Aalborg Universitet

AALBORG UNIVERSITY

DENMARK

Performance analysis of sensor self-localization algorithms

Bøgsted, Martin; Rasmussen, Jakob Gulddahl

AALBORG UNIVERSITY

Performance analysis of sensor self-localization algorithms

by

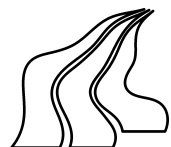Martin Bøgsted Hansen and Jakob Gulddahl Rasmussen

R-2007-03

January 2007

DEPARTMENT OF MATHEMATICAL SCIENCES
AALBORG UNIVERSITY
Fredrik Bajers Vej 7 G ▪ DK - 9220 Aalborg Øst ▪ Denmark
Phone: +45 96 35 80 80 ▪ Telefax: +45 98 15 81 29
URL: www.math.auc.dk/research/reports/reports.htm

Performance analysis of sensor self-localization algorithms

Martin B. Hansen and Jakob G. Rasmussen

Abstract

In this paper the self-localization problem for sensor networks is discussed. We suggest to use the configuration of sensors that has overall maximum probability, given the observations. In a Bayesian framework this corresponds to maximum a posteriori (MAP) estimation. However, there is a main reservation concerning this approach: The computational problem of solving the complex nonlinear optimization problem seems at first glance to be enormous. We suggest in the present paper to reduce the computational burden by a simple coordinatewise greedy algorithm, which is nothing else than the successful iterative conditional modes (ICM) algorithm from spatial statistics and image analysis. The advantages are that it is 1) distributed, 2) simple and 3) easy to implement. A theoretical lower bound on the average mean square error (AMSE) for all localization estimators in multihop sensor networks is presented under suitable regularity conditions on the sensor positions. A simulation study is conducted and it is shown that the AMSE of the proposed estimator for a variety of parameters is close to the lower bound on the AMSE.

Index Terms

Mean square error methods, MAP estimation, distance measurements, position measurements, networks.

I. INTRODUCTION

SELF-LOCALIZATION of wireless devices is interesting for a number of military and civil applications. The basic problem is to deploy a large number of sensors that acquire and process data. In the situation where the location of the sensors is not available there is an interest in developing methods where the sensors are localizing themselves based on observations from their neighbourhood. Furthermore, it is desirable to distribute the computational burden across the network and minimize the amount of inter-sensor communication to avoid congestion problems.

Generally we assume in the present paper a localization problem in which each sensor has available noisy distance measurements to neighbouring sensors. These measurements can be e.g. time of arrival (TOA) or received signal strength (RRS) suitably inverted to distance measurements. A straightforward approach to solving such a localization problem is to minimize the sum of the squares of the differences between inter-sensor distances and measured distances, leading to a nonlinear least-squares (NLS) optimization problem [1], [2]. A more formal and model-based approach is to use maximum likelihood (ML) estimators [3], [4]. We suggest to use the configuration of sensors that has overall maximum probability, given the observations. In a Bayesian framework this corresponds to maximum a posteriori (MAP) estimation.

In general, finding the NLS, ML and MAP estimates of the sensor locations all results in complex nonlinear optimization problems. This motivates a systematic study of the whole battery of nonlinear-optimization techniques invented in e.g. image analysis, optimal control and spatial statistics. Various methods have been suggested e.g. simulated annealing [5]. Simulated annealing is tempting as convergence results can be proved. However, annealing with a theoretically optimal cooling schedule may work very slowly, which is in contrast with the desire for fast computational schemes. Therefore, in practice faster cooling schedules like 'zero temperature sampling' are adopted. These algorithms usually terminate in a local minimum close to the initial configuration after a few iterations. The results depend on the initial configuration and on the visiting scheme. Despite its obvious drawbacks, it is popular since it is fast and

easy to implement. Performance of the estimators is often investigated and compared by the average mean square error [6], [7].

In Section II, we formalize the problem and discuss the signal propagation model used in this paper. Section III introduces Bayesian estimation theory as inferential tool and we suggest to use the average mean square error to evaluate the performance of the estimator. By means of van Trees' inequality [8], [9] we are able to derive a lower bound on the AMSE for all estimators under suitable regularity conditions on the sensor positions. To our knowledge results along these lines have been restricted to Cramér-Rao lower bounds [3]. The Cramér-Rao bound is limited to unbiased estimates whereas the estimators are typically biased in sensor localization problems. Although this can be remedied by analyzing the bias sensitivity of the position estimates [10, Section 3.5] or focusing on the one-dimensional case [11], our approach works in general. We suggest in Section IV to use 'coordinatewise' maximal descent, which is common in combinatorial optimization. In the statistical community, it is usually called ICM and credited to Besag in [12] and the further development in [13] (it was suggested independently by [14]). It was Green who in the discussion of [13] pointed out the close relation to 'zero temperature sampling' in simulated annealing. In Section V we treat the case where all pairwise distances have been measured. An estimator is suggested and the van Trees lower bound is calculated for a bivariate beta a priori distribution on the sensor positions. The situation where every sensor only detects a distance from a subset of the remaining sensors is treated in Section VI. Simulation results are reported in Section VII and it is shown how the AMSE of the estimator, for realistic parameter settings, comes close to the theoretical lower bound.

II. Model formulation

A. Notation

Assume we have a set of N sensors, labeled $V = \{1, \ldots, N\}$, scattered around $S \subseteq \mathbb{R}^2$, each with an associated location $z_1, \ldots, z_N$. The positions of the first M sensors, $K = \{1, \ldots, M\}$, $z_K = (z_1, \ldots, z_M)$ are known, whereas the positions of the rest of the sensors, $U = \{M + 1, \ldots, N\}$, $z_U = (z_{M+1}, \ldots, z_N)$ are unknown.

B. Signal propagation model

For the distance measurements in the model we adopt the approach of [15] and [16] in the following way. Sensor $i \in U$ obtains a noisy measurement d_{ij} of its distance from sensor $j \in V$. We assume in the present paper a Gaussian measurement model, that is $d_{ij} \sim \mathrm{N}(\| z_i - z_j \|, \sigma^2)$-distributed. We use a binary random variable o_{ij} to indicate whether the distance between sensor i and j is observed or not. The indicator o_{ij} is assumed to be Bernoulli distributed with a probability depending on the distance between sensor i and j, in the following way

$$P_o(z_i, z_j) = \exp\left(-\frac{\| z_i - z_j \|^2}{2R^2}\right). \tag{1}$$

This model assumes that the probability of detecting nearby sensors falls off exponentially with squared distances. Throughout the paper we will assume the measurements and indicators to be independently distributed.

C. A Bayesian hierarchical model

In reality the measured distances d_{ij} and d_{ji} may be different, and it is even possible to have $o_{ij} \neq o_{ji}$. It will later be a convenient assumption to assume $d_{ij} = d_{ji}$, but the set-up is easily generalized to the non-symmetric case at the cost of additional computations. Moreover, we assume the vector of unknown

nodes is an $N - M$-dimensional random vector with a priori probability density function p. Hence, the model is naturally formulated as a Bayesian hierarchical model [17],

$$
\begin{aligned}
z_U &\sim p \\
o_{ij}|z_i, z_j &\sim_{\text{iid}} \text{bin}(1, P_o(z_i, z_j)), \ i \in U, 1 \leq j < i \\
d_{ij}|z_i, z_j &\sim_{\text{iid}} \text{N}\left(\| z_i - z_j \|, \sigma^2\right), \ i \in U, 1 \leq j < i.
\end{aligned}
$$

Now the set of *observed/available* distances $d = \{d_{ij}|o_{ij} = 1\}$ is a random variable taking values in $\cup_{n=0}^{L}\mathbb{R}^n$, where $\mathbb{R}^0 = \emptyset$ and $L = (N - M)(N + M - 1)/2$. The latter number is easily obtained by induction.

III. STATISTICAL INFERENCE

We will in this section approach the inferential problem by Bayesian estimation theory. This is a huge area and a total review is out of scope of the present paper, we refer to e.g. [18] for a comprehensive treatment.

A. Position estimates

There are many ways to pick a $\hat{z}_U$ which represents the true configuration of positions, i.e. which is in proper balance between the a priori expectation and the fidelity to the data. By Bayes theorem the a posteriori distribution of the sensor localization given the measurements can be expressed by

$$
p(z_U|d) \propto_{z_U} p(d|z_U)p(z_U),
$$

where $f(x, y) \propto_x g(x, y)$ means there exists a constant $c(y)$ such that $f(x, y) = c(y)g(x, y)$.

One possible rule is to choose a $\hat{z}_U$ which is most favourable for $p(z_U|d)$, i.e. to maximize the a posteriori distribution. Henceforth, we define $\hat{z}_U$ to be the maximum a posteriori estimate of z_U given d.

B. Performance of the estimator

In Bayesian estimation theory estimators are studied in terms of loss functions. The "loss" of estimating a true z_U by $\hat{z}_U$ or the "distance" between $\hat{z}_U$ and z_U is measured by a loss function $L(z_U, \hat{z}_U)$. The choice of L is problem specific. In accordance with previous literature [6], [7] on the subject we choose the average square error loss function

$$
L(z_U, \hat{z}_U) = \frac{1}{N - M} \sum_{i \in U} \| z_i - \hat{z}_{U,i} \|^2.
$$

The Bayes risk of the estimator $\hat{z}_U$ is the mean loss or in our context the AMSE

$$
\begin{aligned}
\text{AMSE} &= \frac{1}{N - M} \sum_{i \in U} \mathbf{E}_z \mathbf{E}_{d|z} \| \hat{z}_{U,i}(d) - z_{U,i} \|^2 \\
&= \frac{1}{N - M} \sum_{i \in U} \mathbf{E}_z \mathbf{E}_{d|z} \left((\hat{x}_{U,i}(d) - x_{U,i})^2 + (\hat{y}_{U,i}(d) - y_{U,i})^2 \right).
\end{aligned}
$$

Here $\mathbf{E}_{d|z}$ denotes expectation with respect to possible realizations of data given positions z of the sensors and $\mathbf{E}_z$ denotes expectation with respect to the a priori distribution of the sensor positions. A natural estimator to choose is the one which minimizes this risk. It can be proved that the Bayes estimator for the average square error loss function is the expected mean of the a posteriori distribution. It turns out, however, that it is possible to make a distributed algorithm to calculate this estimator, but the method is quite computer intensive, see [19]. We therefore study various approximations of the MAP in more detail and show by simulation that they work quite well according to the lower bound on the AMSE.

The Cramér-Rao lower bound has turned out be a popular means to characterize uncertainty of position estimates. As noted by [3], this approach is questionable as the Cramér-Rao bound only holds if the estimator is unbiased, whereas most estimators in general and the maximum likelihood estimator for localization in particular are biased.

Here we show that a Bayesian version of the Cramér-Rao bound due to van Trees [8] remedies some of the problems and directly provides a lower bound on the AMSE for any estimator $\hat{z}_U$. First we should note that van Trees' inequality is derived under suitable regularity conditions on the functions $p(d|z_U)$ and $p(z_U)$, see [9, Section 4] and references therein for details. These rather technical conditions can easily be verified for the situations we are considering in the present paper and are left for the interested reader. Then define the expected Fisher information matrix for the position parameter and a priori distribution

$$I(z_U) = -\mathbf{E}_{d|z} \left(\frac{\partial^2 \log p(d|z_U)}{\partial(x_U, y_U)^{\mathrm{T}} \partial(x_U, y_U)} \right),$$

and

$$I(p) = -\mathbf{E}_z \left(\frac{\partial^2 \log p(z_U)}{\partial(x_U, y_U)^{\mathrm{T}} \partial(x_U, y_U)} \right).$$

Finally, van Trees' inequality [8, page 84] yields

$$\text{AMSE} \geq \frac{1}{N - M} \operatorname{tr} \left(\mathbf{E}_z I(z_U) + I(p) \right)^{-1}. \tag{2}$$

This is similar to the Cramér-Rao lower bound but does not require that the estimator is unbiased, as long as we put a suitable a priori distribution on the sensor positions.

IV. The ICM algorithm

As indicated in Section I there exists a wealth of methods to approximate the MAP estimate, but we choose in this paper to focus on the ICM algorithm, which was introduced in Section I as the 'coordinatewise' steepest descent algorithm. Specifically, for each $i \in U$ choose the localization of sensor i to be the maximizer of the marginal a posteriori distribution

$$p(z_i|z_{V \setminus \{i\}}, d) \propto_{z_i} p(d|z_i, z_{V \setminus \{i\}}, d) p(z_i|z_{V \setminus \{i\}}). \tag{3}$$

This method has a computational advantage if both terms of the right-hand side have local dependence properties. This will indeed be the case for the models introduced below. The ICM scheme is iterated until sufficiently converged. The sensors will be visited according to some visiting scheme, i.e. a sequence $(U_n)_{n \geq 1}$ of nonempty subsets of U. The visiting scheme has to cover U again and again and therefore we require

$$U = \bigcup_{j=\tau(k-1)+1}^{\tau(k)} U_j, \text{ for every } k \geq 1 \tag{4}$$

for some increasing sequence $\tau(k)$, $k \geq 1$, of times. Finally, we set $\tau(0) = 0$. Such a visiting scheme could be either deterministic or stochastic. In the present paper we simply for convenience use sequential visiting, but in reality the network should use some sort of pseudo-random visiting scheme to avoid complicated clock-synchronisation problems.

In the literature several suggestions have been proposed for choosing the initial configuration of the ICM scheme. In the present paper we choose the sum-dist algorithm [20], which is a simple solution for determining the distance to the sensors with known positions by adding the ranges encountered at each hop during a initial network broadcast. For more details, see Section VII. The ICM algorithm for sensor self-localization is schematically given in Algorithm IV.1.

Finally we note that

$$p(z_U|d) = p(z_i|z_{V \setminus \{i\}}, d) p(z_{V \setminus \{i\}}|d),$$

Algorithm IV.1 ICM for sensor self-localization

INITIALIZE $z_U^{(0)}$ by the sum-dist algorithm (see Section VII)
$n := 0$
REPEAT
      FOR $i \in U_n\{$
$$z_i^{(n)} := \arg\max_{z_i} p\left(d|z_i, z_{V\setminus\{i\}}^{(n)}\right) p\left(z_i|z_{V\setminus\{i\}}^{(n)}\right)$$
      $\}$
      $z^{(n+1)} := z^{(n)}$
      $n := n + 1$
UNTIL sufficiently converged (see Section VII)

---

so by (3) $p(\hat{z}_U^{(n)}|d)$ never decreases. It is our experience that convergence, to what must therefore at least be a local minimum of $p(z_U|d)$, seems extremely rapid, with few changes after the 10th cycle, see Section VII.

V. FULL OBSERVATIONS

First we consider the situation where $P_o \equiv 1$, i.e. each sensor detects a distance measurement from every other sensor, called the full observations case. Under these circumstances, by Bayes theorem, the a posteriori distribution is given by

$$
\begin{aligned}
p(z_U|d) \quad &\propto_{z_U} \quad p(d|z_U)p(z_U) \\
&= \prod_{i \in U, 1 \le j < i} \left(\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(d_{ij} - \| z_i - z_j \|)^2}{2\sigma^2} \right) \right) \times p(z_U),
\end{aligned} \tag{5}
$$

whereby the marginal a posteriori pdf, to be plugged into the ICM Algorithm IV.1, is given by

$$
\begin{aligned}
&p(z_i|z_{V\setminus\{i\}}, d) \\
&\propto_{z_i} \prod_{j \in V\setminus\{i\}} \left(\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(d_{ij} - \| z_i - z_j \|)^2}{2\sigma^2} \right) \right) \times p\left(z_i|z_{V\setminus\{i\}}\right).
\end{aligned} \tag{6}
$$

Although locally specified point processes can be a powerful modelling tool, we shall only consider a simple example here.

Example 1 (Binomial point process): Let p be a probability density function (pdf) on $S \subset \mathbb{R}^2$. A point process of n iid points with pdf p is called a binomial point process of n points in S with pdf p. If we assume z_U is a binomial point process of $N - M$ points with pdf p, we get the following simultaneous pdf for z_U

$$p(z_U) = \prod_{i \in U} p(z_i).$$

In this case we get the following plug-in for the ICM Algorithm IV.1

$$p\left(z_i|z_{V\setminus\{i\}}\right) = p(z_i).$$

This expression in turn obviously only involves a local computation. Hence, running the ICM algorithm is straightforward.

To derive a van Trees lower bound on the AMSE, form three $(N - M) \times (N - M)$ matrices $I(z_U)_{xx}$, $I(z_U)_{xy}$ and $I(z_U)_{yy}$ (see also [3, (7)]) with elements

$$[I(z_U)_{xx}]_{ij} = \frac{1}{\sigma^2} \begin{cases} \sum_{k \in K} \frac{(x_i - x_k)^2}{\| z_i - z_k \|^2} + \sum_{k \in U \setminus \{i\}} \frac{(x_i - x_k)^2}{\| z_i - z_k \|^2} & i = j \\ -\frac{(x_i - x_j)^2}{\| z_i - z_j \|^2} & i \neq j \end{cases}$$

$$[I(z_U)_{xy}]_{ij} = \frac{1}{\sigma^2} \begin{cases} \sum_{k \in K} \frac{(x_i - x_k)(y_i - y_k)}{\| z_i - z_k \|^2} + \sum_{k \in U \setminus \{i\}} \frac{(x_i - x_k)(y_i - y_k)}{\| z_i - z_k \|^2} & i = j \\ -\frac{(x_i - x_j)(y_i - y_j)}{\| z_i - z_j \|^2} & i \neq j \end{cases}$$

$$[I(z_U)_{yy}]_{ij} = \frac{1}{\sigma^2} \begin{cases} \sum_{k \in K} \frac{(y_i - y_k)^2}{\| z_i - z_k \|^2} + \sum_{k \in U \setminus \{i\}} \frac{(y_i - y_k)^2}{\| z_i - z_k \|^2} & i = j \\ -\frac{(y_i - y_j)^2}{\| z_i - z_j \|^2} & i \neq j \end{cases}.$$

Next, we form the $2(N - M) \times 2(N - M)$ expected Fisher information matrix for the position parameter

$$I(z_U) = \begin{bmatrix} I(z_U)_{xx} & I(z_U)_{xy} \\ I(z_U)_{xy} & I(z_U)_{yy} \end{bmatrix}. \tag{7}$$

In accordance with previous simulation studies we now assume the sensors to be distributed within a square area $S = [0, 1] \times [0, 1]$, (e.g. [4], [6] and [7]). A natural and quite flexible class of a priori distributions on S is bivariate beta distributions, see [21, Chapter 49] for a good review. Although we could develop our methods for these general classes of distributions, we for ease of exposition assume that the positions of the $N - M$ sensors with unknown positions follow a bivariate beta distribution with density

$$f(z_i) = \frac{\Gamma(\alpha)^4}{\Gamma(2\alpha)^2} (x_i y_i)^{\alpha - 1} ((1 - x_i)(1 - y_i))^{\alpha - 1}$$

for $z_i \in S$ and $\alpha > 0$, i.e. all coordinates are mutually independent and identically distributed as well as symmetrically distributed around $1/2$. By varying α we can get a uniform distribution of positions with $\alpha = 1$, a "bell-shaped" pdf by $\alpha > 1$ and a "well-shaped" pdf by $\alpha < 1$. It is now easy to show that the following expectations exist and by symmetry we have

$$\mathbf{E}_z \frac{(x_i - x_k)^2}{\| z_i - z_k \|^2} = 1/2$$

$$\mathbf{E}_z \frac{(x_i - x_k)(y_i - y_k)}{\| z_i - z_k \|^2} = 0$$

for all $i, k \in U$. If we let X and Y denote two independent and Beta(α, α)-distributed random variables, we also have

$$\eta_k = \mathbf{E}_z \frac{(x_i - x_k)^2}{\| z_i - z_k \|^2} = \mathbf{E}_{X,Y} \left(1 + \left(\frac{Y - y_k}{X - x_k} \right)^2 \right)^{-1} \tag{8}$$

$$\zeta_k = \mathbf{E}_z \frac{(x_i - x_k)(y_i - y_k)}{\| z_i - z_k \|^2} = \mathbf{E}_{X,Y} \left(\frac{X - x_k}{Y - y_k} + \frac{Y - y_k}{X - x_k} \right)^{-1} \tag{9}$$

for all $i \in U$ and $k \in K$. The mean values in (8) and (9) are easily calculated by numerical integration. Now, let $\eta = \sum_{k \in K} \eta_k$ and $\zeta = \sum_{k \in K} \zeta_k$, then we obtain

$$\mathbf{E}_z I(z_U)_{xx} = \left(\eta + \frac{N - M}{2} \right) I - \frac{1}{2} H \tag{10}$$

$$\mathbf{E}_z I(z_U)_{xy} = \zeta I, \tag{11}$$

where I denotes the $(N - M) \times (N - M)$ identity matrix and H is the $(N - M) \times (N - M)$ matrix of all ones. As $\mathbf{E}_z I(z_U)_{yy} = \mathbf{E}_z I(z_U)_{xx}$ and $\mathbf{E}_z I(z_U)_{yx} = \mathbf{E}_z I(z_U)_{xy}$, the matrix $\mathbf{E}_z I(z_U)$ can easily be

constructed by use of (7). The expected Fisher information matrix for the a priori distribution is now derived for $\alpha > 2$ in the following way

$$
\begin{aligned}
-\mathbf{E}_z \frac{\partial^2}{\partial x_i^2} \log p(z_U) &= 2(\alpha - 1) \int_0^1 \frac{\Gamma(\alpha)^2}{\Gamma(2\alpha)} x^{\alpha-3}(1-x)^{\alpha-1} \, dx \\
&= \frac{(\alpha - 1)(\alpha - 2)}{2\alpha - 1} \\
-\mathbf{E}_z \frac{\partial^2}{\partial x_i \partial x_j} \log p(z_U) &= 0, \ i \neq j \\
-\mathbf{E}_z \frac{\partial^2}{\partial x_i \partial y_j} \log p(z_U) &= 0.
\end{aligned}
$$

Hence,

$$
I(p) = \xi \begin{bmatrix} I & O \\ O & I \end{bmatrix}, \tag{12}
$$

where $\xi = \frac{(\alpha-1)(\alpha-2)}{2\alpha-1}$ and O is the $(N-M) \times (N-M)$ zero matrix. Inserting (10), (11) and (12) into (2), we obtain

$$
\text{AMSE} \geq \frac{1}{N-M} \operatorname{tr} \left[\begin{matrix} \left(\eta + \frac{N-M}{2} + \xi\right) I - \frac{1}{2}H & \zeta I \\ \zeta I & \left(\eta + \frac{N-M}{2} + \xi\right) I - \frac{1}{2}H \end{matrix} \right]^{-1}. \tag{13}
$$

VI. PARTIAL OBSERVATIONS

Now consider the situation where every sensor only detects a distance from a subset of the remaining sensors, modelled by the probability P_o discussed in Section II-B. We call this the partial observations case as opposed to the previous full observations case. We split the partial observations case into two separate situations. First, we consider the situation where the sensors are able to broadcast distance information throughout the entire network, called full information available, see Section VI-A. Second, we consider a case where we only obtain position information from the k-order neighbours, see Section VI-B. Although the van Trees lower bound derived in Example 1 strictly speaking only applies to the full observations case, we also expect it to hold for estimators in the partial observations case, since estimation in the full observations case is expected to perform better than the partial observations case. Hence we compare the estimators in the partial observation case with van Trees' inequality for the full observations case, see Section VII.

A. Full information available

In this case the a posteriori distribution is given by

$$
\begin{aligned}
p(z_U|d) \ \propto_{z_U} \ & p(d|z_U)p(z_U) \\
= \ & \prod_{i \in U, 1 \leq j < i} \left[\left(\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(d_{ij} - \| z_i - z_j \|)^2}{2\sigma^2} \right) P_o(z_i, z_j) \right)^{o_{ij}} \right. \\
& \left. \times (1 - P_o(z_i, z_j))^{1-o_{ij}} \right] \times p(z_U).
\end{aligned} \tag{14}
$$

Define a random set $E \subset V \times V$ in the following way:

$$
e_{ij} \in E \text{ if and only if } o_{ij} = 1.
$$

The tuple $\mathcal{G} = (V, E)$ induces an undirected random graph with vertices V and edges E. For a graph $\mathcal{G} = (V, E)$, we define for each $i \in V$ the set of k-order neighbours of i as

$$
\partial_i^k = \{ j \in V \setminus \{i\} : i_1, \ldots, i_l \in V, e_{ii_1}, \ldots, e_{i_l j} \in E, 1 \leq l < k \}.
$$

Note that ∂_i^k is a random variable with values in $2^{V\setminus\{i\}}$, the set of all subsets of $V \setminus \{i\}$. It is now convenient to formulate the marginal a posteriori density, to be plugged into the ICM Algorithm IV.1, in the following way

$$
\begin{aligned}
p(z_i|z_{V\setminus\{i\}}, d) &= p\left(z_i|z_{\partial_i^1}, (d_{ij})_{j\in\partial_i^1}, z_{V\setminus(\partial_i^1\cup\{i\})}, \partial_i^1\right) \\
&\propto_{z_i} \prod_{j\in\partial_i^k}\left(\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(d_{ij}-\parallel z_i - z_j \parallel)^2}{2\sigma^2}\right)P_o(z_i, z_j)\right) \\
&\times \prod_{j\in V\setminus(\partial_i^1\cup\{i\})}(1-P_o(z_i, z_j)) \times p(z_i|z_{V\setminus\{i\}}).
\end{aligned}
$$

B. Information available from k-order neighbours, including non-local information

Inspired by the approach in Sections V and VI-A, we now suggest iteratively to maximize the following marginal a posteriori pdf, based on available local information only (i.e. information passed on from k-order neighbours)

$$
\begin{aligned}
p\left(z_i|z_{\partial_i^k}, (d_{ij})_{j\in\partial_i^k}, \partial_i^k\right) &\propto_{z_i} p\left((d_{ij})_{j\in\partial_i^1}|z_i, z_{\partial_i^k}, \partial_i^k\right)p\left(z_i, z_{\partial_i^k}, \partial_i^k\right) \\
&= \prod_{j\in\partial_i^k}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(d_{ij}-\parallel z_i - z_j \parallel)^2}{2\sigma^2}\right) \\
&\times \int_{\mathbb{R}^{2|V\setminus(\partial_i^k\cup\{i\})|}}p(z_i, z_{\partial_i^k}, z, \partial_i^k)dz \\
&= \prod_{j\in\partial_i^k}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(d_{ij}-\parallel z_i - z_j \parallel)^2}{2\sigma^2}\right) \\
&\times \int_{\mathbb{R}^{2|V\setminus(\partial_i^k\cup\{i\})|}}p(\partial_i^k|z_i, z_{\partial_i^k}, z)p(z_i, z_{\partial_i^k}, z)dz.
\end{aligned}
$$

By using the Bernoulli structure of the included edges in the random graph specified by P_o, we arrive at

$$
\begin{aligned}
&p\left(z_i|z_{\partial_i^k}, (d_{ij})_{j\in\partial_i^k}, \partial_i^k\right) \\
&\propto_{z_i} \prod_{j\in\partial_i^k}\left(\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(d_{ij}-\parallel z_i - z_j \parallel)^2}{2\sigma^2}\right)P_o(z_i, z_j)\right) \\
&\times \int_{\mathbb{R}^{2|V\setminus(\{i\}\cup\partial_i^k)|}}\prod_{j\in V\setminus(\partial_i^k\cup\{i\})}(1-P_o(z_i, z_j))p(z_i, z_{\partial_i^k}, z)dz.
\end{aligned}
\tag{15}
$$

If we assume an inhomogeneous binomial point process as a priori distribution, we get the following simplification

$$
\begin{aligned}
&p\left(z_i|z_{\partial_i^k}, (d_{ij})_{j\in\partial_i^k}, \partial_i^k\right) \\
&\propto_{z_i} \prod_{j\in\partial_i^k}\left(\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(d_{ij}-\parallel z_i - z_j \parallel)^2}{2\sigma^2}\right)P_o(z_i, z_j)\right) \\
&\times \left(\int_{\mathbb{R}^2}(1-P_o(z_i, z))p(z)dz\right)^{|V\setminus(\partial_i^k\cup\{i\})|} \times p(z_i).
\end{aligned}
\tag{16}
$$

We should here note that the integral in (16) in most cases has to be calculated by numerical integration for each and every step in the ICM algorithm. Hence, it seems worthwhile to iteratively use the following approximation in the ICM algorithm:

$$p\left(z_i | z_{\partial_i^k}, (d_{ij})_{j \in \partial_i^k}, \partial_i^k\right)$$
$$\propto_{z_i} \prod_{j \in \partial_i^k} \left(\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(d_{ij} - \| z_i - z_j \|)^2}{2\sigma^2} \right) P_o(z_i, z_j) \right) p(z_i). \tag{17}$$

This is indeed the approximation we use in our simulation studies in Section VII.

VII. SIMULATION RESULTS

In this section we test the ICM algorithm on simulated examples of sensor networks on $S = [0,1] \times [0,1]$. We let $M = 3$ and place the three sensors with known positions at $(0.2, 0.2)$, $(0.8, 0.2)$, and $(0.5, 0.8)$. The $N - M$ sensors with unknown position follow the bivariate beta a priori distribution from Example 1 with $\alpha = 5$. The pdf of this a priori distribution is shown in Figure 1. The rest of the parameters in the model are varied in each of the examples.
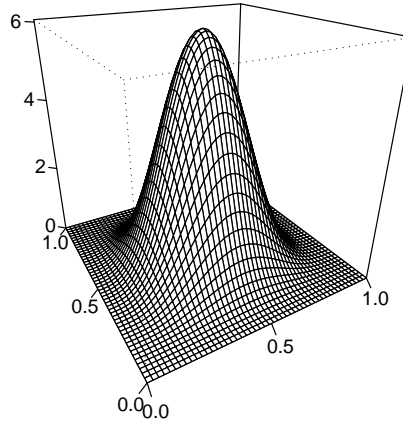

Fig. 1. Density of the bivariate beta a priori distribution with $\alpha = 5$.

For estimation the first problem we need to solve is how to obtain the initial positions $z_{M+1}^{(0)}, \ldots, z_N^{(0)}$ used as a starting point by the ICM Algorithm IV.1. These are obtained by the sum-dist algorithm [20]. In this algorithm a rough estimate of the position of sensor i is obtained by finding the shortest path from sensor i to each sensor with known position in the graph of observed distances, where "shortest" refers to the path having the minimal sum of observed distances. In practice this is done by letting each sensor with known position send out its identity, position and a path length set to zero. Each sensor i receiving such a message from sensor j adds the observed distance between sensor i and j to the path length and sends a message with this path length to its neighbours, unless it has previously received a shorter path length in which case it does nothing. Once all sensors with unknown position have received the shortest path length to each sensor with known position, we estimate the position of sensor i by maximizing (6) using the sensors with known position rather than all the other sensors and the lengths of the minimal paths instead of the observed distances. Naturally this algorithm yields only very rough estimates, but it provides a good starting point for the ICM algorithm. Furthermore, we adopt the following convergence criterion for the ICM algorithm: we say that $z_i^{(n)}$ has converged and do not update the estimated position of sensor i anymore if the distance $\|z_j^{(n)} - z_j^{(n-1)}\| < \epsilon$ for some prespecified $\epsilon > 0$ and for $j \in \{i\} \cup \partial i$. Once all estimated positions have converged, the ICM algorithm terminates. For our examples we use $\epsilon = 0.001$.

The left plot in Figure 2 shows a simulation of a sensor network, where known and unknown positions and the graph of observed distances are shown. The parameters used here are $N = 20$, $\sigma^2 = 0.004$ and $R = 0.3$ in addition to the parameters mentioned above. In the right plot in Figure 2, the estimates obtained by the ICM algorithm are shown in addition to the true positions. This illustrates that the ICM algorithm has provided a reasonable estimate of the positions for this particular example.
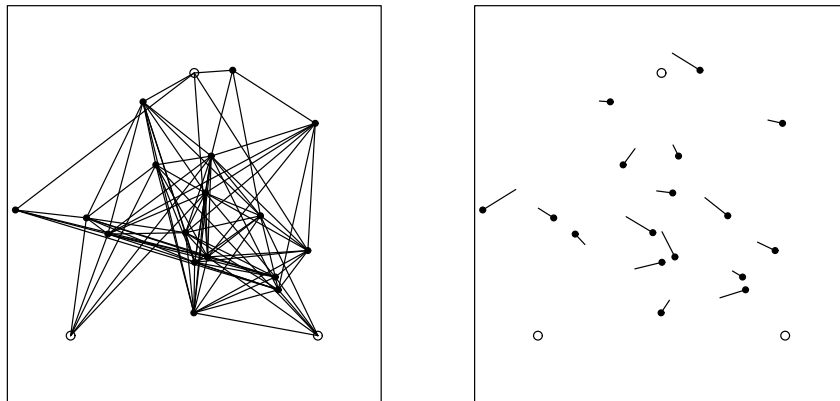

Fig. 2. Left: a simulation of a sensor network where small circles indicate known positions, dots indicate unknown positions, and lines indicate an observed distance. Right: the same simulation, where lines segments indicate the difference between the true positions and the positions estimated by the ICM algorithm.

To test how well the ICM algorithm performs in general, we compare the AMSE estimated from a number of simulations with the lower bound given by (13) while varying parameters in the model. In the first example, we fix $N = 20$, $\sigma^2 = 0.004$, and the parameters mentioned at the beginning of this section, and let R take the values $0.05, 0.10, \ldots, 0.50$. Furthermore, we consider both the case with first order neighbourhood and the case with second order neighbourhood. For each value of R and both neighbourhoods we estimate the AMSE as the average of the average square error calculated for 500 simulations. This is shown in Figure 3, together with confidence bands estimated as the average plus/minus two times the standard deviation. The lower bound for the AMSE is also included in the plot, and since the lower bound is only available in the case of full observations (i.e. corresponding $R = \infty$), it is constant in this plot. In the plot we can see that for low values of R the error is much larger than the lower bound for both cases of neighbourhood orders. This is to be expected since for low R there are a lot fewer observations than in the case of full observations. On the other hand, for high values of R the AMSE for both cases gets very close to the bound, so this indicates that the ICM algorithm performs well in such cases.

If we compare the two orders of neighbourhoods, the only significant difference occurs at $R = 0.01$, and surprisingly the second order neighborhood yields the worst result. One possible explanation for this is that for $R = 0.01$ we often get chains of pairs of sensors observing each other; for example, consider the three sensors i_1, i_2, i_3 where i_2 observes both i_1 and i_3, but i_1 and i_3 do not observe each other. In this example the estimated positions of sensors i_1 and i_3 will tend to be located far away from each other when we employ second order information, since the sensors have not observed each other. However, since the probability of observing sensors is low for the values of R considered here, it is more likely that the sensors do not observe each other simply by chance. Thus estimation using the second order neighbourhood often yields a worse estimate than the first order neighbourhood, which treats the positions of i_1 and i_3 as independent given the position of i_2. For $R = 0.015, \ldots, 0.050$, there seems to be a tendency that the second order information slightly improves the estimate compared to the first order information, but this tendency is not significant. In short, there is nothing to be gained from including the second order neighbours into the ICM algorithm for the cases considered here.
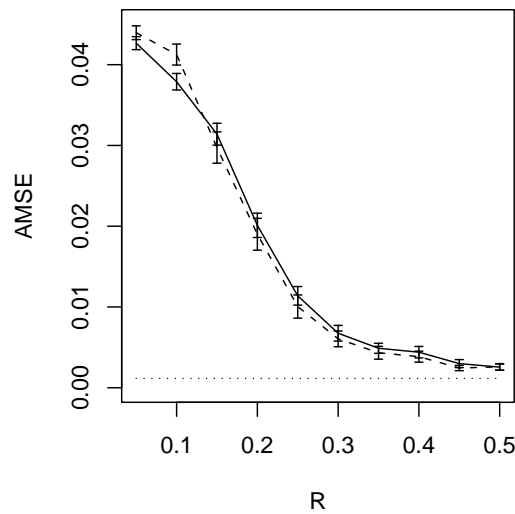
Fig. 3. Plot of the AMSE for different values of R using first order neighbourhood (solid line) and second order neighbourhood (dashed line) with confidence bands (bars) and theoretical lower bound (dotted line).

In the left plot in Figure 4, we consider the case of full observations, since the lower bound yields a more useful comparison in this case. Here, we fix $\sigma^2 = 0.004$ and let N take the values $5, 10, \ldots, 50$. In the plot both the estimated AMSE and the lower bound decrease as N increases. This illustrates that the position estimates are substantially improved by adding more sensors, even though the positions of the new sensors are unknown, since the number of observations are increased. Furthermore, the AMSE is close to the lower bound for all the values of N included in the plot, so this indicates that the ICM algorithm performs well here.

In the right plot in Figure 4, we consider the case of full observations, fix $N = 20$ and let $\sigma^2 = 0.001, 0.002, \ldots, 0.010$. Both the estimated AMSE and the lower bound increase as σ^2 increases. The estimated AMSE grows a bit faster than the lower bound, but nevertheless it stays fairly close to the lower bound for the values considered.
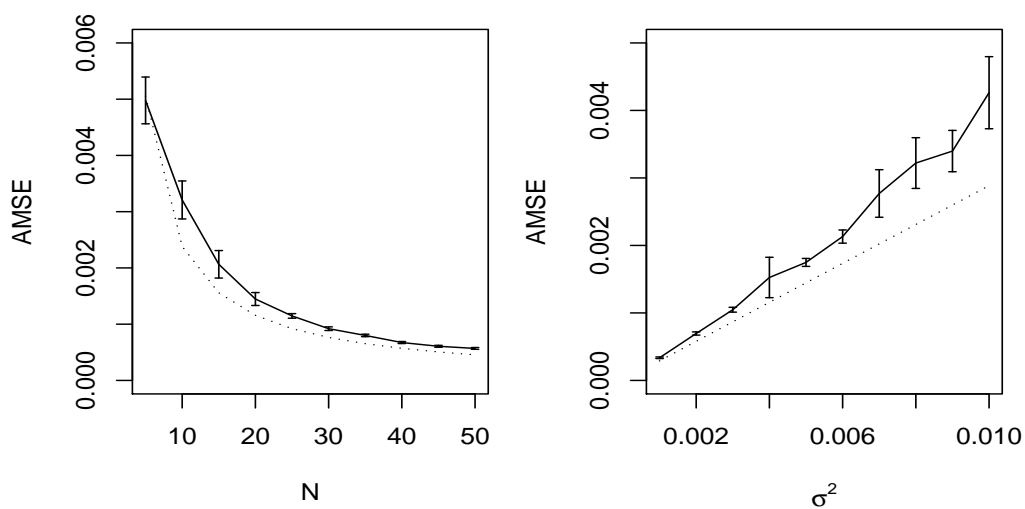

Fig. 4. Left: plot of the AMSE for different values of N (solid line) with confidence bands (bars) and theoretical lower bound (dotted line). Right: as left, but for different values of σ^2.

ACKNOWLEDGEMENTS

REFERENCES

[1] V. Ramadurai and M. Sichitiu, "Localization in wireless sensor networks: A probabilistic approach," in *Proceedings of OPNETWORK 2003*, Washington, DC, 2003.

[2] ——, "Simulation-based analysis of a localization algorithm for wireless ad-hoc sensor networks," in *Proceedings of the International Conference on Wireless Networks*, Las Vegas, NV, 2003.

[3] N. Patwari, J. N. Ash, S. Kyperountas, A. Hero III, R. L. Moses, and N. S. Correal, "Locating the nodes," *IEEE Signal Processing Magazine*, vol. 22, July 2005, pp. 54–69, 2005.

[4] R. L. Moses, D. Krishnamurthy, and R. Patterson, "A self-localization method for wireless sensor networks," *EURASIP Journal of Applied Signal Processing*, vol. 4, pp. 348–358, 2003.

[5] B. V. A.A. Kannan, G. Mao, "Simulated annealing based wireless sensor network localization with flip ambiguity mitigation," in *IEEE 63rd Vehicular Technology Conference (VTC)*, vol. 2, Spring 2006, pp. 1022–1026.

[6] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Transactions on Sensor Networks*, vol. 2, pp. 188–220, 2006.

[7] G. Destino and G. T. F. de Abreu, "Sensor localization from WLS optimization with closed-form gradient and Hessian," in *Proceedings of IEEE Globecom*, 2006.

[8] H. L. van Trees, *Detection, Estimation, and Modulation Theory, Part 1.* John Wiley and Sons, New York, NY, 1968.

[9] R. D. Gill and B. Y. Levit, "Applications of the van Trees inequality: a Bayesian Cramér-Rao bound," *Bernoulli*, vol. 1, pp. 59–79, 1995.

[10] N. Patwari, "Location estimation in sensor networks," Ph.D. dissertation, University of Michigan, 2005, available at http://www.eecs.umich.edu/~hero/Preprints/patwari_thesis_final.pdf.

[11] Q. Shi, S. Kyperountas, N. S. Correal, and F. Niu, "Performance analysis of relative location estimation for multihop wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 830–838, 2005.

[12] J. E. Besag, "Discussion of a paper by P. Switzer," *Bulletin of the International Statistical Institute*, vol. 50, pp. 422–425, 1983.

[13] ——, "On the statisical analysis of dirty pictures," *Journal of the Royal Statistical Society B*, vol. 48, pp. 259–302, 1986.

[14] J. Kitler and J. Föglein, "Contextual classification of multispectral pixel data," *Image and Vision Computing*, vol. 2, pp. 13–29, 1984.

[15] A. T. Ihler, J. W. Fisher III, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 809–819, 2005.

[16] R. L. Moses, D. Krishnamurthy, and R. Patterson, "Self-calibration of sensor networks," in *Proceedings of SPIE, vol. 4743, Unattended Ground Sensor Technologies and Applications IV*, 2002, pp. 108–119.

[17] A. Gelman, J. Carlin, H. Stern, and D. Rubin, *Bayesian Data Analysis*, 2nd ed. Chapman and Hall/CRC,Boca Raton, FL, 2003.

[18] J. Berger, *Statistical Decision Theory and Bayesian Analysis*, 2nd ed. Springer Verlag, New York, NY, 2003.

[19] D. Marinakis and G. Dudek, "Probabilistic self-localization for sensor networks," in *Proceedings of the AAAI National Conference on Artificial Intelligence*, Boston, Massachusetts, July 2006.

[20] A. Savvides, H. Park, and M. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA), Atlanta, GA*, 2002, pp. 112–121.

[21] S. Kotz, N. Balakrishnan, and N. Johnson, *Continuous Multivariate Distributions*, 2nd ed. John Wiley and Sons, New York, NY, 2000.