

LOCATION ESTIMATION IN WIRELESS NETWORKS: A BAYESIAN APPROACH

David Madigan^{1,2}, Wen-Hua Ju², P. Krishnan², A. S. Krishnakumar²
and Ivan Zorych¹

¹*Rutgers University and* ²*Avaya Labs Research*

Abstract: We present a Bayesian hierarchical model for indoor location estimation in wireless networks. We demonstrate that our model achieves accuracy that is similar to other published models and algorithms. By harnessing prior knowledge, our model drastically reduces the requirement for training data as compared with existing approaches.

Key words and phrases: Bayesian graphical models, BUGS, localization, Markov chain Monte Carlo, sensor networks.

1. Introduction

The growth of wireless networking has generated commercial and research interest in statistical methods to track people and things. Inside stores, hospitals, warehouses and factories, where Global Positioning System devices generally do not work, Indoor Positioning Systems (IPS) aim to provide location estimates for wireless devices such as laptop computers, handheld devices and electronic badges. The proliferation of “Wi-Fi” (IEEE 802.11b) wireless internet access in cafes, college campuses, airports, hotels, and homes has generated particular interest in indoor positioning systems that utilize physical attributes of Wi-Fi signals. Typical applications include tracking equipment and personnel in hospitals, providing location-specific information in supermarkets, museums and libraries, and location-based access control.

In a standard Wi-Fi setup, one or more access points serve end-users. In what follows we focus on networks with multiple access points (typical of networks in office buildings or large public spaces). Wi-Fi location estimation can employ one or more of several physical attributes of the medium. Typical features include received signal strength (RSS) from the access points, the angle of arrival of the signal, and the time difference of arrival. Among these, RSS is the only feature that reasonably priced hardware can currently measure. There exists a substantial literature on using RSS for location estimation in wireless networks – see, for example, Bahl and Padmanabhan (2000), Ladd, Bekris, Rudys, Marceau, Kavraki

and Dan, (2002) and Roos, Myllymaki and Tirri (2002). Related websites include www.ekahau.com, www.bluesoft-inc.com and www.newburynetworks.com. In a laboratory setting, RSS decays linearly with log distance and a simple triangulation using RSS from three access points can uniquely identify a location in a two-dimensional space. In practice, physical characteristics of a building such as walls, elevators, and furniture, as well as human activity, add significant noise to RSS measurements. Consequently statistical approaches to location estimation prevail.

The standard approach uses supervised learning techniques. The training data comprise vectors of signal strengths, one for each of a collection of known locations. The dimension of each vector equals the number of access points. The corresponding location could be one-dimensional (e.g., location on a long airport corridor), two-dimensional (e.g., location on one floor of a museum), or three-dimensional (e.g., location within a multi-storey office building). Collection of the location data is labor intensive, requiring physical distance measurements with respect to a reference object such as a wall. The model building phase then learns a predictive model that maps signal strength vectors to locations. Researchers have applied many supervised learning methods to this problem, including nearest neighbor methods, support vector machines, and assorted probabilistic techniques. In this paper we explore the use of hierarchical Bayesian graphical models (Spiegelhalter (1998) and Gelman, Carlin, Stern and Rubin (2003)) for wireless location. Our objective is to use the hierarchical Bayesian framework to incorporate important prior information and the graphical model framework to facilitate the construction of realistically complex models.

Gathering extensive training data and the requisite physical measures of location (“profiling”) involves a steep upfront cost and deployment effort (Smailagic, Siewiorek, Anhalt, Kogan and Wang (2001)). Furthermore, even in normal office environments, changing environmental, building, and occupancy conditions can affect signal propagation and require repeated data gathering to maintain predictive accuracy (Bahl, Padmanabhan and Balachandran (2000)). Consequently, minimizing the number of training observations needed to adequately profile a particular site is an important objective. Similarly we seek to minimize data requirements concerning internal wall materials, flooring, occupancy, etc.

Two types of location estimation systems exist. In a client-based deployment, the client measures the signal strengths as seen by it from various access points. The client uses this information to locate itself. The cost to an enterprise for such deployments is the cost of profiling the site, building the model, and maintaining the model. In an infrastructure-based deployment, the administrator deploys so-called sniffing devices that monitor the signal strength from clients. The cost to enterprises in such deployments is the typically modest cost of deploying the

necessary hardware and software, and the time and effort to build and maintain the model (if it is not completely automated).

Our key finding is that a hierarchical Bayesian approach, incorporating prior physical knowledge about the nature of Wi-Fi signals, can provide accurate location estimates without *any* location information in the training data. In the context of an infrastructure-based deployment, our proposed model can thus eliminate profiling entirely.

Section 2 provides some additional background. Our approach uses probabilistic graphical models and Section 3 provides describes the framework we use. Sections 4 and 5 describe the datasets we used for experimentation as well as various results. Section 6 describes some potential future work.

We focus on static location estimation. That is, we consider models that estimate location at a particular timepoint and we do not attempt to track moving objects. In a companion paper we will describe extensions to dynamic tracking models.

2. Background

2.1. Related Work

Location estimation techniques in wireless networks can be broadly classified based on the methods used to build models and methods used to search the models in the online phase. For building models, most techniques profile the entire site and collect one or more signal strength samples from all visible access points at each sample point. Each point is mapped to either a signal strength vector (Bahl and Padmanabhan (2000), Ladd et al. (2002), Prasithsangaree, Krishnamurthy and Chrysanthis (2002) and Saha, Chaudhuri, Sanghi and Bhagwat (2003) or a signal strength probability distribution (Battiti, Brunato and Villani (2002), Roos et al. (2002), Thrun (2000) and Youssef, Agrawala and Udaya Shankar (2003)). Such profiling techniques require considerable investment in data gathering. Alternatively, a parametric model that uses signal propagation physics and calculates signal degradation based on a detailed map of the building, the walls, obstructions and their construction material, has been proposed (Bahl and Padmanabhan (2000)). Obtaining detailed maps of the building and its changes over time is, however, a hurdle that needs to be overcome for the use of this method.

In Smailagic et al. (2001), the authors emphasize a client-based location model and raise interesting privacy issues in location-based services. We expect that in enterprises, based on current privacy policies used for other electronic transmissions like email and web-access, the preference would be for an infrastructure-based solution. If privacy is desired, in our case, on entering a

site a client device could download the model for that site and use it to determine its own location. As mentioned in Smailagic et al. (2001), client-based approaches must also be concerned about the power requirements on the client devices that are inherently power constrained. Sniffing for clients to provide an infrastructure-based system has also been proposed (Christ and Godwin (1993), Want et al. (1992) and Werb and Lanzl (1998)).

Custom sensors have been used for location estimation in other interesting ways (Priyantha, Chakraborty and Balakrishnan (2000), Want et al. (1992) and Werb and Lanzl (1998)). In Want et al. (1992) and similar systems, infra-red (IR) wireless technology is used; IR technology has limited range and hence has not become very popular. In Priyantha et al. (2000), a decentralized (client-based) approach using time difference of arrival between ultrasound and RF signals from custom sensors is used for location estimation. The system in Werb and Lanzl (1998) uses expensive custom RF-based hardware for location estimation, and an approach based on time difference of signal arrival, which is inherently more expensive to measure. In contrast, our approach is easier to bootstrap, is based on RSS and can be built with off-the-shelf components. Recent advances in sensor technology (Huang (2003)) and projected decreases in the manufacturing cost allow us to provide a cost-effective solution.

2.2. Radio frequency signal propagation in wireless ethernet

Ladd et al. (2002) provide an introduction to the behavior of Wi-Fi signals and here we present a brief summary. The IEEE 802.11b High-Rate standard uses radio frequencies in the 2.4 GHz band. Wi-Fi adaptors use spread-spectrum technology that spreads the signal over several frequencies. In this way, interference on a single frequency does not entirely block the signal. The signal itself propagates in a complex manner (Hassan-Ali and Pahlavan (2002)). Reflection, absorption, and diffraction occur when the signal's waves encounter opaque obstacles resulting in essentially random variations of signal strength. A variety of other factors such as noise, interference from other sources, and interference between channels also affect the signal. The resonant frequency of water happens to be 2.4 GHz so people also absorb the radio waves and impact the signal strength. Other common devices using the 2.4 GHz band include microwave ovens, Bluetooth devices and 2.4 GHz cordless phones.

The consequence of all this is that received signal strength varies over time at a single location and varies across different locations. However, signal profiles corresponding to spatially adjacent locations are similar, as the various external variables remain approximately the same over short distances. Furthermore, the local average of the signal strength varies slowly over time and the signal strength decays approximately in proportion to log distance (Howard, Siddiqi and Sukhatme (2003)).

3. Bayesian Graphical Models

A *graphical model* is a multivariate statistical model embodying a set of conditional independence relationships. A graph displays the independence relationships. The vertices of the graph correspond to random variables and the edges encode the relationships. To date, most graphical models research has focused on acyclic digraphs, chordal undirected graphs, and chain graphs that allow both directed and undirected edges, but have no partially directed cycles (Lauritzen (1996)).

Here we focus on acyclic digraphs (ADGs) with both continuous and categorical random variables. In an ADG, *all* the edges are directed and the graph represents them with arrows (see Figure 1). A directed graph is acyclic if it contains no cycles. Each vertex in the graph corresponds to a random variable $X_v, v \in V$ taking values in a sample space \mathcal{X}_v . To simplify notation, we use v in place of X_v in what follows. In an ADG, the *parents* of a vertex v , $\text{pa}(v)$, are those vertices from which vertices point into v . The *descendants* of a vertex v are the vertices which are reachable from v along a directed path. A vertex w is a *child* of v if there is an edge from v to w . The parents of v are taken to be the only direct influences on v , so that v is independent of its non-descendants given its parents. This property implies a factorization of the joint density of $X_v, v \in V$, which we denote by $p(V)$, given by

$$p(V) = \prod_{v \in V} p(v|\text{pa}(v)). \quad (1)$$

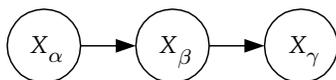


Figure 1. A simple acyclic directed graphical model.

Figure 1 shows a simple example. This directed graph represent the assumption that X_γ and X_α are conditionally independent given X_β . The joint density of the three variables factors accordingly as $p(X_\alpha, X_\beta, X_\gamma) = p(X_\alpha)p(X_\beta|X_\alpha)p(X_\gamma|X_\beta)$.

For graphical models where all variables are discrete, Spiegelhalter and Lauritzen (1990) presented a Bayesian analysis and showed how independent Dirichlet prior distributions can be updated locally to form posterior distributions as data arrive. Heckerman, Geiger, and Chickering (1995) provided corresponding closed-form expressions for complete-data likelihoods and posterior model probabilities. Madigan and York (1995) described corresponding Bayesian model averaging procedures. In the Bayesian framework, model parameters are random variables and appear as vertices in the graph.

When some variables are discrete and others continuous, or when some of the variables are latent or have missing values, a closed-form Bayesian analysis generally does not exist. Analysis then requires either analytic approximations of some kind or simulation methods. Here we consider a Markov chain Monte Carlo (MCMC) simulation method. Spiegelhalter (1988) provides a brief introduction to a particular MCMC algorithm, the univariate Gibbs sampler, for Bayesian graphical models as follows.

The Gibbs sampler starts with some initial values for each unknown quantity (that is, model parameters, missing values, and latent variables), and then cycles through the graph simulating each variable v in turn from its conditional probability distribution, given all the other quantities, denoted $V \setminus v$, fixed at their current values (known as the “full conditional”). The simulated v replaces the old value and the simulation shifts to the next quantity. After sufficient iterations of the procedure one assumes that the Markov chain has reached its stationary distribution, and then future simulated values for vertices of interest are monitored. Inferences concerning unknown quantities are then based on data analytic summaries of these monitored values, such as empirical medians and 95% intervals. Some delicate issues do arise with the Gibbs sampler such as assessment of convergence, sampling routines, etc. Gilks, Richardson and Spiegelhalter (1996) provide a full discussion.

The crucial connection between directed graphical models and Gibbs sampling lies in expression (1). The full conditional distribution for any vertex v is:

$$\begin{aligned} p(v|V \setminus v) &\propto p(v, V \setminus v) \\ &\propto \text{terms in } p(V) \text{ containing } v \\ &= p(v|\text{pa}(v)) \prod_{w \in \text{child}(v)} p(w|\text{pa}(w)), \end{aligned}$$

i.e., a prior term and a set of likelihood terms, one for each child of v . Thus, when sampling from the full conditional for v , we need only consider vertices which are parents, children, or parents of children of v , and we can perform local computations. The BUGS language and software (Spiegelhalter, Thomas, and Best (1999)) implements a version of the Gibbs sampler for Bayesian graphical models. We utilized BUGS for the experiments we report below.

4. Datasets

We collected RSS data from three floors at two sites, referred to in this paper as BR, CA Up and CA Down. Both the BR and CA sites are office buildings and have deployed 802.11b wireless networks. Figure 2 shows the floor plans for the two sites.

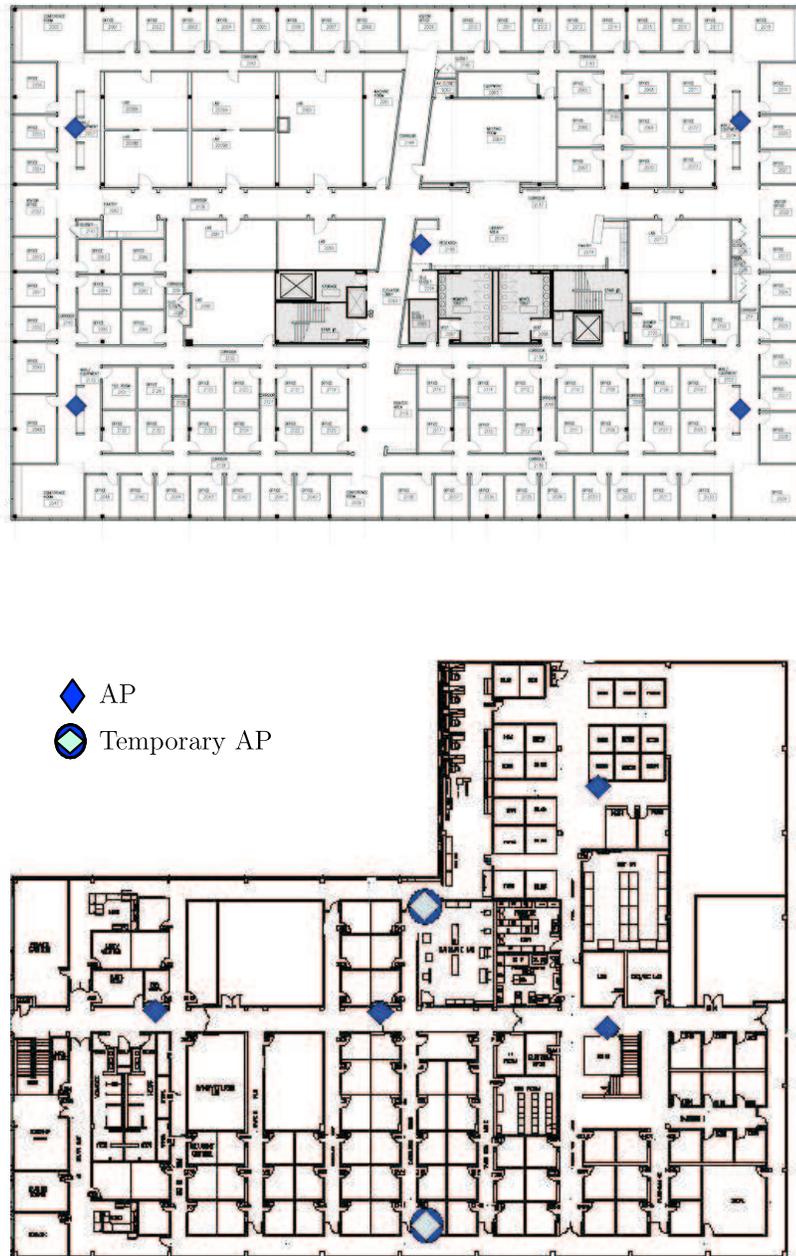


Figure 2. Floor plans for the BR and CA sites showing the access points (APs).

To make our RSS measurements, we used a Linux IPAQ with a modified driver updated to scan for access points. The IPAQ had a custom client and a standard Konqueror web browser. The user making RSS measurements clicked

on current location in an image of the floor as displayed on the browser. The posting of this information triggered an RSS measurement request at the client from the web server on a separate TCP channel. The web server then recorded the coordinate and RSS vector information at that location. We did not specifically orient the IPAQ in any way while taking measurements.

The BR site has 5 access points and measures 225 ft X 144 ft. We made 254 RSS measurements along the corridors of this site. The measurements were made over different sessions spanning several days.

The CA Down floor has 4 access points, three of which are colinear, and measures 250 ft X 175 ft, with a “slice” removed. Due to the colinearity of the three access points, we installed two temporary access points. The CA Up floor has 4 access points. At the CA site, a colleague took 146 measurements on the “Down” floor and 56 measurements on the “Up” floor in the corridors.

5. Models and Experiments

Our goal is to construct a model that embodies extant knowledge about Wi-Fi signals as well as physical constraints implied by the target building. We present a series of models of increasing complexity, in each case showing results with varying training dataset sizes. We focus throughout on predictive accuracy.

5.1. A non-hierarchical Bayesian graphical model

Figure 3 shows a particular graphical model for a two-dimensional location estimation problem in a building with four access points. In what follows we refer to this model as M_1 (although the number of access points varies).

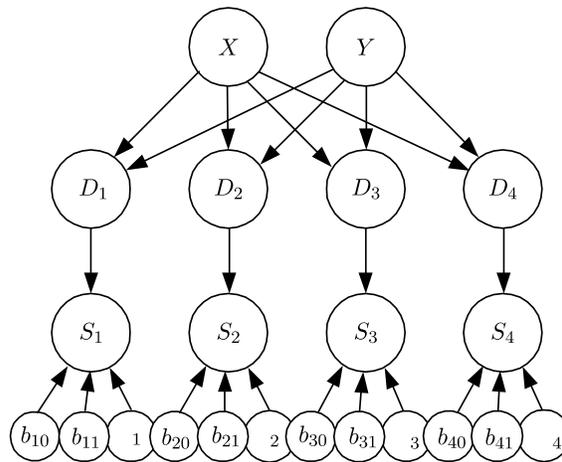


Figure 3. A Bayesian graphical model for location estimation. This is model M_1 .

The vertices X and Y represent location. The vertex D_1 (respectively D_2, D_3 , and D_4) represents the euclidean distance between the location specified by X and Y and the first (respectively second, third, and fourth) access point. Since we assume the locations of the access points are known, the D_i 's are deterministic functions of X and Y . The vertex S_i represents the signal strength measured at (X, Y) with respect to the i th access point, $i = 1, \dots, 4$. The model assumes that X and Y are marginally independent.

Specification of the model requires a conditional density for each vertex given its parents, here taken as follows:

$$\begin{aligned} X &\sim \text{uniform}(0, L), \\ Y &\sim \text{uniform}(0, B), \\ S_i &\sim N(b_{i0} + b_{i1} \log D_i, \tau_i), i = 1, 2, 3, 4, \\ b_{i0} &\sim N(0, 0.001), i = 1, 2, 3, 4, \\ b_{i1} &\sim N(0, 0.001), i = 1, 2, 3, 4. \end{aligned}$$

Here L and B denote the length and breadth of the building respectively. The distributions for X and Y reflect the physical constraints of the building. The model for S_i reflects the fact that signal strength, decays approximately linearly with log distance. Note that we use $N(\mu, \tau)$ to denote a Gaussian distribution with mean μ and precision τ so that the prior distributions for b_{i0} and b_{i1} have large variance.

Figure 4 shows a more compact representation for M_1 using the BUGS plate notation for replicated sub-models, and with d denoting the number of access points.

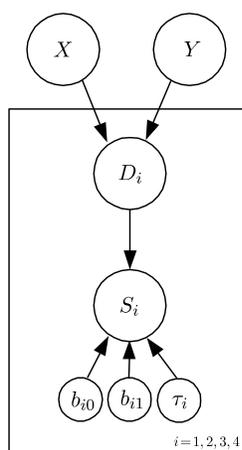


Figure 4. A Bayesian graphical model using plate notation. This is also model M_1 .

Figure 5 shows the predictive performance of model M_1 on the BR data, as a function of training set size. Specifically, for each training set size N , we plot the average performance for 30 replications of a random test-training split, using N observations for training and one observation for testing. The red solid curve shows the results for M_1 . In each case, and throughout the paper, the estimates resulted from 110,000 MCMC iterations, discarding the first 10,000. This seemed to provide adequate convergence in most cases, according to standard BUGS diagnostics. We return to this issue at the end of the paper. For comparison purposes, the blue dotted curve shows the equivalent results for the smoothed nearest-neighbor “SmoothNN” model of Krishnan et al. (2003). The SmoothNN model proved highly competitive in comparison with two other benchmark systems and hence we use it for comparison purposes in this paper. Figure 5 shows that M_1 outperforms the SmoothNN model with smaller training sample sizes, but underperforms the SmoothNN model at the larger sample sizes.

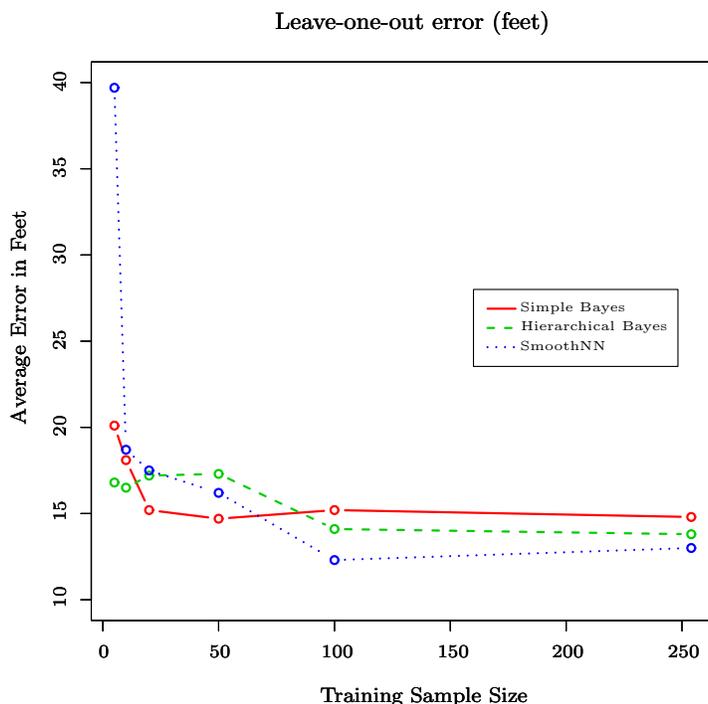


Figure 5. Average predictive accuracy of the non-hierarchical Bayesian graphical model M_1 , the hierarchical model M_2 , and the SmoothNN model on the BR data.

Figure 6 provides more detail and also shows results for the other two datasets. The results for the three different datasets are qualitatively similar.

Tables 1, 2 and 3 provide corresponding summary statistics. Note that predictive accuracy does tend to improve with training sample size, although not in every case.

SmoothNN (S) versus Bayesian (B) Model, Error in Feet

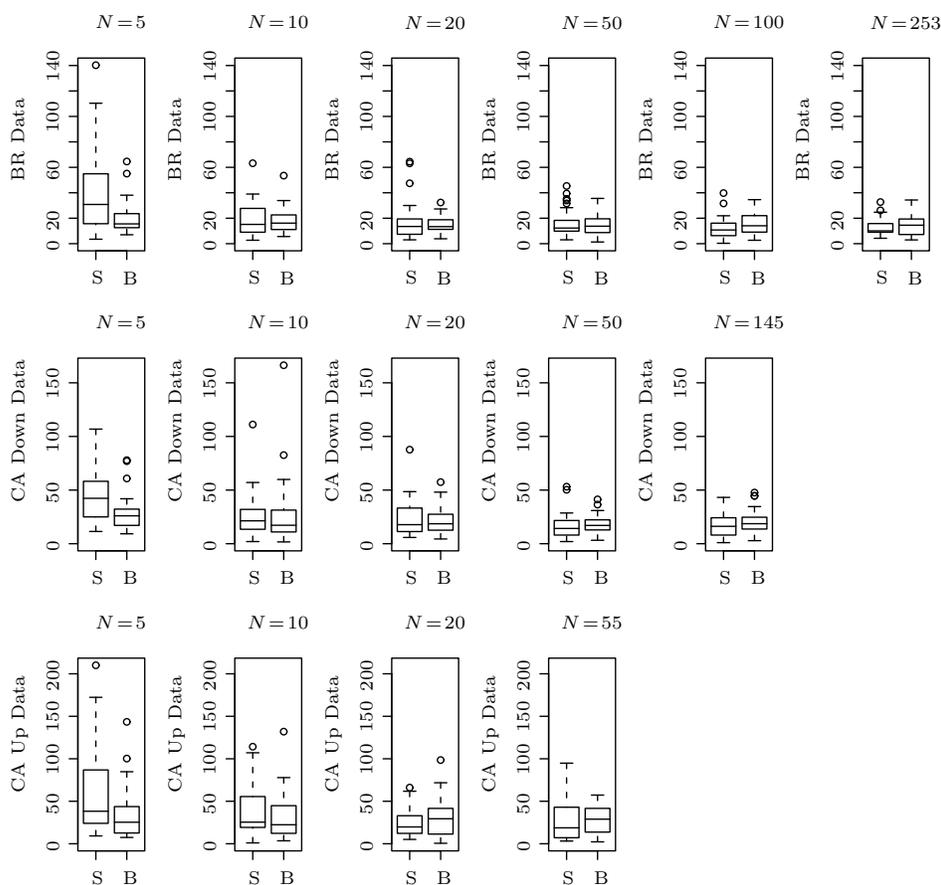


Figure 6. Predictive accuracy of the SmoothNN model versus the non-hierarchical Bayesian graphical model (M_1) for the BR data, CA Down data and CA Up Data.

Table 1. Leave-one-out average accuracy in feet for the BR data. Results are averaged over 30 replications. The corresponding standard errors range from about 1.5 to 2.5.

Model	Training Sample Size					
	5	10	20	50	100	253
Bayesian M_1	20.1	18.1	15.2	14.7	15.2	14.8
SmoothNN	39.7	18.7	17.5	16.2	12.3	13.0

Table 2. Leave-one-out average accuracy in feet for the CA Down data. Results are averaged over 30 replications. The corresponding standard errors range from about 1.5 to 5.5.

Model	Training Sample Size				
	5	10	20	50	145
Bayesian M_1	28.8	27.4	21.6	18.2	19.9
SmoothNN	46.3	26.7	24.3	17.1	17.4

Table 3. Leave-one-out average accuracy in feet for the CA Up data. Results are averaged over 30 replications. The corresponding standard errors range from about 2.7 to 5.7.

Model	Training Sample Size			
	5	10	20	55
Bayesian M_1	35.4	31.7	30.5	28.5
SmoothNN	59.9	36.3	25.2	28.2

5.2. A hierarchical Bayesian graphical model

Next we seek to incorporate the knowledge that the coefficients of the linear regression models corresponding to each of the access points should be similar since the similar physical processes are in play at each access point. Physical differences between locations of the different access points will tend to mitigate the similarity but nonetheless, borrowing strength across the different regression models might provide some predictive benefits.

Figure 7 shows the hierarchical model M_2 . The conditional densities for this model are

$$\begin{aligned}
 X &\sim \text{uniform}(0, L), \\
 Y &\sim \text{uniform}(0, B), \\
 S_i &\sim N(b_{i0} + b_{i1} \log D_i, \tau_i), i = 1, \dots, d, \\
 b_{i0} &\sim N(b_0, \tau_{b0}), i = 1, \dots, d, \\
 b_{i1} &\sim N(b_1, \tau_{b1}), i = 1, \dots, d, \\
 b_0 &\sim N(0, 0.001), \\
 b_1 &\sim N(0, 0.001), \\
 \tau_{b0} &\sim \text{Gamma}(0.001, 0.001), \\
 \tau_{b1} &\sim \text{Gamma}(0.001, 0.001).
 \end{aligned}$$

The green dashed curve in Figure 5 shows the predictive accuracy of M_2 on the BR data. A comparison of M_1 and M_2 shows that the hierarchical model performs similarly to its non-hierarchical counterpart, although M_2 does provide improvement in average error for the smallest training sample size of 5.

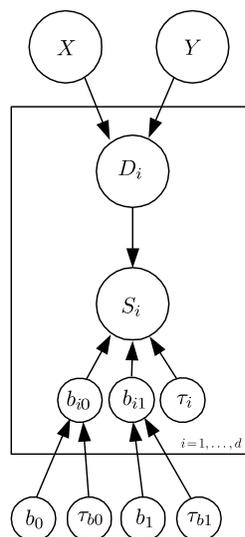


Figure 7. A Bayesian hierarchical graphical model using plate notation. This is model M_2 .

Figure 8 provides more detail and also shows results for the other two datasets. Again, the results for the three different datasets are qualitatively similar. Tables 4, 5 and 6 provide corresponding summary statistics. In general, the results show small differences between the non-hierarchical model M_1 and the hierarchical model M_2 .

Table 4. Leave-one-out average accuracy in feet for the BR data. Results are averaged over 30 replications. The corresponding standard errors range from about 1.3 to 2.5.

Model	Training Sample Size					
	5	10	20	50	100	253
Bayesian M_1	20.1	18.1	15.2	14.7	15.2	14.8
Bayesian M_2	16.8	16.5	17.2	17.3	14.1	13.8

Table 5. Leave-one-out average accuracy in feet for the CA Down data. Results are averaged over 30 replications. The corresponding standard errors range from about 1.5 to 7.5.

Model	Training Sample Size				
	5	10	20	50	145
Bayesian M_1	28.8	27.4	21.6	18.2	19.9
Bayesian M_2	21.3	26.3	25.0	20.3	18.7

Table 6. Leave-one-out average accuracy in feet for the CA Up data. Results are averaged over 30 replications. The corresponding standard errors range from about 2.7 to 6.8.

	Training Sample Size			
Model	5	10	20	55
Bayesian M_1	35.4	31.7	30.5	28.5
Bayesian M_2	30.6	37.9	33.0	33.5

Simple Bayesian (B) versus Hierarchical Bayesian (H) Model, Error in Feet

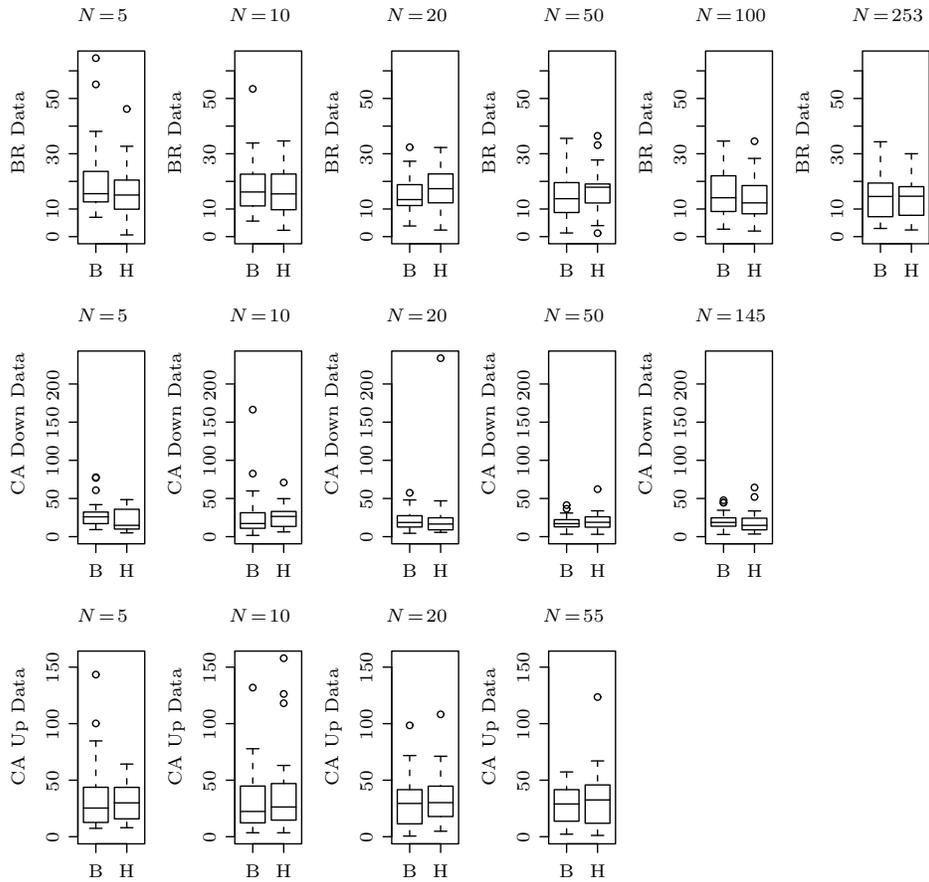


Figure 8. Predictive accuracy of the non-hierarchical Bayesian graphical model (M_1) versus the hierarchical Bayesian graphical model (M_2) for the BR data, CA Down data and CA Up Data.

5.3. Training data with no location information

Model M_2 incorporates two sources of prior knowledge. First, M_2 embodies

the knowledge that signal strength decays approximately linearly with log distance. Second, the hierarchical portion of M_2 reflects prior knowledge that the different access points behave similarly. Here we pursue the idea that perhaps this prior knowledge provides sufficient constraints to obviate the need to know the actual locations of the training data observations. Specifically, the training data now comprise vectors of signal strengths with *unknown* locations: X and Y in M_1 and M_2 become latent variables.

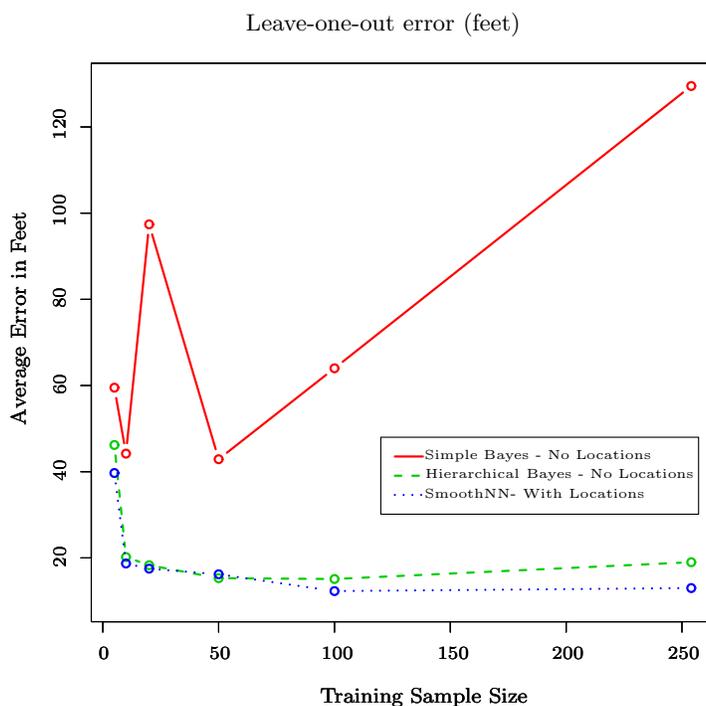


Figure 9. Average predictive accuracy of the non-hierarchical Bayesian graphical model on the BR data with no location data.

Figure 9 shows the average predictive performance for the BR data with different numbers of random sampled signal strength vectors. In each case the results shows averages over 30 replications, except for the maximal case (254 for BR, 146 for CA Down, 56 for CA Up) which uses all the signal strength vectors in the training data. The red solid curve corresponds to M_1 and the green dashed curve to M_2 . The results for the SmoothNN model are reproduced from Figure 5 and reflect training data *with known locations*. These results show some striking features. With no location information, M_1 performs poorly and shows no improvement with increasing numbers of signal strength vectors. Model M_2 , however, from about 10 training vectors onwards, performs almost as well

as the SmoothNN model trained on data with complete location information for each signal strength vector.

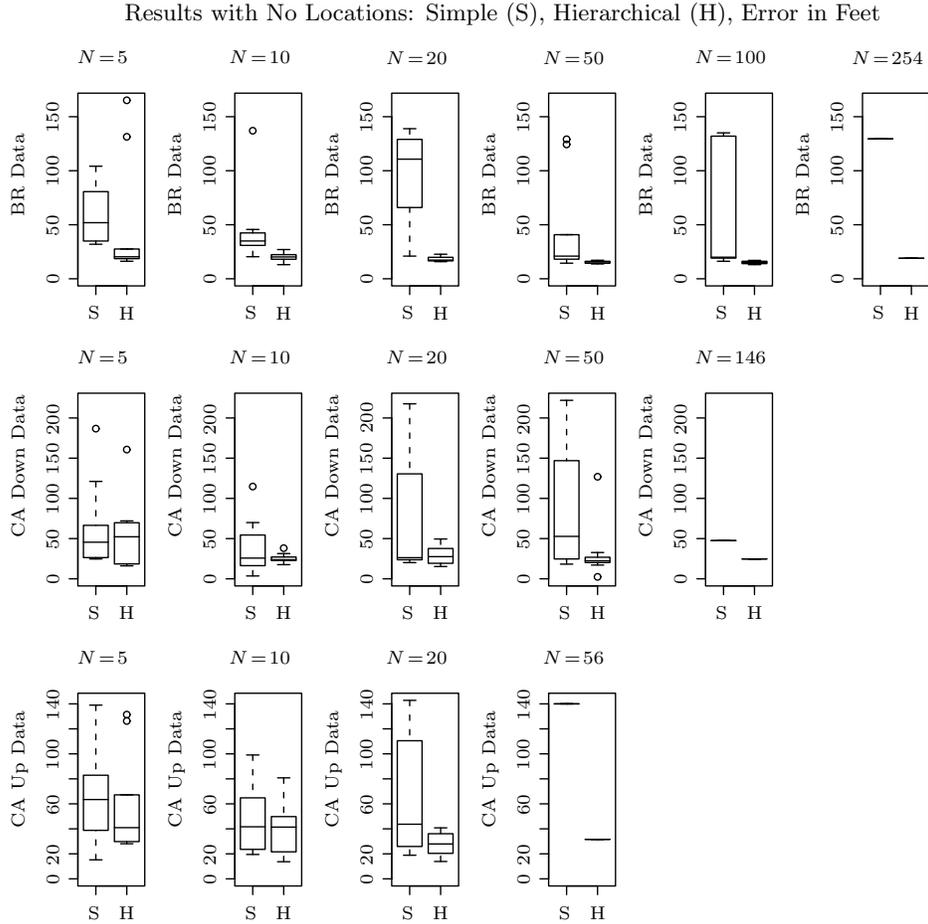


Figure 10. Predictive accuracy of the Bayesian graphical model with no location information. Non-hierarchical model (M_1) versus the hierarchical Bayesian graphical model (M_2) for the BR data, CA Down data and CA Up Data.

Figure 10 provides more detail and also shows results for the other two datasets. Once again, the results for the three different datasets are qualitatively similar. Tables 7, 8 and 9 provide corresponding summary statistics. In each case the hierarchical model, even with no location information, provides predictive performance that is close to, although not as good as, the state-of-the-art SmoothNN model.

Dropping the location data requirement affords significant practical benefits. As discussed in Section 1, the location measurement process is slow and human-intensive. By contrast, gathering signal strengths vectors without the corresponding locations does not require human intervention; in the infrastructure approach, suitably instrumented access points or sniffing devices can solicit signal strength measurements from existing Wi-Fi devices and can do this repeatedly at essentially no cost.

We note the existing location estimation algorithms that we are aware of all require location information in the training data to produce *any* estimates.

Table 7. Average accuracy in feet for the BR data. No location information in the training data. Results are averaged over 30 replications.

Model	Training Sample Size					
	5	10	20	50	100	254
Bayesian M_1 , No Locations	59.5	44.2	97.4	42.9	64.0	129.5
Bayesian M_2 , No Locations	46.2	20.2	18.3	15.3	15.1	19.0
SmoothNN, With Locations	39.7	18.7	17.5	16.2	12.3	13.0

Table 8. Average accuracy in feet for the CA Down data. No location information in the training data. Results are averaged over 30 replications.

Model	Sample Size				
	5	10	20	50	146
Bayesian M_1 , No Locations	62.6	32.1	54.6	71.9	47.8
Bayesian M_2 , No Locations	54.4	26.2	27.8	33.3	25.0
SmoothNN, With Locations	46.3	26.7	24.3	17.1	17.4

Table 9. Average accuracy in feet for the CA Up data. No location information in the training data. Results are averaged over 30 replications.

Model	Sample Size			
	5	10	20	56
Bayesian M_1 , No Locations	61.9	42.9	56.5	140.9
Bayesian M_2 , No Locations	41.7	40.9	30.6	31.2
SmoothNN, With Locations	59.9	36.3	25.2	28.2

5.4. Incorporating corridor effects and other prior knowledge

The graphical modeling framework coupled with MCMC provides a very flexible tool for multivariate modeling. Here we pursue two ideas that demonstrate this flexibility and aim to improve predictive accuracy, especially when the training data contain no location information.

Corridor Model

All three datasets show striking corridor effects. That is, when an access point is located in a corridor, the signal strength tends to be substantially stronger along the entire corridor. In the three office building floors we have examined, corridors are mostly parallel to the walls. Hence, a location that shares either an x -coordinate or a y -coordinate with an access point (at least approximately) tends to be in the same corridor as that access point.

Figure 11 shows a model (M_3) with a corridor effect, C_i . The variable C_i takes the value 1 if the location (X, Y) shares a corridor with access point i and 0 otherwise. We define “sharing a corridor” as having an x - or y -coordinate within three feet of the corresponding access point coordinate. Since corridor width varies from building to building, this definition should vary accordingly, although we do not pursue this here.

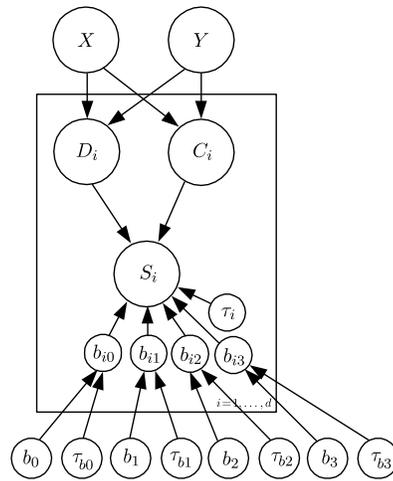


Figure 11. An extension of M_2 to include a corridor main effect. This is model M_3 .

The conditional densities for model M_3 are

$$\begin{aligned}
 X &\sim \text{uniform}(0, L), \\
 Y &\sim \text{uniform}(0, B), \\
 S_i &\sim N(b_{i0} + b_{i1} \log D_i + b_{i2} C_i + b_{i3} C_i D_i, \tau_i), i = 1, \dots, d, \\
 b_{ij} &\sim N(b_j, \tau_{bj}), i = 1, \dots, d, j = 0, 1, 2, 3, \\
 b_j &\sim N(0, 0.001), j = 0, 1, 2, 3, \\
 \tau_{bj} &\sim \text{Gamma}(0.001, 0.001), j = 0, 1, 2, 3.
 \end{aligned}$$

Note we have included a corridor main effect as well as a corridor-distance interaction term. Figure 12 shows the results with various (labeled) training sample sizes. Tables 10, 11 and 12 provide more details.

Table 10. Leave-one-out average accuracy in feet for the BR data. Results are averaged over 30 replications. The corresponding standard errors range from about 1.4 to 7.5. “M,” “I,” and “B” refer to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Model	Training Sample Size					
	5	10	20	50	100	253
Bayesian M_2	16.8	16.5	17.2	17.3	14.1	13.8
Bayesian M_3^M	17.0	16.8	17.9	15.9	12.9	14.4
Bayesian M_3^I	19.5	18.2	15.9	18.0	15.2	15.6
Bayesian M_3^B	20.0	16.5	12.3	13.6	14.0	15.6

Table 11. Leave-one-out average accuracy in feet for the CA Down data. Results are averaged over 30 replications. The corresponding standard errors range from about 1.5 to 9.1. “M,” “I,” and “B” refer to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Model	Training Sample Size				
	5	10	20	50	145
Bayesian M_2	21.3	26.3	25.0	20.3	18.7
Bayesian M_3^M	20.8	17.5	17.4	16.9	18.3
Bayesian M_3^I	23.2	22.4	15.4	20.9	17.5
Bayesian M_3^B	24.4	25.0	17.4	41.7	31.9

Table 12. Leave-one-out average accuracy in feet for the CA Up data. Results are averaged over 30 replications. The corresponding standard errors range from about 2.7 to 6.8. “M,” “I,” and “B” refer to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Model	Training Sample Size			
	5	10	20	55
Bayesian M_2	30.6	37.9	33.0	33.5
Bayesian M_3^M	31.8	34.2	31.4	32.5
Bayesian M_3^I	35.7	28.8	31.4	34.7
Bayesian M_3^B	36.6	32.0	31.7	30.0

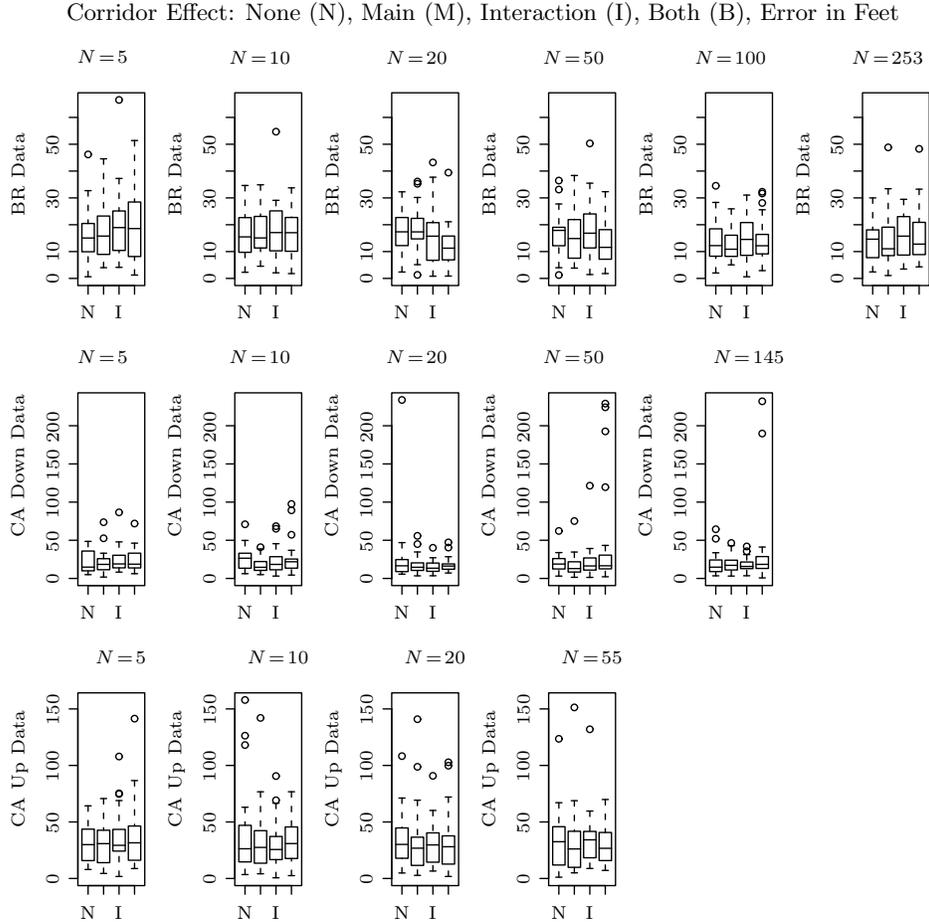


Figure 12. Predictive accuracy of the hierarchical Bayesian graphical model (M_2) versus the hierarchical Bayesian graphical model with corridor effects (M_3) for the BR data, CA Down data and CA Up Data. “N” corresponds to no corridor effect and is the same as M_2 . “M,” “I,” and “B” correspond to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Figure 13, Tables 13, 14 and 15 provide corresponding results with no location information.

These analyses suggest that our particular approach to modeling a corridor effect does not improve predictive performance.

Informative Priors for the Regression Coefficients

A second direction we considered was the incorporation of mildly informative prior distributions for the regression coefficients. Specifically, we used a

Table 13. Average accuracy in feet for the BR data. No location information. Results are averaged over 30 replications. The corresponding standard errors range from about 1.9 to 4.5. “M,” “I,” and “B” refer to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Model	Training Sample Size					
	5	10	20	50	100	254
Bayesian M_2 , No Locations	46.2	20.2	18.3	15.3	15.1	19.0
Bayesian M_3^M , No Locations	29.1	20.4	17.5	15.9	15.6	17.6
Bayesian M_3^I , No Locations	34.1	18.2	17.7	16.1	15.5	16.3
Bayesian M_3^B , No Locations	34.4	26.1	20.3	21.4	16.9	14.4

Corridor Effect: None (N), Main (M), Interaction (I), Both (B), Error in Feet

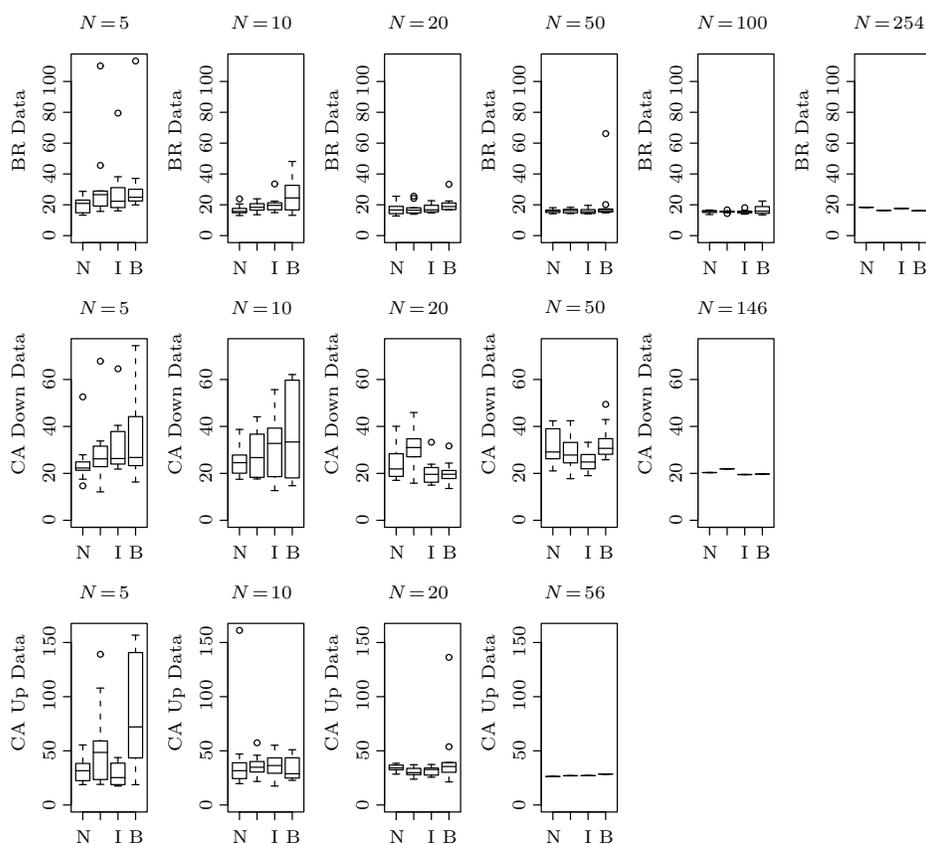


Figure 13. Predictive accuracy of the hierarchical Bayesian graphical model (M_2) versus the hierarchical Bayesian graphical model with corridor effects (M_3) for the BR data, CA Down data and CA Up Data. No location information. “N” corresponds to no corridor effect and is the same as M_2 . “M,” “I,” and “B” correspond to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Table 14. Average accuracy in feet for the CA Down data. No location information. Results are averaged over 30 replications. The corresponding standard errors range from about 2.1 to 10.8. “M,” “I,” and “B” refer to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Model	Training Sample Size				
	5	10	20	50	146
Bayesian M_2 , No Locations	23.9	29.4	29.2	29.8	21.9
Bayesian M_3^M , No Locations	29.8	29.9	24.7	26.5	34.2
Bayesian M_3^I , No Locations	29.1	25.0	25.9	32.7	33.4
Bayesian M_3^B , No Locations	30.0	38.9	25.6	32.1	27.1

Table 15. Average accuracy in feet for the CA Up data. No location information. Results are averaged over 30 replications. The corresponding standard errors range from about 3.4 to 36.8. “M,” “I,” and “B” refer to model M_3 with main effect only, interaction only, and both main effect and interaction, respectively.

Model	Training Sample Size			
	5	10	20	55
Bayesian M_2 , No Locations	63.6	38.1	28.9	30.6
Bayesian M_3^M , No Locations	28.5	36.3	31.6	27.1
Bayesian M_3^I , No Locations	46.6	36.2	30.1	27.0
Bayesian M_3^B , No Locations	62.6	32.7	44.7	28.6

$N(10, 0.1)$ prior for b_0 and a $N(-19.5, 0.1)$ prior for b_1 in Model M_2 . The means of these priors correspond to the average intercept and slope from a maximum likelihood analysis of the combined data over all access points from all three locations. The precisions of 0.1 permit considerable posterior variability around these values.

Figure 14 shows the results and Tables 16, 17 and 18 provide more details. The informative priors do provide improved predictive performance, especially for the experiments with no location data and small numbers of signal strength vectors.

Table 16. Leave-one-out average accuracy in feet for the BR data. Results are averaged over 10 replications.

Model	Training Sample Size					
	5	10	20	50	100	253
Bayesian M_2	16.8	16.5	17.2	17.3	14.1	13.8
Bayesian M_2^{Inf}	13.8	19.2	18.2	15.3	15.2	17.5

Table 17. Leave-one-out average accuracy in feet for the CA Down data. Results are averaged over 10 replications.

Model	Training Sample Size				
	5	10	20	50	145
Bayesian M_2	21.3	26.3	25.0	20.3	18.7
Bayesian M_2^{Inf}	19.4	25.4	24.3	30.8	34.0

Table 18. Leave-one-out average accuracy in feet for the CA Up data. Results are averaged over 10 replications.

Model	Training Sample Size			
	5	10	20	55
Bayesian M_2	30.6	37.9	33.0	33.5
Bayesian M_2^{Inf}	29.3	29.4	33.4	31.3

Bayesian Hierarchical Diffuse (D) versus Informative (I) Priors, Error in Feet



Figure 14. Predictive accuracy of the hierarchical Bayesian graphical model (M_2) versus the hierarchical Bayesian graphical model with informative priors, CA Down data and CA Up Data.

Figure 15, Tables 19, 20 and 21 provide corresponding results for training data with no location information.

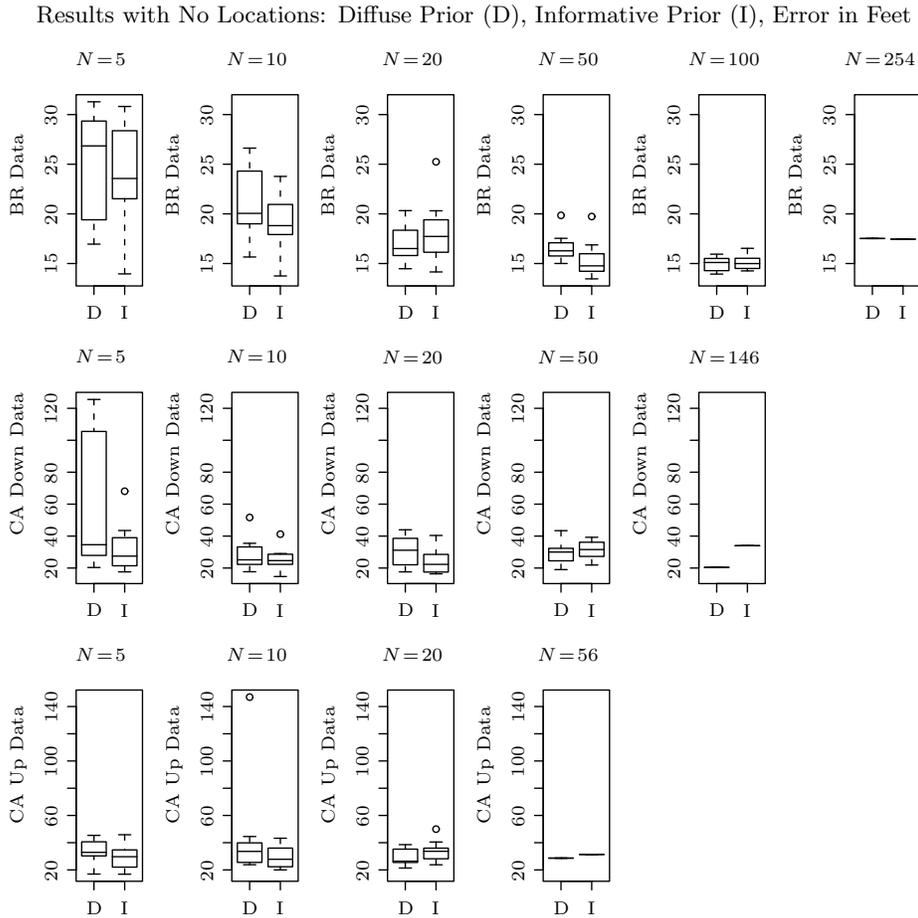


Figure 15. Predictive accuracy of the hierarchical Bayesian graphical model (M_2) versus the hierarchical Bayesian graphical model with informative priors and No Location Information, BR data, CA Down data and CA Up data.

Table 19. Leave-one-out average accuracy in feet for the BR data. No Location Information. Results are averaged over 30 replications. The corresponding standard errors range from about 1.3 to 2.5.

Model	Training Sample Size					
	5	10	20	50	100	254
Bayesian M_2 , No Locations	46.2	20.2	18.3	15.3	15.1	19.0
Bayesian M_2^{Inf} , No Locations	18.8	14.2	13.7	15.7	12.2	13.7

Table 20. Leave-one-out average accuracy in feet for the CA Down data. No Location Information. Results are averaged over 30 replications. The corresponding standard errors range from about 1.5 to 9.1.

Model	Training Sample Size				
	5	10	20	50	146
Bayesian M_2 , No Locations	23.9	29.4	29.2	29.8	21.9
Bayesian M_2^{Inf} , No Locations	18.0	31.8	18.0	22.4	20.0

Table 21. Leave-one-out average accuracy in feet for the CA Up data. No Location Information. Results are averaged over 30 replications. The corresponding standard errors range from about 2.7 to 6.8.

Model	Training Sample Size			
	5	10	20	56
Bayesian M_2 , No Locations	63.6	38.1	28.9	30.6.5
Bayesian M_2^{Inf} , No Locations	28.5	35.7	26.1	30.3

6. Future Work

Several directions for future work suggest themselves. In the first instance, we will explore several generalizations of the current model to include the following.

- Piecewise linear or spline-based models for the core signal strength-log distance relationship. The data show some evidence of non-linearity, especially at shorter distances. In particular we will explore the transformation selection algorithm of Hoeting, Raftery, and Madigan (2002).
- Models that can incorporate approximate location information. For example, when sensors are attached to wireline telephones, the room location may be available but not the location of the sensor within the room.
- Extensions of the corridor effects we discussed above to include more detailed information concerning wall locations as well as locations of potentially interfering objects such as elevators, kitchens, printers, etc.
- Incorporation of other data pertaining to the signal, such as angle of arrival.

Currently Markov chain Monte Carlo algorithms estimate the parameters and produce location estimates. For real-time or for larger-scale applications such simulation-based approaches may prove impractical and we will explore alternatives. In particular, variational approximations (see, for example, Jaakola and Jordan (2000)) may prove useful.

Since our experiments involve multiple test-training splits, manual MCMC convergence checking is not possible. We carried out several runs of 1,000,000 iterations for a few of the models and observed that predictive performance did

not improve. Nonetheless, some more systematic, automated approach to convergence diagnostics would be more satisfactory.

A major future thrust of our work will be to extend the current model to dynamic tracking applications. Such applications may begin with a known location (e.g., a location where a user takes a wireless device off a power rack) or not. In either case, the model will assume that the true location varies smoothly over time according to a low-order hidden stochastic process. Robotics has stimulated prior work in this direction and Monte Carlo algorithms for such applications exist. Work on so-called “particle filters” is relevant – see, for example, Thrun (2000), Gilks and Berzuini (2001) and Ridgeway and Madigan (2003). Again, alternatives to simulation such as online EM algorithms or quasi-Bayes (Opper (1998)) procedures may prove necessary.

Acknowledgements

The U.S. National Science Foundation supports Madigan’s work. Perl and BUGS code for the various models this paper describes are available from the first author.

References

- Bahl, P. and Padmanabhan, V. N. (2000). RADAR: An in-building RF-based user location and tracking system. In *Proceedings of IEEE Infocom 2000*. Tel Aviv, Israel.
- Bahl, P., Padmanabhan, V. N. and Balachandran, A. (2000). Enhancements to the RADAR User Location and Tracking System. Microsoft Research Technical Report, February 2000.
- Battiti, R., Brunato, M. and Villani, A. (2002). Statistical Learning Theory for Location Fingerprinting in Wireless LANs. Dipartimento di Informatica e Telecomunicazioni, Università di Trento, Technical Report DIT-020086.
- Christ, T. W. and Godwin, P. A. (1993). A Prison Guard Duress Alarm Location System. *Proceedings of the IEEE International Carnahan Conference on Security Technology*.
- Gelman, A., Carlin, J. B., Stern, H. S. and Rubin, D. B. (2003). *Bayesian Data Analysis*. Chapman and Hall, London.
- Gilks, W. R. and Berzuini, C. (2001). Following a moving target. *J. Roy. Statist. Soc. Ser. B* **63**, 127-146.
- Gilks, W. R., Richardson, S. and Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in Practice*. Chapman and Hall, London.
- Hassan-Ali, M. and Pahlavan, K. (2002). A new statistical model for site-specific indoor radio propagation prediction based on geometric optics and geometric probability. *IEEE Trans. Wireless Communication* **1**, 112-124.
- Heckerman, D., Geigen, D. and Chickering, D. (1995). Learning Bayesian network: The combination of knowledge and statistical data. *Machine Learning* **20**, 197-243.
- Hoeting, J., Raftery, A. E. and Madigan, D. (2002). A method for simultaneous variable and transformation selection in linear regression. *J. Comput. Graph. Statist.* **11**, 485-507.
- Howard, A., Siddiqi, S. and Sukhatme, G. S. (2003). An experimental study of localization using wireless ethernet. *The 4th International Conference on Field and Service Robotics*.

- Huang, G. T. (2003). Casting the Wireless Sensor Net. Technology Review, MIT's Magazine of Innovation, 51-56.
- Jaakola, T. and Jordan, M. I. (2000). Bayesian parameter estimation via variational methods. *Statist. Comput.* **10**, 25-37.
- Krishnan, P., Krishnakumar, A. S., Ju, W., Mallows, C., Ganu, S. (2003). A system for lease: System location estimation assisted by stationary emitters for indoor RF wireless networks. *Infocom*. To appear.
- Ladd, A. M., Bekris, K. E., Rudys, A., Marceau, G., Kavradi, L. E. and Dan, S. (2002). Robotics-based location sensing using wireless ethernet. In *Proceedings of the Eighth International Conference on Mobile Computing and Networking (MOBICOM)*. Atlanta, GA.
- Lauritzen, S. L. (1996). *Graphic Models*. Charendon Press, Oxford, 1996.
- Madigan, D. and York, J. (1995). Bayesian graphical models for discrete data. *Internat. Statist. Rev.* **63**, 215-232.
- Opper, M. (1998). A Bayesian approach to online learning. In *On-Line Learning in Neural Networks* (Edited by D. Saad). Cambridge University Press.
- Prasithsangaree, Krishnamurthy, P. and Chrysanthos, P.K. (2002). On Indoor Position Location With Wireless LANs, *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2002)*. Lisbon, Portugal.
- Priyantha, N. B., Chakraborty, A. and Balakrishnan, H. (2000). The Cricket Location Support System. *Proceedings of the Sixth Annual ACM International Conference on Mobile Computing and Networking*. Boston, MA.
- Ridgeway, G. and Madigan, D. (2003). A sequential Monte Carlo Method for Bayesian analysis of massive datasets. *J. Knowledge Discovery and Data Mining* **7**, 301-319.
- Roos, T., Myllymaki, P. and Tirri, H. (2002). A statistical modeling approach to location estimation. *IEEE Trans. Mobile Computing* **1**, 59-69.
- Saha, S., Chaudhuri, K., Sanghi, D. and Bhagwat, P. (2003). Location Determination of a Mobile Device using IEEE 802.11 Access Point Signals. *IEEE Wireless Communications and Networking Conference (WCNC) 2003*. New Orleans, Louisiana.
- Sato, M. and Ishii, S. (2000). On-line EM algorithm for the normalized gaussian network. *Neural Computation* **12**, 407-432.
- Smailagic, A., Siewiorek, D. P., Anhalt, J., Kogan, D. and Wang, Y. (2001). Location sensing and privacy in a context aware computing environment. *Pervasive Computing* 2001.
- Spiegelhalter, D. J. (1998). Bayesian graphical modeling: a case study in monitoring health outcomes. *Appl. Statist.* **47**, 115-133.
- Spiegelhalter, D. J. and Lauritzen S. L. (1990). Sequential updating of conditional probabilities on directed graphic structures. *Networks* **20**, 579-605.
- Spiegelhalter, D. J., Thomas, A. and Best, N. G. (1999). *WinBUGS Version 1.2 User Manual*. MRC Biostatistics Unit.
- Thrun, S. (2000). Probabilistic Algorithms in Robotics. Technical Report CMU-CS-00-126, Computer Science Department, Carnegie Mellon University.
- Want, R., Hopper, A., Falcao, V. and Gibbons, J. (1992). The Active Badge Location System. *ACM Trans. Information Systems* **10**, 91-102.
- Werb, J. and Lanzl, C. (1998). Designing a Positioning System for Finding Things and People Indoors. *IEEE Spectrum*, 71-78.
- Youssef, M., Agrawala, A. and Udaya Shankar, A. (2003). WLAN Location Determination via Clustering and Probability Distributions. *IEEE International Conference on Pervasive Computing and Communications (PerCom) 2003*. Fort Worth, Texas.

Department of Statistics, 501 Hill Center, Rutgers University, Piscataway, NJ 08854, U.S.A.
E-mail: madigan@rutgers.edu

Avaya Labs Research, 233 Mount Airy Rd. Basking Ridge, NJ 08852, U.S.A.

E-mail: whju@avaya.com

Avaya Labs Research, 233 Mount Airy Rd. Basking Ridge, NJ 08852, U.S.A.

E-mail: pk@avaya.com

Avaya Labs Research, 233 Mount Airy Rd., Basking Ridge, NJ 08852, U.S.A.

E-mail: ask@avaya.com

DIMACS Center, Rutgers University, 96 Frelinghuysen Rd., Piscataway, NJ 08854-8018, U.S.A.

E-mail: zorych@dimacs.rutgers.edu

(Received June 2005; accepted October 2005)