# Sensor Selection for Cooperative Spectrum Sensing

Yngve Selén, Hugo Tullberg, and Jonas Kronander
Ericsson Research
Torshamnsgatan 23, Kista
SE - 164 80 Stockholm, Sweden
Email: yngve.selen@ericsson.com, hugo.tullberg@ericsson.com, jonas.kronander@ericsson.com

***Abstract*—This article considers spectrum-on-demand in a cellular system. A communication system that wants to access spectrum to which it does not own a license must perform spectrum sensing to identify spectrum opportunities, and to guarantee that it does not cause unacceptable interference to the license owner.**

**Because a single sensor may be in a fading dip, cooperative sensing among multiple sensors which experience uncorrelated fading is required to guarantee reliable sensing performance. At the same time, as few sensors as possible should be used to reduce the battery consumption, while still employing enough many for the sensing to be reliable. Since shadow fading is correlated for closely spaced sensors, it is desired to select sensors which are sufficiently spatially separated.**

**The present article addresses the problem of selecting appropriate sensors from a candidate set to engage in cooperative sensing, using different degrees of knowledge about the sensor positions. Three different algorithms for sensor selection are presented and evaluated by means of simulation. It is shown that all algorithms outperform random selection of the sensors.**

## I. Introduction

The usage of radio spectrum is being liberalized in the sense that regulators get more and more favorable to the idea of frequency bands not exclusively assigned to a particular transmission system [1]. This development is largely based on the observation that spectrum assigned to licensees is often not fully utilized, see e.g., [2]. One approach pursued in regulations is to license the spectrum to a licensee, or *primary* user, while at the same time the frequency band may be used by other users, *secondary* users, under the condition that they do not cause harmful interference to the system operation of the primary user. Another approach under discussion is to have completely unlicensed spectrum, which has to be shared with equal right among many users.

At the same time as some licensed frequency bands lie underutilized, the increasing demand for broadband wireless services has made part of the radio spectrum allocated to mobile communications a scarce resource. One of the greatest challenges faced by the wireless industry today is to devise methods for inexpensive and efficient spectrum utilization. To meet the demand, International Mobile Telephony Advanced (IMT-Advanced) envisions peak data rates up to 100 Mbps for high mobility, and 1 Gbps for low mobility, using up to 100 MHz bandwidth. Due to regulatory and practical issues, it is highly unlikely that a 100 MHz bandwidth can be allocated as a contiguous part of the spectrum. It will therefore be necessary to allocate spectrum in a non-contiguous manner. In future non-contiguous allocation, it is likely that different parts of the spectrum are of different "quality", e.g., primary licensed spectrum and spectrum accessed opportunistically as a secondary user.

In this work we study a spectrum-on-demand scenario as a means to provide the peak data rates. We consider a cellular system where a user has access to a certain amount of primary, licensed spectrum and wants to utilize additional secondary accessed spectrum to increase the instantaneous bandwidth. Spectrum-on-demand is of interest to both the user and the operator. For a user, the increased bandwidth provides a better wireless experience, and an operator can provide more advanced services and adapt the bandwidth to the traffic dynamics.

As an example of the operation of a system that employs the spectrum-on-demand approach, consider a system S1 (e.g., a TV broadcasting system) that has a license for the spectrum band B1. Another system S2 (e.g., a cellular system) which has a licensed, reliable spectrum band B2 wants to exploit opportunities in the spectrum band B1 as a secondary user. As long as the network load in S2 is low relative to the bandwidth of B2, it is probably not necessary for S2 to use resources in B1. However, when the network load becomes high, B1 can be used by S2 for, e.g., non time-critical transmissions, such as large file transfers.

A secondary user operating in B1 needs awareness of the spectrum opportunities in B1, i.e., resources (time/frequency resources or codes) in B1 which are currently not used by S1, or by any other secondary system. In our scenario, S2 has to detect the opportunities by spectrum sensing. If the system S2 is confident that there are resources in B1 which are not being used, it may choose to use those resources for its own traffic.

Fig. 1 illustrates the idea of spectrum-on-demand. At time $t_1$ system S2 experiences an increased spectrum demand and its band B1 becomes fully utilized. S2 starts to sense band B1 in search for spectrum opportunities, i.e., available spectrum. At time $t_2$ system S2 has detected a spectrum opportunity and starts to use part of B1 in a secondary manner. At time $t_3$ the spectrum demand decreases but S2 still utilizes resources in B1. At time $t_4$ the spectrum demand decreases further and S2 abandons B1.

Secondary use of spectrum, in general, involves (i) identifying spectrum opportunities, (ii) using the identified opportunities for communication, and (iii) vacating the spectrum when the primary user starts transmitting or when the extra bandwidth is no longer needed. As seen from the above

978-1-4244-2017-9/08/$25.00 ©2008 IEEE

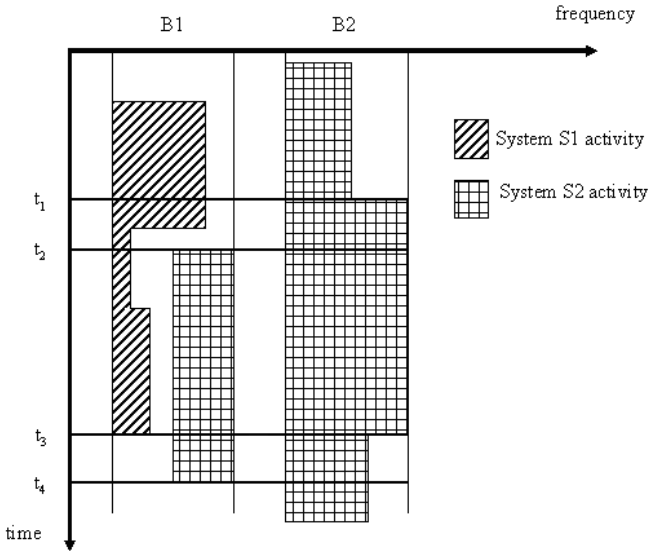


Fig. 1. Illustration of spectrum-on-demand system operation.

example, a key element in opportunistic spectrum access is to reliably detect secondary spectrum opportunities.

To enable spectrum-on-demand in a centralized network, we consider distributed cooperative sensing of the spectrum where the user equipments (UEs) in the system act as sensor nodes[1] in a collaborative cluster, and the sensing is coordinated by the base station (BS), which acts as a main node for the cluster. The sensor nodes receive a sensing request from the BS, and decide whether they are willing to take part in a collaborative sensing or not. This decision may, e.g., be based on the capabilities of the node, the available battery, etc. The nodes which take part in the sensing then each send a sensing report to the base station. This report may contain, e.g., hard decisions, statistics, or raw data. Based upon the received sensing reports, the BS performs the required signal processing to decide which secondary resources, if any, are available for secondary usage.

In order to perform reliable spectrum sensing, several sensors which experience (at least to some extent) uncorrelated fading, with respect to the possible signals they are sensing for, are required [3]. This is because a single sensor may be in a deep fade which makes it virtually impossible to detect usage of spectrum resources [4]. Further, a node performing spectrum sensing will require power for its receiver and baseband circuitry, which will reduce the battery life-time. It is therefore desirable to use as few sensors as possible, while still having enough for the sensing to be reliable. In this sense, the number of sensors to use is a trade-off between having a high reliability of the sensing and having a low battery consumption of the partaking nodes.

Current research focus on detection methods, e.g. [5], and not on sensor selection. To the best of our knowledge, there has been very little work on how to distribute the sensing between the different nodes in the network. If this aspect is not treated properly, there is a risk that several sensors will experience correlated fading, which will decrease the sensing performance [5]. This means that; (i) the number of sensors will either have to be increased, to maintain the confidence in the sensing, or; (ii) that the sensing performance decreases in the sense that the risk of missing a spectrum opportunity, and/or the risk of causing interference to the primary system, will increase. In case (i) above, the total power consumption of the sensing goes up, which makes it less attractive for nodes with limited power (such as nodes operating on battery) to take part in a sensing. In the above case (ii), the throughput of the secondary system decreases or, even more seriously, the interference to the primary system becomes unacceptable in which case secondary operation within the spectrum bands may not be permitted at all.

In this article the selection of which sensors to use in a cooperative sensing is considered. The goal is to find a set of sensors which experience uncorrelated shadow fading, and the underlying assumption is that such a set of sensors will perform better in a subsequent sensing than a set of sensors selected purely at random. Evaluation of the sensor selection methods presented herein using actual spectrum opportunity detection algorithms falls outside the scope of the present article (although discussions on how the detection algorithms will affect the sensor selection methods are included), but will be investigated elsewhere. The article is organized as follows; in Section II the system model considered in this study is specified. In Section III the problem of selecting the sensors to take part in the cooperative sensing is detailed and three algorithms for sensor selection based on different amount of information available to the deciding entity are presented. Simulation results are presented in Section IV and the article ends with conclusions in Section V.

## II. System Model

We consider cooperative spectrum sensing in a centralized system. In the following we describe a cellular system but the methods also apply to an ad hoc network with a master node. In Figure 2, $BS_1$ is the coordinating BS, $BS_2$ is the BS in a neighboring cell in the own system, $UE_1$ and the other shaded UEs are the UEs selected for sensing, $UE_2$ and the other clear UEs are UEs not selected for sensing, $pTx_1$ is a primary (incumbent) transmitter far away from the cell (e.g., a TV transmitter), $pTx_2$ is a primary (incumbent) transmitter within or close to the cell. Incumbent transmitters far away are assumed to have comparably high output power (or the transmitted signals will not reach the sensors) resulting in large incumbent cells and effectively equal received power in a large fraction of the cell. For incumbent transmitters in, or close to, the cell, the assumption of equal received power does not hold. This may result in a situation where parts of the spectrum can be used in parts of the cell far away from the incumbent transmitter. The relative cell size[2] of the incumbent system and the secondary system may thus implicitly affect our sensor selection algorithms. This is further described in Section III.

[1]We will use the terms "sensor nodes", "sensors", and "nodes" interchangeably throughout the article.

[2]The term "cell size" will be used throughout to denote "the area covered by a transmitter". The discussion is therefore not limited to cellular systems.

The sensor selection algorithms described in this article depend on the assumption that the secondary system works on a faster time scale than the primary system. This is necessary, because the secondary system needs to collect the sensing data and form a decision on which resources are available prior to using them. During this timeframe, the resource allocation of the primary system must not change.

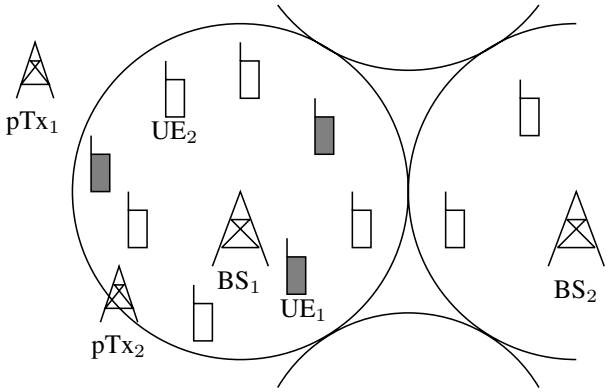


Fig. 2. System Model

### A. Channel Propagation Model

The radio propagation channel fundamentally limits the performance of any sensing system. Knowledge of the channel characteristics and a reasonable channel model are therefore important in order to analyze and simulate the performance of the primary and secondary communication systems and the sensing system. Typically, the effects of a mobile communication radio channel can be divided into three parts; path loss, large-scale fading (shadowing) and small-scale fading. If the maximum sensor separation is small compared to the distance to the primary transmitter, then the path loss will be approximately the same for all sensors. Additionally, if the location of the transmitter is unknown, no information on the path loss is available. This makes it difficult to exploit any path loss model in the sensor selection algorithms. However, path loss models may affect the sensor selection implicitly by influencing the targeted number of active sensors, see Section III.

Small-scale fading causes rapid, random variations in signal strength at the sensors which appear difficult, if not impossible, to consider in sensor selection. It's not unusual [6] to discard the effect of small-scale fading when designing sensing systems based on energy detection [7].

Shadowing, or large-scale fading, is the slow variation of received power (signal strength) as the receiver moves in and out of radio propagation shadows behind mountains, buildings and similar large structures. Shadow fading *will* affect the sensor selection algorithms to be developed herein. Shadowing is often modeled as a log-normal distributed random process around a local mean. The path loss is often used as the local mean and the standard deviation $\sigma_{dB}$ in dB, which depends on the environment, is commonly in the range 4-10 dB [8], [9].

The shadowing fading is spatially correlated, which means that we can exploit information about the sensor positions to reduce the risk of correlated fading, as will be done in the algorithms in Section III. The correlation as a function of distance (e.g., between two sensors), $R(d)$, can be modeled by an exponential function [3]

$$R(d) = e^{-ad} \tag{1}$$

where $d$ is the distance and $a$ is an environment parameter. In an urban non-line-of-sight environment, $a \approx 0.1204/\text{m}$, and in suburban environments $a \approx 0.002/\text{m}$ (assuming the distance is given in meters) [3], [10]. Since the correlation is modeled by an exponential function, it can not be negative and it approaches zero in the limit.

From (1) we can establish the "decorrelation distance" $d_0$, i.e., the minimum separation between terminals required for the shadowing correlation to fall below a determined threshold. In the following we will use the term *uncorrelated shadowing* for shadowing correlation that has fallen below this threshold.

It should be noted that decorrelation distances are not necessarily angle agnostic, e.g., if there is a line-of-sight path between the primary transmitter and a sensor. In this case a sensor which is directly behind another sensor relative to the primary transmitter needs a larger decorrelation distance as compared to another sensor that is perpendicular to the line-of-sight line joining the primary transmitter with the first sensor. Since the direction to the primary transmitter is unknown in the current application this direction-dependent effect can not be directly exploited. Instead, an average over all angles is incorporated in the environment parameter $a$ in (1).

### B. Sensor Position Models

We consider two position models for the sensors: radius information only (i.e., distance from the BS), and full position information (with variable uncertainty). This is further detailed in Sections III and IV.

## III. Sensor Selection

The basic idea in this work is to select appropriate sensors to participate in cooperative sensing. The set of all nodes in a geographical area is referred to as the *total set*. In a cellular system the total set can be all nodes associated with a BS. The *candidate set* is a subset of the total set and consists of all nodes that can perform sensing. The reasons for why a node is not a member of the candidate set can be due to permanent factors, e.g., the node may lack the necessary functionalities, and due to temporary factors, e.g., the battery level of the node is too low. The sensor selection algorithms partition the candidate set into two sets: one *active set* and one *passive set*. The active set contains the sensors that will participate in the cooperative sensing at this particular time, and the passive set contains the sensors in the candidate set that will not participate in the cooperative sensing at this particular time. To reduce the power consumption in the individual nodes, the partitioning of the candidate set into the active set and the passive set is allowed to vary over time. Furthermore, if the

spectrum range to sense is divided into sub-ranges, there can be a separate active set for each frequency sub-range.

In [4] it is shown that the number of active sensors has a large effect on the sensing performance if the sensors are few. However, once the number of active sensors reaches a certain number, the sensing performance is only marginally increased if additional sensors are used. Hence, a suitable compromise between sensing performance and battery consumption in the sensors should give an active set consisting of $N$ sensors, where $N$ is chosen such that the sensing performance is only marginally improved when using $N+1$ sensors, but discernibly worsened with $N-1$ sensors. In [4], suitable sizes $N$ of the active set were 25 – 50, provided the sensors did not experience correlated shadow fading.

The algorithms described herein aim at selecting $N$ sensors for the active set, $N$ being a user parameter, which have as little correlation as possible to one another. The choice of $N$ will depend on the sensing algorithms and detection thresholds. Furthermore, the relative cell sizes of the primary and secondary system can have an effect on the choice of $N$: If $\bar{N}$ uncorrelated sensors within the range of the primary transmission are required by the detection algorithms, then more than $\bar{N}$ sensors may be requested if the cell size of the primary system is on the same order as (or smaller than) that of the secondary system. This is because $\bar{N}$ sensors evenly spread in the secondary cell will not all be within the primary system's transmission range. If, on the other hand, the primary system has significantly larger cells than the secondary system, then all sensors within a cell of the secondary system can be assumed to be within the transmission range of the primary system, and $N := \bar{N}$ sensors will suffice. Since we aim at describing a general methodology for sensor selection which is not tied to any specific detection algorithms, thresholds and systems, we will not further discuss the choice of $N$ herein.

In the rest of this section we first derive approximations on the possible number of sensors in a cell. We then propose three different algorithms for partitioning the candidate set into an active set and a passive set in such a way that the sensors in the active set, i.e., the sensors that perform sensing, do not experience correlated shadow fading.

### A. Maximum and Minimum Numbers of Available Sensors

We establish two bounds which can be used for evaluating the difficulty of a partitioning problem, or for evaluating the quality of a sensor selection algorithm aiming at selecting as many sensors as possible[3]. The upper bound is derived using sphere packing. The lower bound is not a strict bound but rather the expected number of uncorrelated sensors when sensors are selected purely at random.

*1) Sphere Packing Upper Bound:* We use sphere packing to establish an upper bound on the number of sensors that can experience uncorrelated shadowing. Maximizing the number of uncorrelated sensors is equivalent to packing of spheres (or, rather, circles) with radius $r_0 = d_0/2$ centered around the sensor, where $d_0$ is the decorrelation distance i.e., the minimum distance between sensors that experience uncorrelated shadow fading. The densest two-dimensional packing is the hexagonal lattice $A_2$, shown in Fig. 3 [11].

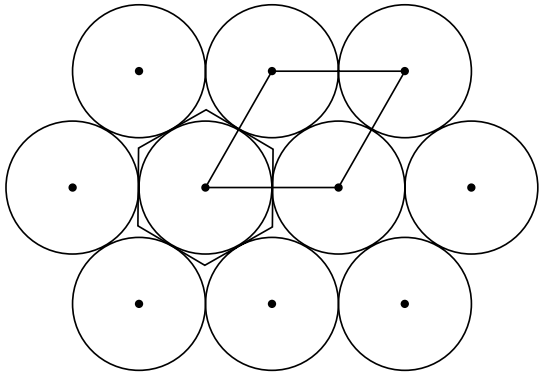

Fig. 3. Sphere packing on a hexagonal lattice. The fundamental region can either be rhomboidal or hexagonal.

The fundamental region of the $A_2$ lattice, which can be either be a rhomboid or a hexagon, has area $A_h = \sqrt{3}/2$. The area of a circle inscribed in a fundamental hexagon has area $A_c = \pi/4$. Hence, the fraction of the area covered by circles over the total area is $\rho = A_c/A_h = \pi/(2\sqrt{3})$. If we ignore effects on the cell edges[4], we get the maximum number of uncorrelated sensors in a cell of area $A_C$ as $n^* = \lfloor \rho A_C/(\pi r_0^2) \rfloor$ where $\lfloor \cdot \rfloor$ denotes rounding down to the nearest integer. For a circular cell of radius $R$ we get

$$n^* = \left\lfloor \frac{2\pi}{\sqrt{3}} \frac{R^2}{d_0^2} \right\rfloor. \tag{2}$$

*2) Random Selection Lower Bound:* We use the expected number of uncorrelated, randomly placed sensors in a cell as a lower "bound" on the minimum number of uncorrelated sensors. Simulations suggest that for a cell of radius $R$, a decorrelation distance $d_0$ and uniform user distribution over the cell, the number of uncorrelated sensors, $n$, is a random variable (r.v.) with a Rayleigh probability density function (pdf) with parameter $R/d_0$. Note that this is not strictly a bound, but rather a number which any sensible sensor selection algorithm aiming at maximizing the number of uncorrelated sensors should exceed.

In Table I we give the upper and lower estimates on the number of sensors for different fractions $R/d_0$, rounded to the nearest lower integer. In the rest of this section we will devise methods to select UEs for collaborative sensing. However, in the following we use selection algorithms to select a predetermined number of sensors while minimizing the correlation probability. In this case we use Table I as an indication of how difficult it will be to find the desired number of uncorrelated sensors, or whether it is possible at all.

[3]The sensor selection algorithms we will present, however, aim at selecting a given number of sensors. These sensors should have a low probability of experiencing correlated shadow fading.

[4]The number of sensors is given by the number of lattice points in a cell of area $A_C$. If lattice points (sensors) are located close to the cell edge, then a small change in cell radius $R$ can result in inclusion/exclusion of a number of sensors. However, since it is hard to determine the exact cell size and since cells are not exactly circular, we argue that these edge effects can be ignored.

TABLE I
COMPARISON OF UPPER BOUND AND MINIMUM EXPECTED NUMBER OF SENSORS.

| $R/d_0$ | Upper bound | Minimum number |
|---|---|---|
| 2 | 13 | 1 |
| 5 | 89 | 5 |
| 10 | 361 | 11 |
| 20 | 1450 | 24 |
| 50 | 9068 | 61 |
| 100 | 36275 | 124 |

### B. Correlation Measure Based Sensor Selection Algorithm

*1) Problem Formulation and Algorithm Description:* This sensor selection algorithm uses information on the location of the sensors and the associated uncertainty to select the sensors in the active set. The algorithm is based on the following optimization problem:

$$\begin{array}{rl} \min & \sum_{i=1}^{M}\sum_{j=1}^{M} a_i a_j c_{ij} \\ \text{subject to} & \sum_{i=1}^{M} a_i = N \\ & a_i \in \{0,1\},\ i=1,\ldots,M \end{array} \quad (3)$$

where $a_i$ represents activity of the sensor $i$: $a_i = 1$ if the sensor belongs to the active set and $a_i = 0$ if the sensor belongs to the passive set. $M$ is the total number of sensors available for sensing (i.e., the number of sensors in the candidate set) and the user parameter $N$ is the desired number of sensors to use in the sensing. The term $c_{ij} \geqslant 0$ is a *correlation measure*[5] between the sensors $i$ and $j$. This correlation measure could, e.g., be a correlation function based on the Euclidian distance between the estimated positions of two sensors, such as $R(d)$ (see (1)). However, the correlation measure should be a combination of a correlation function, the positions of the sensor pair, and the associated positioning uncertainty.

Alternatively, $c_{ij}$ can be seen as a cost measure describing the cost of having sensor $i$ and $j$ active simultaneously. The minimization problem is then concerned with finding the combination of sensors which gives the lowest cost. Since the correlation will generally be one (and perhaps the dominating) component in this cost measure, the term *correlation measure* will be used throughout this section.

The problem above is easily recognized as an integer optimization problem and it can likely only be solved exactly by an exhaustive search; i.e., by testing all possible values of $\{a_i\}_{i=1}^{M}$ which fulfill the constraints. Such an exhaustive search is, at least for large $M$, very time-consuming and is not recommended for implementation in the current context. Instead, we herein describe an algorithm which finds an approximate solution to the above problem by a greedy approach: The algorithm starts with all sensors active. Sensors are then successively removed, one-by-one, from the active set (the sensor which has the largest summed correlation measure relative to the remaining sensors is removed) until the active set contains the desired number of sensors

[5]Note the difference between correlation function and correlation measure. The correlation function is a function describing the correlation between two sensors given the (known) distance between them; see (1). The correlation measure used in this algorithm is more general, and the correlation function is only one possible component of it. Different suggestions for correlation measures are described below, in Section III-B2.

The algorithm can run in one of two modes. The appropriate mode should be selected based on the size of the candidate set, on the processing power of the system running the algorithm, and on how fast the solution needs to be obtained. E.g., the slower Mode 1 can be used if the number of sensors available for sensing is lower than a pre-determined number $K$ ($K$ may be 0 or $\infty$, if one desires to always use one of the modes). Otherwise, Mode 2 will be used. Other criteria for selecting one of the two modes, or for combining them, are of course perfectly possible.

*a) Mode 1:* In this mode, the estimated positions of the sensors are used directly, which means that all the pair-wise correlation measures between sensors have to be computed. Assuming that $M$ sensors are available in a cell, there will be $\sum_{i=1}^{M-1} i = \frac{M(M-1)}{2}$ pair-wise correlation measures to compute.

*b) Mode 2:* If Mode 1 is deemed too computationally intensive (e.g., if there are too many sensors between which the pair-wise correlation measures have to be computed), the estimated sensor positions can be "rounded off" to the closest (according to some distance measure) grid points in the cell. The grid points are some pre-determined locations which are stored in the coordinating node. In this manner, each sensor becomes associated with one of the grid points. The coordinating node has the pre-computed pair-wise correlation measures between all grid points stored in memory, and therefore no computation of the pair-wise correlation measures is necessary. Only the grid points which have sensors associated to them are used, and they correspond to the candidate set. Thus, the pair-wise correlation measures which are not between two active grid points can be ignored. The grid points are assumed to lie closely enough spaced, so that the positioning error introduced by rounding off the sensor position has a relatively small effect on the pair-wise correlation measure. This positioning error should be included in the correlation measure.

Herein, we will only discuss and evaluate Mode 1 of the algorithm.

A sensible correlation measure should be symmetric, so that $c_{ij} = c_{ji}$. In this case, only one of these two correlation measures needs to be stored in the memory. Also, usually $c_{ii} = c_{jj}$, which also reduces the memory requirements. However, $c_{ii}$ can be used to describe the attractiveness of using sensor $i$ as compared to the other sensors. E.g., if sensor $i$ is known to have larger battery and better sensing performance than sensor $j$, then one could set $c_{ii} < c_{jj}$. Similarly, if the coordinating node has very good sensing capabilities (e.g., the coordinating node is a BS in a cellular system), the corresponding $c_{ii}$ value can be set very low, even $c_{ii} = -\infty$ can be used to guarantee that the sensor will remain active.

The correlation measure will be discussed in detail below. Assuming values of the correlation measure have been obtained, the algorithm runs as follows:

1) Set $P$ equal to the number of candidate sensors. Number those sensors from 1 up to $P$.

2) Let

$$i := \arg \max_{i \in \{1,\ldots,P\}} \sum_{j=1}^{P} c_{ij}. \quad (4)$$

3) Remove the sensor $i$ from consideration, number the remaining sensors from 1 up to $P-1$, and then set $P := P-1$.
4) If $P \leq N$, terminate. Otherwise, go to 2.

After termination of the algorithm, $N$ sensors remain and constitute the active set.

The sensor $i$ which maximizes the equation in Step 2 is the sensor which has the greatest summed correlation measure to the remaining sensors. Therefore, if the goal is to minimize the total correlation between the remaining sensors, sensor $i$ should be removed.

It is straightforward to derive other, but similar, versions of algorithm described herein. E.g., the algorithm could start with a single active sensor (e.g., the sensor of the coordinating node) and the other sensors passive. The passive sensors are then turned active in a one-by-one fashion, where the passive sensor is activated which has the minimum value of the summed correlation to the already active sensors. Similarly, one could start with $N$ active, randomly selected, sensors and then take turns to add and remove (or vice versa) a number of sensors for a number of iterations until a stopping criterion is reached. E.g., one can stop after a fixed number of sensor additions or removals, or when the algorithm reaches a local minimum, such that the same sensor(s) that where added are then directly removed (or vice versa).

In the above algorithm the aim is to find the $N$ sensors which have as small summed correlation measure as possible. Also other stopping criteria are perfectly viable. Such stopping criteria may be based, e.g., on the number of remaining sensors, the respective correlation of those, the mean correlation value of the remaining sensors, etc.

The above algorithm assumes that $M > N$. If this is not the case, the solution to the sensor selection problem is trivial: All $M$ available sensors should be used. However, the sensing system then needs to take a decision on whether $M$ sensors are enough to obtain the required sensing accuracy. Possibly, the detection thresholds have to be increased, or spectrum opportunities have to be ignored completely.

*2) The Correlation Measure:* The algorithm itself does not require any specific form of the correlation measure. However, to illustrate what types of correlation measures can be used, and how to combine positioning uncertainty and correlation function, some examples of correlation measures are given below.

Assume that the position of sensor $i$ is a stochastic variable $\boldsymbol{x}_i$ and that the position of sensor $j$ is a stochastic variable $\boldsymbol{x}_j$. The distance vector between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is then a stochastic variable $\boldsymbol{d}_{ij} = \boldsymbol{x}_i - \boldsymbol{x}_j$ with the pdf $p(\boldsymbol{d}_{ij})$. The correlation *function*, which can be different for different types of environment (urban, suburban, etc), is generally a function of the (scalar) distance $d$ between two locations, $R(d)$, see (1). A vector-valued distance, $\boldsymbol{d}$, can be mapped to a scalar distance by taking an appropriate norm: $d := \|\boldsymbol{d}\|$. Since the sensor distance $\boldsymbol{d}_{ij}$ is a stochastic variable one can use, as correlation measure, the expected value of the correlation function conditioned on the distribution of $\boldsymbol{d}_{ij}$:

$$c_{ij}|p(\boldsymbol{d}_{ij}) := E\{R(\|\boldsymbol{d}_{ij}\|)|p(\boldsymbol{d}_{ij})\} = \int R(\|\boldsymbol{d}_{ij}\|)p(\boldsymbol{d}_{ij})d\boldsymbol{d}_{ij}. \quad (5)$$

If the algorithm above runs in Mode 2 with fixed locations (i.e., the grid points), the above equation can be pre-computed for all location pairs. However, it appears intractable for Mode 1 to compute (5) on-line for all sensor pairs.

Under the following simplifying assumption and methodology the above expression can be pre-computed and tabulated over a grid of distance values, which facilitates a (fast) table-lookup mapping between sensor distance and correlation measure. This mapping can be used for both algorithm modes. The required assumption is that the pdfs $\{p(\boldsymbol{d}_{ij})\}$ are, for all sensor pairs, circularly symmetric around their means, independent and identically distributed with exception of the mean values. This assumption holds if the same assumption is made on the pdfs describing the sensor locations in the cell.

Let $\boldsymbol{m}_{ij}$ denote the mean value of $p(\boldsymbol{d}_{ij})$. Then, by using the notation $R_v(\boldsymbol{d})$ for the vector input version of $R(d)$ (i.e., $R_v(\boldsymbol{d}) := R(\|d\|)$), the correlation measure (5) gets the form

$$\begin{aligned} c_{ij}|p(\boldsymbol{d}_{ij}) &:= E\{R_v(\boldsymbol{d}_{ij})|p(\boldsymbol{d}_{ij})\} \\ &= E\{R_v(\boldsymbol{d}_{ij})|p_{\mathbf{0}}(\boldsymbol{d}_{ij} - \boldsymbol{m}_{ij})\} \\ &= \int R_v(\boldsymbol{d}_{ij})p_{\mathbf{0}}(\boldsymbol{m}_{ij} - \boldsymbol{d}_{ij})d\boldsymbol{d}_{ij} \end{aligned} \quad (6)$$

where $p_{\mathbf{0}}(\cdot)$ is the distribution obtained by shifting $p(\cdot)$ to zero mean. The argument of $p_{\mathbf{0}}(\cdot)$ has been multiplied by "$-1$" after the second equality, which is allowed because $p_{\mathbf{0}}(\cdot)$ is circularly symmetric around $\mathbf{0}$. The above equation is easily recognized as a convolution, and since both $R_v(\cdot)$ and $p_{\mathbf{0}}(\cdot)$ are circularly symmetric around $\mathbf{0}$ and since $\{p(\boldsymbol{d}_{ij})\}$ are identically distributed, except for their mean values, for all sensor pairs, the above equation will be equal for all distributions $p(\boldsymbol{d}_{ij})$ for which $\boldsymbol{m}_{ij}$ has identical norm $m = \|\boldsymbol{m}\|$. The equation (6) can thus be pre-computed over a fine grid of $K$ values of $m$, $\{c(m_k)\}_{k=1}^{K}$, and stored in the memory. Then, assuming two sensors $i$ and $j$ are located at a distance $d_{ij} = \|\boldsymbol{d}_{ij}\|$ from one another, one can use

$$c_{ij} := c(m_l),\ m_l = \arg \min_{m \in \{m_k\}_{k=1}^{K}} |d_{ij} - m|. \quad (7)$$

In this manner, only a table lookup (i.e., no on-line computation) is necessary to obtain the $c_{ij}$-values in (4).

As mentioned earlier, $c_{ij}$ can be viewed as a cost measure. Then, $R(d)$ can be a function which does not directly describe the correlation, but rather gives a notion of how the sensing performance is hurt by sensor distance. E.g., $R(d)$ can be much larger and more slowly decaying within the decorrelation distance $d_0$ than what is suggested by the exponential decay in Section II-A.

Finally, it should be noted that there can be a rather large uncertainty inherent in the correlation model $R(d)$ and other factors. Therefore, approximations of the above correlation measures, or ad hoc solutions, may also perform well.

### C. Iterative Partitioning Based on Estimates of the Sensor Positions

Another algorithm for partitioning the sensors such that, with high probability, $N$ uncorrelated sensors are obtained is based on estimates of the positions of the sensor nodes available for sensing and on the decorrelation distance $d_0$.

This algorithm iteratively partitions the sensors into the two subsets; the active set and the passive set. The algorithm starts with the active set containing only the BS. Then the algorithm investigates, sensor by sensor, whether or not the sensor is spatially separated by more than $d_0$ from all the present sensors in the active set. Also an extra margin for positioning uncertainty may be added. If the sensor is uncorrelated in this sense with all the nodes in the active set, it is added to the active set. Otherwise it is added to the passive set. This procedure iteratively increases the size of the active set until it has reached a predefined target number, or until all sensor positions have been investigated and no further sensors are available for consideration. If a predefined target number of sensors exists and is reached before all the sensor positions have been compared to the active set, the remaining sensors are added to the passive set.

The steps of the algorithm are outlined below:

1) Obtain, by some means, estimates of the decorrelation distance $d_0$ and of the positions of the sensors $\boldsymbol{r}_i$, $i = 1, \ldots, M$, where $M$ is the total number of sensors in the candidate set.
2) Add the BS to the active set, if it is to take part in the sensing.
3) Organize the remaining sensors in a candidate set.
4) Choose one sensor from the candidate set at random and remove the chosen sensor from the set.
5) Let $i'$ be the index of the selected sensor. If
$$\|\boldsymbol{r}_{i'} - \boldsymbol{r}_j\| > d_0, \ \forall j \in A,$$
where $A$ is the set of indices of the sensors in the active set, the selected sensor (with index $i'$) is added to the active set. Otherwise, the sensor is added to the passive set.
6) Repeat Steps 4 and 5 until the target number of sensors to be selected has been added to the active set or the candidate set is empty, so that there are no more sensors to choose from.

The sensors in the passive set may be used as input to the same algorithm if more than one frequency band is to be sensed and the sensing load must be spread over many sensors.

### D. Sensor Selection Based on Radius Information

In this section we assume that only radius information is available, i.e., the distances from the coordinating node to the other available sensors. Distance information can be obtained from propagation times[6], timing advance or similar features. We further implicitly assume that the sensor angles are uniformly distributed over the interval $[0, \ 2\pi)$ radians.

To ensure uncorrelated shadowing we should pick sensors with a radial separation of at least $d_0$, and ideally the sensor radii should be integer multiples of $d_0$. The maximum number of sensors is then $\leqslant \lfloor R/d_0 \rfloor + 1$ with equality only if there exists a sensor at each integer multiple of $d_0$. As $R/d_0$ increases this selection method becomes increasingly inefficient as we can accommodate multiple sensors at equal or approximately equal radii towards the cell edge (provided the sensors are well spread in angle). In the following we derive a probabilistic model of correlated shadowing when only radius information is available and use this to devise an algorithm to partition the candidate set into an active and a passive set.

We establish the probability that two sensors at distances $r_1$ and $r_2$ from the coordinating node experience correlated shadowing. Without loss of generality, we assume $r_2 \geqslant r_1$. For the sensors to experience uncorrelated shadowing the distance between them must be at least $d_0$, as indicated in Fig. 4.

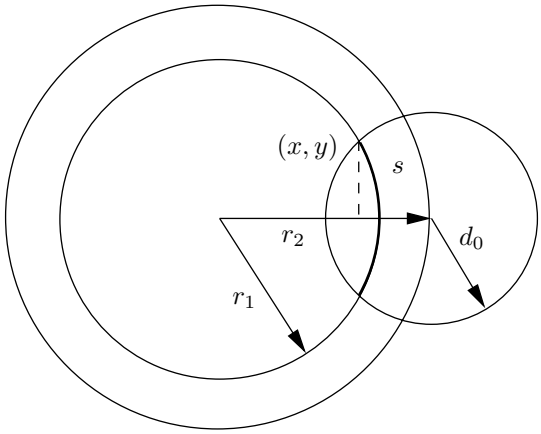


Fig. 4. Visualization of the decorrelation length $d_0$. The figure shows a schematic of two sensors at radii $r_1$ and $r_2$ from the BS in the center, and the radius (around the UE at radius $r_2$) within which the correlation is non-negligible.

We have three cases depending on the values of $r_1$, $r_2$ and $d_0$. If $d_0 < r_2 - r_1$, then probability of correlated shadowing is zero. If $d_0 > r_1 + r_2$, then the probability of correlated shadowing is 1. In other cases, the probability of correlation is given by the length of the circle segment $s$ divided by the circumference of the circle with radius $r_1$. To find $s$ we note that the circles of radii $r_1$ and $d_0$ intersect in the upper half plane at the point $(x, y)$, and we get

$$\begin{aligned} r_1^2 &= x^2 + y^2 \\ d_0^2 &= (r_2 - x)^2 + y^2. \end{aligned} \tag{8}$$

Solving for $x$ yields

$$x = \frac{r_1^2 + r_2^2 - d_0^2}{2r_2} \tag{9}$$

and the angle $\theta$

$$\theta = \cos^{-1}\left(\frac{r_1^2 + r_2^2 - d_0^2}{2r_1 r_2}\right). \tag{10}$$

[6]This gives the distance the electromagnetic waves have traveled, not the physical distance between units, and is an upper bound on the separation between the base station and the UE. However, the difference between the two distances is usually small and the propagation distance can thus be used to determine whether units experience uncorrelated shadowing or not. It is more likely that insufficient timing resolution in the receiver will cause larger errors than the difference between propagation distance and physical distance.

The length of $s$ is

$$s = 2r_1 \cos^{-1}\left(\frac{r_1^2 + r_2^2 - d_0^2}{2r_1 r_2}\right) \tag{11}$$

and hence

$$\mathrm{P}[\text{corr}] = \frac{s}{2\pi r_1} = \frac{1}{\pi}\cos^{-1}\left(\frac{r_1^2 + r_2^2 - d_0^2}{2r_1 r_2}\right). \tag{12}$$

If we assume that both sensor 1 and sensor 2 are located at (approximately) equal radii $r_1 \approx r_2$, then the probability of correlated shadowing can be simplified to

$$\mathrm{P}[\text{corr}] \approx \frac{d_0}{\pi r_1}. \tag{13}$$

So far we have discussed the probability of correlated shadowing between two sensors. With multiple sensors we can extend the geometrical argument and find the probability of $K$ uncorrelated sensors at a given radius $r_1$. Using a Union Bound argument, we approximate the probability that the $k$-th user is correlated to any of previous $k-1$ by

$$\mathrm{P}[k\text{-th user correlated}] = (k-1)\frac{s}{2\pi r_1} \approx (k-1)\frac{d_0}{\pi r_1}. \tag{14}$$

The number of uncorrelated sensors that can be accommodated at (approximately) equal radius $r_1$ from the coordinating node is the largest integer $K$ such that the probability does not fall below a particular design threshold T. We find $K$ by solving

$$\mathrm{P}[\text{K uncorrelated users}] = \prod_{k=1}^{K}\left[1 - (k-1)\frac{s}{2\pi r_1}\right] \geqslant T. \tag{15}$$

Using (15) we compute the number of sensors, including the coordinating BS, that can be accommodated in circular cells of different sizes with certain probabilities of being uncorrelated. The results are given in Table II. We note that as $R/d_0$ increases the number of possible sensors increases significantly if we allow the same probability of uncorrelated shadowing.

TABLE II
NUMBER OF USERS, INCLUDING THE BS, THAT CAN BE ACCOMMODATED IN CELLS OF DIFFERENT SIZES FOR DIFFERENT PROBABILITIES OF UNCORRELATED SHADOWING.

| Design | $R/d_0$ | | | | | |
|---|---|---|---|---|---|---|
| prob. | 2 | 5 | 10 | 20 | 50 | 100 |
| 1.0 | 3 | 6 | 11 | 21 | 51 | 101 |
| 0.99 | 3 | 6 | 11 | 21 | 70 | 175 |
| 0.95 | 3 | 6 | 15 | 37 | 140 | 385 |
| 0.90 | 3 | 8 | 19 | 51 | 196 | 547 |

*1) Algorithm to Select Sensors Based on Radius Information:* The algorithm takes as its input the candidate set of sensors, where the distances (radii) from the central node to each candidate node is known, and a given design probability of correlation, and returns as its ouput an active and a passive set of sensors. The algorithm is initialized by setting the "next radius" $r := 0$, and the "number of sensors to pick" (at next radius $r$) $k := 1$. The coordinating node (BS in a cellular system or master node in an ad hoc network) is then selected as the first node in the active set.

Given $r$ and a desired maximum value of the correlation probability $\mathrm{P}[\text{corr}_{\text{design}}]$, which we for simplicity can set equal to $T$, the next minimum radius $r'$ is computed by solving (12) for $r_2$ given $r_1 := r$. The value $r'$ is then checked against some constraints. One example of such constraints is a minimum increase in radius between $r$ and $r'$, $r' \geqslant r + c$, where $c$ can be related to the decorrelation distance $d_0$. Another example of constraints is that for certain values of the correlation probability $\mathrm{P}[\text{corr}_{\text{design}}]$, radii $r$ and $r'$, and decorrelation distance $d_0$, there exist no real solutions to (12), in which case some parameter, e.g., the design $\mathrm{P}[\text{corr}_{\text{design}}]$, must be adjusted. When a valid value for $r'$ has been found, the algorithm sets $r := r'$. The algorithm then computes the number of sensors $k$ that can be accommodated at the radius $r$ without the probability of correlation exceeding the design threshold by solving (15).

Given updated values of $r$ and $k$, the algorithm proceeds to selecting the $k$ sensors with smallest radii still greater than $r$. If $k$ such sensors exist they are selected to the active set and the algorithm returns to the step of updating $r$ and $k$ as described above. If less than $k$ sensors with radii greater or equal to $r$ exist in the candidate set, we have now exhausted the candidate sets and no more sensors remain to explore. These remaining sensors are added to the active set and the algorithm terminates.

This algorithm may optionally terminate when a specified value of active sensors has been reached. If no such value is specified then the algorithm below runs without modification. Another option is to run the algorithm without limitation on the number of active sensors and, if the number of selected sensors exceeds the desired number, purge the active set, e.g. by removing the sensor(s) with the smallest radial difference(s).

The algorithm is summarized in the steps below:

1) The candidate set and design correlation probability $\mathrm{P}[\text{corr}_{\text{design}}]$ are inputs to the algorithm.
2) Set $r := 0$.
3) Compute the next radius $r'$ by solving (12) for $r_2$ given $r_1 := r$ and $\mathrm{P}[\text{corr}_{\text{design}}]$.
4) If $r'$ meets all constraints then continue to next step. If not, adjust $\mathrm{P}[\text{corr}_{\text{design}}]$ and return to Step 3.
5) Set $r := r'$
6) Determine the number of sensors $k$ at radius $r$ using (15).
7) Add the $k$ sensors with smallest radii $\geqslant r$ to the active set if they exist, and return to Step 4. If less than $k$ sensors exist, add these to the active set and stop as we now have no further sensors to explore in the candidate set.

If it is desirable from a computational complexity standpoint we can pre-compute the next radius $r'$ given $r$ and the number of sensors at $r$, given the maximum allowed probability of correlation.

### *E. Information Exploited by the Algorithms*

For clarity of exposition, the amount of positioning information exploited by the presented selection algorithms is listed in Table III.

TABLE III
AMOUNT OF POSITIONING INFORMATION EXPLOITED BY THE ALGORITHMS.

| | Position information | Position uncertainty |
|---|---|---|
| Correlation III-B | Full position | Yes |
| Iter part III-C | Full position | No |
| Radius based III-D | Radius only | No |

## IV. SIMULATION RESULTS

This section contains a simulation study of the partitioning algorithms for three different scenarios. We consider circular cells having a radius of $R = 250$ m (all positioning units are given in meters) with the BS (also containing a sensor) placed in the center of the circle. The cells are assumed flat: no height information is used. The considered scenarios are:

1) *Many uniformly distributed sensors*: In this scenario the total number of sensors in the candidate set is $M = 501$. The sensors are randomly uniformly distributed over the cell (except for the BS-sensor located in the origin).
2) *Few uniformly distributed sensors*: This scenario is the same as the above, except that the number of sensors in the candidate set is $M = 101$.
3) *Few sensors clustered in two regions*: Here, the total number of sensors in the candidate set is $M = 101$. The BS-sensor is placed in the origin, 20 sensor positions are independently drawn from a circular Gaussian distribution with mean $(x, y) = (R/2, 0)$ and a standard deviation of 71 meters, and 80 sensor positions are uniformly distributed within a circle of radius $R/6$ positioned at $(x, y) = (-R/2, -R/2)$.

The true positions of the sensors are unknown to the partitioning algorithms, except for the position of the BS-sensor. Instead, the algorithms are given position estimates, modeled by adding independent random variables generated from a circular Gaussian distribution with standard deviation varying from 0 up to $1.5\sqrt{2}d_0$ to all true sensor positions. The decay factor of the correlation function is $a = 0.1204/\text{m}$ (corresponding to the environment parameter in an urban scenario; see Section II), and a correlation of 0.2 is assumed "small enough", such that the decorrelation distance becomes $d_0 = -\log(0.2)/a \approx 13$ m (the distance above which the correlation becomes smaller than 0.2). The choice of the correlation value 0.2 is *ad hoc*, but it can be motivated by the fact that the measurements which are the base for the correlation model (1) fit well for correlations $> 0.2$, but less so for smaller correlation values [3]. Also, for the values of the decay factor $a$ considered here, the decrease of $R(d)$ becomes small with increasing distance for correlations $> 0.2$, so the "distance penalty" for using lower correlation values would be large. The target number of active sensors is set to $N = 30$, which is consistent with [4]. The fraction $R/d_0 \approx 20$, and from Table I we see that the upper bound of uncorrelated sensors is 1450, whereas purely random locations of the sensors would give 24 uncorrelated sensors on average. This means that random sensor selection will likely not fulfill the target of $N = 30$ uncorrelated sensors. Note that the upper bound in Table I assumes that the sensors can be positioned arbitrarily, which is not the case here.

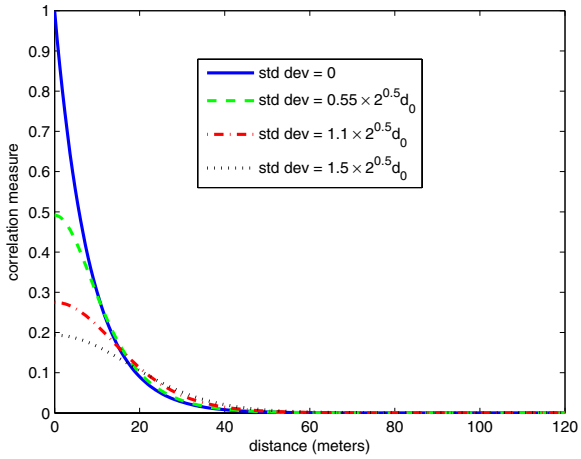


Fig. 5. Correlation measure, used by the method described in Section III-B, for a few positioning error standard deviations.

The simulations are run for 40 different position estimation error variances. For each such variance, 500 different sensor distributions are generated for each of the above scenarios. The following measures are used to evaluate the partitioning methods:

*a) Mean number of uncorrelated sensors:* After the methods have selected the sensors in the active set based on the estimated positions, the true positions of the selected sensors are used to compute the number of sensors in the active set for which all other active sensors are further away than $d_0$. This is then averaged over the 500 sensor distribution realizations.

*b) Mean correlation to other active sensors:* Similar to the above measure, but here the function $R(d)$ is computed and averaged over all sensor pairs (using their true positions) in the active set for each iteration. The average of these averages is then computed over the 500 sensor distribution realizations.

In all the Figures 6–11 the following abbreviations are used: "Corr Meas" refers to the correlation measure based algorithm described in Section III-B, "Iter Part" is the iterative partitioning algorithm described in Section III-C, "Radius based" is the algorithm to select sensors based on radius information in Section III-D1 and "Random" refers to random selection of 30 sensors, which is considered as a reference case. The correlation measure based algorithm described in Section III-B runs with a correlation measure computed by numerically convolving (over a fine grid) the vector-input version of $R(d)$ with a Gaussian pdf of the distance vector $\boldsymbol{d}$ using the true positioning error variance, according to (6). Correlation measures for a few positioning error standard deviations are shown in Figure 5, where it can be clearly seen how the positioning uncertainty smears the correlation function. Also, in the correlation measure based algorithm, the BS-sensor is given a weight $c_{ii} = -\infty$ (assuming the BS-sensor is the $i$-th sensor), so that it will always remain in the active set.

The results for Scenario 1 are shown in Figures 6–7. In Figures 8–9, the results for Scenario 2 are shown. Figures 10–11 show the results for Scenario 3.

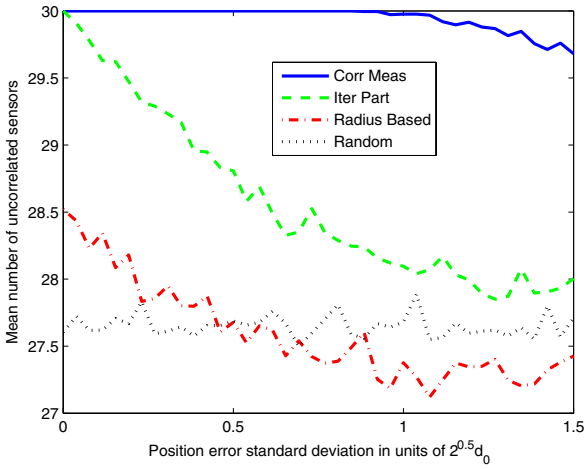


Fig. 6. The mean number of uncorrelated sensors for Scenario 1: many uniformly distributed sensors.

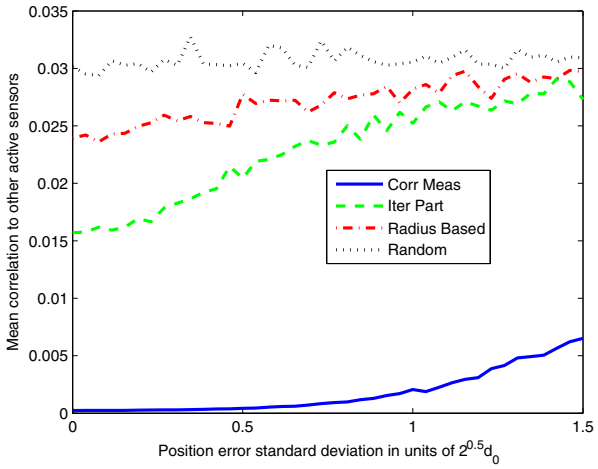


Fig. 9. The mean correlation for Scenario 2: few uniformly distributed sensors.

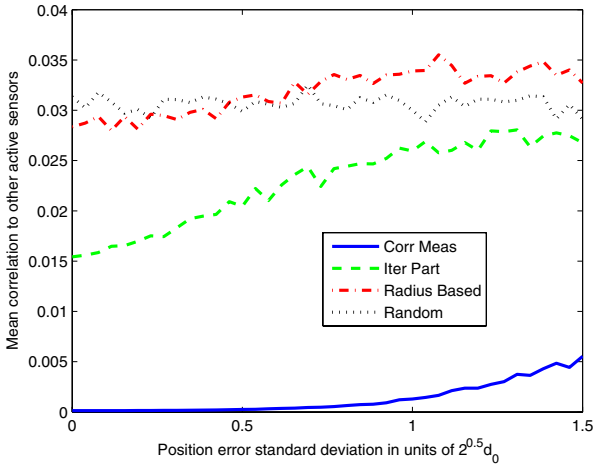


Fig. 7. The mean correlation for Scenario 1: many uniformly distributed sensors.

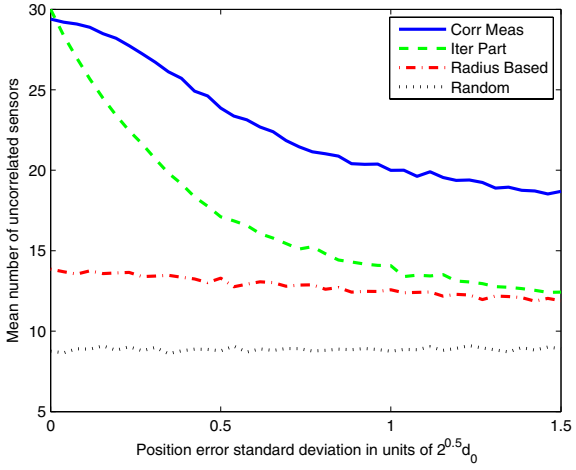


Fig. 10. The mean number of uncorrelated sensors for Scenario 3: few sensors clustered in two regions.

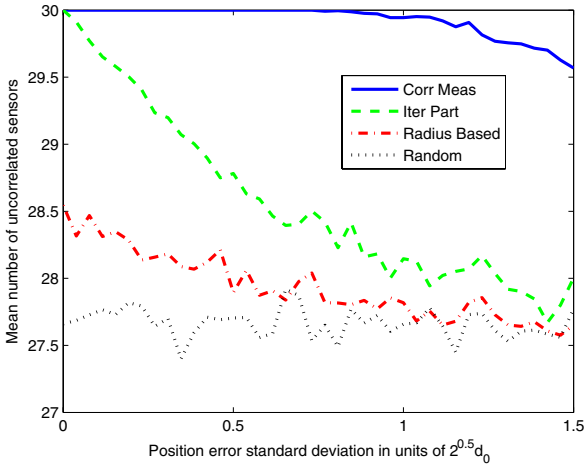


Fig. 8. The mean number of uncorrelated sensors for Scenario 2: few uniformly distributed sensors.

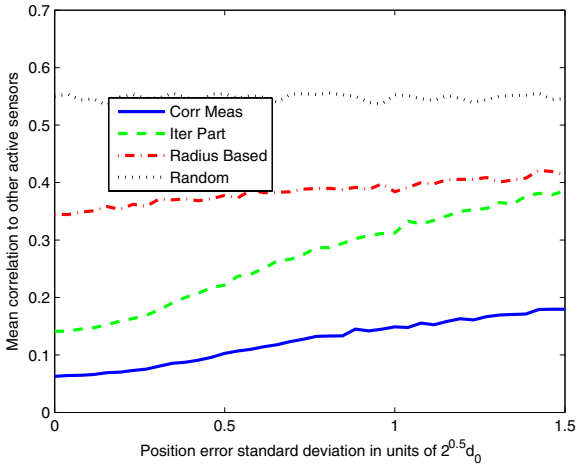


Fig. 11. The mean correlation for Scenario 3: few sensors clustered in two regions.

TABLE IV
AVERAGE RUNTIMES (IN SECONDS) OF THE ALGORITHMS FOR THE DIFFERENT SCENARIOS.

| | Corr Meas | Iter Part | Radius Based | Random |
|---|---|---|---|---|
| Scenario 1 | 2.853 | 0.3619 | 0.0317 | $2.25 \cdot 10^{-4}$ |
| Scenario 2 | 0.0215 | 0.0529 | 0.0171 | $8.32 \cdot 10^{-5}$ |
| Scenario 3 | 0.0210 | 0.0532 | 0.0190 | $8.14 \cdot 10^{-5}$ |

The figures immediately reveal that the three algorithms that take into account some level of positioning of the sensors outperform the random selection algorithm. The only exception is for Scenario 1, where random selection performs better than the radius based method for large positioning errors. This is likely due to mismodeling; the radius based method does not consider the positioning uncertainty, which dominates over the decorrelation distance $d_0$ for the cases where random selection performs better. The plots further reveal that the correlation measure based algorithm performs very well, especially in the scenarios with uniform distribution of the sensors.

It can be seen in Figures 6-9 that the correlation measure based algorithm performs very well even when the uncertainty in the sensor positions are as large as the decorrelation length. Only when the uncertainty is increased beyond the decorrelation distance it becomes more difficult to obtain a decent sensor selection. But for these cases the correlation measure based algorithm still clearly outperforms the other algorithms.

Scenario 3 presents the most difficult task for the algorithms, as seen in Figures 10-11, since the sensors are clustered in two different regions of the cell. The algorithm based on only radius information has difficulties to return a good selection, even when there is no uncertainty in the sensor positions at all (note that this algorithm implicitly assumes an even angular distribution of the sensors; see Section III-D). The sensor selection performed by the iterative partitioning and the correlation measure based algorithms are of good quality when the positioning errors are small, but the quality of the selections decreases faster than for the scenarios with uniform sensor distribution.

In Table IV, the average runtimes (in seconds) of the different algorithms are shown. The algorithms were implemented in Matlab v7.3 and run on an Intel Pentium 4 2.80 GHz processor with 3 Gb RAM. The runtimes where averaged over 5000 iterations (the positioning error did not affect any of the runtimes of the algorithms). The algorithms have not been optimized for speed, but Table IV at least gives an indication of the relative runtimes of the algorithms. In particular, the correlation measure based algorithm described in Section III-B becomes slow when the number of sensors becomes large. The incremental version of this algorithm, where sensors are added one-by-one starting with the BS runs faster at similar performance. Also, as already stated in Section III-B, Mode 2 of the algorithm should be preferred if the number of sensors is large. If the sensors do not move at high velocities, a partitioning runtime of a few seconds may not be a problem. Completely random selection is, of course, the fastest way of selecting sensors, but the risk of selecting sensors which experience correlated shadow fading is much larger than for the other methods.

## V. CONCLUSIONS

In this article the sensor selection problem has been emphasized and three algorithms have been evaluated. Each of the three outperforms the random sensor selection method and all produce good distributions of the selected sensors in the case when the sensors are uniformly distributed over a cell. For scenarios when the sensors are clustered in limited regions of a cell, the two selection algorithms that are based on the estimated position of the sensors perform well while the algorithm that only assumes knowledge of the sensor distances to the base station has problems to obtain a good selection.

Extensions to this work could be, e.g., to address the problem of sensor selection when no centralized coordination is assumed, i.e., the nodes make their own decisions on whether to participate in the sensing or not. Further, to distribute the sensing load fairly over the sensors when repeated sensing is demanded while maintaining reliability of the sensing outcome would also be an interesting problem to address.

## VI. ACKNOWLEDGEMENTS

The authors thank the reviewers for their insightful comments which improved the quality of this paper.

This work was conducted partly within the $E^3$ - End-to-End Efficiency project of the 7th Framework Programme of the European Commission. Further, Hugo Tullberg gratefully acknowledges the support from the Swedish Foundation for Strategic Research (SSF).

## REFERENCES

[1] B. Romano, "FCC adopts rule changes for smart radios," 2005, press Release.

[2] T. Erpek, M. Lofquist, and K. Patton, "Spectrum occupancy measurements: Loring commerce centre, Limestone, Maine, september 18-20, 2007," 2007, shared Spectrum Company Report.

[3] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronic Letters*, vol. 27, no. 23, pp. 2145–2146, November 1991.

[4] S. Mishra, A. Sahai, and R. Brodersen, "Cooperative sensing among cognitive radios," *Communications, 2006. ICC '06. IEEE International Conference on*, vol. 4, pp. 1658–1663, June 2006.

[5] A. Ghasemi and E. S. Sousa, "Opportunistic spectrum access in fading channels through collaborative sensing," *Journal of Communications*, vol. 2, no. 2, pp. 71–82, March 2007.

[6] S. T. Jones, "It's not unusual," 2000, the Millenium Collection: The Best of Tom Jones.

[7] E. G. Larsson and M. Skoglund, "Cognitive radio in a frequency planned environment: some basic limits," 2007, to appear in IEEE Trans. on Wireless Commun.

[8] B. Sklar, "Rayleigh fading channels in mobile digital communication systems part I: Characterization," *IEEE Communications Magazine*, vol. 35, no. 9, pp. 136–146, September 1997.

[9] J. D. Parsons, *The Mobile Radio Propagation Channel*, 2nd ed. Chichester, England: John Wiley & Sons, 2000, author full name John David Parsons.

[10] A. Ghasemi and E. S. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," in *IEEE Int. Symp. New Frontiers in Dynamic Spectrum Access Networks, DySPAN*, Baltimore, MD, USA, November 2005, pp. 131–136.

[11] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*, 3rd ed. New York, NY, USA: Springer-Verlag, 1999, ISBN 0-387-98585-9.