

Distributed Target Tracking Using Signal Strength Measurements by a Wireless Sensor Network

Anand Oka, *Member, IEEE*, and Lutz Lampe, *Senior Member, IEEE*

Abstract—Wireless Sensor Networks are well suited for tracking targets carrying RFID tags in indoor environments. Tracking based on the received signal strength indication (RSSI) is by far the cheapest and simplest option, but suffers from secular biases due to effects of multi-path, occlusions and decalibration, as well as large unbiased errors due to measurement noise. We propose a novel algorithm that solves these problems in a distributed, scalable and power-efficient manner. Firstly, our proposal includes a tandem incremental estimator that learns and tracks the radio environment of the network, and provides this knowledge for the use of the tracking algorithm, which eliminates the secular biases due to radio occlusions etc. Secondly, we reduce the unbiased tracking error by exploiting the co-dependencies in the motion of several targets (as in crowds or herds) via a fully distributed and tractable particle filter. We thereby extract a significant “diversity gain” while still allowing the network to scale seamlessly to a large tracking area. In particular, we avoid the pitfalls of network congestion and severely shortened battery lifetimes that plague procedures based on the joint multi-target probability density.

Index Terms—Wireless Sensor Networks, Distributed Tracking, RSSI, Radio Environment Estimation, Particle Filter

I. INTRODUCTION

The ability to remotely locate and track mobile targets is crucial in many applications like security, access control, and robotics [1, 2]. Global Positioning Systems (GPS) tend to be relatively costly, and do not give adequate precision inside buildings, mines etc. A more accurate and inexpensive alternative is a *Wireless Sensor Network* (WSN) [3], an *ad hoc* network of battery powered *motes* having modest capabilities of sensing, computation and multi-hop wireless communication. The *sensor* in each mote is a radio receiver that detects radio signals from cheap unobtrusive radio frequency identification (RFID) tags mounted on the targets to be tracked. While such a WSN can be set up quickly and cheaply, its computation and communication resources need to be used frugally due to the severe power constraints on the motes.

A WSN can achieve localization of the targets by interpreting several metrics of the tag transmissions such as the time of arrival (TOA), time difference of arrival (TDOA), angle of arrival (AOA), multi-path signature (MPS), and received signal strength indication (RSSI) [1, 2]. RSSI based tracking has

by far the lowest cost since it does not require multi-antenna reception, accurate clocks, duplex communication with the tag, or expensive signal processing by the motes. However, if the operating environment has significant occlusions, shadowing or multi-path propagation, it gives large biases in the position estimates *which cannot be eliminated even with an abundance of observation data*. The biasing effect also occurs if the transmit power from the tags or the gains of the antennae lose calibration. Furthermore, even the unbiased component of the tracking error can be large due to the noisy nature of RSSI measurements.

We present an advanced RSSI based tracking technique that significantly ameliorates these problems by incorporating two novel features: (1) It recursively *learns and compensates* for the radio environment of the tags and the motes based on maximum likelihood principles, thereby eliminating the secular biases in the tracking errors. (2) It reduces the unbiased tracking error by exploiting the dependencies in the motion of targets that are moving in crowds or herds, thus achieving a *diversity gain*. For this it uses a particle filtering approach [4–9], which is known to be efficient in representing the highly multi-modal a-posteriori probability density of the target states. However, unlike [4–9], it implements a *bank of distributed particle filters* on the *marginal* densities of the targets, that interact only through conditional marginal expectations [10]. This enables it to avoid the intractable manipulation of the joint multi-target probability density (JMPD). Furthermore, each filter operates on data aggregated solely from a local neighborhood of the tag. In summary, we present a simple, inexpensive, distributed and scalable tracking technique that works robustly in indoor environments like hospitals or factories.

Outline: After defining the system model in Section II, we present the tracking sub-algorithm in Section III, and the radio-environment estimation sub-algorithm in Section IV. Section V discusses the practical aspects of scalable implementation. In Section VI we present simulation results. We use the following conventions: x^T denotes transpose of x . $\{\Upsilon_b^a : a = 1, 2, \dots, A; b = 1, 2, \dots, B\}$ denotes a *set* of objects formed by letting the indices run over the prescribed ranges. Where no confusion can arise we denote it by $\{\Upsilon_b^a\}$ or $\{\Upsilon\}$. *Partially* enumerated indices always specify a *subset*. E.g., $\{\Upsilon_b^a : b = 1, 2, \dots, B\}$ will denote a subset of B elements, where the superscript index is pinned to a .

II. SYSTEM MODEL AND OVERVIEW

Suppose we wish to track in D dimensions ($D \in \{1, 2, 3\}$) in some bounded tracking region, using a WSN. Within the tracking area, N motes are placed at more or less uniformly

This work was supported in part by the National Sciences and Engineering Research Council of Canada (STPGP 364962-08, PGS-D3-348380-2007) and the British Columbia Innovation Council (BCII-2007-IS-UBC-008). It was presented in part at the IEEE Intl. Conf. on Communications, 2009.

The first author was with the Department of Electrical and Computer Engineering, University of British Columbia, Canada, and is currently with Research in Motion, Canada. (e-mail: anando@ece.ubc.ca). The second author is with the Department of Electrical and Computer Engineering, University of British Columbia, Canada. (e-mail: Lampe@ece.ubc.ca).

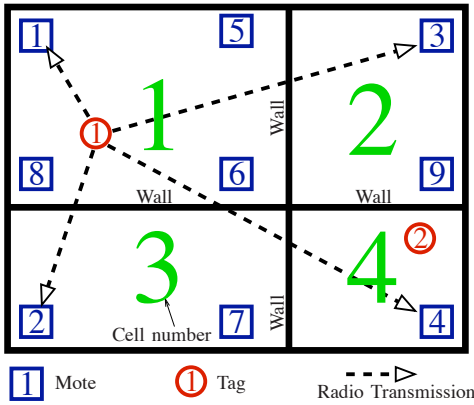


Fig. 1. RSSI based target tracking in an indoor environment in a plane ($D = 2$ dimensions), using a WSN. There are nine installed motes (of which only $N = 4$ are shown to be active in the figure), $M = 2$ targets and $L = 4$ cells. The cells are defined to be the rooms of the building.

spread-out but arbitrary locations, with position vectors $r_n \in \mathbb{R}^D$, $n = 1, 2, \dots, N$. All these position vectors are assumed to be known to each mote. We divide the tracking region into L cells which are chosen such that there is no appreciable spatial variability in the radio transmission characteristic from any points within a cell to a mote, apart from the standard distance dependent path-loss due to geometric spreading (this will be clarified shortly). In many cases the cells may be chosen to be the rooms in the building. The knowledge of the boundaries of the cells is assumed to be available to all the motes. For example, consider the system setup illustrated in Figure 1, which we also use in some simulations in Section VI. The tracking area is one floor of a building with four rooms of dissimilar sizes, within which the mobile targets are confined. The number of targets M is assumed fixed and known to the tracking application. Each target carries an active RFID tag. The position of target number $m \in \{1, 2, \dots, M\}$ at discrete time index $t \in \{1, 2, \dots\}$ is denoted by $X_m^t \in \mathbb{R}^D$. $\{X_m^t\}$ is viewed as a random process governed by a known *maneuver model*, which will be described in Section II-A. Let $Y_{m,n}^t \in \mathbb{R}$ denote the RSSI in decibel-milliwatt (dBmW) of the signal from tag m received at mote n at time t . Then $\{Y_{m,n}^t\} | \{X_m^t\}$ is governed by a known *observation model*, which will be described in Section II-B. Let the power of the signal transmitted by tag m at time t be Φ_m^t dBmW, and let the *gain* of the path from cell $l \in \{1, 2, \dots, L\}$ to mote n at time t be $\Gamma_{l,n}^t$ dB. $\Gamma_{l,n}^t$ does not include the propagation path loss due to geometric spreading. It may be helpful to think of it as the (negative of the) loss due to attenuation by intervening objects like walls. The quantities $\{\Phi_m^t\}$ and $\{\Gamma_{l,n}^t\}$ are viewed as unknown but deterministic *quasi-static* parameters. For accurate tracking, the true values of these parameters need to be estimated and used. Let $\hat{\Phi}_m^t$ and $\hat{\Gamma}_{l,n}^t$ denote the estimates of Φ_m^t and $\Gamma_{l,n}^t$ respectively.

The aim of the tracking system is to find an estimate \hat{X}_m^t for the position X_m^t of each tag $m = 1, 2, \dots, M$ at time t , based on all the observations up to and including that epoch, namely $\{Y_{m,n}^\tau : \text{all } m, n; \tau = 1, 2, \dots, t\}$. Ideally, the estimates $\{\hat{X}_m^t : \text{all } m\}$ should be optimal in the sense of the

normalized total mean squared tracking error at time t ,

$$(\Delta^2)^t = \frac{1}{M} \sum_{m=1}^M \mathbb{E} \left[\|\hat{X}_m^t - X_m^t\|^2 \right]. \quad (1)$$

The over-all tracking system consists of a target tracking algorithm based on particle filtering, and a stochastic incremental radio environment estimation algorithm based on Maximum Likelihood (ML) principles. This hybrid solution compares favorably to a fully Bayesian approach that appends all unknown parameters into an expanded ‘‘meta-state’’.

A. Maneuver Model

Let $V_m^t \in \mathbb{R}^D$ denote the velocity of target m at time t . Define the *sub-state* corresponding to tag m at time t as $Z_m^t \doteq [X_{m,1}^t V_{m,1}^t X_{m,2}^t V_{m,2}^t \dots X_{m,D}^t V_{m,D}^t]^T$, and the *total-state* of all the targets as $Z^t \doteq [Z_1^t Z_2^t \dots Z_M^t]^T$. In this paper we will use a class of maneuver models that is popular in the literature of target tracking [4, 11–13], where the total-state is assumed to be governed by a first-order Markov Chain (MC)

$$Z^{t+1} \doteq A Z^t + B U^t + \epsilon_{\text{depen}} g(BC Z^t). \quad (2)$$

In this equation the term $A Z^t$ captures the inertia of each individual target and the term $B U^t$ represents the effect of the individual independent accelerations (‘maneuvers’) of the targets. Hence $A \in \mathbb{R}^{2MD \times 2MD}$ is a block-diagonal matrix with the sub-matrix $\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$ repeated along the diagonal, and $B \in \mathbb{R}^{2MD \times MD}$ is a block-diagonal matrix with the sub-matrix $[T^2/2, T]^T$ repeated along the diagonal, with T being the time interval between successive maneuvers. The acceleration $U^t \in \mathbb{R}^{MD}$ is assumed to be independently and identically distributed for all t according to $\mathcal{N}(0, \sigma_U^2 I_{MD})$, and is assumed to be active throughout the real-time interval $[(t-1)T, tT]$ (i.e. the physical acceleration is piecewise constant). The term $g(BC Z^t)$ captures the effect of causal interactions in-between the targets, with the scalar $\epsilon_{\text{depen}} \geq 0$ controlling the relative strength of these interactions. C is some suitably chosen *interaction feedback* matrix, and $g(\cdot)$ is some suitable *compressive* non-linearity applied component-wise to its vector argument.

Clearly, if we let $\epsilon_{\text{depen}} = 0$, we obtain a model where the tags move independently of each other, with tag m accelerating independently in each dimension according to component numbers $(m-1)D + 1, \dots, (m-1)D + D$ respectively of U^t . On the other hand if $\epsilon_{\text{depen}} > 0$, and $C, g(\cdot)$ are chosen to mimic certain heuristic rules known as Reynolds’ *flocking rules* [13], we can model the motion of a variety of herds and swarms, and even deterministically constrained motion like ‘convoy’ or ‘leapfrog’ [4]. In general, we can improve the tracking accuracy by exploiting the mutual information among such co-dependently moving targets.

B. Observation Model

A single transmission from a tag consists of a packet of bits that uniquely identify it relative to other tags. An incorrect decoding of these bits can be reliably detected using a Cyclic Redundancy Check (CRC) code, and such packets can be deemed *invalid* and rejected by the algorithm. For *valid* packets, we can always correctly associate the RSSI

with the source of transmission, thus avoiding the problem of *data association* [12]. For simplicity we will now assume that the interval between the periodic transmissions from the tags is also T seconds, and that these transmissions are synchronous.¹ A commonly used model for the strength of the radio transmissions in the far-field of the transmitting tag is the *Friis model* [14]. In the tracking scenario [12] this model implies that the observations made by the motes at real time tT are given (in dBmW) by

$$Y_{m,n}^t = \Phi_m^t + \Gamma_{f(X_m^t),n}^t - \rho 10 \log_{10} \|X_m^t - r_n\| + W_{m,n}^t. \quad (3)$$

Here ρ is a known path loss exponent ($\rho = 2.0$ in free-space) and $W_{m,n}^t$ is a process of additive perturbations in the RSSI measurements (expressed in dB) with i.i.d. components for all t, m, n distributed according to $\mathcal{N}(0, \sigma_W^2)$. These perturbations can be attributed to several causes like the data noise in the signal, fast fading and other imperfections of the RSSI measurement circuitry, and the thermal noise in the front-end receiver of the motes, c.f. e.g. [1, 2, 4]. The function $f(\cdot)$ is a *position quantizer* that takes a position vector in the tracking area and returns its cell index:

$$f: \mathbb{R}^D \rightarrow \{1, 2, \dots, L\}, \quad x \mapsto f(x). \quad (4)$$

III. TRACKING WITH A SAMPLING IMPORTANCE-RESAMPLING PARTICLE FILTER

In this section we will use the current parameter estimates, namely $\{\hat{\Phi}_m^t: \text{all } m\}$ and $\{\hat{\Gamma}_{l,n}^t: \text{all } l, n\}$, as if they are the true parameters, and perform the filter update of the propagated probability density function (p.d.f.) of the total system state. The optimal estimate of the state in the sense of minimum mean squared error (MMSE) is $\hat{Z}^t = \int_{\mathbb{R}^{2MD}} z^t q(z^t) dz^t$, where $q^t(z^t)$ is the a-posteriori density of the total-state at time t conditioned on all the observations up to and including sampling time index t , given by the Bayesian recursion

$$q^t(z^t) \propto p(y^t|z^t) \int_{\mathbb{R}^{2MD}} p(z^t|z^{t-1}) q^{t-1}(z^{t-1}) dz^{t-1}. \quad (5)$$

This recursion has no closed form solution, and in fact becomes intractable as M increases. Hence, for achieving a tractable algorithm, we must use some type of approximation. A particle filter is known to give the best approximation of the highly multi-modal $q^t(z^t)$ when tracking with RSSI [4–6]. In particular, we will use a bank of particle filters, one each for the sub-state of the corresponding target. Let $\mathbb{R}^{2D} \ni \zeta_{m,\pi}^t, \pi = 1, 2, \dots, \Pi$ be the Π particles representing $q_m^t(z_m^t)$, the marginal a-posteriori distribution at time t of the sub-state of target m . The particle representation is made via the identity $q_m^t(z_m^t) \approx \frac{1}{\Pi} \sum_{\pi=1}^{\Pi} \delta_{\zeta_{m,\pi}^t}(z_m^t)$, where $\delta_{\zeta}(z)$ is the Dirac-delta function centered at ζ , that satisfies the sifting property $\int h(z) \delta_{\zeta}(z) dz = h(\zeta)$. Then the estimate of the sub-state of target m is given by $\hat{Z}_m^t = \int_{\mathbb{R}^{2D}} z_m^t q_m(z_m^t) dz_m^t \approx \frac{1}{\Pi} \sum_{\pi=1}^{\Pi} \zeta_{m,\pi}^t$.

For a given target m , the particles $\{\zeta_{m,\pi}^t, \pi = 1, \dots, \Pi\}$ are calculated using the previous particles $\{\zeta_{m,\pi}^{t-1}, \pi = 1, 2, \dots, \Pi\}$ and the a-posteriori *marginal expectations* of all the other

Algorithm 1 SIR particle filter-bank

- 1: For each $m = 1, 2, \dots, M$, do steps 2 through 6.
- 2: Initialize sub-state particles $\{\zeta_{m,\pi}^{t-1}, \pi = 1, 2, \dots, \Pi\}$ from a *uniform distribution* over the (finite) tracking region. Then for each $t = 1, 2, \dots$, execute steps 3 through 6.
- 3: From the Π sub-state particles $\{\zeta_{m,\pi}^{t-1}, \pi = 1, 2, \dots, \Pi\}$ create Π full-state particles $\{\xi_{m,\pi}^{t-1}, \pi = 1, 2, \dots, \Pi\}$ as follows:

$$\xi_{m,\pi}^{t-1T} = [\hat{Z}_1^{t-1T} \dots \hat{Z}_{m-1}^{t-1T} \zeta_{m,\pi}^{t-1T} \hat{Z}_{m+1}^{t-1T} \dots \hat{Z}_M^{t-1T}].$$

- 4: Propagate according to the maneuver model:

$$\xi_{m,\pi}^t \leftarrow A \xi_{m,\pi}^{t-1} + B \hat{U}^t + \epsilon_{depen} g(B F \xi_{m,\pi}^{t-1}). \quad (6)$$

In doing so we generate an innovation process \hat{U}^t having the same statistical properties as U^t . (This constitutes sampling from the conditionally Gaussian proposal $p(z^t|z^{t-1})$, also known as the ‘kinematic prior’ [7].)

- 5: Importance-Resample based on the observation likelihood. That is, calculate the *weight* of particle π as

$$\begin{aligned} w_{m,\pi}^t &\doteq p(\{Y_{m,n}^t: \text{all } n\} | \xi_{m,\pi}^t) \\ &= \prod_{n=1}^N \exp \left\{ \frac{-1}{2\sigma_W^2} (Y_{m,n}^t - \hat{\Phi}_m^t - \hat{\Gamma}_{f(\xi_{m,\pi}^t),n}^t)^2 \right. \\ &\quad \left. + \rho 10 \log \|\xi_{m,\pi}^t - r_n\|^2 \right\} \end{aligned} \quad (7)$$

Normalize these weights so that they sum to one. Then draw Π numbers from $1, \dots, \Pi$ (with replacement) using the distribution $\{w_{m,\pi}^t: \pi = 1, 2, \dots, \Pi\}$. Use these as indices to select Π particles from the pool $\{\xi_{m,\pi}^t: \pi = 1, 2, \dots, \Pi\}$. Call them $\{\chi_{m,\pi}^t: \pi = 1, 2, \dots, \Pi\}$.

- 6: Project $\{\chi_{m,\pi}^t: \pi = 1, 2, \dots, \Pi\}$ to the m^{th} sub-state, by discarding the components corresponding to $m' \neq m$. Call them $\{\zeta_{m,\pi}^t: \pi = 1, 2, \dots, \Pi\}$.
-

targets at time $t-1$, namely $\{\hat{Z}_{m'}^{t-1}: m' = 1, 2, \dots, M, m' \neq m\}$. The calculation is done by using a Sampling-Importance-Resampling (SIR) particle filter [15], whose pseudo-code is given in Algorithm 1. While the SIR particle filter is well-known, our algorithm involves certain important modifications that allow the consistent integration of the information available from the other cooperating sub-state filters. This involves uplifting particles to the full state-space and merging the information about the other targets (steps 2 and 3), before performing the SIR update (step 4 and 5). Subsequently we project the particles back to the marginal state-space (step 6).

It is known [15] that while the re-sampling operation eliminates the need to explicitly maintain particle weights and hence avoids the problem of *degeneracy*, it can in turn lead to *particle impoverishment*. One can use more elaborate proposals to combat this problem, and the ‘optimal’ proposal in this sense is $p(z^t|z^{t-1}, y^t)$ which minimizes the variance of the weights [6]. Unfortunately, it cannot be written in a simple tractable closed form and some kind of approximation needs to be calculated. Comparing the complexity of drawing fewer particles from a complex prior vs the complexity of drawing somewhat more particles from a simple conditional normal, we concluded that the latter was a better option. Moreover

¹These assumptions are not critical and can be relaxed in practice.

we found that impoverishment does not seem to be a very serious problem under the choice of $p(z^t|z^{t-1})$, and can be easily side-stepped by the simple stratagem of ensuring that the particle innovation process $\{\hat{U}^t\}$ has a significant power by design. That is, even though $\{\hat{U}^t\}$ is deemed to be statistically identical to the true innovation process $\{U^t\}$, we choose its power to be $\max(\sigma_{thresh}^2, \sigma_U^2)$. By using a σ_{thresh}^2 as large as possible while still being compatible with the maneuver interval T and the end application, we ensure that we do not suffer from particle impoverishment even if σ_U^2 is itself small.

IV. STOCHASTIC INCREMENTAL ESTIMATOR OF THE RADIO ENVIRONMENT

Now we consider the problem of incrementally estimating the radio environment, utilizing the observations made by the motes as well as the results of the tracking algorithm described in Section III. Unlike [16] we use a mixed ML-Bayesian approach, i.e. while the target tracking is done with a Bayesian particle filter, the radio environment estimation is done via ML principles. This is because the radio environment parameters are *quasi-static*, almost nothing is known about their priors or dynamics, and a fully Bayesian implementation is not amenable to a distributed power efficient implementation.

Recall that $q_m^t(z_m^t), m = 1, 2, \dots, M$ are the a-posteriori marginal distributions of all the tag sub-states at time t , and these are approximately calculated by the bank of particle filters and presented in the form of particles $\{\zeta_{m,\pi}^t : \pi = 1, 2, \dots, \Pi\}, m = 1, 2, \dots, M$. As far as the estimator is concerned the joint distribution of the tags is *assumed* to be the product of these marginals, hence the sub-states of the tags are assumed to be *independent* of each other and distributed according to $q_m^t(z_m^t), m = 1, 2, \dots, M$ respectively. (This is a commonly used strategy for combating the exponential complexity of working with joint distributions. For example, the same idea is used in the sum-product decoding of *cyclic* factor graphs like Low density Parity Check codes, and in the Mean-Field approximation in statistical physics.) Similarly, we know the current estimate of the parameters, namely $\hat{\Phi}_m^t$ and $\hat{\Gamma}_{l,n}^t$, and we have access to the current observations $\{Y_{m,n}^t : \text{all } m, n\}$. Based on this data, and the knowledge of the observation model, we want to calculate the updated estimates of the parameters, namely $\hat{\Phi}_m^{t+1}$ and $\hat{\Gamma}_{l,n}^{t+1}$.

For conciseness, let us stack the parameters $\{\hat{\Phi}_m^t : \text{all } m\}$ and $\{\hat{\Gamma}_{l,n}^t : \text{all } l, n\}$ into a single vector parameter $\hat{\Upsilon}^t$ using a stacking function $f_{stack} : \mathbb{R}^M \times \mathbb{R}^{L \times N} \rightarrow \mathbb{R}^{M+LN}$. Now consider a stochastic incremental ‘partial-M step’ Expectation Maximization (EM) algorithm [17, 18] for updating $\hat{\Upsilon}^t$,

$$\hat{\Upsilon}^{t+1} = \hat{\Upsilon}^t + \epsilon F^{-1} S(\hat{\Upsilon}^t; \{Y_{m,n}^t : \text{all } m, n\}). \quad (9)$$

Here $S(\hat{\Upsilon}^t; \{Y_{m,n}^t : \text{all } m, n\})$ denotes the *score* of the log-likelihood of the observations $\{Y_{l,n}^t : \text{all } m, n\}$ viewed as a function of the parameter $\hat{\Upsilon}^t$. That is, letting $\lambda(\hat{\Upsilon}^t; \{Y_{m,n}^t : \text{all } m, n\}) \doteq \log(p(\{Y_{m,n}^t : \text{all } m, n\}|\hat{\Upsilon}^t))$,

$$S(\hat{\Upsilon}^t; \{Y_{l,n}^t : \text{all } m, n\}) \doteq \frac{\partial \lambda(\hat{\Upsilon}^t; \{Y_{m,n}^t : \text{all } m, n\})}{\partial \Upsilon}. \quad (10)$$

The expression for the score has been evaluated in Appendix A, where we also indicate how it can be calculated in practice by using the particles provided by the tracking algorithm. The scalar ϵ is a step size that controls the trade-off between the parameter tracking speed and steady-state error variance, and F^{-1} is a suitably chosen positive definite pre-scaling matrix for the score. Note that (9) is a stochastic recursion since the score $S(\hat{\Upsilon}^t; \{Y_{m,n}^t : \text{all } m, n\})$ is a random vector. It is known that the score, when evaluated at the true parameter value governing the observations, yields a random vector of *zero mean* [19]. Hence it follows that the *averaged gradient* [20] version of recursion (9) respects the true parameter as a fixed point. Its stability is determined by the covariance matrix of the score, i.e. the Fisher information F_Υ of the parameter Υ , which can be equivalently written as

$$F_\Upsilon = -\mathbb{E} \left[\frac{\partial^2 \lambda(\Upsilon; \{Y_{m,n}^t\})}{\partial \Upsilon^2} \right]. \quad (11)$$

Provided $F_\Upsilon > 0$, and we start from a suitable initial estimate $\hat{\Upsilon}^0$, recursion (9) converges to the true (quasi-static) parameter and stays in its vicinity with high probability [17, 18].

An exact closed form derivation of the Fisher information in the present scenario seems infeasible, hence in Appendix A we calculate an optimistic approximation

$$F_\Upsilon \lesssim \hat{F}_\Upsilon = \frac{1}{\sigma_W^2} \begin{pmatrix} NI_M & \frac{1}{L} \mathbf{1}_{M \times LN} \\ \frac{1}{L} \mathbf{1}_{LN \times M} & \frac{M}{L} I_{LN} \end{pmatrix}, \quad (12)$$

where I_M denotes an $M \times M$ identity matrix and $\mathbf{1}_{M \times LN}$ denotes an $M \times LN$ matrix whose entries are all ones. Closer inspection of this matrix reveals that it is rank deficient by one. That is, it is a square matrix of size $M + LN$ but has rank $M + LN - 1$. Hence there is one degree of ambiguity regarding the parameter, given the observations. The physical reason for this is easily understood by noting that if we increase all the transmit powers $\{\Phi\}$ by an arbitrary amount and reduce all the gains $\{\Gamma\}$ by an equal amount, we will get exactly the same set of observations from the model in equation (3). In other words, the observations cannot uniquely determine the model. This also implies that a gradient ascent algorithm (with or without pre-scaling), where the gradient is averaged w.r.t. all sources of stochasticity, is *not asymptotically stable* at the true parameter even though it respects it as a fixed point. As a result, the underlying stochastic algorithm will also be unstable.

Fortunately, this problem is easily solved by fixing at least one parameter from the set Υ and postulating that it does not need to be re-estimated. This reduces the size of the Fisher information matrix, which is now some principal submatrix of the above matrix and is therefore guaranteed to have full rank, thus ensuring asymptotic stability. For example one could postulate that one tag has been very accurately calibrated so that its transmit power is perfectly known. Another good choice of the parameters that can be fixed is the following:

$$\hat{\Gamma}_{f(r_n),n}^t = \Gamma_{f(r_n),n}^t = 0, \quad n = 1, 2, \dots, N, \forall t \quad (13)$$

That is, we assume that there is no attenuation (excluding geometric spreading loss) of the radio transmission from a tag to a mote whenever the tag is in the same cell as the mote.

Whatever the choice of the parameters to be fixed, let $\mu(\hat{F}_\Upsilon)$ denote the corresponding principal sub-matrix of \hat{F}_Υ . Let $COV(\hat{\Upsilon}^t - \Upsilon) = \mathbb{E}[(\hat{\Upsilon}^t - \Upsilon)(\hat{\Upsilon}^t - \Upsilon)^T]$ denote the expected covariance matrix of the error in estimating a constant truth parameter Υ . Then, in steady state and under the simplifying assumptions of Appendix A, we have an optimistic approximation of the Cramer-Rao lower bound (CRLB) [19]

$$\mu(COV(\hat{\Upsilon}^t - \Upsilon)) \geq \frac{\epsilon}{2 - \epsilon} \mu(\hat{F}_\Upsilon)^{-1}. \quad (14)$$

In the above expression, the factor $\frac{\epsilon}{2 - \epsilon}$ accounts for the time-constant (data-window) of the incremental estimation algorithm. Note that the bound (14) is not necessarily achievable since we are using an optimistic approximation of F_Υ . In recursion (9), if F is chosen to be an identity matrix, the algorithm performs a regular gradient ascent. On the other hand, if F is chosen to be F_Υ , the algorithm performs a natural gradient ascent which has better stability and acquisition properties. Furthermore, natural gradient ascent of the log-likelihood asymptotically achieves the minimal possible steady state estimation variance specified by the CRLB [18].

A. Initialization in Tough Radio Environments

The recursive estimator of equation (9) will blindly converge to the correct solution from the initial value $\hat{\Upsilon}^0 = 0$ provided the radio environment is such that even without accounting for the true parameter value Υ the estimated positions of the targets fall in the *correct cells*. (This is analogous to how a decision directed LMS equalizer will converge blindly provided the “eye” is not closed.) However in a tough radio environment, where the above condition is not satisfied, it is likely that the tandem operation of the estimator and filter will result in a false lock, in which case the corresponding position estimates will also remain highly biased.

This problem can be solved by a one-time coarse calibration of the system during the first set-up. A few targets (humans or mobile robots) are equipped with RFID tags and made to move to several predetermined locations in the tracking area at predetermined times (each cell should be visited at least once). This *known* trajectory of their motion (i.e. a *training sequence*) is used in the calculation of the score and Fisher information for the estimator recursion of equation (9), rather than the estimates provided by the PF in the form of sub-state particles (cf. Appendix A). Once the estimator converges coarsely, the parameter is recorded for future use as the initial value $\hat{\Upsilon}^0$. This calibration need not be repeated because the radio environment (walls, shelves, doors, windows etc) shows only slow variation with time which can be subsequently tracked in the normal (“decision directed”) mode of operation.

V. DISTRIBUTED IMPLEMENTATION AND SCALABILITY

In a centralized implementation of the filter bank of Section III, all valid observations of all motes need to be streamed to a *Fusion Center* via multi-hop communication. Hence, as the network scales, the communication load of each mote increases monotonically and unboundedly, and a regime inevitably occurs when the network is pushed into congestion

and/or the lifetime of the motes becomes unacceptably small. This serious drawback of centralized tracking is inescapable whether or not we choose to exploit inter-target dependencies. As a solution to this problem, we will now propose a distributed power efficient implementation of the filter bank which allows the system to scale seamlessly to larger tracking regions without encountering problems of network congestion and short lifetimes. Our proposal is based on two ideas: (A) *Localized ownership*: At any point in time, a single ‘owner’ mote lying in close proximity to a target is responsible for tracking that target. (B) *Localized data aggregation*: While tracking, an owner mote relies on measurements and sub-state information solely from a local neighborhood of the target. In the following, a *transmission* by a mote means a single-hop broadcast transmission to one or more of its nearest neighbors. The *normalized communication load* means the average number of transmissions made by one mote in each observation interval T .

A. Localized Ownership

Definition: At each time t , every target m has a unique owner mote $\Omega^t(m) \in \{1, 2, \dots, N\}$, who is solely responsible for maintaining and updating the sub-state particles $\{\zeta_{m,p}^t : p = 1, 2, \dots, \Pi\}$, according to the algorithm of Section III. \square Initial ownership may be assigned arbitrarily. Later, the decision to change the ownership of tag m can be made only by its current owner $\Omega^t(m)$. It periodically examines the tag’s current position, which it knows with high certainty from the particles it maintains. If tag m has moved out of its proximity, $\Omega^t(m)$ polls the mote that is now nearest to the target, and transfers ownership to that mote, provided it is willing to take on the responsibility. Otherwise it polls the next closest one, and so on. Assuming that the targets visit various parts of the tracking region equi-probably, this simple change-of-ownership protocol ensures an equitable distribution of the computation and communication load across the sensor network. We will require that the owner $\Omega^t(m)$ should make his ownership of m known to the motes that belong to a certain set which we call the *tracking neighborhood* of target m at time t . It is defined as

$$\eta_\rho^t(m) = \{n \in \{1, 2, \dots, N\} : \|r_{\Omega^t(m)} - r_n\| < \rho\}, \quad (15)$$

where ρ is a user specified *aggregation radius* (in meters), that controls the size of the tracking neighborhood. Using an efficient localized broadcast algorithm like [21], which assures guaranteed delivery throughout the neighborhood using $O(|\eta_\rho^t(m)|)$ *fixed-length* messages, the normalized communication load of publishing the ownership information can be shown to be $O(\frac{M \rho^D}{N t_{ch}})$, where t_{ch} is the average time between the change of ownership of a target.

B. Localized Data Aggregation

The propagation of the sub-state particles for tag m via equation (6) requires that the previous sub-state estimates of all the other tags, namely $\hat{Z}_{m'}^{t-1}$, $m' \neq m$, be made available for the creation of the full-state particles, which entails a normalized communication load of $O(M)$. (Of course, if the

tags are known, or presumed, to be moving *independently*, this contribution to the communication load will be zero.). Similarly, the importance sampling via equation (7) requires that all the valid observations $\{Y_{m,n}^t : \text{all } n\}$ be made available for calculation of the weights, which entails a normalized communication load of $O(N)$. For achieving a truly scalable tracking system we need to ensure that both these contributions are reduced to $O(1)$, via some kind of approximation.

To this end, we postulate that data aggregation at $\Omega^t(m)$ be performed only from the tracking neighborhood $\eta_\varrho^t(m)$. That is, the propagation of the sub-state particles $\{\zeta_{m,\pi}^{t-1}, \pi = 1, 2, \dots, \Pi\}$ is based only on the sub-states $\{\hat{X}_{m'}^t : \Omega^t(m') \in \eta_\varrho^t(m)\}$, with the unavailable sub-states being set to zero. (The interaction feedback matrix C also needs to be appropriately modified; for an example see Section VI.) Similarly, the importance re-sampling of the particles of target m at time t is performed only on the basis of the measurements $\{Y_{m,n}^t : n \in \eta_\varrho^t(m)\}$. Our basic premise is that we can use a very localized neighborhood (i.e. a small ϱ), and still achieve a performance close to what can be achieved with non-localized tracking ($\varrho = \infty$). This is possible because the signal received at the motes in $\eta_\varrho^t(m)$ is considerably more informative about the position of tag m at time t , while the signal received by motes outside $\eta_\varrho^t(m)$ is dominated by the measurement noise and contains relatively little position information. (Errors in ranging and angle estimation are more deleterious in measurements from far-away motes.)

We will give convincing simulation evidence of this hypothesis in Section VI for the case of RSS based tracking, though, of course, this principle applies to TOA, DOA etc as well. Assuming for now that there is no loss in choosing a small constant value for ϱ , we now proceed to note that the data transfer from each mote in $\eta_\varrho^t(m)$ to $\Omega^t(m)$ can be accomplished with a small *constant* number of hops proportional to ϱ , possibly even a single hop. Hence it follows that the contribution to the normalized communication load from data aggregation is reduced to $O(\frac{M\varrho^{D+1}}{N})$. Thus it follows that our proposal for a distributed implementation of the particle filter bank has a total normalized communication load of $O(\frac{M\varrho^D}{N^{t_{ch}}} + \frac{M\varrho^{D+1}}{N})$, which implies that we have a truly scalable implementation. That is, provided the tracking area, the number of motes and the number of targets are all increased in the same proportion, i.e. $M \propto N \propto \aleph, \aleph \uparrow \infty$, and the spatial density of the motes is kept roughly constant over the tracking region, the normalized communication load, and hence the power drain in each mote, remains *invariant*. Furthermore, since typically $t_{ch} \gg 1$ in sparsely deployed networks, data aggregation dominates ownership publication.

A fully distributed implementation of the incremental parameter estimator is also possible, with a somewhat larger complexity of inter-mote coordination and two kinds of owners. We will not discuss the details except noting that: (a) An $O(M)$ normalized communication load is sufficient in the most general case. (b) By restricting the gradient calculation of $\hat{\Gamma}_{l,n}$ to be based on particles only from tags that are (estimated to be) in cell l , the load can be reduced to $O(1)$.

VI. SIMULATIONS AND DISCUSSION

In Section VI-A we simulate the tracking system of Sections III and IV, and in Section VI-B its distributed implementation as proposed in Section V.

A. Simulations of Tracking and Parameter Estimation

The simulations in this section are done in two parts. In the first part (*target tracking*) we assume a static radio environment, initialize the incremental estimator with the true environment parameters $\{\Phi\}, \{\Gamma\}$ and *keep it operational*, and investigate the root-mean-squared (RMS) tracking error. In the second part (*parameter estimation*) we investigate the parameter estimator's performance in static as well as dynamic environments. We use a realistic synthetic two-dimensional radio environment consisting of four rooms of unequal sizes, as displayed in Figure 1. The total tracking area is a square of side 20.0 meters, with nine motes placed in it on a uniform square grid of minimum distance 10 meters, and labelled as shown. The number of *active* motes is denoted by $1 \leq N \leq 9$, with a default value $N = 9$ unless otherwise specified. We assume that each room is a cell (thus $L = 4$) and that the nominal gain of the tag's signal because of a wall is $\gamma_{wall} = -2.0$ dB. Let $\nu_{l,n}$ denote the number of walls crossed by a straight-line path from the cell l to the mote n . The numbers $\nu_{l,n}$ can be gleaned from the building plan in Figure 1. As discussed in Section IV, in order to ensure a full rank for the Fisher information, we will freeze at zero a few components of Γ and $\hat{\Gamma}$ according to equation (13). We assume that $M = 8$ targets are moving around in the tracking area, with an innovation power of $\sigma_U^2 = 10^{-35/10}$ (i.e. -35 dB). In the tracking algorithm we use $\sigma_{thresh}^2 = 0.1$ (i.e. -10 dB) and $\Pi = 128$. For now we will consider only independent motion of targets by choosing $\epsilon_{depend} = 0$.

1) *Static Radio Environment Model*: The transmission from cell l to mote n is modelled to undergo a time-invariant gain given by $\Gamma_{l,n} = \nu_{l,n}\gamma_{wall} + \eta_{l,n}(1 - \mathbb{I}_l(r_n))$, where $\eta_{l,n}$ are independently drawn from a normal distribution of zero mean and standard deviation $\sigma_\Gamma = 3.0$ dB. (The function $\mathbb{I}_l(\cdot)$ is defined in Appendix A.) Similarly, the actual transmit power of tag m is modelled to be a time-invariant quantity given by $\Phi_m = \Phi_{nominal} + \phi_m$ where ϕ_m are independently drawn, for all m , from a normal distribution with zero mean and a standard deviation $\sigma_\Phi = 3.0$ dB. Once the parameters $\{\Gamma_{l,n}\}$ and $\{\Phi_m\}$ are drawn, they are kept fixed for the duration of an experiment.

2) *Dynamic Radio Environment Model*: The cell-to-mote gains are modelled to be functions of time, given by $\Gamma_{ln}^t = \nu_{l,n}\gamma_{wall} + \sigma_\Gamma \sin(2\pi t/t_\Gamma)(1 - \mathbb{I}_l(r_n))$. Similarly the transmit powers are modelled as $\Phi_m^t = \Phi_{nominal} + \sigma_\Phi \sin(2\pi t/t_\Phi)$. We choose the hyper-parameter values $\sigma_\Phi = \sigma_\Gamma = 3.0$. Also, since our experiments on a real-life WSN test-bed suggest that the transmit power changes more slowly than the radio gains, we choose the following practically relevant time constants: $t_\Gamma = 5 \times 10^4$ and $t_\Phi = 2 \times 10^5$. For the static as well as the dynamic model, the estimated parameters $\{\hat{\Gamma}_{l,n}\}$ and $\{\hat{\Phi}\}$ are initialized to their known nominal values, namely $\{\nu_{l,n}\gamma_{wall}\}$ and $\Phi_{nominal}$ respectively.

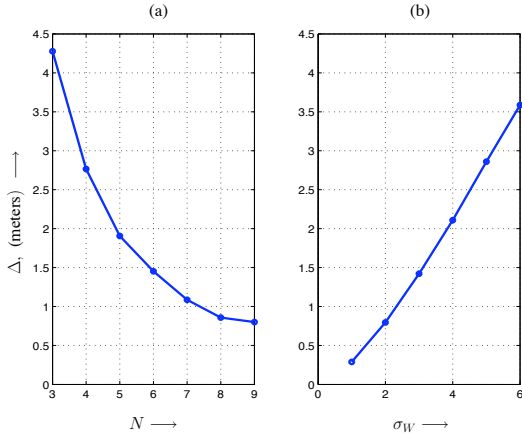


Fig. 2. Effect on tracking accuracy of (a) the number of nodes N , with σ_W fixed at 2.0, and (b) the measurement noise standard deviation σ_W , with N fixed at 9.

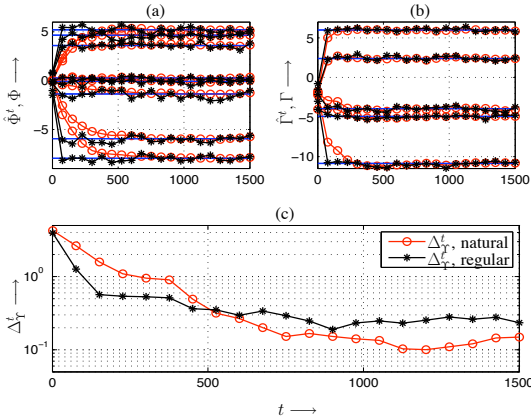


Fig. 3. An example of acquisition of a static radio environment. (a) Estimated parameter $\hat{\Phi}^t$ with a natural gradient recursion (circle markers), estimated parameter $\hat{\Phi}^t$ with a regular gradient recursion (star markers), and true parameter Φ (no markers). (b) Estimated parameter $\hat{\Gamma}^t$ with natural gradient (circle), estimated parameter $\hat{\Gamma}^t$ with regular gradient (stars) and true parameter Γ (no markers). (c) $\Delta_{\hat{\Upsilon}}^t$, the total parameter estimation error.

3) *Simulation Results: Target Tracking*: Figure 2 displays the effect on the RMS tracking accuracy Δ in meters, as defined in equation (1), of the number of nodes N and the measurement noise standard deviation σ_W , in a static radio environment.

The mean squared tracking accuracy Δ^2 , in principle an ensemble expectation, is estimated by doing a single sufficiently long run and taking a time-average of the squared error. It is seen that the dependence on both quantities N and σ_W is roughly linear, except when N is large and σ_W is small. The saturation effects can be attributed to our use of a finite number of particles and an artificially inflated power for the particle innovation process (cf. Section III). The results indicate that we can exchange the mote density for an improved RSSI measurement accuracy, without sacrificing the tracking error.

4) *Simulation results: Parameter Estimation*: In Figure 3 we illustrate an example of acquisition of a randomly chosen static radio environment, as described in Section VI-A1. We have chosen the system parameters $N = 9$ and $\sigma_W = 2.0$. The subplots (a) and (b) show the trajectories of the components of estimated parameters $\hat{\Phi}$ and $\hat{\Gamma}$ respectively. The lines with circle markers indicate a natural gradient algorithm

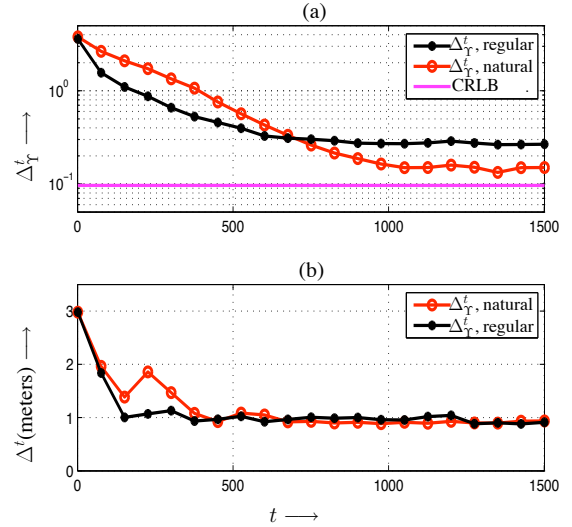


Fig. 4. Acquisition and tracking performance averaged over ten independent experiments. (a) $\Delta_{\hat{\Upsilon}}^t$, the RMS estimation error in Υ , along with the corresponding Cramer-Rao lower bound estimate given by equation (14). (b) Δ^t , the normalized RMS tracking error in the positions of the targets, in meters.

($F^{-1} = F_{\Upsilon}^{-1}$) while lines with star markers indicate a regular gradient algorithm ($F^{-1} = \|F_{\Upsilon}^{-1}\| I$). The true values are also shown (as solid lines) for reference. (In the case of $\Gamma, \hat{\Gamma}$, for clarity we have shown only the components corresponding to the gains from the first cell to all the nodes.) We see that this ‘step-response’ has rapid acquisition (within 500 samples). With natural gradient all components tend to converge together, while with regular gradient the speeds are more varied. Similarly, subplot (c) shows the absolute error in the total estimated parameter, $\Delta_{\hat{\Upsilon}}^t = \|\hat{\Upsilon}^t - \Upsilon^t\|$. We see that the MSE remains uniformly small after acquisition.

In Figure 4, we provide the average acquisition performance where the averaging is done over ten experiments. In each experiment a new random static environment is chosen, and a new measurement noise process is simulated. Subplot (a) shows the RMS error in the parameter estimate, and the optimistic estimate of the CRLB resulting from equation (14), while subplot (b) shows the RMS tracking error (in meters) of the particle filter as it uses the parameter values provided by the estimator. We see that we practically achieve the estimate of the CRLB when a natural gradient is used, indicating that our estimator is close to being efficient. A small loss is suffered w.r.t. the estimated CRLB when a regular gradient is used. Similarly, we see that initially, when the radio environment is not yet acquired, the average tracking error is quite large (of the order of four meters), but after acquisition is completed it drops to a fraction of a meter.

Finally, in Figure 5, we illustrate an example of tracking a time-varying radio environment of Section VI-A2. We use the natural gradient, and $N = 9$ and $\sigma_W = 2.0$. The trajectories of the true parameters (solid lines) and the estimated parameter (lines with markers) are displayed in subplots (a) and (b) respectively for Φ and Γ . Subplot (c) shows the absolute parameter estimation error and subplot (d) shows the target tracking error (in meters) during the same time interval. We

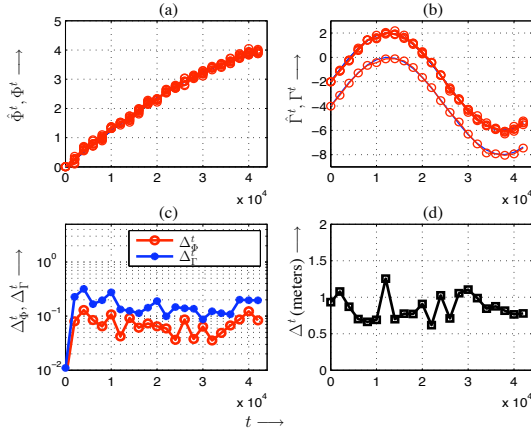


Fig. 5. Tracking a time-varying radio environment with a natural gradient recursion. (a) Estimated parameter $\hat{\Phi}^t$ (lines with circle markers), and the true parameter Φ^t (solid lines). (b) Estimated parameter $\hat{\Gamma}^t$ (lines with circle markers) and the true parameter Γ^t (solid lines). (c) The parameter estimation errors $\Delta_{\hat{\Phi}}^t, \Delta_{\hat{\Gamma}}^t$. (d) The normalized RMS position tracking error Δ^t .

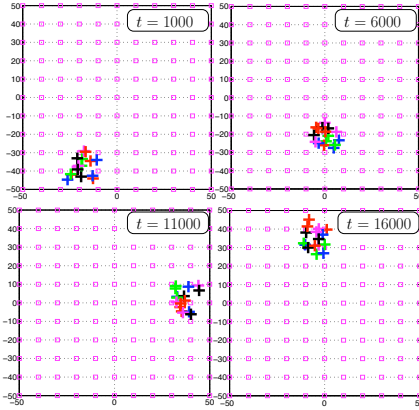


Fig. 6. Snapshots of herd motion of $M = 16$ targets governed by maneuver model of Section II-A with selection of parameters as in Section VI-B.

see that the estimated parameter agrees well with the true parameter, yielding a target tracking error comparable to the experiments done with a static environment (Figure 4).

B. Simulations of Distributed Implementation

In this section we will present simulation results for RSS based tracking of *co-dependently* moving targets with the distributed implementation of the cooperative particle filter as described in Section V. For simplicity, we will assume that the radio parameters are perfectly known to the tracking algorithm. We simulate $M = 16$ targets moving co-dependently in a large square tracking region of dimensions 100×100 meters, within which $N = 121$ motes are deployed on a square grid. We choose $\epsilon_{depen} = 0.25$, $\sigma_U^2 = -20$ dB, $\sigma_{thresh}^2 = -10$ dB and $\Pi = 128$. We use an interaction feedback matrix $C = \text{KRON}(\text{KRON}(\mathbf{1}_{M \times M} - M I_M, I_D), [1, 0])$ (KRON is the ©MATLAB function for the Kronecker tensor product). The non-linearity used is $g(s) = \tanh(\sinh(x)/100)$, which is characterized by saturation to ± 1.0 for large inputs, and a ‘dead zone’ for small inputs. These choices mimic the flocking rules postulated by [13], and result in a type of ‘herd’ motion, whose snapshots are displayed in Figure 6.

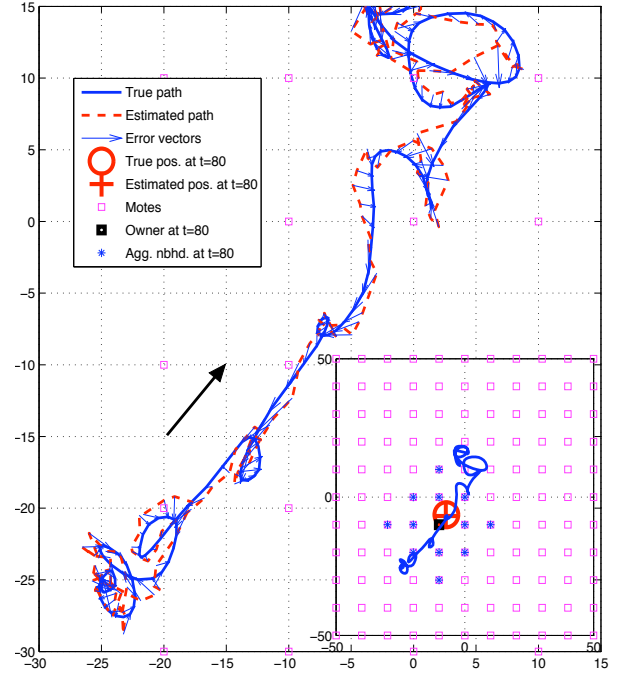


Fig. 7. Target tracking with $\sigma_W = 3.0$. Big figure shows the true path of target $m = 1$, and its estimate. The arrow shows the direction of travel. The inset shows an illustration of target ownership and tracking neighborhood at epoch $t = 80$, with $\varrho = 20$ meters.

In Figure 7, we first illustrate the notion of localized tracking. For this experiment, we choose the tracking neighborhood radius $\varrho = 20$ meters and noise deviation $\sigma_W = 3.0$. The filter update for target m at time t uses a modified interaction feedback matrix $C = \text{KRON}(\text{KRON}(\mathbf{1}_{M \times M} - \alpha^t M I_M, I_D), [1, 0])$, where α^t denotes the fraction of the sub-states that were available at time t from the tracking neighborhood $\eta_\rho^t(m)$. For clarity we only display the path of target number $m = 1$, though all $M = 16$ targets have been being simulated. Also shown are the estimated path and the error vectors at various points in time. The inset graph shows a zoomed out picture with the following elements at an arbitrary time $t = 80$: the true position of the target X_m^t (circle), the estimated position \hat{X}_m^t (plus), the tracking neighborhood $\eta_\rho^t(m)$ (stars) and the owner mote $\Omega^t(m)$ (square).

Finally, in Figure 8 we come to the crucial observation of these simulations. We consider the RMS tracking error Δ as a function of σ_W , and let ϱ be a parameter. We simulate the cooperative filter bank proposed in this paper, as well as a ‘non-cooperative’ filter bank as a benchmark. By non-cooperative, we mean that although the targets move with dependencies, the filter bank presumes independent motion and hence uses $\epsilon_{depen} = 0$, thus implementing an independent filter for each target. Our main interest is in estimating (i) what aggregation radius ϱ is sufficient to extract almost all the tracking accuracy possible, and (ii) how much ‘gain’ our cooperative filter bank gives over the non-cooperative scheme. Both these questions are answered in Figure 8 (note that the ordinate axis is logarithmic). Firstly, we see that a very modest value of $\varrho = 20$ meters is sufficient to give most of the tracking accuracy. (Recall from Figure 7 that this corresponds to a

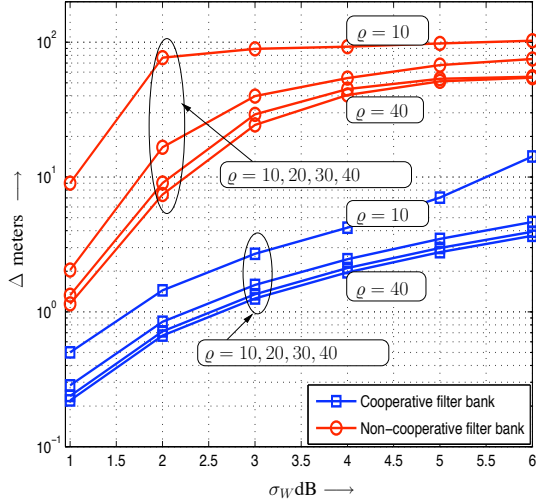


Fig. 8. RMS tracking error Δ (meters) as a function of measurement noise deviation σ_W (dB), of a cooperative distributed filter bank and a non-cooperative distributed filter bank, for various values of the aggregation radius $\rho = 10, 20, 30, 40$ meters.

tracking neighborhood of size $|\eta_\rho^t(m)| \approx 13$ nodes.) Secondly, we see that there is a huge gain in tracking accuracy due to the use of the cooperative filter bank, ranging from a factor of 8.0 to 20.0. Furthermore, this gain is enhanced for large values of measurement noise, which is exactly the regime of interest for low cost WSN tracking systems. (The saturation of the curves at high noise levels is an artifact of the finite square tracking.)

In summary, we have demonstrated a simple WSN based solution for tracking multiple co-dependently moving targets in indoor environments. The tracking component of the solution exploits the temporal and spatial dependencies in the motion of the targets via a distributed tractable particle filter, and achieves good tracking accuracy and stability. At the same time, by using a tandem incremental estimator for the radio environment, the solution achieves a high level of robustness to effects like multi-path and occlusions that are commonly found in indoor environments.

APPENDIX A

CALCULATION OF THE SCORE AND FISHER INFORMATION

Denote the projection of a sub-state to its spatial position by $\psi: \mathbb{R}^{2D} \rightarrow \mathbb{R}^D$, $\psi(a) = [a_1, a_3, \dots, a_{2D-1}]^T$. In the following development, for clarity we drop the temporal index for the various quantities. Using the presumed mutual independence of the sub-states, the log-likelihood of the parameters is

$$\lambda(\Upsilon; \{Y_{l,n}\}) \sim \sum_{m=1}^M \log \int p(\{Y_{m,n} : \text{all } n\} | \Phi_m, \{\Gamma\}, z_m) q_m(z_m) dz_m \quad (16)$$

Due to equation (3),

$$p(\{Y_{m,n} : \text{all } n\} | \Phi_m, \{\Gamma\}, z_m) = \frac{\exp \left\{ \frac{-1}{2\sigma_W^2} \sum_{n=1}^N \left(\frac{Y_{m,n} - \Phi_m - \Gamma_{f(\psi(z_m)),n}}{+\rho 10 \log \|\psi(z_m) - r_n\|} \right)^2 \right\}}{(2\pi\sigma_W^2)^{N/2}}.$$

The score of the parameter components is given by

$$S_{\Phi; m} \doteq \frac{\partial \lambda(\{Y_{l,n}\})}{\partial \Phi_m} = \frac{1}{D} \int \frac{dz_m q_m(z_m) p(\{Y_{m,n'} : \text{all } n'\} | \Phi_m, \{\Gamma\}, z_m)}{\sum_{n=1}^N (Y_{m,n} - \Phi_m - \Gamma_{f(\psi(z_m)),n} + \rho 10 \log \|\psi(z_m) - r_n\|)},$$

$$S_{\Gamma; l,n} \doteq \frac{\partial \lambda(\{Y_{l,n}\})}{\partial \Gamma_{l,n}} = \frac{1}{D} \sum_{m=1}^M \int \frac{dz_m q_m(z_m) \mathbb{I}_l(\psi(z_m)) p(\{Y_{m,n'} : \text{all } n'\} | \Phi_m, \{\Gamma\}, z_m)}{(Y_{m,n} - \Phi_m - \Gamma_{f(\psi(z_m)),n} + \rho 10 \log \|\psi(z_m) - r_n\|)},$$

where $\mathbb{I}_l(a) = 1$ if $l = f(a)$, zero otherwise (i.e. an indicator function for the condition that position a lies in cell number l), and $D = \sigma_W^2 \int q_m(z_m) p(\{Y_{m,n'} : \text{all } n'\} | \Phi_m, \{\Gamma\}, z_m) dz_m$. Hence the total score is given by

$$S(\Upsilon; \{Y_{l,n}\}) \doteq f_{stack}(\{S_{\Phi; m}\}, \{S_{\Gamma; l,n}\}). \quad (17)$$

The integrations above are approximated in practice by taking sample averages over the set of sub-state particles. For calculating the Fisher information F_Υ we need to further differentiate the score and take an expectation w.r.t. the joint distribution of the state and the observations made available to the estimator. Since no closed form expression is available for the propagated density $q(\cdot)$ of the total-state (which is the reason why we use a particle representation in the first place), it appears infeasible to obtain a closed form expression for F_Υ . However we can calculate an optimistic approximation by differentiating the conditional likelihood assuming *perfect* state information at the estimator. That is,

$$F_\Upsilon \approx \hat{F}_\Upsilon = \mathbb{E} \left[-\frac{\partial^2}{\partial \Upsilon^2} \log p(\{Y_{m,n}\} | \Upsilon, Z_{true}) \right], \quad (18)$$

where the expectation is w.r.t. the joint distribution of the true state and the observations, $p(Z_{true})p(\{Y_{m,n}\} | \Upsilon, Z_{true})$. For clarity let us denote $p(\{Y_{m,n}\} | \Upsilon, Z_{true})$ simply as $p(\Upsilon)$. Then the expectations w.r.t. $\{Y_{m,n}\} | \Upsilon, Z_{true}$ of the various second derivatives of the log-likelihood $\log p(\Upsilon)$ are given by

$$\mathbb{E} \left[\frac{\partial^2 \log p(\Upsilon)}{\partial \Phi_m \partial \Phi_{m'}} \right] = \begin{cases} \frac{-N}{\sigma_W^2}, & \text{if } m = m' \\ 0, & \text{otherwise} \end{cases}, \quad (19)$$

$$\mathbb{E} \left[\frac{\partial^2 \log p(\Upsilon)}{\partial \Gamma_{l,n} \partial \Gamma_{l',n'}} \right] = \begin{cases} \frac{-\sum_{m=1}^M \mathbb{I}_l(\psi(Z_{true,m}))}{\sigma_W^2}, & \text{if } (l,n) = (l',n') \\ 0, & \text{otherwise} \end{cases}, \quad (20)$$

$$\mathbb{E} \left[\frac{\partial^2 \log p(\Upsilon)}{\partial \Phi_m \partial \Gamma_{l,n}} \right] = \frac{-\mathbb{I}_l(\psi(Z_{true,m}))}{\sigma_W^2}. \quad (21)$$

Finally, let us make the simplifying assumption that all the cells are of equal area and that, after the state MC of equation (2) becomes stationary, the true positions of the tags are a-priori uniformly distributed in the tracking region (the velocity distribution is immaterial). By taking the expectations of the (negative of the) RHS of the equations above under such a $p(Z_{true})$, and packing them into a matrix form compatible with the stacking operation f_{stack} , we get the final form of the optimistic approximation \hat{F}_Υ given by equation (12).

REFERENCES

- [1] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Localization Systems for Wireless Sensor Networks," *IEEE Wireless Communications Magazine*, vol. 14, no. 6, pp. 6–12, Dec. 2007.
- [2] N. Patwari, J. N. Ash, S. Kyperountas, A. O. H. III, R. L. Moses, and N. S. Correal, "Locating the Nodes: Cooperative Localization in Wireless Sensor Networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, July 2005.
- [3] D. Culler, D. Estrin, and M. Srivastava, "Guest Editors' Introduction: Overview of Sensor Network," *Computer*, vol. 37, no. 8, pp. 41–49, Aug. 2004.
- [4] I. Kyriakides, D. Morrell, and A. Papandreou-Suppappola, "Sequential Monte Carlo Methods for Tracking Multiple Targets With Deterministic and Stochastic Constraints," *IEEE Trans. Signal Processing*, vol. 56, no. 3, pp. 937–948, March 2008.
- [5] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund, "Particle Filters for Positioning, Navigation, and Tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 425–437, Feb. 2002.
- [6] M. Orton and W. Fitzgerald, "A Bayesian Approach to Tracking Multiple Targets Using Sensor Arrays and Particle Filters," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 216–223, Feb. 2002.
- [7] C. Kreucher, K. Kastella, and A. O. H. III, "Tracking Multiple Targets Using a Particle Filter Representation of the Joint Multitarget Probability Density," *SPIE Int. Symp. on Optical Science and Techn.*, vol. 5204, pp. 258–259, 2003.
- [8] K. Yuna and D. Kim, "Robust location tracking using a dual layer particle filter," *Pervasive and Mobile Computing*, vol. 3, no. 3, pp. 209–232, Jun. 2007.
- [9] I. Recio, J. M. Moya, A. Araujo, J. C. Vallejo, and P. Malagon, "Analysis and Design of an Object Tracking Service for Intelligent Environments," *Lecture Notes in Computer Science (Springer Berlin/Heidelberg)*, vol. 5518, pp. 914–921, Jun. 2009.
- [10] P. Djuric, T. Lu, and M. Bugallo, "Multiple Particle Filtering," *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol. 3, pp. III–1181–III–1184, April 2007.
- [11] X. R. Li and V. P. Jilkov, "A Survey of Manuevering Target Tracking: Dynamic Models," *IEEE Trans. Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.
- [12] T. Vercauteren and X. Wang, "Decentralized Sigma-Point Information Filters for Target Tracking in Collaborative Sensor Networks," *IEEE Trans. Signal Processing*, vol. 53, no. 8, pp. 2997–3009, Aug. 2005.
- [13] C. W. Reynolds, "Flocks, Herds and Schools: A Distributed Behavioral Model," *Comput. Graph. (Proc. ACM SIGGRAPH'87)*, vol. 21, pp. 25–34, Jul. 1987.
- [14] T. Rappaport, *Wireless Communications*. Prentice-Hall, 2002.
- [15] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
- [16] C. Morelli, M. Nicoli, V. Rampa, and U. Spagnolini, "Hidden Markov Models for Radio Localization in Mixed LOS/NLOS Conditions," *IEEE Trans. Signal Processing*, vol. 55, no. 4, pp. 1525–1542, April 2007.
- [17] R. M. Neal and G. E. Hinton, "A View of the EM Algorithm That Justifies Incremental, Sparse, and Other Variants," pp. 355–368, 1998.
- [18] A. Oka and L. Lampe, "Incremental Distributed Identification of Markov Random Field Models in Wireless Sensor Networks," *IEEE Trans. Signal Processing*, vol. 57, no. 6, pp. 2396–2405, June 2009.
- [19] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.
- [20] H. J. Kushner and J. Yin, *Stochastic Approximation Algorithms and Applications*. Springer-Verlag, 1997.
- [21] M. Khabbazi and V. K. Bhargava, "Localized Broadcasting with Guaranteed Delivery and Bounded Transmission Redundancy," *IEEE Trans. on Computers*, vol. 57, no. 8, pp. 1072–1086, Aug. 2008.



Anand Oka (S'01, M'09) earned a B.E. in Electronics and Telecommunications from the Government College of Engineering - Pune, India in 1995, an M.Sc. in Electrical Engineering from the Technion - Israel Institute of Technology in 1999, and a Ph.D. in Electrical and Computer Engineering from the University of British Columbia, Canada in 2009. He held the position of Algorithms Research Engineer with Millimetrix Broadband Networks, in Israel and USA, from 1998 to 2006. Currently he is employed as a DSP Developer with Research In Motion, Canada. His main research interests are in information theory, communications, source/channel coding and iterative statistical inference techniques, with applications to distributed signal processing for wireless sensor networks. He is a recipient of the British Columbia Innovation Council scholarship in Intelligent Systems (2007), the NSERC Post Graduate Scholarship (2007), the UBC Graduate Entrance Scholarship (2006), the Technion Aaron and Miriam Gutwirth Scholarship (1996) and the NCERT National Talent Search Scholarship (1989).



Lutz Lampe Lutz Lampe (M'02, SM'08) received the Diplom (Univ.) and the Ph.D. degrees in electrical engineering from the University of Erlangen, Germany, in 1998 and 2002, respectively. Since 2003 he has been with the Department of Electrical and Computer Engineering at the University of British Columbia, where he is currently an Associate Professor.

He is co-recipient of the Eurasp Signal Processing Journal Best Paper Award 2005 and the Best Paper Award at the 2006 IEEE International Conference on Ultra-Wideband. In 2003, he received the Dissertation Award of the German Society of Information Techniques. He was awarded the UBC Killam Research Prize in 2008 and the Friedrich Wilhelm Bessel Research Award by the Alexander von Humboldt Foundation in 2009.

He is an Editor for the IEEE Transactions on Wireless Communications and the International Journal on Electronics and Communications (AEUE), and he has served as Associate Editor for the IEEE Transactions on Vehicular Technology from 2004 to 2008. He is Vice-Chair of the IEEE Communications Society Technical Committee on Power Line Communications. He was General Chair of the 2005 International Symposium on Power Line Communications and the 2009 IEEE International Conference on Ultra-Wideband.