} . The basic idea is to use the measurements and the three sample points to form the intersection set. To this end, the updating node x_w uses the ordering of the measurement to determine the common focus of the two hyperbolae (hyperboloids). For example, for the measurements d_{TUVW} and d_{TVUW} the common focus is x_t since we have

$$d_{TUVW} = \||x_t^i - x_w^i\| - \||x_u^i - x_w^i\| + \||x_u^i - x_v^i\| - \||x_t^i - x_v^i\| \quad (31)$$

$$= \||x_t^i - x_w^i\| - \||x_u^i - x_w^i\| + k_{TUVW} \quad (32)$$

and

$$d_{TVUW} = ||x_t^i - x_w^i|| - ||x_v^i - x_w^i|| + ||x_u^i - x_v^i|| - ||x_t^i - x_u^i|| \quad (33)$$

$$= ||x_t^i - x_w^i|| - ||x_v^i - x_w^i|| + k_{TVUW} \quad (34)$$

since using the i -th sample from current beliefs sets $\{x_t^i, x_u^i, x_v^i\}$, the last two terms in each expression evaluate to constants. Note that the ordering of the measurement matters. In the interferometric ranging procedure this ordering is determined by which nodes are transmitters and which are receivers and therefore easily obtained and stored in memory. The sample set is translated and rotated so that the common focus is at the origin and another focus lies on the x axis. Now for each measurement a sample is drawn from the noise model $\eta_j \sim p_\eta$ and the constants defining the hyperbolae (hyperboloids) are perturbed by this sample. Thus we have the quadratic equations defining our constraints on the location of the updating node as

$$d_{TVUW} - k_{TVUW} + \eta_1 = ||x_t^i - x_w^i|| - ||x_u^i - x_w^i|| \quad (35)$$

and

$$d_{TVUW} - k_{TVUW} + \eta_2 = ||x_t^i - x_w^i|| - ||x_v^i - x_w^i||. \quad (36)$$

From the left hand sides of these equations and the samples defining the foci, the intersection set of the hyperbolae (hyperboloids) can be computed. In the 2D case, it is possible that the intersection set is a single point, however in general the intersection set itself must be sampled to produce the message sample m_{tuvw}^i . Note that this point must now be transformed back to the original coordinates system of the input data. Recall that the notation m_{tuvw} denotes the message “sent” from $\{x_t, x_u, x_v\}$ to x_w .

Finally the samples from the intersection set are weighted to complete a faithful approximation to (24).

$$\alpha_{tuvw}^i = \frac{\alpha_t^i \alpha_u^i \alpha_v^i \mathcal{R}(m_{tuvw}^i)}{m_{uvwt}(x_t) m_{vwtu}(x_u) m_{wtuv}(x_v)}. \quad (37)$$

In the expression defining the weight for the message sample m_{tuvw}^i we have introduced the function $\mathcal{R}(\cdot)$. This function serves to weight samples based on the notion of *range*. Because the intersection sets are sensitive to noise in the defining data, it can be the case that some samples are placed far outside the sensor field. This function limits the impact of these outliers by taking the max distance of the new sample from the incoming beliefs samples and evaluates an exponentially decreasing function on that distance. In the 3D case, the notion of maximum range also defines intervals to sample the unbounded hyperbola intersection curves so that only a segment of that hyperbola is ever used in the message update. This procedure is performed for all samples to construct a sample based estimate of the message $m_{tuvw} \approx \{m_{tuvw}^i, \alpha_{tuvw}^i\}_{i=1}^N$. When all messages have been similarly constructed, samples are drawn from the message product (23) to form the estimate of the marginal $\beta(x_w)$ as described in the previous section. Finally,

now that node x_w has an updated belief, it broadcasts its belief to neighbors and the messages appearing in the denominator of the weighting expression (37) are computed and stored in memory for use in the next iteration of belief updating. This completes one iteration of belief propagation for a single node.

6 Broadcast Scheduling

After the interferometric ranges have been computed, the message passing algorithm is initiated by localized nodes broadcasting their beliefs to neighbors (neighbors with respect to the graph defining the probabilistic model). Since non-anchor nodes are initialized with an empty prior distribution, they are silent until updating their beliefs. Clearly, since anchors are the only nodes initialized with their location, they initiate the message passing. Since in general, a single quad measurement does not suffice to localize an unknown node, it is likely that the first nodes receiving messages from the anchors will not be uniquely localized. In any case, these nodes broadcast their (perhaps multi-modal) belief. In this way, the belief updating grows out from the anchor nodes as shown in Figure 6. In the figure, the anchor nodes are depicted by diamonds and labeled as 1, 2, and 3. The complete graph on four nodes denotes the first quad and therefore the first belief update. Subsequent nodes in range can use utilize the computed beliefs or the locations of the anchors to update their own belief. This process continues until covering the entire graph and repeats with the next iteration, however now that all nodes have a nonempty belief there will likely be more quads available to refine their belief estimates. Note that we also assume that at least 3 anchors are involved in at least one quad measurement, otherwise the process would not initiate. This assumption will be relaxed in future implementations by giving all nodes some prior distribution, however it is not currently implemented and it is expected that many iterations of belief propagation will be required to localize the node sufficiently. Even with 3 anchors sharing a measurement with a non-anchor node, it is reasonable to ask under what conditions the algorithm will grow out to cover the entire network. For a partial answer we quote a result derived for localization with trilateration with pairwise range estimates. In [20], necessary and sufficient conditions were derived for network localizability using trilateration. Using a *random geometric graphs* model of the ad-hoc configuration of sensor nodes and the existence of

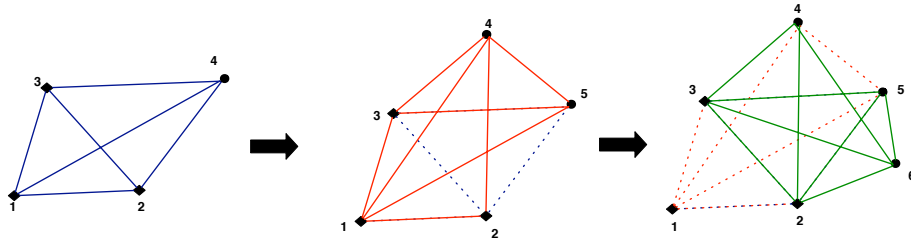


Fig. 6. Graph growing out from the anchors

pairwise range measurements between nodes, asymptotic results were obtained for determining the existence of a so-called *trilaterative ordering* of the vertices in a graph. A trilaterative ordering in dimension d for a graph is an ordering of the vertices $1, \dots, d+1, \dots, n$ such that $1 \dots d+1$ are fully connected and from every vertex $j > d+1$, there are least $d+1$ edges to vertices in the ordering. By appealing to graph rigidity theory, the authors show that the existence of trilaterative ordering is a necessary and sufficient condition for unique localizability of the network. Moreover, they establish an asymptotic result that for a network of n nodes with measurement range r , if $\lim_{n \rightarrow \infty} \frac{nr^2}{\log n} > 8$, then there exists a trilaterative ordering with high probability. In our case, this is a *necessary* condition for the broadcast schedule to cover the entire graph in the first iteration of belief propagation. Given the underlying geometry of the ranging procedure and the uncertainty in the measurements, additional iterations of message passing are needed for sufficient localization. However, this result which is satisfied by dense (measurement) graphs yields a theoretical assurance that our algorithm will terminate with all nodes being involved.

7 Simulation Results

To assess the performance of the algorithm we performed simulations with real and simulated data. We implemented the algorithm as described in the previous sections in MATLAB. This centralized version of the algorithm retains all the components necessary for a distributed implementation, but the simplicity of a centralized algorithm allowed for the focus to be on the algorithm and not on technical (albeit important) issues regarding wireless communications and limited computational power. We used the KDE Toolbox [21], a MATLAB toolbox with optimized data structures and sampling procedures, for the Gaussian mixture product sampling.

For a point of comparison, the real data used in our experiments was the “football field” data provided by Vanderbilt University [19]. This data set contains over 7000 RIM’s for a network of 16 nodes placed in an approximate grid.

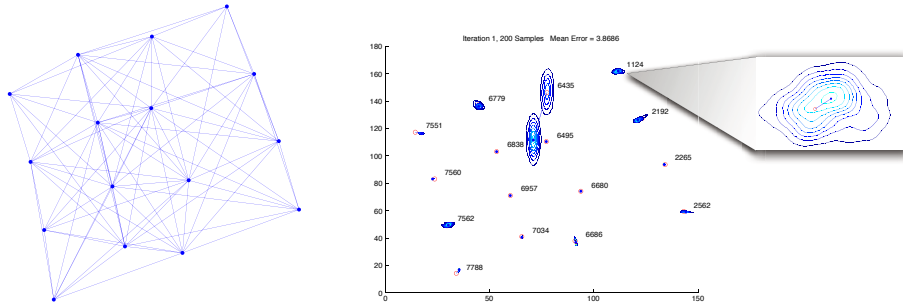


Fig. 7. Nearest neighbor quads graph from the football field data set and first iteration marginals

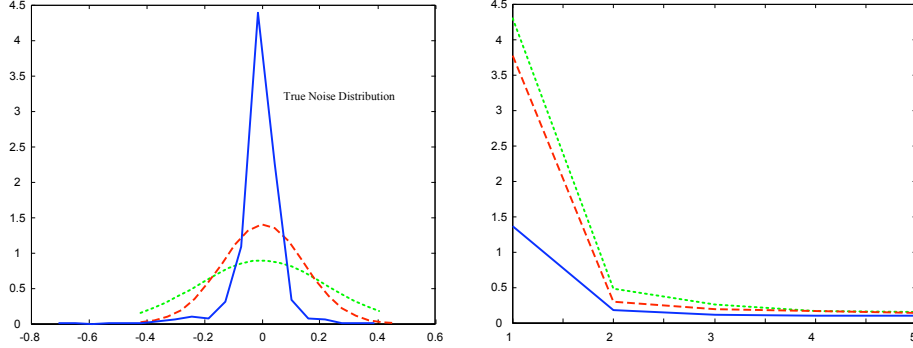


Fig. 8. Noise distributions used in simulations and average localization results

This data set benefits from the filtering technique proposed in [8] to refine the range estimates resulting in an error distribution that is nearly Gaussian with variance approximately .007, depicted by the solid (blue) curve in Figure 8. Excellent localization results with this data set were obtained in [8] using a centralized genetic algorithm executed at a base station. The genetic algorithm of [8] does not exploit the geometric structure of the problem, but rather does something akin to exhaustive search to find a minimum of the associated optimization problem. To obtain the centimeter localization accuracy reported in [8], this 7000 element data set was used.

From this data set we generated a random *nearest neighbor* graph simulating measurements in our simulation. This graph represents just 53 quads or equivalently 106 interferometric measurements. Three central nodes, labeled {6680, 6838, 6957} were chosen as anchors. Figure 7 shows the quad graph constructed from the football field data set and the marginals from the first iteration of belief propagation. As in all our simulations, the final localization results is taken as the maximum of the marginal distribution. Note that in the results plot, node 6435 has a multimodal marginal distribution. It turns out that node 6435 is the first updating node and with only 3 anchors, there is only one measurement with which to update its belief resulting in the bimodal distribution. Samples approximating this distribution are broadcast to neighbors for their updates. Its important to note that even with the bimodality, neighboring nodes are able to refine their estimates fairly well in the first iteration. Note also that nodes on the boundary are localized but with some uncertainty as shown in the close-up. In this particular simulation, successive iterations of the message passing drove down the mean error to less than 15 centimeters.

The approach is sensitive to noisy messages. Taking the product of messages in equations (22, 23) is effectively equivalent to performing an AND operation on the messages and looking mostly at the intersection region of all messages. A single noisy message has a heavy hand in altering this region. The result is a smaller region of support from the message product that leads to samples that

are closely clustered, possibly lending misplaced confidence in their locations. Additionally, these locations can be removed from the true location of the node, leading to a situation where a node is confidently localized to the wrong location. Setting the bandwidths to capture the spread of the incoming messages may help to alleviate this situation.

From a Bayesian point of view, our algorithm relies on two sources of prior knowledge: the noise distribution and the maximum measurement range. Since these quantities can be estimated but never known with certainty before deployment, it is interesting to investigate the impact of our certainty of these quantities on the localization results. To test this, we performed 5 iterations of the belief propagation over 10 trials to get average localization results for various noise distributions. The results of this experiment are shown in Figure 8. The true noise distribution calculated from ground truth information from the entire data set is depicted by the solid (blue) curve in the left panel of Figure 8. The solid (blue) curve in the right panel shows the mean error per iteration in meters when the true noise distribution is used in the algorithm. Recall that we use our knowledge of the noise distribution by perturbing the measurements before solving for the intersection set (Eqs. (35) and (36)). Similarly, if we assume *a priori* that the noise distribution is given by the dashed (red and green) curves in the right panel, the corresponding localization estimates suffer somewhat. Not surprisingly, perfect knowledge of the noise distribution sharpens the localization results. However, in these experiments, even though our noise distribution assumptions are qualitatively different from the true distribution, the results are not affected too severely. A similar analysis with respect to our prior knowledge regarding the maximum range showed similar robustness.

In an effort to understand the impact of the grid layout of the football field data set, we created simulated data sets with ad-hoc deployments by placing nodes according to a uniform distribution over the field. A grid layout supports favorable geometries of the quads and limits the adverse situations depicted in Figure 2. These experiments exposed the dependence on the message schedule – if

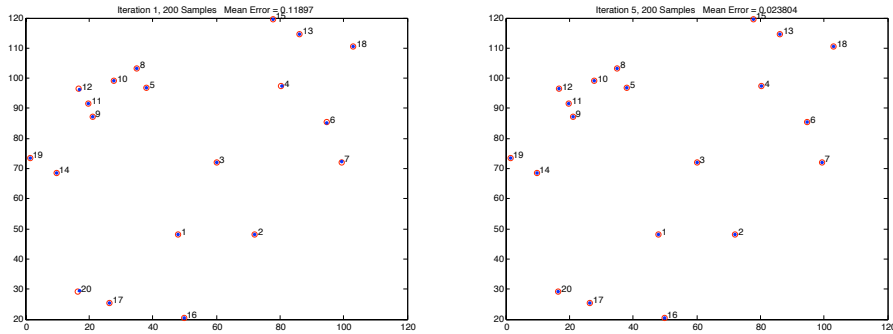


Fig. 9. Localization results ad-hoc deployment

in the first iteration of belief propagation most nodes update with only one quad measurement, it can cause instability in the updates of nodes not involved in measurements with anchors. Thus, merely satisfying the necessary condition stated in Section 6, may result in poor estimates. This is especially true with noisy measurements and “unfavorable” geometries as depicted in the first quadrant of Figure 2. However, we found empirically that if in the first iteration of belief propagation most nodes had at least 2 quad measurements, results were comparable to the football field data set simulations. As an example, Figure 9 depicts a scenario where most nodes updated with almost 3 quads in the first iteration with an mean of 7.4 quads for subsequent iterations.

As a final simulation, we investigated the performance of the algorithm on a three dimensional network. Initial results show success as a proof of concept, however more work is needed for the algorithm to be a viable method for three dimensional localization. However, given that there is no physical limitation to precise interferometric ranging in 3D and the scarcity of non-planar localization techniques, we find these initial simulations promising. As a test set we created a 3D lattice of 27 nodes and designated nodes 1 through 5 as anchors. Ideally, four non-planar anchors should suffice, however in our initial simulations with 4 anchors a fraction of the nodes could not localize with less than 2 units of error. This test set contained only 66 quads (for a total of 112 measurements). Results from 2 iterations of belief propagation are shown in figure 10. We also experimented with irregular configurations as well, with similar performance, however it was difficult to avoid nearly co-planar quads that adversely affected some nodes localization. Figure 11 is an output from the simulation showing 3 messages contributing to the belief update of node 12. Though perhaps difficult to see, there are 3 hyperbola segments contributing the belief pictured in the left panel of the figure. Two of these are messages constructed from triples consisting entirely of anchors, hence the thin curve representing the message. One message is constructed from messages from non-anchor nodes that have updated previously in the iteration of belief propagation. This message is clearly corrupted by noise and the location uncertainty of the sending nodes.

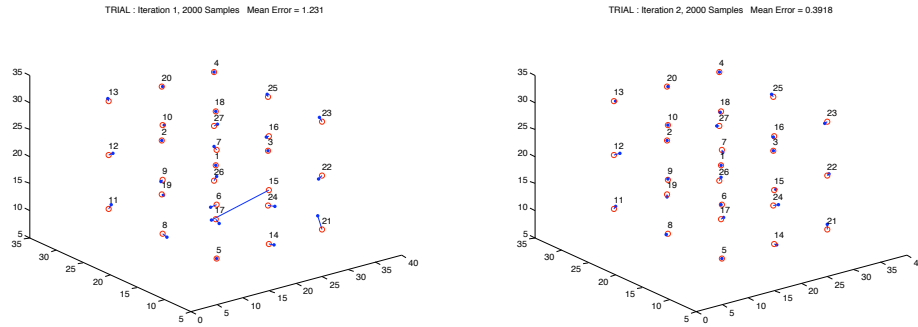


Fig. 10. Localization results for 4 iterations of belief propagation

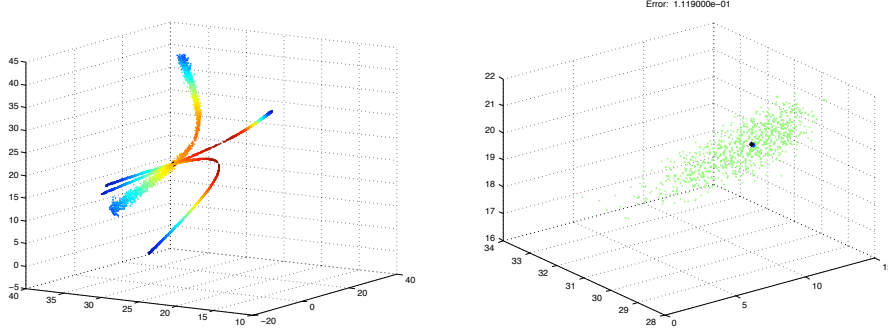


Fig. 11. Three messages and the marginal estimate from the product sampling

A significant drawback in the 3D case is the number of samples required for sufficient approximation of the messages. In our simulations we used 2000 samples. Associated with this high number of samples would be a significant communication cost that would likely limit the algorithm's effectiveness in a sensor network deployment. It would be interesting, though not considered here, to apply a message compression technique as in [22] to limit the number of samples transmitted.

8 Conclusion

The localization problem in sensor networks generally involves two separate tasks: ranging and the localization algorithm itself. Ranging is a fundamentally physical problem limited by power constraints, process noise and device characteristics. Radio interferometry is a significant advance that does not produce pairwise ranges, but rather a distance measurement that is a function of the locations of four nodes. This technique can produce very precise measurements at relatively long ranges. In this paper, we have contributed to the other half of the localization problem. Namely, we have proposed an algorithm that exploits the radio interferometry technique and we have shown centimeter localization accuracy on real and simulated data sets. We have defined a flexible probabilistic model that can account for non-Gaussian noise models and lends itself to distributed computation. Aside from the advantages of a distributed implementation, we have shown that the performance of our method compares favorably with the current centralized algorithm while using far fewer interferometry measurements. We have proposed nonparametric belief propagation as the machinery that enables an efficient solution. Nonparametric belief propagation is an approximation based on Monte Carlo sampling whose trade-off between efficiency and accuracy is dependent on the number of samples being used. As technological improvements continue to make faster computation cheaper and smaller, distributed sensor systems will increasingly be able to perform the necessary calculations associated with nonparametric belief propagation to satisfy

approximation error requirements. As for future work, it would be interesting to investigate additional interferometry data sets exhibiting non-Gaussian errors to assess the possibility of using the technique in a multipath environment and explore designs that can be implemented on the current generation of sensor network devices.

Acknowledgments. The authors thank Andreas Terzis and Dan Wilt for helpful discussions. This work was supported by Independent Research and Development funding.

References

1. Priyantha, N., Chakraborty, A., Balakrishnan, H.: The cricket location-support system. In: Proceedings of the 6th ACM MOBICOM Conference (2000)
2. Girod, L., Estrin, D.: Robust range estimation using acoustic and multimodal sensing. In: IEEE International Conference on Intelligent Robots and Systems (2001)
3. Bahl, P., Padmanabhan, V.N.: RADAR: An in-building RF-based user location and tracking system. In: Proceedings of INFOCOM 2000, pp. 775–784 (March 2000)
4. Barton-Sweeney, A., Lymberopoulos, D., Savvides, A.: Sensor Localization and Camera Calibration in Distributed Camera Sensor Networks. In: Proceedings of IEEE BaseNets (October 2006)
5. Stoleru, R., He, T., Stankovic, J.A., Luebke, D.: A high-accuracy, low-cost localization system for wireless sensor networks. In: SenSys 2005. Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, pp. 13–26. ACM Press, New York (2005)
6. Farrell, R., Garcia, R., Lucarelli, D., Terzis, A., Wang, I.-J.: Localization in multimodal sensor networks. In: Third International Conference on Intelligent Sensors, Sensor Networks, and Information Processing (to appear, 2007)
7. Maróti, M., Völgyesi, P., Dóra, S., Kusý, B., Nádas, A., Lédeczi, Á., Balogh, G., Molnár, K.: Radio interferometric geolocation. In: SenSys 2005. Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, pp. 1–12. ACM Press, New York (2005)
8. Kusý, B., Maróti, Á.L.M., Meertens, L.: Node density independent localization. In: IPSN 2006. Proceedings of the Fifth International Conference on Information Processing in Sensor Networks, pp. 441–448. ACM Press, New York (2006)
9. Ihler, A.T., Moses, R.L., Fischer III, J.W., Willsky, A.S.: Nonparametric belief propagation for self-localization of sensor networks. *IEEE Journal on Selected Areas in Communications* 23(4), 809–819 (2005)
10. Fang, B.: Simple solutions for hyperbolic and related position fixes. *IEEE Transactions on Aerospace and Electronic Systems* 26(5), 748–753 (1990)
11. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Understanding Belief Propagation and its Generalizations. In: International Joint Conference on Artificial Intelligence (August 2001)
12. Paskin, M.A., Guestrin, C., McFadden, J.: A robust architecture for distributed inference in sensor networks. In: IPSN. Proceedings of the Fourth International Conference on Information Processing in Sensor Networks, pp. 55–62 (2005)
13. Zhang, D.-Q., Chang, S.-F.: Learning to Detect Scene Text Using Higher-order MRF with Belief Propagation. In: IEEE Workshop on Learning in Computer Vision and Pattern Recognition (June 2004)

14. Koller, D., Lerner, U., Angelov, D.: A general algorithm for approximate inference and its application to hybrid bayes nets. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence (1999)
15. Bickson, D., Dolev, D., Weiss, Y.: Modified belief propagation algorithm for energy saving in wireless sensor networks. Technical Report TR-2005-85, The Hebrew University (2005)
16. Sudderth, E., Ihler, A., Freeman, W., Willsky, A.: Nonparametric belief propagation. In: CVPR (2003)
17. Isard, M.: Pampas: Real-valued graphical models for computer vision. In: Proceedings of CVPR (2003)
18. Ihler, A.T., Sudderth, E.B., Freeman, W.T., Willsky, A.S.: Efficient multiscale sampling from products of Gaussian mixtures. In: Thrun, S., Saul, L., Schölkopf, B. (eds.) Neural Information Processing Systems 16, MIT Press, Cambridge (2004)
19. Kusy, B., Balogh, Gy., Ledeczi, A., Sallai, J., Maroti, M.: <http://tinyos.cvs.sourceforge.net/tinyos/tinyos-1.x/contrib/vu/tools/java/isis/nest/localization/rips/>
20. Eren, T., Aspnes, J., Whiteley, W., Yang, Y.R.: A theory of network localization. IEEE Transactions on Mobile Computing 5(12), 1663–1678 (2006)
21. Ihler, A.: Kde toolbox, <http://ttic.uchicago.edu/~ihler/code/kde.php>
22. Ihler, A.T., Fisher III, J.W., Willsky, A.S.: Communication-constrained inference. Technical Report 2601, MIT, Laboratory for Information and Decision Systems (2004)