Statistical Location Detection With Sensor Networks

Saikat Ray, *Student Member, IEEE*, Wei Lai, *Student Member, IEEE*, and Ioannis Ch. Paschalidis, *Member, IEEE*

***Abstract*—The paper develops a systematic framework for designing a stochastic location detection system with associated performance guarantees using a wireless sensor network. To detect the location of a mobile sensor, the system relies on RF-characteristics of the signal transmitted by the mobile sensor, as it is received by stationary sensors (clusterheads). Location detection is posed as a hypothesis testing problem over a discretized space. Large deviations results enable the characterization of the probability of error leading to a placement problem that maximizes an information-theoretic distance (Chernoff distance) among all pairs of probability distributions of observations conditional on the sensor locations. The placement problem is shown to be NP-hard and is formulated as a linear integer programming problem; yet, large instances can be solved efficiently by leveraging special-purpose algorithms from the theory of discrete facility location. The resultant optimal placement is shown to provide asymptotic guarantees on the probability of error in location detection under quite general conditions by minimizing an upper bound of the error-exponent. Numerical results show that the proposed framework is computationally feasible and the resultant clusterhead placement performs near-optimal even with a small number of observation samples in a simulation environment.**

***Index Terms*—Hypothesis testing, information theory, mathematical programming/optimization, sensor networks, stochastic processes.**

I. Introduction

RECENT advances in sensor technologies have enabled a plethora of applications of wireless sensor networks including some novel ones; e.g., ecological observation [1] and "smart kindergarten" [2]. A wireless sensor network is deployed either in a *flat* manner, or in a *hierarchical* manner. In the former case, the sensor nodes have identical capabilities. In a hierarchical network, on the other hand, subsets of nodes form logical *clusters*. The *clusterheads*, which act like intermediate data fusion centers, are usually more sophisticated. The hierarchical deployment lends to greater scalability [3]. Clusterheads remain stationary, whereas other sensors in the network maybe moving. In such a deployment, a wireless sensor network can provide *location detection* service.

A location detection system locates the approximate physical position of users (or "assets") on a site. The global positioning system (GPS) is an efficient solution outdoors [4]. However, GPS works poorly indoors, is expensive, and in some military scenarios it can be jammed. Thus, a non GPS-based location detection system, especially indoors, is of considerable interest. Various interesting applications are enabled by such services: locating mobile equipment or personnel in a hospital; intelligent audio players in self-guided museum tours; intelligent maps for large malls and offices, smart homes [5], [6]; as well as surveillance, military and homeland security related applications. Moreover, a location detection service is an invaluable tool for counteraction and rescue [7] in disaster situations.

The key idea underlying location detection is as follows: when a packet is transmitted by a mobile sensor (simply "sensor" henceforth), associated RF characteristics observed by the clusterheads (e.g., signal-strength, angle-of-arrival) depend on the location of the transmitting sensor. This is due to the fact that these characteristics depend on—among many other factors—the distance between the sensor and the clusterhead and attenuations and reflections of various radio-paths existing between them. Therefore, the observation contains information about the location of the sensor node. A wireless local area network (WLAN) based location detection system works analogously: base-stations play the role of clusterheads and the client nodes play the role of mobile sensors.

Several non-GPS location detection systems have been proposed in the literature. The *RADAR* system provides RF-based location detection service [8] using a precomputed signal-strength map of the building. This system declares the position of the sensor to be the location whose mean signal-strength vector is closest to the observed vector in Euclidean sense. A similar system is SpotOn [9]. The *Nibble* system improves up on *RADAR* by taking the probabilistic nature of the problem into account [10]. A similar system is presented in [11]. In [12], the location detection problem is cast in statistical learning framework to enhance the models. Performance trade-off and deployment issues are explored in [13]. References to many other systems can be found in [14].

The location detection systems proposed so far primarily focus on various pragmatic issues that arise while building a working system, but shed little light on the fundamental character of the problem. In particular, let the user position be denoted by the random variable $\mathbf{X} \in \mathcal{X}$, and given a user position, let the observation made by the system be the random variable $\mathbf{Y}$. The ensemble of conditional distributions $\{p_{\mathbf{Y}|\mathbf{X}}(\cdot|\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}$ characterizes the system. The problem of designing a location detection system then reduces to designing

Manuscript received February 17, 2005; revised December 1, 2005. The work of S. Ray was supported in part by the National Science Foundation under CAREER Award ANI-0132802. The work of I. Ch. Paschalidis was supported in part by the National Science Foundation under CAREER Award ANI-9983221 and Grants DMI-0330171, ECS-0426453, CNS-0435312, and DMI-0300359, and by the ARO under the ODDR&E MURI2001 Program Grant DAAD19-01-1-0465 to the Center for Networked Communicating Control Systems. The material in this paper was presented in part at IEEE INFOCOM 2005, Miami, FL, March 2005.

S. Ray is with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: saikat@seas.upenn.edu).

W. Lai and I. Ch. Paschalidis are with the Center for Information and Systems Engineering, and Department of Manufacturing Engineering, Boston University, Brookline, MA 02446 USA (e-mail: wei@bu.edu; yannisp@bu.edu).

Communicated by B. Prabhakar, Guest Editor.

Digital Object Identifier 10.1109/TIT.2006.874376

a system for which $\mathbf{Y}$ contains maximum amount of "information" about $\mathbf{X}$. What is then the right design that achieves this goal?

The designer neither has control over the user location—and hence the distribution of $\mathbf{X}$—nor can achieve the right distribution by means of "coding". However, the ensemble of conditional distributions $\{p_{\mathbf{Y}|\mathbf{X}}(\cdot|\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}$ depends on the positions of the clusterheads. On a typical site, there are many suitable locations for placing clusterheads. Thus, a location detection system can be optimized over possible clusterhead positions. Note that previous works considered clusterhead (base-station) positions to be given [11], [15]. Thus, optimization along this line has hitherto been unexplored.

The problem of designing a location detection system boils down to choosing clusterhead positions so that the ensemble of conditional distributions improves the system performance.[1] Consider the case when the possible user locations are finite, i.e., $|\mathcal{X}| = N$ for some N. Moreover, assume that we have a sequence of observations forming an observation vector $\mathbf{Y} = (\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n)$. Suppose now that the system decodes the user position to be $\hat{\mathbf{X}}$ using the observation $\mathbf{Y}$. Then a plausible design is to choose the clusterhead placement that minimizes the probability of error, $\mathbf{P}[\mathbf{X} \neq \hat{\mathbf{X}}]$.

This approach is however not practical since the probability of error depends on the the distribution of $\mathbf{X}$, which is usually not known in practice. Moreover, the probability of error also depends on the number of observations, n. It may happen that a clusterhead placement that is optimal for small n is not so for large n. The probability of error is therefore not an intrinsic property of the system.

We resort to information theory to measure the intrinsic ability of a system to distinguish between user locations. Intuitively, if two conditional distributions are "alike", then the system cannot easily distinguish between them. Thus, the conditional distributions should be "disparate" in a well designed system. We use *Chernoff distances* between distributions to make this notion precise. The most intuitive definition of Chernoff distance between two distributions is in terms of their Kullback–Leibler (KL) distance (or KL-divergence, or relative entropy) [16]. Recall that the KL distance between two densities $p_1(\cdot) = p(\cdot|\mathbf{X} = \mathbf{x}_1)$ and $p_0(\cdot) = p(\cdot|\mathbf{X} = \mathbf{x}_0)$ is given by

$$\mathcal{D}\left(p_1(\cdot)\|p_0(\cdot)\right) \triangleq \int_{\Re^n} p_1(\mathbf{y}) \log\left[\frac{p_1(\mathbf{y})}{p_0(\mathbf{y})}\right] d\mathbf{y}.$$

We do not use the KL distance since it is not symmetrical. The design of the location detection system, on the other hand, should be symmetric; reordering the user locations does not change anything physically.

Now consider the manifold defined by the distributions of the form

$$p_\lambda(\cdot) = \frac{p_0^\lambda(\cdot)p_1^{1-\lambda}(\cdot)}{\int p_0^\lambda(y)p_1^{1-\lambda}(y)dy}.$$

The reader can visualize it as a curve connecting $p_1(\cdot)$ and $p_0(\cdot)$ parameterized by λ. Let $\lambda = \lambda^*$ correspond to the density on the manifold satisfying

$$\mathcal{D}\left(p_{\lambda^*}(\cdot)\|p_0(\cdot)\right) = \mathcal{D}\left(p_{\lambda^*}(\cdot)\|p_1(\cdot)\right).$$

Intuitively, $p_{\lambda^*}(\cdot)$ is midway between the densities $p_1(\cdot)$ and $p_0(\cdot)$ in the KL distance sense. Then, the Chernoff distance between the densities $p_1(\cdot)$ and $p_0(\cdot)$ is defined as

$$d(p_0, p_1) \triangleq \mathcal{D}\left(p_{\lambda^*}(\cdot)\|p_0(\cdot)\right). \tag{1}$$

The Chernoff distance measures the separation between these densities. It is nonnegative and vanishes if and only if the densities are equal. Moreover, the Chernoff distance is symmetric.

In this paper, we propose to design a location detection system such that the worst case Chernoff distance between the conditional densities is maximized. It turns out that the asymptotic (as $n \to \infty$) probability of error decreases exponentially. The decay rate is intrinsic to the system in the sense that it is independent of the distribution of $\mathbf{X}$ as well as n. It only depends on the worst case Chernoff distance. Thus, this approach optimizes the inherent ability of the system to distinguish between user locations.

We consider the case where the possible user locations as well as the possible clusterhead positions are finite. The corresponding location detection system design problem is combinatorial. We formulate the clusterhead placement as a linear integer programming problem with the objective of maximizing the worst case Chernoff distance between the conditional densities over all possible location pairs. Further, in this paper, we develop a special purpose algorithm for solving large instances efficiently. For the resulting placement we provide an asymptotic upper bound and a lower bound on the probability of error in location detection under quite general conditions.

The main contributions of this paper that substantially differ from other existing works are as follows. We

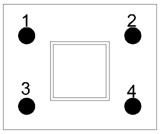
- pose the location detection problem in a rigorous statistical hypothesis testing framework;
- *characterize* system performance;
- *optimize* the intrinsic ability of the system by systematic clusterhead positioning and are able to solve large problem instances efficiently; and
- provide *performance guarantees* on the computed deployment plan.

The rest of this paper is organized as follows. We introduce our system model in Section II. The mathematical results underpinning our proposed framework are contained in Section III. Section IV discusses the clusterhead placement methodology: in particular, the mathematical programming formulation, a fast algorithm, and related performance bounds. We provide numerical results in Section V. Conclusions are in Section VI.

II. Overview of the System

In this section we introduce our system model and some of our notation. Consider a wireless sensor network to be deployed

[1] The other possibility is to favorably change the conditional distributions by designing the physical observable signal. Here we assume that the observable signal is given.



(a) Topology.

$$\begin{bmatrix} 100 & 10 & 10 & 2 \\ 10 & 100 & 1 & 12 \\ 10 & 2 & 67 & 14 \\ 3 & 10 & 10 & 120 \end{bmatrix}$$

(b) Mean Matrix.

$$\begin{bmatrix} 1 & 10 & 15 & 17 \\ 7 & 2 & 18 & 10 \\ 8 & 12 & 7 & 1 \\ 14 & 9 & 7 & 3 \end{bmatrix}$$

(c) Variance Matrix.

Clusterhead Positions	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$
Probability of Error	0	0	0.0011

Clusterhead Positions	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$
Probability of Error	0.0989	0	0

(d) Probability of error of various clusterhead placements.

Fig. 1. A simple example.

on a given site. The reader may assume the site to be the interior of a building. There are K available clusterheads. A clusterhead maybe placed at one of M distinct positions in the set $\mathcal{B} = \{B_1, B_2, \ldots, B_M\}$; each position B_i holds at most one clusterhead. A sensor node (e.g., a sensor on some mobile host) moves around in the building and the objective of the system is to resolve the position of the sensor. The sensor moves slow enough so that it can be considered static over the time-scale of the measurements. We assume that it is sufficient to resolve the position of the sensor to one of N distinct locations in the set $\mathcal{L} = \{L_1, L_2, \ldots, L_N\}$. The sets $\mathcal{L}$ and $\mathcal{B}$ are given as system requirements and they need not be distinct. In practice, B_i's could correspond to positions we have access to and L_i's could correspond to the rooms in the building.

Discretization of the system is also necessary for the data collection process. Reflections and occlusions make it virtually impossible to model received signal strength accurately by analytic functions in indoor environments. Computationally expensive methods like ray-tracing yield good approximations [17]; but they require extensive data about material properties, such as, reflection coefficients of surrounding walls. More importantly, signal strength and angle-of-arrival are inherently statistical due to movement of people, and reliable estimation of probability density functions calls for actual measurements over a discrete set of points.

As an aside, we note that trilateration/triangulation based location detection methods, such as [18], [19], are expected to perform poorly indoors, since the received signal-strength is heavily affected by reflections and occlusions [20]. A hypothesis testing framework is therefore the key for achieving an acceptable level of system performance.

We now continue our system description. Fig. 1(a) provides a simple example of a site. The floor shown in this figure is discretized into four positions. A sensor could be located anywhere on the floor; each of the four positions can hold one clusterhead. That is, $\mathcal{B} = \mathcal{L} = \{1, 2, 3, 4\}$, and $N = M = 4$. To simplify the description, we depict a 2-D site. However, our methodology handles 3-D sites equally well; in fact, we expect greater accuracy for 3-D sites since higher attenuation between locations separated by one or more floors provide wider variation in signal strength.

Suppose a sensor node is located at position $L \in \mathcal{L}$ and transmits a packet. Each clusterhead observes some physical quantities associated with the packet. Often, the observed quantities are signal-strength and angle-of-arrival. [2] However, our methodology applies to any set of physical observations.

Let $\mathbf{y}^{(i)}$ denote the vector of observations made by a clusterhead at position B_i. These observations are random quantities; $\mathbf{Y}^{(i)}$ denotes the corresponding random variable. A series of n consecutive observations is denoted by $\mathbf{y}_1^{(i)}, \mathbf{y}_2^{(i)}, \ldots, \mathbf{y}_n^{(i)}$. The conditional probability density (pdf) of the observation $\mathbf{Y}^{(i)}$ given that $L = L_j$ is denoted by $p_{\mathbf{Y}^{(i)}|L_j}(\mathbf{y}|L_j)$. We assume that the set of probability density functions for each distinct pair (B_i, L_j) is known to the system. Our methodology does not impose any constraint on the conditional pdfs; they could be arbitrary.

We assume that the sequence of consecutive observations made by a clusterhead contains a subsequence, $\{\mathbf{y}_m^{(i)}\}$, that is independent and identically distributed (i.i.d.) conditioned on the position of the sensor node. This assumption is justified when there are enough movements in the site so that the lengths of various radio-paths between the receiver and the transmitter change on the order of a wavelength between consecutive observations. For example, if a wireless sensor network operates at the 900-MHz ISM band, then the half-wavelength is only about 17 cm, and body movements of the user may alone cause observations separated in time by a few seconds to be i.i.d. The observations made by different clusterheads at the *same* instant need not be independent.

As discussed later on, our system uses an optimal hypothesis testing scheme to distinguish between locations $L \in \mathcal{L}$. Each clusterhead first makes n observations. The system then aggregates these nK observations and uses the known conditional density functions to choose the optimal hypothesis based on a maximum *a posteriori* probability (MAP) rule.

A. A Simple Example

Let us observe the result of placing $K = 2$ clusterheads in various positions in the small topology shown in Fig. 1(a). We simulate the system assuming the conditional distribution of the signal strength (in dB) to be Gaussian distributed. The mean and the variance of the distribution when the transmitter is placed at position i and the receiver at position j are given by the ij-th element of the mean and variance matrices shown in Fig. 1(b) and (c), respectively. Details of our simulation settings are described later in Section V. The probability of error observed in our simulations for all 6 possible placements is shown in Fig. 1(d). The results show that if the clusterheads are placed diagonally, then the location detection system performs poorly; however, when the two clusterheads are placed along any edge, the probability of error becomes negligible. An intuitive explanation is as follows. If the clusterheads are placed along a diagonal, say at positions 1 and 4, then both clusterheads are placed symmetrically with respect to positions 2 and 3, and thus, neither of them can reliably distinguish between these positions, which degrades the system performance. Therefore, symmetrical clusterhead placement is not conducive to accurate location detection. From a coverage point of view, however, diagonal placement

[2] The case where a clusterhead does not receive a packet can also be handled by assigning a pre-determined value to the observed quantities; for instance, the signal strength can be assigned a value below the receiver sensitivity.

of the clusterheads is better. Therefore, clusterhead placements that optimize system coverage are not necessarily optimal with regard to location detection. This point will emerge again when we describe our simulation of a realistic system in Section V.

III. Mathematical Foundation

In this section, we take the clusterhead locations as given and formulate the hypothesis testing problem for determining the location of sensors. We also review some results on binary hypothesis testing that will be useful later on in assessing the performance of our proposed clusterhead placement.

Suppose we place clusterheads in K out of the M available positions in $\mathcal{B}$. Without loss of generality let these positions be $B_1, \ldots, B_K$. Suppose also that a sensor is in some location $L \in \mathcal{L}$ and transmitting packets. As before, let $\mathbf{y}^{(i)}$ be the vector of observations at each clusterhead $i = 1, \ldots, K$; we write $\mathbf{y} = (\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(K)})$ for the vector of observations at all K clusterheads. These observations are random; let $\mathbf{Y}$ denote the random variable corresponding to $\mathbf{y}$ and $p_{\mathbf{Y}|L_j}(\mathbf{y}|L_j)$ the pdf of $\mathbf{Y}$ conditional on the sensor being in location $L_j \in \mathcal{L}$. Observations $\mathbf{y}^{(i)}$ and $\mathbf{y}^{(j)}$ made at the same instant need not be independent. If they are, however, it follows that

$$p_{\mathbf{Y}|L_j}(\mathbf{y}|L_j) = p_{\mathbf{Y}^{(1)}|L_j}(\mathbf{y}^{(1)}|L_j) \cdots p_{\mathbf{Y}^{(K)}|L_j}(\mathbf{y}^{(K)}|L_j).$$

Suppose that the clusterheads make n consecutive observations $\mathbf{y}_1, \ldots, \mathbf{y}_n$, which are assumed i.i.d. Based on these observations we want to determine the location L of the sensor.

The problem at hand is a standard N-ary hypothesis testing problem. It is known that the MAP rule is *optimal* in the sense of minimizing the probability of error. (Henceforth, the term optimality should be interpreted as minimization of the probability of error.) More specifically, we declare $L = L_j$ if

$$j = \underset{i=1,\ldots,N}{\arg\max} \left[\pi_i p_{\mathbf{Y}|L_i}(\mathbf{y}_1|L_i) \cdots p_{\mathbf{Y}|L_i}(\mathbf{y}_n|L_i)\right] \quad (2)$$

(ties are broken arbitrarily), where π_i denotes the prior probability that the sensor is in location L_i.

Next we turn our attention to binary hypothesis testing for which tight asymptotic results on the probability of error are available. These results will be useful in establishing performance guarantees for our proposed clusterhead placement later on.

Suppose now that the sensor's position is either L_i or L_j. A clusterhead located at B_k makes n i.i.d. observations $\mathbf{y}_1^{(k)}, \ldots, \mathbf{y}_n^{(k)}$. Let

$$X_{ijk}(\mathbf{y}) = \log \left[p_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}|L_i) / p_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}|L_j)\right]$$

be the log-likelihood ratio. Define

$$\Lambda_{ijk}(\lambda) \triangleq \log \mathbf{E}_{L_j}\left[e^{\lambda X_{ijk}(\mathbf{y})}\right] \quad (3)$$

where the expectation is taken with respect to the density $p_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}|L_j)$. It follows that

$$\Lambda_{ijk}(\lambda) = \log \int_{-\infty}^{\infty} p^{\lambda}_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}|L_i) p^{1-\lambda}_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}|L_j)\, d\mathbf{y}. \quad (4)$$

The function $\Lambda_{ijk}(\lambda)$ is a log-moment generating function of the random variable X_{ijk}, hence convex (see [21, Lemma 2.2.5] for a proof). Let d_{ijk} be the Fenchel-Legendre transform (or convex dual) of $\Lambda_{ijk}(\lambda)$ evaluated at zero, i.e.,

$$d_{ijk} \triangleq \sup_{\lambda \in [0,1]} \left[-\Lambda_{ijk}(\lambda)\right]. \quad (5)$$

This quantity is known as the Chernoff distance[3] between the densities $p_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}|L_i)$ and $p_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}|L_j)$ ([22], [21, § 3.4]). The definitions (1) and (5) are equivalent [16], although (5) is easier to handle computationally.

Consider next the probability of error in this binary hypothesis testing problem when we only use the observations made by clusterhead B_k. Suppose we make decisions optimally and let $\mathcal{S}^n$ denote the optimal decision rule (i.e., a mapping of $\mathbf{y}_1^{(k)}, \ldots, \mathbf{y}_n^{(k)}$ onto either "accept L_i" or "accept L_j"). We have two types of errors with probabilities

$$\alpha_n \triangleq \mathbf{P}_{L_i}[\mathcal{S}^n \text{ rejects } L_i], \;\; \beta_n \triangleq \mathbf{P}_{L_j}[\mathcal{S}^n \text{ rejects } L_j]. \quad (6)$$

The first probability is evaluated under $p_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}|L_i)$ and the second under $p_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}|L_j)$. The probability of error, $P_n^{(e)}$, of the rule $\mathcal{S}^n$ is simply $P_n^{(e)} = \pi_i \alpha_n + \pi_j \beta_n$. Large deviations asymptotes for the probability of error under the optimal rule $\mathcal{S}^n$ have been established by Chernoff [22], [21, Corollary 3.4.6] and are summarized in the following theorem.

Theorem III.1 (Chernoff's Bound): If $0 < \pi_i < 1$, then

$$\begin{aligned}\lim_{n\to\infty} \frac{1}{n} \log \alpha_n &= \lim_{n\to\infty} \frac{1}{n} \log \beta_n \\ &= \liminf_{n\to\infty} \frac{1}{n} \log P_n^{(e)} = -d_{ijk}.\end{aligned}$$

In other words, all these probabilities approach zero exponentially fast as n grows and the exponential decay rate equals the Chernoff distance d_{ijk}. Intuitively, these probabilities behave as $f(n)e^{-nd_{ijk}}$ for sufficiently large n, where $f(n)$ is a slowly growing function in the sense that $\lim_{n\to\infty}(\log f(n))/n = 0$. Moreover, when Maximum Likelihood (ML) rule is optimal (i.e., prior probabilities of the hypotheses are equal), we have the following.

Theorem III.2: Suppose $\pi_i = 1/N$ for all i. Then $P_n^{(e)} \leq e^{-nd_{ijk}}$ for all n.

The proof is given in Appendix III. Note the interesting fact that Chernoff distances, and thus the exponents of the probability of errors do not depend on the priors π_i.

The Chernoff distances between the joint densities of the data observed by all the clusterheads can be defined similarly by replacing $p_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}|L_j)$ and $p_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}|L_i)$ by $p_{\mathbf{Y}|L_j}(\mathbf{y}|L_j)$ and $p_{\mathbf{Y}|L_i}(\mathbf{y}|L_i)$, respectively. However, computation of the clusterhead placement that optimizes the Chernoff distances between the joint densities turns out to be a nonlinear problem with integral constraints. It quickly becomes intractable with increasing problem size and the optimum clusterhead placement for realistic sites cannot be computed using such a formulation. Optimization of the clusterhead placement in our formulation, on the other hand, reduces to a linear optimization

[3] It is nonnegative, symmetric, but it does not satisfy the triangle inequality.

$$\max \quad \epsilon \tag{7}$$

$$\text{s.t.} \quad \sum_{k=1}^{M} x_k = K, \tag{8}$$

$$\sum_{k=1}^{M} y_{ijk} = 1, \quad i,j = 1,\ldots,N,\ i<j, \tag{9}$$

$$y_{ijk} \le x_k, \quad \forall i,j, i<j, k=1,\ldots,M, \tag{10}$$

$$\epsilon \le \sum_{k=1}^{M} d_{ijk}\ y_{ijk}, \quad \forall\ i,j,\ i<j, \tag{11}$$

$$y_{ijk} \ge 0, \quad \forall\ i,j,\ i<j, \forall k, \tag{12}$$

$$x_k \in \{0,1\}, \quad k=1,\ldots,M. \tag{13}$$

Fig. 2. Mixed linear integer programming formulation.

problem (although still with integral constraints), for which large problem instances can be solved within reasonable time. For the resultant placement, we will be able to derive bounds on the probability of error of the decision rule that uses joint distributions. We will later see that the computed placement performs near-optimal in our simulation environment.

A couple of remarks on how various Chernoff distances can be computed are in order. First conditional densities, $p_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}|L_i)$ and $p_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}|L_j)$ are estimated using training data. In some cases, the density functions may also be modeled as parametrized densities, such as the Gaussian density, and the corresponding parameters—mean and variance for the Gaussian case—can be measured. The integration in (4) and the minimization in (5) can be done numerically; the latter one using standard methods, such as steepest descent, since the log-moment generating function is convex. For the important special case of Gaussian conditional densities analytical results can be obtained; we derive them in Appendix II. Note that, although $N \times M$ measurements are needed to compute the conditional densities, these measurements can be made in parallel by using multiple clusterheads and sensors.

IV. Clusterhead Placement Methodology

We next discuss how to place clusterheads to facilitate location detection. We place clusterheads so that for any pair of locations, at least one clusterhead can distinguish between them with an error exponent greater or equal to ϵ, and then maximize ϵ.

A. MIP Formulation

We formulate the clusterhead placement problem as a mixed integer linear programming problem (MIP). The formulation is in Fig. 2. The decision variables are x_k, y_{ijk}, and ϵ ($k = 1,\ldots,M, i,j = 1,\ldots,N, i<j$). x_k is the indicator function of placing a clusterhead at location B_k; i.e., $x_k = 1$ indicates that a clusterhead is placed at location B_k and $x_k = 0$ suggests otherwise. The variables y_{ijk} and ϵ are *not* constrained to take integer values. Equation (8) in Fig. 2 represents the constraint that K clusterheads are to be placed. Let x_k^*, y_{ijk}^*, and ϵ^* ($k = 1,\ldots,M, i,j = 1,\ldots,N, i<j$) be an optimal solution of this MIP. The locations where clusterheads are to be placed are in the set $\mathcal{Y}^* \triangleq \{B_k : x_k^* = 1\}$. Although this problem is NP-hard (proved in Appendix I), it can be solved efficiently for sites with more than 100 locations by using a special purpose algorithm proposed in Section IV-B.

The next proposition establishes a useful property for the optimal solution and value of the MIP in Fig. 2. In preparation for that result consider an arbitrary placement of K clusterheads. More specifically, let $\mathcal{Y}$ be any subset of the set of potential clusterhead positions $\mathcal{B}$ with cardinality K. Let $\mathbf{x}(\mathcal{Y}) = (x_1(\mathcal{Y}),\ldots,x_M(\mathcal{Y}))$ where $x_k(\mathcal{Y})$ is the indicator function of B_k being in $\mathcal{Y}$. Define

$$\epsilon(\mathcal{Y}) = \min_{\substack{i,j=1,\ldots,N\\ i<j}} \max_{k:x_k(\mathcal{Y})=1} d_{ijk}. \tag{14}$$

We can interpret $\max_{k:x_k(\mathcal{Y})=1} d_{ijk}$ as the best decay rate for the probability of error in distinguishing between locations L_i and L_j from some clusterhead in $\mathcal{Y}$. Then $\epsilon(\mathcal{Y})$ is simply the worst pairwise decay rate.

Proposition IV.1: For any clusterhead placement $\mathcal{Y}$ we have

$$\epsilon^* \ge \epsilon(\mathcal{Y}). \tag{15}$$

Moreover, the selected placement achieves equality; i.e.

$$\epsilon^* = \min_{\substack{i,j=1,\ldots,N\\ i<j}} \max_{k:x_k^*=1} d_{ijk}. \tag{16}$$

Proof: Consider the placement $\mathcal{Y}$ and let

$$y_{ijk} = \begin{cases} 1, & \text{if } k = \underset{k:x_k(\mathcal{Y})=1}{\arg\max}\ d_{ijk}, \\ 0, & \text{otherwise,} \end{cases} \qquad \forall i,j; i<j, \forall k.$$

If more than one of the y_{ijk}'s is 1 for a given pair (i,j), we arbitrarily set all but one of them to 0 to satisfy Eq. (9) in Fig. 2. Then

$$\min_{\substack{i,j\\ i<j}} \sum_{k=1}^{M} d_{ijk} y_{ijk} = \min_{\substack{i,j\\ i<j}} \max_{k:x_k(\mathcal{Y})=1} d_{ijk} = \epsilon(\mathcal{Y}).$$

Observe that $\mathbf{x}(\mathcal{Y})$, y_{ijk}'s (as defined above), and $\epsilon(\mathcal{Y})$ form a feasible solution of the MIP in Fig. 2. Clearly, the value of this feasible solution can be no more than the optimal ϵ^*, which establishes (15).

Next note that (11) in Fig. 2 is the only constraint on ϵ. So, we have

$$\epsilon^* = \min_{\substack{i,j\\ i<j}} \sum_{k=1}^{M} d_{ijk} y_{ijk}^* = \min_{\substack{i,j\\ i<j}} \sum_{k:x_k^*=1} d_{ijk} y_{ijk}^*. \tag{17}$$

The second equality is due to (10) in Fig. 2. The final observation is that the right-hand side of the above is maximized when

$$y_{ijk}^* = \begin{cases} 1, & \text{if } k = \underset{k:x_k^*=1}{\arg\max}\, d_{ijk}, \\ 0, & \text{otherwise,} \end{cases} \qquad \forall i,j, i<j, \forall k.$$

(Again, at most one y_{ijk}^* is set to 1 for a given (i,j) pair.) Thus, an optimal solution satisfies the above. This, along with (17) establishes (16). □

$$
\begin{aligned}
\min\quad & \pi && (19)\\
\text{s. t.}\quad & \sum_{k=1}^{M} s_k = K, && (20)\\
& \sum_{k=1}^{M} t_{ijk} = 1, \quad \forall i,j = 1,\ldots,N, i<j, && (21)\\
& t_{ijk} \le s_k, \quad \forall i<j,\ k=1,\ldots,M, && (22)\\
& \pi \ge \sum_{k=1}^{M} b_{ijk} t_{ijk}, \quad \forall i,j,i<j, && (23)\\
& t_{ijk} \ge 0, \quad \forall i,j,k,i<j, && (24)\\
& s_k \in \{0,1\}, \quad k=1,\ldots,M && (25)
\end{aligned}
$$

Fig. 3. Equivalent formulation of the MIP of Fig. 2.

As before, $\max_{k:x_k^*=1} d_{ijk}$ is the best decay rate for the probability of error in distinguishing between locations L_i and L_j from some clusterhead in the set $\mathcal{Y}^*$. Then ϵ^* is simply the worst such decay rate over all pairs of locations. Moreover, according to Proposition IV.1, this worst decay rate is no worse than the corresponding quantity $\epsilon(\mathcal{Y})$ achieved by any other clusterhead placement $\mathcal{Y}$.

B. Efficient Computation of the Proposed MIP

In this section, we propose an algorithm that solves the MIP presented in Fig. 2 faster than a general purpose MIP-solver such as CPLEX [23]. Our approach is to construct an alternate formulation of the proposed MIP first, and then solve it using an iterative algorithm. The computational advantage of this approach lies in the fact that we solve a *feasibility* problem in each iteration that contains only $O(M)$ variables and $O(N^2)$ constraints instead of $O(N^2M)$ variables and $O(N^2M)$ constraints that appear in the formulation in Fig. 2, and thus can be solved much faster.

1) Alternate Formulation: Let us sort the Chernoff distances, d_{ijk}'s, in nonincreasing order, and let b_{ijk} denote the index of d_{ijk}. We let equal distances have the same index. Note that b_{ijk} is a positive integer upper bounded by $MN(N-1)/2$.

Now consider the MIP problem shown in Fig. 3. This problem is actually the MIP formulation of the vertex K-center problem [24]. The following proposition establishes that the formulations of Figs. 2 and 3 are indeed equivalent.

Proposition IV.2: Suppose $(\mathbf{s}^*, \mathbf{t}^*, \pi^*)$ is an optimal solution to the problem in Fig. 3. Then $(\mathbf{x}^* = \mathbf{s}^*, \mathbf{y}^* = \mathbf{t}^*, \epsilon^* = d_{i^*j^*k^*})$ is an optimal solution to the MIP problem in Fig. 2, where (i^*, j^*, k^*) is such that $b_{i^*j^*k^*} = \pi^*$.

Proof: A proof analogous to the one of Proposition IV.1 establishes

$$\pi^* = \max_{\substack{i,j\\ i<j}} \min_{k:s_k^*=1} b_{ijk}. \tag{18}$$

Let (i^*, j^*, k^*) be such that $\pi^* = b_{i^*j^*k^*}$. Then, $(\mathbf{x}^*, \mathbf{y}^*, \epsilon^*) = (\mathbf{s}^*, \mathbf{t}^*, d_{i^*j^*k^*})$ is an optimal solution of the MIP problem in Fig. 2. To that end, observe that $\mathbf{x}^*$, $\mathbf{y}^*$ satisfy constraints (8)–(10), (12) and (13) in Fig. 2. Moreover, since b_{ijk} was defined as the index of d_{ijk}, (18) and (16) imply the optimality of $(\mathbf{x}^*, \mathbf{y}^*, \epsilon^*)$; namely, the min-max of the d_{ijk}'s is equivalent to the max-min of their rank. □

$$
\begin{aligned}
\max\quad & 0 && (26)\\
\text{s.t.}\quad & \sum_{k=1}^{M} c_{ijk} w_k \ge 1,\ \forall i,j = 1,\ldots,N, i<j, && (27)\\
& \sum_{k=1}^{M} w_k = K, && (28)\\
& w_k \in \{0,1\}, \quad \forall k=1,\ldots,M, && (29)
\end{aligned}
$$

Fig. 4. The feasibility problem. c_{ijk}'s are defined by (30).

```
1) Set $\theta = 1$.
2) Determine $c_{ijk}, \forall i,j,k$ and solve the problem shown in Fig. 4.
    a) If the problem of Fig. 4 is infeasible, set $\theta := \theta + 1$ and go to step 2.
    b) Else, if the problem is feasible, stop.
```

Fig. 5. Iterative feasibility algorithm.

We remark that it is also true that there is a corresponding optimal solution to the problem of Fig. 3 for every optimal solution to the problem of Fig. 2.

2) Iterative Algorithm: Proposition IV.2 allows us to solve the problem of Fig. 3 instead of the problem of Fig. 2. So we will concentrate on the former. Our approach is to solve this problem by an iterative *feasibility* algorithm along the lines proposed in [24]. In particular, we use a slightly modified version of a *two-phase* algorithm proposed in [25], [26].

The core idea of the iterative algorithm is to solve the feasibility problem shown in Fig. 4. The problem of Fig. 4 depends on a parameter θ by the following equation:

$$c_{ijk} = \begin{cases} 1, & \text{if } b_{ijk} \le \theta \\ 0, & \text{otherwise.} \end{cases} \tag{30}$$

Intuitively, θ represents the index of some Chernoff distance in the nonincreasingly sorted list that we initially created, and the feasibility problem checks whether all pairs of locations can be distinguished (by at least one clusterhead) with an error exponent greater or equal to the Chernoff distance pointed by θ. If not, θ is increased, which means that it now points to a smaller Chernoff distance, and the process is repeated. At termination, θ, which corresponds to the largest feasible Chernoff distance, provides the optimal value of the problem in Fig. 3. The formal iterative algorithm is shown in Fig. 5. It is clear that this algorithm terminates in a finite number of steps. In particular, if $\theta = MN(N-1)/2$, we see that $c_{ijk} = 1$ for all i,j,k, and the feasibility conditions of problem of Fig. 4 are trivially satisfied. Next we show that at termination, we obtain the optimal solution to the problem of Fig. 3.

Proposition IV.3: Let θ^* be the value of θ when the algorithm of Fig. 5 terminates and $\mathbf{w}^*$ the optimal solution to problem of Fig. 4 at the last iteration. Then $\mathbf{s}^* = \mathbf{w}^*$ induces an optimal solution to the problem of Fig. 3 with optimal objective function value $\pi^* = \theta^*$.

Proof: First note that at the last iteration, for any i,j ($i<j$), there exists at least one k such that $b_{ijk} \le \theta^*$ with $w_k^* = 1$; otherwise the problem is infeasible. Next we construct a feasible solution $(\mathbf{s}^*, \mathbf{t}^*, \pi^*)$ to the problem of Fig. 3 as follows. Given

```
1) (LP phase) Set $l := 1$, $u := \frac{MN(N-1)}{2}$.
2) Let $\theta := \lceil \frac{l+u}{2} \rceil$, determine $c_{ijk}$ $\forall i, j, k$ and solve the relaxed version of the problem of Fig. 4.
    a) If the relaxed version of the problem of Fig. 4 is feasible, set $u := \theta$.
    b) Else, set $l := \theta$.
3) If $u - l \leq 1$, go to step 4. Else go to step 2.
4) (IP phase) If the relaxed version of the problem of Fig. 4 is feasible, set $\theta := l$, else set $\theta := u$. Then set $l := \theta, u := \frac{MN(N-1)}{2}$.
5) Determine $c_{ijk}, \forall i, j, k$ and solve the problem of Fig. 4.
    a) If the problem of Fig. 4 is infeasible, set $l := \theta, \theta := \lceil \frac{l+u}{2} \rceil$ and go to step 5.
    b) Else, if the problem is feasible
        i) If $\theta \neq u$, set $u := \theta, \theta := \lceil \frac{l+u}{2} \rceil$ and go to step 5.
        ii) Otherwise, stop.
```

Fig. 6. Two-phase iterative feasibility algorithm.

a pair (i, j), we select one k such that $b_{ijk} \leq \theta^*$ and $w_k^* = 1$. Then we set $t_{ijk}^* = 1$ and $t_{ijl}^* = 0$ for any $l \neq k$. We repeat this process for all pairs (i, j). Finally, we set $\mathbf{s}^* = \mathbf{w}^*$ and $\pi^* = \theta^*$. Then, the triplet $(\mathbf{s}^*, \mathbf{t}^*, \pi^*)$ satisfies all the constraints of the problem of Fig. 3 and is therefore a feasible solution.

Next we prove the optimality of $(\mathbf{s}^*, \mathbf{t}^*, \pi^*)$ by contradiction. Suppose that there exists a feasible solution $(\tilde{\mathbf{s}}, \tilde{\mathbf{t}}, \tilde{\pi})$ to the problem of Fig. 3 such that $\tilde{\pi} < \pi^*$. Then according to the algorithm in Fig. 5, there is a step where $\theta = \tilde{\pi}$. This implies that the corresponding problem of Fig. 4 was infeasible; otherwise the algorithm would not have increased the value of θ beyond $\tilde{\pi}$. However, since $(\tilde{\mathbf{s}}, \tilde{\mathbf{t}}, \tilde{\pi})$ is feasible for the problem of Fig. 3 we have

$$\tilde{\pi} \geq \sum_{k:\tilde{s}_k=1} b_{ijk} \tilde{t}_{ijk}, \quad \forall i, j, i < j$$

which implies that for all (i, j) with $i < j$ there exists at least one k with $\tilde{s}_k = 1$ such that $b_{ijk} \leq \tilde{\pi}$. Hence, $\mathbf{w} = \tilde{\mathbf{s}}$ is feasible for the problem of Fig. 4 when $\theta = \tilde{\pi}$. We arrived at a contradiction. □

To expedite the convergence in the actual implementation, we use the two-phase feasibility algorithm shown in Fig. 6, which is a modified version of the algorithm proposed in [25]. The two-phase algorithm consists of two parts. In the first part (*LP phase*), we construct the linear programming (LP) relaxation of the problem of Fig. 4 by replacing the binary constraint (29) in Fig. 4 by

$$w_k \in [0, 1], \quad \forall k = 1, \ldots, M. \tag{31}$$

Then we solve the relaxed problem and compute the smallest integer $\theta = \theta_0$ such that the LP relaxation is feasible. In the second part (*IP phase*), we compute θ^* by executing the iterative algorithm starting from $\theta = \theta_0$, but this time we solve the integer programming problem instead of the LP relaxation. The important difference between our algorithm and the algorithm proposed in [25], [26] is that we employ binary search both in the LP-phase and IP-phase whereas the authors of [25], [26] use linear search in the LP-phase. Our use of binary search in the IP-phase further decreases the computation time for large problem instances almost by an order of magnitude.

C. Performance Guarantees

1) Upper Bound on Probability of Error: The optimal value ϵ^* of the MIP in Fig. 2 can be used to provide an upper bound on the probability of error achieved by the corresponding clusterhead placement $\mathcal{Y}^*$.

Suppose we place clusterheads according to $\mathcal{Y}^*$. As discussed in Section III the optimal location detection device uses the MAP rule to accept one hypothesis out of N possibilities (cf. (2)). We adopt the notation of Section III with the only exception that now we let $\mathbf{y}^{(i)}$ denote the vector of observations at the ith element of $\mathcal{Y}^*$, for $i = 1, \ldots, K$. We write $\mathbf{y} = (\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(K)})$. We use $P_n^{(e),\mathrm{opt}}$ to denote the probability of error under the optimal rule based on n consecutive i.i.d. observations $\mathbf{y}_1, \ldots, \mathbf{y}_n$. Let $E_n(i, j)$ denote the event that

$$\pi_i p_{\mathbf{Y}|L_i}(\mathbf{y}_1|L_i) \cdots p_{\mathbf{Y}|L_i}(\mathbf{y}_n|L_i) \leq \pi_j p_{\mathbf{Y}|L_j}(\mathbf{y}_1|L_j) \cdots p_{\mathbf{Y}|L_j}(\mathbf{y}_n|L_j).$$

It follows that

$$\begin{aligned} P_n^{(e),\mathrm{opt}} &= \sum_{i=1}^{N} \pi_i \mathbf{P}[\cup_{j \neq i} E_n(i, j)] \\ &\leq \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} \pi_i \mathbf{P}[E_n(i, j)]. \end{aligned} \tag{32}$$

The last inequality is due to the union bound.

Next note that $\mathbf{P}[E_n(i, j)]$ has the same exponential decay rate as the probability of error α_n (cf. (6)) in a binary hypothesis testing problem that seeks to distinguish between locations L_i and L_j. Furthermore, this latter probability can be no larger than the probability of error achieved when we discard most of the observation vector $\mathbf{y}$ and use only the observations $\mathbf{y}^{(k^*)}$ made at a single clusterhead k^*, where $k^* = \arg\max_{k:x_k^*=1} d_{ijk}$ (ties are broken arbitrarily). As a result, using Chernoff's bound (Theorem III.1), we obtain

$$\lim_{n\to\infty} \frac{1}{n} \log \mathbf{P}[E_n(i, j)] \leq - \max_{k:x_k^*=1} d_{ijk}. \tag{33}$$

To conclude our argument, notice that the right hand side of (32) is a finite sum of exponentials from which the one with the largest exponent dominates in the limit $n \to \infty$. Combining (32) and (33), we obtain

$$\limsup_{n\to\infty} \frac{1}{n} \log P_n^{(e),\mathrm{opt}} \leq - \min_{\substack{i,j \\ i<j}} \max_{k:x_k^*=1} d_{ijk} = -\epsilon^*. \tag{34}$$

The following proposition summarizes our conclusion.

Proposition IV.4: Let $\mathbf{x}^*$ be an optimal solution of the MIP in Fig. 2 with corresponding optimal value ϵ^* given in (16). Then the probability of error of the optimal location detection system under clusterhead placement $\mathcal{Y}^* \triangleq \{B_k : x_k^* = 1\}$ satisfies

$$\limsup_{n\to\infty} \frac{1}{n} \log P_n^{(e),\mathrm{opt}} \leq -\epsilon^*.$$

Step 1: Estimate conditional densities $p_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}|L_i)$ for all possible clusterhead and sensor node placements.
Step 2: Compute the numerical values of d_{ijk} for all possible cases using the densities estimated in Step 1.
Step 3: Determine the clusterhead placement by using the algorithm developed in Section IV-B.

Fig. 7. Proposed clusterhead placement framework.

More intuitively, $P_n^{(e),\text{opt}} \leq f(n)e^{-n\epsilon^*}$, for sufficiently large n, where $f(n)$ is some function satisfying

$$\lim_{n\to\infty} (\log f(n))/n = 0.$$

2) Lower Bound on Probability of Error: Recall that the system declares $L = L_j$ if

$$j = \arg\max_{i=1,\ldots,N} \left[\pi_i p_{\mathbf{Y}|L_i}(\mathbf{y}_1|L_i) \cdots p_{\mathbf{Y}|L_i}(\mathbf{y}_n|L_i)\right] \tag{35}$$

where $\mathbf{Y}$ is the *joint* observation vector at all clusterheads, i.e., the system uses the *joint* probability distributions for likelihood calculations. Therefore the asymptotic probability of error depends on the Chernoff distances among the joint conditional probability distributions. However, in order to linearize our MIP, we used the *marginal* Chernoff distances for our optimization problem. Thus, we need to infer the joint probability distributions from the marginals to provide a lower bound on the probability of error in location detection.

In general marginal distributions do not possess information about the joint distributions. But if the observations made at the clusterheads happen to be statistically independent, then the joint pdf is simply the product of the marginals. In this case we can provide a lower bound on the probability of error. Assuming independence, we have

$$p_{\mathbf{Y}|L_j}(\mathbf{y}|L_j) = p_{\mathbf{Y}^{(1)}|L_j}(\mathbf{y}^{(1)}|L_j) \cdots p_{\mathbf{Y}^{(K)}|L_j}(\mathbf{y}^{(K)}|L_j)$$

for all j. The joint log-likelihood ratio is

$$\begin{aligned} X_{ij}(\mathbf{y}) &= \log\left[p_{\mathbf{Y}|L_i}(\mathbf{y}|L_i)/p_{\mathbf{Y}|L_j}(\mathbf{y}|L_j)\right] \\ &= \sum_{k:x_k^*=1} \log\left[p_{\mathbf{Y}^{(k)}|L_i}(\mathbf{y}^{(k)}|L_i)/p_{\mathbf{Y}^{(k)}|L_j}(\mathbf{y}^{(k)}|L_j)\right] \\ &= \sum_{k:x_k^*=1} X_{ijk}(\mathbf{y}^{(k)}). \end{aligned}$$

Note that, in order to avoid more tedious notation, we do not explicitly show the dependence of $X_{ij}(\mathbf{y})$ on the placement of the clusterheads.

Correspondingly, we have

$$\Lambda_{ij}(\lambda) \triangleq \log \mathbf{E}_{L_j}\left[e^{\lambda X_{ij}(\mathbf{y})}\right] = \sum_{k:x_k^*=1} \Lambda_{ijk}(\lambda). \tag{36}$$

Consider the convex dual of $\Lambda_{ij}(\lambda)$,

$$\Lambda_{ij}^*(\alpha) \triangleq \sup_{\lambda\in[0,1]} \left[\lambda\alpha - \Lambda_{ij}(\lambda)\right], \tag{37}$$

and define $d_{ij} \triangleq \Lambda_{ij}^*(0)$. However, using (36)

$$\begin{aligned} \Lambda_{ij}^*(\alpha) &= \sup_{\lambda\in[0,1]} \left[\lambda\alpha - \sum_{k:x_k^*=1} \Lambda_{ijk}(\lambda)\right] \\ &= \inf_{\sum_{k:x_k^*=1} \alpha_k = \alpha} \sum_{k:x_k^*=1} \Lambda_{ijk}^*(\alpha_j) \end{aligned} \tag{38}$$

where $\Lambda_{ijk}^*(\alpha_k)$ is the convex dual of $\Lambda_{ijk}(\lambda)$ and the last equality is due to a convex duality property [27].

At this point one can compute $\Lambda_{ijk}^*(\alpha)$ and solve for $d_{ij} = \Lambda_{ij}^*(0)$ using (38). However, here we bound d_{ij} in terms of d_{ijk}'s using (38) and establish a connection with ϵ^*. In particular, since $\alpha_k = 0$ for all k form a feasible solution to the minimization problem that appears in (38), we get

$$\begin{aligned} d_{ij} = \Lambda_{ij}^*(0) &\leq \sum_{k:x_k^*=1} \Lambda_{ijk}^*(0) = \sum_{k:x_k^*=1} d_{ijk}, \\ \text{or } d_{ij} &\leq K \max_{k:x_k^*=1} d_{ijk} \\ \text{or } \min_{\substack{i,j \\ i<j}} d_{ij} &\leq K \min_{\substack{i,j \\ i<j}} \max_{k:x_k^*=1} d_{ijk} = K\epsilon^*. \end{aligned} \tag{39}$$

Now to compute a lower bound on the probability of error $P_n^{(e),\text{opt}}$, note that

$$P_n^{(e),\text{opt}} = \sum_{i=1}^{N} \pi_i \mathbf{P}[\cup_{j\neq i} E_n(i,j)] \geq \sum_{i=1}^{N} \pi_i \mathbf{P}[E_n(i,j)] \tag{40}$$

where j is any index not equal to i. From Theorem III.1, asymptotically each term in the right-hand side of (40) is of the form $f(n)e^{-nd_{ij}}$, thus, the term with the largest exponent will dominate the sum. Namely

$$\liminf_{n\to\infty} \frac{1}{n} \log P_n^{(e),\text{opt}} \geq -d_{ij}\ \forall\, i,j,\ i<j \tag{41}$$

$$\text{or } \liminf_{n\to\infty} \frac{1}{n} \log P_n^{(e),\text{opt}} \geq -\min_{\substack{i,j \\ i<j}} d_{ij} \tag{42}$$

$$\text{or } \liminf_{n\to\infty} \frac{1}{n} \log P_n^{(e),\text{opt}} \geq -K\epsilon^*. \tag{43}$$

The transition from (42) to (43) is due to (39).

In summary, our proposed clusterhead placement framework is a 3-step process outlined in Fig. 7. Proposition IV.4 provides a upper bound on probability of error on the computed placement. For the case of independent observations, (43) provides a lower bound on the probability of error.

D. Weighted Cost of Error Events

In the development above, we seek to minimize the probability of error. This criterion is appropriate when we treat all error events equally. However, in practice, some error events may have more impact than others. For example, in the case of the error event in which the system declares the position to be L_j while the true position is L_i, it is reasonable to treat this error less harmful if L_i and L_j are close-by than the case when L_j is located far from L_i. Hence, it is natural to generalize our setting for the case where each error has an associated cost C_{ji} and the objective is to minimize the expected cost.

At first, it seems that one needs to modify our design criteria to accommodate different costs. But it turns out that our design (and associated performance guarantees) remains valid without any modification even if arbitrary nonnegative costs are used as long as correct decisions have zero cost. This surprising conclusion is based on the results obtained in [28]. In particular, let $\mathbf{P}_{L_i}[\mathcal{S}^n \text{ chooses } L_j]$ be the probability the the optimum decision rule $\mathcal{S}^n$ declares hypothesis L_j when hypothesis L_i is true, C_{ji} the associated cost, and π_i the *a priori* probability of hypothesis L_i. Then

$$\lim_{n\to\infty} -\frac{1}{n}\log\left(\sum_i \sum_{j\neq i} \pi_i C_{ji} \mathbf{P}_{L_i}[\mathcal{S}^n \text{ chooses } L_j]\right) = \min_{\substack{s,t\\ s<t}} d_{st}. \quad (44)$$

Note that the argument of log in (44) is the average cost. That is, (44) shows that the average cost decreases exponentially with large n and the exponent is the minimum pairwise Chernoff distance. Since the ϵ^* computed in the MIP shown in Fig. 2 gives a lower bound on the minimum pairwise Chernoff distance, the performance guarantees obtained in Section IV-C remain valid.

E. Connectivity

In the formulation shown in Fig. 2, we have not imposed any *connectivity* constraint. This leaves open the possibility that all clusterheads would receive packets from one of the locations at a very low power. Let us now impose additional constraints to avoid such a case. In particular, let ζ_{ij} be the indicator function of whether the received signal strength at location B_j exceeds a threshold Θ when the transmitter is at location L_i. The parameter Θ can be chosen to meet quality requirements. Then, adding the constraint

$$\sum_{k=1}^{M} \zeta_{ik} x_k \geq p, \qquad \forall\, i = 1, 2, \ldots N \quad (45)$$

in the formulation of Fig. 2 ensures that each location is *covered* by at least p clusterheads. Similarly, the fast algorithm we developed in Section IV-B can be modified to include a connectivity constraint by adding the constraint

$$\sum_{k=1}^{M} \zeta_{ik} w_k \geq p, \qquad \forall\, i = 1, 2, \ldots, N \quad (46)$$

in the feasibility problem of Fig. 4. Note, however, these additional constraints may make the MIP infeasible for large Θ or p.

V. Numerical Results

There are two numerical aspects of our proposed methodology: i) the scalability of our proposed MIP, and ii) the quality of the clusterhead placement selected by our method. We have simulated realistic sites to evaluate both these aspects, which we discuss in this section.

A. Setup

Our simulation models the fourth floor of the Photonics Building at Boston University. Fig. 8(a) schematically represents the site showing only the primary walls. Clusterheads observe the received signal-strength of each packet. The conditional distribution of the received signal-strength, predicated on the location of the sensor, is assumed to be log-normal. Therefore, in dB scale, received signal-strengths are Gaussian distributed. Note that log-normal shadowing is widely used in the literature [29]. The transmit power is assumed to be held constant at 100 mW (20 dBm), and the path-loss exponent is assumed to be 3.5 uniformly. Moreover, each primary wall between the transmitter and the receiver is assumed to introduce an additional 3 dB loss. So, the mean received power μ (in dBm) at the receiver is calculated using the formula: $\mu = 20 - (40 + 10 \times 3.5 \times \log_{10}(d) + 3 \times n)$; here, d is the distance between the transmitter and the receiver (in meters), n is the number of walls separating them, and the term 40 corresponds to the power measured at a close-in reference distance. The standard deviation, σ, (in dBm) is given by $\sigma = 5 + 1 \times n$. The parameters used in the simulation are typical [29]. Consecutive observations are assumed to be i.i.d.

B. Scalability of the Proposed Integer Program

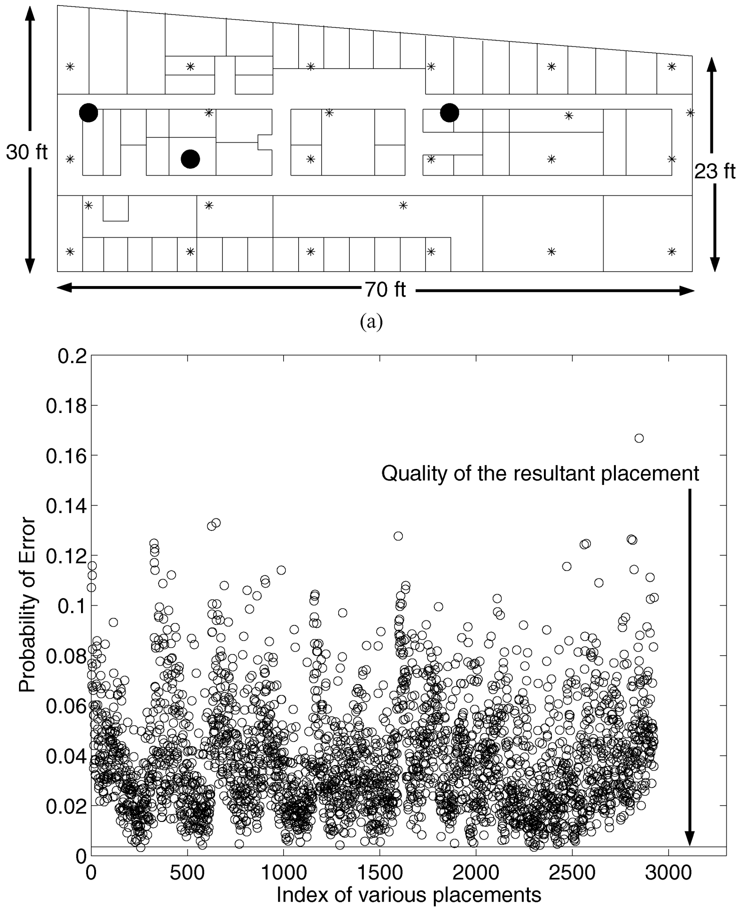
First we examine whether our proposed MIP can be used to solve realistic problems. We implemented our proposed MIP (shown in Fig. 2) in CPLEX. It is a commercial linear/quadratic program solver that also supports integer programming [23]. CPLEX solves MIP's using a branch-and-bound method, solving the LP relaxations by a dual-simplex algorithm at each step. Our fast algorithm (cf. Section IV-B) is implemented in C; the subproblems are solved using CPLEX. Both the programs were run on a computer with an Intel® Xeon™CPU running at 3.06 GHz, and 3.6 Gbytes of main memory; background processes were minimal.

In the site model shown in Fig. 8(a), a set of 100 points are chosen on a grid (not shown in the figure). From this set, N points are chosen randomly with uniform distribution; these constitute both $\mathcal{L}$ and $\mathcal{B}$. Therefore, the size of the problem is proportional to N. We let N increase from 3 with a stepsize of 1. For each pair of positions, the conditional densities and the associated Chernoff distances are calculated based on the model described in Section V-A. For each value of N, each program finds the best placement for $K = 3$ clusterheads.

Fig. 9 shows the run-time of both the approaches as a function of $|\mathcal{L}| = |\mathcal{B}| = N$. The nonmonotonicity of the plot arises from the fact that the graphs are random at each point, and that CPLEX uses various heuristics, which in some cases can find an optimal solution faster. From the figure, we see our proposed fast algorithm solved problems of size $N = 100$—a cardinality that can represent large sites—in less than 5 min. Within this time CPLEX can solve problems of size $N = 15$. In fact, within a time limit of 24 h (86400 s), CPLEX could only solve problems with size $N = 45$. So one can safely say that CPLEX would not be able to solve a problem of size $N = 100$ within a reasonable time-frame. This clearly demonstrates the necessity of the proposed fast algorithm.

C. Quality of the Selected Clusterhead Placement

As shown in Fig. 8(a), we selected 27 positions on the site so that no two positions are in the same room. Each of the selected positions can hold one clusterhead, and the sensor location



(a)

(b)

Fig. 8. Clusterhead placement using the proposed methodology in the fourth floor of the Photonics Building, Boston University. (a) Site layout. (b) Probability of error for various placements. (a) Site layout. (b) Probability of error for various placements.

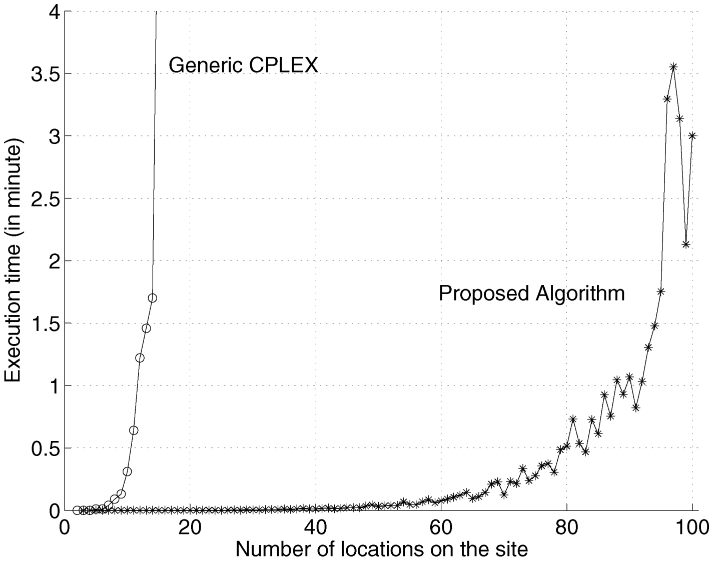


Fig. 9. Scalability of the proposed linear MIP problem.

needs to be resolved to one of them; i.e., these 27 points constitute both $\mathcal{L}$ and $\mathcal{B}$.

As described previously, Chernoff distances for all possible combinations are computed by assuming the conditional densities to be Gaussian with mean and variances according to the model described before. We use $N = 27$ in our proposed integer program to compute the best placement for $K = 3$ clusterheads. Computation time is about 0.52 s. The outcome is shown in Fig. 8(a) by solid circles. As expected, the clusterheads are not located symmetrically. This is because, as mentioned in Sec-

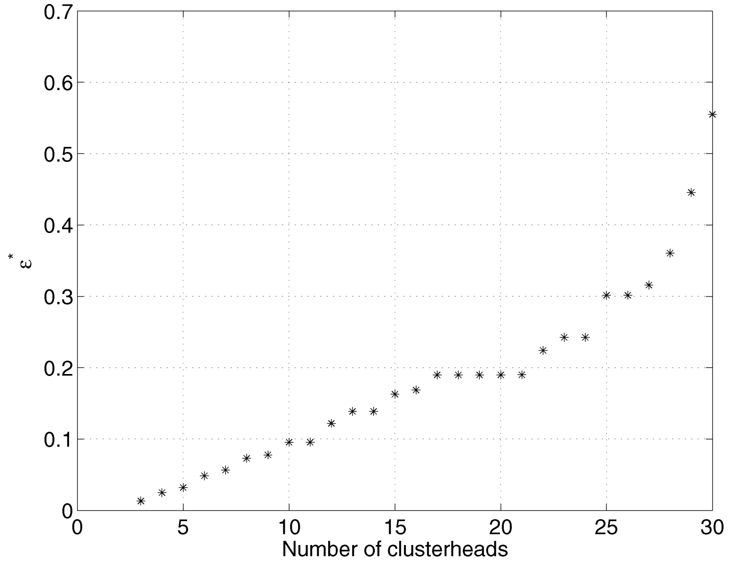


Fig. 10. Number of clusterheads versus $\epsilon^{\star}$.

tion II, positions that are located symmetrically to all clusterheads, are difficult to distinguish, and they degrade the performance of the system.

Now we evaluate the probability of error in location detection for all possible placements of three clusterheads in our site. There are in total $\binom{27}{3} = 2925$ possible placements. First, clusterheads are placed according to a given placement. Then, one position is chosen with uniform probability from $\mathcal{L}$, and the sensor node is placed at that position. Next, 10 observation samples are generated for each clusterhead based on the statistical model described above. (Assuming a 1-s sampling interval, this represents an observation period of only 10 s.) All these samples are considered i.i.d. Thus, in aggregate there are 30 observations. These 30 samples are collected together and maximum-likelihood decision about the location of the sensor is taken. Note that, since all sensor locations are equally likely, ML decision is also the MAP decision. For each given placement of clusterheads, the trial is repeated 10 000 times. The fraction of erroneous decisions then provides the probability of error in location detection associated with that placement of clusterheads.

Fig. 8(b) shows the probability of error, P_e, for all possible clusterhead placements. We find that P_e varies widely with clusterhead placement: P_e is as high as 16% for some placements, whereas there are placements which reduces P_e to less than 0.5%. This clearly demonstrates the need for intelligent clusterhead placement.

The horizontal line in Fig. 8(b) refers to $P_e = 0.0035$, the observed probability of error when clusterheads are positioned according to the placement selected by our proposed integer program. It is clear that the selected placement performs near optimal.

D. Evaluation of Design Alternatives

One of the important advantage of our framework is that it provides answers to many "what-if" questions that are often used for designing practical systems. Below we explore a few dimensions to illustrate the power of our methodology.

1) Number of Clusterheads: First, consider the effect of increasing K, the number of clusterheads. Clusterheads are usually costly. Thus it is important to understand whether increasing K brings about a significant performance gain. We illustrate the tradeoff for the site shown in Fig. 8(a) with $N = M = 100$. We increase K from 3 to 30. For each case, we solve the optimization problem. The behavior of the optimal Chernoff distance, ϵ^*, is shown in Fig. 10. Optimal ϵ^* is nondecreasing as expected, but the gain is not uniform over our chosen range. The increase in ϵ^* is modest (roughly linear) at the beginning. However, there is no gain if we increase K from 17 to 21. On the other hand, ϵ^* increases sharply after $K = 27$. So, a system expansion for this example may not be profitable unless K is quite high. This insight is hard to predict without our proposed framework.

2) Number of Observations: The probability of error decreases with the number of observations. However, the system must wait longer in order to obtain a large number of observations. Recall that the system assumes that the user location can be considered static over the measurement period. Thus, it is desirable to reduce the observation time, especially while tracking a moving user. An estimate of the number of observations–and hence the observation duration–can be made from the upper bound of the probability of error given by Proposition IV.4. Fig. 11 shows the simulation results for the behavior of probability of error as well as the theoretical upper bound as the number of observations, n, is increased. The decrease is exponential as Theorem III.1 postulates. The upper bound is not tight in this regime, but it is indeed an upper bound. In practice, n is not a very tight resource (compared to K, the number of clusterheads) and usually an educated estimation of n using the upper bound will suffice.

3) Coverage Constraint: So far we have not considered the effect of the coverage constraint, p. Table I shows the behavior of ϵ^* with increasing p. For this example, we have used $M = N = 27$, $K = 8$. The threshold Θ is set to -70 dB. Clearly, increasing p can only decrease ϵ^* as Table I shows. However, in this example, there is no effect of p as long as $p \leq 6$. The choice $p = 7$ slightly reduces ϵ^*; but the optimization problem

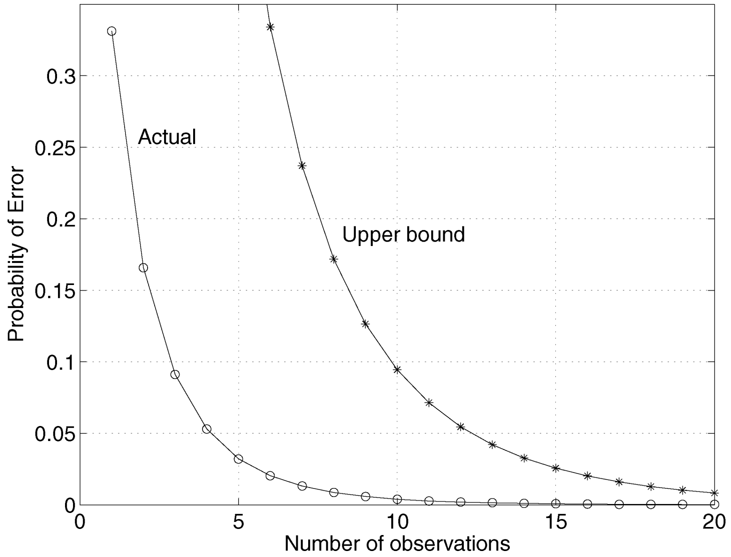


Fig. 11. Number of observations versus the probability of error.

TABLE I
SMALLEST CHERNOFF DISTANCE VERSUS COVERAGE CONSTRAINT. $M = N = 27, K = 8$

Coverage constraint	$\epsilon^\star$
$p = 1, \ldots, 6$	0.39303
$p = 7$	0.38731
$p = 8$	Infeasible

is infeasible for $p = 8$. Usually, $p = 2$ or $p = 3$ provides sufficient robustness under normal operational conditions. Thus, we can conclude that the coverage constraint is not very imposing in this example.

VI. CONCLUSION

A hierarchical wireless sensor network can provide (indoor) location detection service. In this paper, we have proposed a systematic framework to optimally position a given number of clusterheads with the goal of optimizing the performance of the location detection service, thereby complementing other existing location detection schemes that consider clusterhead placements to be given.

We posed the problem of location detection as a hypothesis testing problem by discretizing the space. This is also necessary in practical systems in order to carry out required measurements. Our framework chooses the clusterhead placement that maximizes the worst case Chernoff distance over all pairs of conditional densities. Employing large deviations results, we provided an asymptotic guarantee on the probability of error in location detection. The proposed framework is applicable to a variety of practical situations since it does not impose any restriction on the distribution of the observation sequence, hence allowing the use of any set of physical observations. *Mutatis mutandis*, it is also applicable in a WLAN context.

We developed a mixed integer programming formulation to determine the optimal clusterhead placement, and a fast algorithm for solving it. We evaluated the scalability of the proposed MIP as well as the quality of the resultant placement. Our implementation of the proposed fast algorithm shows that the proposed MIP is capable of solving realistic problems within reasonable time-frames. The quality of the clusterhead placement found by our proposed MIP was assessed through simulation of a realistic site. From the simulation results we found that the proposed placement performs near-optimal in our simulation environment even with a small number of observation samples.

APPENDIX I
NP-HARDNESS OF THE CLUSTERHEAD PLACEMENT PROBLEM

Let us call the recognition version of the optimization problem solved by the MIP of Fig. 2, as the CLUSTERHEAD PLACEMENT problem. In this appendix, we show that CLUSTERHEAD PLACEMENT is NP-hard. In what follows, it will be helpful to visualize CLUSTERHEAD PLACEMENT by means of a complete bipartite graph as shown in Fig. 12. In this figure, there are M nodes in the upper partition, and $N(N-1)/2$ nodes in the lower partition. Node (i,j) of the lower partition is connected to node k of the upper partition with an edge having weight d_{ijk}. Then CLUSTERHEAD PLACEMENT is the following problem.

INSTANCE: A complete bipartite Graph $G = (V, E)$, with M nodes in the first partition $\mathcal{B}$, $N(N-1)/2$ nodes in the other partition $\mathcal{L}$, nonnegative weights d_{ijk} on the edges, positive integer $K \leq |\mathcal{B}|$ and a number ϵ.

QUESTION: Is there a subset $\mathcal{K} \subseteq \mathcal{B}$ such that $|\mathcal{K}| = K$ and such that every vertex $v = (i,j) \in \mathcal{L}$ is joined to at least one node $k \in \mathcal{K}$ by an edge with weight $d_{ijk} \geq \epsilon$?

We transform DOMINATING SET to CLUSTERHEAD PLACEMENT problem. Recall that DOMINATING SET is the following problem [30, p. 75]:

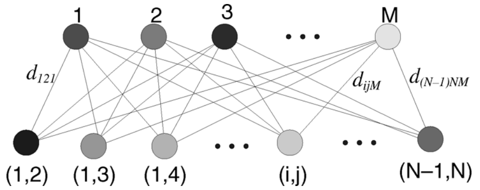


Fig. 12. Bipartite graph illustrating the action of proposed MIP.

INSTANCE: Graph $G = (V, E)$, positive integer $\bar{K} \leq |V|$.

QUESTION: Is there a subset $V' \subseteq V$ such that $|V'| \leq \bar{K}$ and such that every vertex $v \in V \backslash V'$ is joined to at least one member of V' by an edge in E?

DOMINATING SET is known to be NP-hard [30], so this transformation will establish NP-hardness of CLUSTERHEAD PLACEMENT.

Given an instance of the DOMINATING SET problem with $G = (V, E)$, and positive integer $\bar{K}$, put $M = |V|$ and choose smallest N such that $N(N-1)/2 \geq |V|$. Put M nodes in $\mathcal{B}$, $N(N-1)/2$ nodes in $\mathcal{L}$, and create a complete bipartite graph G' as shown in Fig. 12. Index the nodes in $\mathcal{L}$ by $l : 1 \leq l \leq N(N-1)/2$. Assign weight 1 to the edge connecting node $v_l \in \mathcal{L}, 1 \leq l \leq |V|$, and $v_k \in \mathcal{B}$ if $l = k$ or $(v_l, v_k) \in E$; 0 otherwise. Finally assign weight 1 to all edges connecting node $v_l \in \mathcal{L}, |V| < l \leq N(N-1)/2$, and $v_k \in \mathcal{B}$. It is easy to see that this construction is a polynomial time operation.

Now execute CLUSTERHEAD PLACEMENT on G' with $K = \bar{K}$ and $\epsilon = 1$. If CLUSTERHEAD PLACEMENT returns YES, then for each node $v_l \in \mathcal{L}$, there is a node in $v_k \in \mathcal{K}$ such that the weight of (v_l, v_k) is 1. But then v_l and v_k is connected in G; i.e., each node $v_l \in V \setminus \mathcal{K}$ (viewing $\mathcal{K}$ as a subset of nodes of G) is connected to a node in $\mathcal{K}$. Since $|\mathcal{K}| = K$, DOMINATING SET requirement on G is satisfied as well.

Now suppose that CLUSTERHEAD PLACEMENT returns NO. Since for any choice of $\mathcal{K}$, the nodes $v_l \in \mathcal{L}, |V| < l \leq N(N-1)/2$, are connected to a node in $\mathcal{K}$ with weight 1, it must be the case that for every choice of $\mathcal{K}$, there is a node x in the set $\{v_l \in \mathcal{L}, 1 \leq l \leq |V|\}$, for which there is no node in $\mathcal{K}$ that is connected to x by an edge with weight 1. But that means no subset of V with K elements can cover the rest of the nodes; i.e., DOMINATING SET would return NO as well. We therefore have transformed DOMINATING SET to CLUSTERHEAD PLACEMENT and thus CLUSTERHEAD PLACEMENT is NP-hard.

Appendix II
Chernoff Distance Between Gaussian Densities

As we pointed out earlier, our framework can handle arbitrary distributions. However, in many situations, the observation is simply the signal strength and is well modeled as a Gaussian random variable with appropriate mean and variance. In this section, we consider this important special case and provide a closed-form formula for the Chernoff distance d_{ijk}.

Denote the Gaussian conditional densities by

$$\begin{aligned} p_{Y^{(k)}|L_i}(y|L_i) &= \frac{1}{\sigma_{k|i}\sqrt{2\pi}} \exp\left\{-\frac{\left(y-\mu_{k|i}\right)^2}{2\sigma^2_{k|i}}\right\} \\ &\triangleq N(y;\mu_{k|i},\sigma^2_{k|i}) \\ p_{Y^{(k)}|L_j}(y|L_j) &= N(y;\mu_{k|j},\sigma^2_{k|j}). \end{aligned}$$

Then, (3) yields

$$\begin{aligned} \Lambda_{ijk}(\lambda) = \lambda \log\left(\frac{\sigma_{k|j}}{\sigma_{k|i}}\right) &+ \log \sigma_{k|i} \\ &- \frac{1}{2}\log\left[\lambda\sigma^2_{k|j} + (1-\lambda)\sigma^2_{k|i}\right] \\ &- \frac{1}{2}\cdot\frac{\lambda(1-\lambda)(\mu_{k|i}-\mu_{k|j})^2}{\lambda\sigma^2_{k|j}+(1-\lambda)\sigma^2_{k|i}}. \end{aligned}$$

Define

$$\begin{aligned} A &= (\sigma^2_{k|j} - \sigma^2_{k|i})\left[2\log\left(\frac{\sigma_{k|j}}{\sigma_{k|i}}\right)(\sigma^2_{k|j} - \sigma^2_{k|i}) \right. \\ &\qquad \left. + (\mu_{k|i} - \mu_{k|j})^2\right], \\ B &= (\sigma^2_{k|j} - \sigma^2_{k|i})\left[4\sigma^2_{k|i}\log\left(\frac{\sigma_{k|j}}{\sigma_{k|i}}\right) - (\sigma^2_{k|j} - \sigma^2_{k|i})\right] \\ &\quad + 2\sigma^2_{k|i}(\mu_{k|i} - \mu_{k|j})^2 \\ C &= \sigma^2_{k|i}\left[2\sigma^2_{k|i}\log\left(\frac{\sigma_{k|j}}{\sigma_{k|i}}\right) - (\sigma^2_{k|j} - \sigma^2_{k|i}) - (\mu_{k|i} - \mu_{k|j})^2\right]. \end{aligned}$$

The minimum of $\Lambda_{ijk}(\lambda)$ occurs at

$$\lambda^* = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \tag{47}$$

the sign being chosen to ensure that $\lambda^* \in [0, 1]$. Indeed, as $\Lambda_{ijk}(0) = \Lambda_{ijk}(1) = 0$, and $\Lambda_{ijk}(\lambda)$ is a convex function, λ^* minimizes $\Lambda_{ijk}(\lambda)$. Then, we have

$$d_{ijk} = -\Lambda_{ijk}(\lambda^*). \tag{48}$$

Appendix III
Upper Bound for Binary Case

Theorem III.1: Suppose $\pi_i = 1/N$ for all i. Then $P_n^{(e)} \leq e^{-nd_{ijk}}$ for all n.

Proof: From our assumption that $\mathbf{y}_1^{(k)}, \ldots, \mathbf{y}_n^{(k)}$ are i.i.d, we have

$$X_{ijk}\left(\mathbf{y}_1^{(k)}, \ldots, \mathbf{y}_n^{(k)}\right) = \sum_{l=1}^{n} X_{ijk}(\mathbf{y}_l^{(k)}). \tag{49}$$

Since the ML rule rejects L_j if the likelihood ratio is greater than 0, for any n and $\lambda \geq 0$

$$\begin{aligned} \beta_n &= \mathbf{P}_{L_j}[X_{ijk}\left(\mathbf{y}_1^{(k)}, \ldots, \mathbf{y}_n^{(k)}\right) > 0] \\ &\leq \mathbf{P}_{L_j}[e^{\lambda\sum_{l=1}^{n} X_{ijk}(\mathbf{y}_l^{(k)})} > e^{\lambda\cdot 0}] \\ &\leq \mathbf{E}_{L_j}[e^{\lambda\sum_{l=1}^{n} X_{ijk}(\mathbf{y}_l^{(k)})}] \\ &= \left[\mathbf{E}_{L_j}[e^{\lambda X_{ijk}(\mathbf{y}^{(k)})}]\right]^n \\ &= e^{n\cdot\log\left(\mathbf{E}_{L_j}[e^{\lambda X_{ijk}(\mathbf{y}^{(k)})}]\right)}. \end{aligned}$$

The inequality above is the Markov inequality. Optimizing with respect to λ and recognizing that it suffices to optimize over $\lambda \in$

$[0,1]$ (see [21, Exer. 3.4.13]) yields $\beta_n \leq e^{-n \cdot d_{ijk}}$. Similarly, by noting that d_{ijk} is symmetric with respect to the conditional distributions, $\alpha_n \leq e^{-n \cdot d_{ijk}}$, and thus $P_n^{(e)} \leq e^{-n d_{ijk}}$. □

References

[1] E. Biagioni and K. Bridges, "The application of remote sensor technology to assist the recovery of rare and endangered species," *Int. J. High Perform. Comput. Appl., Special Issue on Distributed Sensor Networks*, no. 3, Aug. 2002.

[2] M. B. Srivastava, R. R. Muntz, and M. Potkonjak, "Smart kindergarten: Sensor based wireless networks for smart developmental problem-solving enviroments," *Mobile Comput. Netw.*, pp. 132–138, 2001.

[3] S. Bandyopadhyay and E. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proc. IEEE INFOCOM*, San Francisco, CA, 2003.

[4] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning System: Theory and Practice*, 4th ed. New York: Springer-Verlag, 1997.

[5] T. D. Hodes, R. H. Katx, E. S. Schreiber, and L. Rowe, "Composable ad hoc mobile services for universal interaction," in *Proc. Mobicom'97*, vol. 9, 1997.

[6] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Mobile Comput. Netw.*, 2000, pp. 32–43.

[7] A. Meissner, T. Luckenbach, T. Risse, T. Kirste, and H. Kirchner, "Design challenges for an integrated disaster management communication and information system," in *Proc. 1st IEEE Workshop Disaster Recovery Netw.*, New York, 2002.

[8] P. Bahl and V. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. IEEE INFOCOM*, Tel-Aviv, Israel, Mar. 2000.

[9] J. Hightower, R. Want, and G. Borriello, "SpotON: An Indoor 3d Location Sensing Technology based on RF Signal Strength," University of Washington, Department of Computer Science and Engineering, Seattle, WA, UW CSE 00-02-02, 2000.

[10] P. Castro, P. Chiu, T. Kremenek, and R. Muntz, "A probabilistic location service for wireless network environments," in *Proc. Ubicomp.*, Atlanta, GA, Sep. 2001.

[11] M. Youssef and A. Agrawala, "Handling samples correlation in the Horus system," in *Proc. IEEE INFOCOM*, Hong Kong, Mar. 2004.

[12] R. Battiti, M. Brunato, and A. Villani, "Statistical Learning Theory for Location Fingerprinting in Wireless LANs," University of Trento, Department of Information and Communication Technology, Trento, Italy, DIT 02-086, 2003.

[13] P. Prasithsangaree, P. Krishnamurthy, and P. K. Chrysanthis, "On indoor position location with wireless LANs," in *Proc. 13th IEEE PIMRC Conf.*, Sep. 2002.

[14] [Online]. Available: http://www.cs.umd.edu/~moustafa/location_papers.htm

[15] K. Mechitov, S. Sandresh, Y. Kwon, and G. Agha, "Cooperative Tracking with Binary-Detection Sensor Networks," University of Illinois at Urbana-Champaign Computer Science Department, Tech. Rep. UIUCDCS-R-2003-2379, 2003.

[16] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[17] J. W. McKown and R. L. Hamilton Jr., "Ray tracing as a design tool for radio networks," *IEEE Netw. Mag.*, pp. 27–30, Nov. 1991.

[18] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-Less Low Cost Outdoor Localization for Very Small Devices," University of Southern California/Information Sciences Institute, Tech. Rep. 00-729, 2000.

[19] A. Nasipuri and K. Li, "A directionality based location discovery scheme for wireless sensor networks," in *Proc. First ACM Int. Workshop Wireless Sensor Netw. Appl. (WSNA'02)*, Sep. 2002.

[20] S. Slijepcevic, S. Megerian, and M. Potkonjak, "Location errors in wireless embedded sensor networks: Sources, models, and effects on applications," *ACM Mobile Comput. Commun. Rev.*, vol. 6, no. 3, pp. 67–78, Jul. 2002.

[21] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*, 2nd ed. New York: Springer-Verlag, 1998.

[22] H. Chernoff, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *Ann. Math. Statist.*, vol. 23, pp. 493–507, 1952.

[23] (2002) ILOG CPLEX 8.0. ILOG, Inc., Mountain View, California. [Online]. Available: http://www.ilog.com

[24] M. Daskin, *Network and Discrete Location*. New York: Wiley, 1995.

[25] T. IIhan and M. Pinar. An Efficient Exact Algorithm for the Vertex p-Center Problem. [Online]. Available: http://www.ie.bilkent.edu.tr/mustafap/pubs/

[26] F. A. Ozsoy and M. C. Pinar, "An exact algorithm for the capacitated vertex p-center problem," Comput. Oper. Res., to be published.

[27] R. Rockafellar, *Convex Analysis*. Princeton, NJ: Princeton University Press, 1970.

[28] C. C. Leang and D. H. Johnson, "On the asymptotics of M-hypothesis Bayesian detection," *IEEE Trans. Inf. Theory*, vol. 43, pp. 280–282, Jan. 1997.

[29] J. Liberti, C. Joseph, and T. S. Rappaport, *Smart Antennas for Wireless Commucations: IS-95 and Third Generation CDMA Applications*. Upper Saddle River, NJ: Prentice Hall PTR, 1999, Prentice Hall Communications Engineering and Emerging Technologies Series, ch. 1, p. 37.

[30] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: Freeman, 1979.