

Modeling spatially-correlated data of sensor networks with irregular topologies

Apoorva Jindal

Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, CA - 90089
Email: apoorvaj@usc.edu

Konstantinos Psounis

Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, CA - 90089
Email: kpsounis@usc.edu

Abstract—The physical phenomena monitored by sensor networks, e.g. forest temperature, usually yield sensed data that are strongly correlated in space. We have recently introduced a mathematical model for such data, and used it to generate synthetic traces and study the performance of algorithms whose behavior depends on this spatial correlation [1]. That work studied sensor networks with grid topologies.

This work extends our modeling methodology to sensor networks with irregular topologies. We describe a rigorous mathematical procedure and a simple practical method to extract the model parameters from real traces. We also show how to efficiently generate synthetic traces that correspond to sensor networks with arbitrary topologies using the proposed model. The correctness of the model is verified by statistically comparing synthetic and real data. Further, the model is validated by comparing the performance of algorithms whose behavior depends on the degree of spatial correlation in data, under real and synthetic traces. The real traces are obtained from both publically available sensor data, and sensor networks that we deploy. Finally, we augment our existing trace-generation tool with new functionality suited for sensor networks with irregular topologies.

I. INTRODUCTION

The wireless sensor networks of the near future are envisioned to consist of a large number of inexpensive wireless nodes. These nodes will operate under significant power constraints, which precludes them from using large transmission ranges. This, together with the low cost of individual sensors, implies that sensors will be densely deployed. As a result, it is expected that a high degree of spatial correlation will exist in the sensor network data. Many algorithms have been proposed that exploit this correlation. For example, spatial correlation has been used in data aggregation and routing algorithms [2]–[5], data storage and querying [6], [7], [8], mac protocol design [9], data compression and encoding [10], and calibration [11].

The evaluation of protocols that are sensitive to the spatial features of input data requires data representing a wide range of realistic conditions. However, since very few real systems have been deployed, there is hardly any experimental data available to test the proposed algorithms. As a result, sensor network researchers make different assumptions when generating data inputs to evaluate systems; some assume the data to be jointly Gaussian with the correlation being a function of the distance [9], some assume that the data follows the

diffusion property [8], and some assume a function for the joint entropy of the data [3]. Other researchers propose the use of environmental monitoring data obtained from remote sensing [6], however the granularity of these data sets do not match the expected granularity of sensor networks' data. With this in mind, the authors in [12] proposed a method to interpolate existing experimental data to support irregular topologies and increase granularity.

The goal of this paper and its predecessor [1] is to come up with a parsimonious mathematical model that can capture spatial correlation of any degree irrespective of the granularity, density, number of source nodes or topology. There are many benefits from such an endeavor. First, the model will provide a procedure to synthetically generate sensor networks data exhibiting various degrees of correlation, enabling a meticulous study of the performance of proposed algorithms. Second, it will enable different researchers to evaluate different algorithms using a common trace generation method, which, in turn, will make comparisons between different algorithms meaningful. In other words, the model can serve as a benchmark. Third, it will provide guidelines for designing algorithms that exploit correlations in an optimal manner.

In [1] we introduced such a model and used it to generate synthetic traces. We established the validity of the model by comparing the statistical properties of original (environmental) and synthetic data, and the performance of sensor network algorithms, whose behavior depends on the spatial correlation, under original and synthetic data.

The main drawback of this first attempt is that it requires the sensor network to have a grid topology. Unfortunately, fixing this problem is not straightforward. The simple approach of generating a huge number of grid-based data and then keeping only the values that correspond to sensor locations is computationally very wasteful. Further, when this approach is used, the model parameters depend on the specific node locations; different locations would yield different parameters to describe the same correlation structure. Hence, the current model is unsuitable for irregular topologies.

The main contribution of this work is the introduction of a new, more general flavor of the model in [1] that can be efficiently used for arbitrary topologies. The model comes with a rigorous mathematical procedure and a simple practical

method to extract the model parameters from real traces. It also comes with an efficient method to generate synthetic traces that correspond to irregular topologies. Further, it is shown that prior approaches to model spatial correlations, e.g. assuming that data are jointly Gaussian, are subcases of our more general model. Finally, publicly available trace-generation tools are created as part of this work.

The paper is organized as follows. Section II introduces the variogram, which is a handy metric to characterize spatial correlation in data. Section III summarizes our model for sensor networks with grid topologies and motivates the need for a new version. The new model is presented in Section IV, followed by a mathematically rigorous procedure and a simple, practical method to infer the model parameters in Section V. In Section VI we verify the correctness of the model by statistically comparing synthetic and real data, and validate it by comparing the performance of a well know algorithm, CMAC [9], under the experimental and the synthetic data. The real traces are obtained from both publically available structural health monitoring data, and sensor networks that we deploy. Finally, Section VII describes the trace-generation tools and Section VIII concludes the work.

II. VARIOGRAM: A STATISTIC TO MEASURE CORRELATION IN DATA

A statistic often used to characterize spatial correlation in data is the variogram [12]–[14]. Given a two dimensional stationary process $V(x, y)$, the variogram (also called semi-variance) is defined as

$$\gamma(d_1, d_2) = \frac{1}{2} E[(V(x, y) - V(x + d_1, y + d_2))^2].$$

For isotropic random processes [15] the variogram depends only on the distance $d = \sqrt{d_1^2 + d_2^2}$ between two nodes.¹ In this case, if (x_d, y_d) denotes a point which is d distance away from (x, y) ,

$$\gamma(d) = \frac{1}{2} E[(V(x, y) - V(x_d, y_d))^2], \quad (1)$$

where $(x - x_d)^2 + (y - y_d)^2 = d^2$.

For a set of samples $v(x_i, y_i)$, $i = 1, 2, \dots$ on a regular grid, $\gamma(d)$ can be estimated as follows,

$$\hat{\gamma}(d) = \frac{1}{2m(d)} \sum_1^{m(d)} [v(x_i, y_i) - v(x_j, y_j)]^2,$$

where $m(d)$ is the number of points at a distance d within each other, i.e. the sum is over all points for which $(x_i - x_j)^2 + (y_i - y_j)^2 = d^2$.

A straightforward method to estimate the variogram for a set of samples on an irregular grid consists of the following steps: (i) for every pair of samples, compute the distance between them and the squared difference between their data values, (ii) make a scatter plot of the variogram values against the

distance, and (iii) curve fit the scatter plot to obtain an estimate of the variogram.

A more statistically robust method, traditionally used in Geostatistics [15], [16], consists of the following steps: (i) as before, for every pair of samples compute the distance between them and the squared difference between their data values, (ii) divide the entire range of distance into discrete intervals with an interval size being equal to the average distance to the nearest neighbor, (iii) assign each of the pair of samples to one of the distance intervals and compute the average variance in each interval by dividing the sum of the squared-differences between data-values by the total number of pairs lying in that distance interval, and (iv) assign the average variance to the mid point of each interval and curve fit these points to one of the standard variogram models [15], [16].

In this paper, we will use the second method to estimate the variogram from the experimental traces.

A. Relationship between the variogram and the covariance

Another very commonly used statistic to measure correlation in data is the covariance. For a two dimensional isotropic stationary process $V(x, y)$, the covariance is defined as

$$C(d) = E[(V(x, y) - \mu)(V(x_d, y_d) - \mu)],$$

where (x_d, y_d) is denotes a point d distance away from (x, y) and $\mu = E[V(x, y)]$.

Since both the variogram and the covariance are measures of the correlation in data, we derive the relationship between them and verify that both of them can be used interchangeably. From Equation (1),

$$\begin{aligned} \gamma(d) &= \frac{1}{2} E[(V(x, y) - V(x_d, y_d))^2] \\ &= \frac{1}{2} E[((V(x, y) - \mu) - (V(x_d, y_d) - \mu))^2] \\ &\Rightarrow \gamma(d) = \sigma_V^2 - C(d), \end{aligned} \quad (2)$$

where $\sigma_V^2 = E[(V(x, y) - \mu)^2]$ is the variance of the process $V(x, y)$.

Equation (2) implies that a lower (higher) value of the variogram implies a higher (lower) value of the covariance and correlation. Figure 1 plots the variogram and the covariance for a trace generated by assuming a jointly Gaussian model for the spatial data [17].

A characteristic of the variogram which can be inferred from the plot is that it levels off (becomes parallel to the x-axis) at a distance beyond which the covariance or the correlation between the samples go to zero. Further, the constant value to which the variogram saturates is equal to the variance of the process.

Since both metrics can be interchangeably used, in this paper we will only present variogram plots.

¹We will use the Euclidean distance to measure distances between two points.

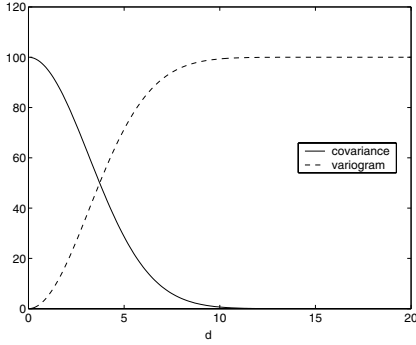


Fig. 1. Variogram and Covariance plots for a trace generated by assuming jointly Gaussian model for the spatial data.

III. MODEL FOR SPATIALLY CORRELATED DATA FOR A NETWORK WITH GRID TOPOLOGY

In this section, we summarize the model proposed in [1]. Consider a sensor network whose nodes reside on a grid topology. Let $V(x, y)$ be the data value at node (x, y) . It is assumed that $V(x, y)$ is a stationary process that has a unique first order distribution whose probability density function (pdf) is denoted by $f_V(v)$. (This is referred to as the long term distribution of the data.)

Let $N(d)$ denote the number of nodes at a distance d from (x, y) . Let V_d denote the data value at a node which is d distance away from (x, y) , and V_d^k denote the data value at the k^{th} node ($1 \leq k \leq N(d)$) at a distance d from (x, y) . Following is the model for generating the data values,

$$V(x, y) = \begin{cases} V_1^1 + Z & \text{with probability } \frac{\alpha_1}{N(1)} \\ \vdots \\ V_1^{N(1)} + Z & \text{w.p. } \frac{\alpha_1}{N(1)} \\ V_2^1 + Z & \text{w.p. } \frac{\alpha_2}{N(2)} \\ \vdots \\ V_2^{N(2)} + Z & \text{w.p. } \frac{\alpha_2}{N(2)} \\ \vdots \\ V_h^1 + Z & \text{w.p. } \frac{\alpha_h}{N(h)} \\ \vdots \\ V_h^{N(h)} + Z & \text{w.p. } \frac{\alpha_h}{N(h)} \\ Y & \text{w.p. } \beta \end{cases}, \quad (3)$$

where Z and Y are random variables independent of each other as well as V , and their pdf's are denoted by $f_Z(z)$ and $f_Y(y)$ respectively. Both Y and Z determine the long term distribution of V , and Z captures the small differences between neighboring data values. The above equation simply says that the probability that $V(x, y)$ is derived from the value of any node which is d distance away from (x, y) is α_d . Further, the probability that $V(x, y)$ is derived from the value of a particular such node is $\frac{\alpha_d}{N(d)}$. The parameters of the model are h , the α_i 's, β , $f_Y(y)$ and $f_Z(z)$. Since the sum of

probabilities should equal one, $\beta + \sum_{i=1}^h \alpha_i = 1$.

Remark: For a grid, the L1 or manhattan distance is a meaningful way to measure distances between two nodes. (The L1 distance between two nodes (x_1, y_1) and (x_2, y_2) is given by $d = |x_1 - x_2| + |y_1 - y_2|$.) Thus, the distances between nodes on a grid are in multiples of the minimum inter-sensor distance which is equal to the size of the grid. This simplifies the model structure as both the variogram and the α 's can be viewed as a discrete function of distance.

A. Motivation for a new model

One way to extend the model in [1] to an irregular topology is to convert the irregular topology to a grid topology by adding more nodes, generate data at all these nodes, and then discard the additional nodes. The problem with this approach is that it is computationally very expensive. We perform a simple calculation to obtain at how many additional nodes we have to generate data. Let us randomly distribute n nodes in a square of side 1. Let the locations of each node (x_i, y_i) , be chosen uniformly and independently of each other. Let p denote the minimum coordinate distance between the nodes, that is $p = \min_{1 \leq i \leq n, 1 \leq j \leq n} [\min(|x_i - x_j|, |y_i - y_j|)]$. It is easy to see that we need to generate data for at least $\frac{1}{p^2}$ nodes.

The probability that k out of n nodes distributed randomly in an interval of size 1 lie in an interval of size d equals $\binom{n}{k} (d)^k (1-d)^{n-k}$ which is Binomial (n, d) . For a very large n , Binomial (n, d) can be approximated by Poisson (nd) . So, the inter sensor distance is distributed as Exponential (nd) . The minimum inter sensor distance in one dimension will be the minimum of n exponentials, and hence is distributed as Exponential $(n^2 d)$. Since the x coordinate and the y coordinate of each node is chosen independently of each other, the minimum coordinate distance p is distributed as Exponential $(2n^2 d)$. Hence, on average, we will have to generate at least $O(n^4)$ additional nodes to get the data values on n nodes. This simple calculation shows that it is computationally very expensive to populate data on an irregular topology by converting it to a grid topology first.

Even if we have the capability to perform these calculations, a change in the node locations will change the model parameters as the value of p depends on the actual node locations. This is problematic, since the model parameters should only depend on the physical phenomenon being monitored, and not on the actual node locations. Hence, modeling data on an irregular topology by converting the irregular topology to a grid topology is not appropriate, which motivates the need to explore more well-suited models.

IV. MODEL FOR AN IRREGULAR GRID

In this section we introduce our model for capturing the statistical properties of sensor networks data. For ease of notation, we use polar coordinates to define node locations. We assume that nodes are distributed in a circle of unit radius. Let $V(r, \theta)$ be the data value at node (r, θ) , where $0 < r < 1$ and

$0 < \theta < 2\pi$. We assume that $V(r, \theta)$ is a stationary isotropic process that has a unique first order distribution denoted by $f_V(v)$.

Without loss of generality and to simplify exposition, we assume that we want to generate the data value at the origin. We propose the following model to do so:

$$V(0,0) = I_{(U=T)}Y + I_{(U=H)} \int_{\theta} \int_r (V(r, \theta) + Z) \delta(R=r) dr \delta(\Theta=\theta | R=r) r d\theta, \quad (4)$$

where:

- U represents a coin that when it lands heads (H), with probability $1 - \beta$, the origin's data value is obtained from neighboring nodes, and when it lands tails (T), with probability β , it is obtained from a random variable Y . (I_A denotes an indicator function that equals one when event A occurs and equals zero otherwise.)
- Similarly to the regular topology model, Y and Z are random variables independent of each other as well as V , with pdf's $f_Y(y)$ and $f_Z(z)$ respectively. Y models the situation where the origin's data value is not obtained from neighboring nodes, Z captures the small differences between neighboring data values, and both of them determine the long term distribution of V .
- R is a random variable with pdf $f_R(r)$. When $R = r$, the origin's data value is obtained from locations at distance r from the origin. ($\delta(R=r)$ denotes a δ -function of R that is non-zero when $R=r$.) $f_R(r)dr$ is the probability of this event. $f_R(r)$ is a parameter of our model and from now on we refer to it as $\alpha(r)$, since it corresponds to the α_i 's of the grid topology model.
- Θ is a random variable with pdf $f_{\Theta}(\theta)$. When $\Theta = \theta | R = r$, the origin's data value is obtained from locations at angle θ given that their distance from the origin is r . $f_{\Theta|R}(\theta | r)r d\theta$ is the probability of this event. We assume that θ is uniformly distributed between angles θ_1 and θ_2 . Thus,

$$f_{\Theta|R}(\theta | r) = \begin{cases} \frac{1}{(\theta_2 - \theta_1)r} & \theta_1 < \theta < \theta_2 \\ 0 & \text{otherwise} \end{cases}$$

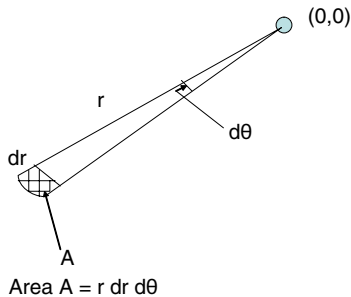


Fig. 2. The probability that the data value at $(0,0)$ is derived from a node in region A is $\frac{\alpha(r)}{(\theta_2 - \theta_1)r} r dr d\theta$.

Given the above, the cdf of $V(0,0)$ can be expressed as follows,

$$P(V(0,0) \leq v) = \beta P(Y \leq v) + (1 - \beta) \int_{\theta} \int_r P(V(r, \theta) + Z \leq v) \frac{\alpha(r)}{(\theta_2 - \theta_1)r} r dr d\theta. \quad (5)$$

Equation (4) and (5) simply say that the the probability that the data value at a node is directly derived from a node lying in the shaded region A in Figure 2 is $\frac{\alpha(r)}{(\theta_2 - \theta_1)r} r dr d\theta$. $\alpha(r)dr$ is the probability that a node's data value is derived from any node at a distance r away from it. The number of nodes r distance away and lying in an arc of $(\theta_2 - \theta_1)$ is proportional to $(\theta_2 - \theta_1)r$. Now, given that the node's data value is derived from a node r distance away, the probability that it is derived from a node in an arc of $d\theta$ is $\frac{r d\theta}{(\theta_2 - \theta_1)r}$.

The parameters of the model are $\alpha(r)$, β , $f_Y(y)$ and $f_Z(z)$. The values of θ_1 and θ_2 depend on the method used to populate data. We will explain their role in more detail in Section IV-A.

$\alpha(r)$ will be a decreasing function of r as the correlation between nodes decreases as their distance increases. Throughout this paper, we assume that $\alpha(r)$ is zero for $r \geq r_{max}$ for some value of r_{max} .

Now, since the pdf's should integrate out to 1, we get the following equation,

$$\int_0^{r_{max}} \int_{\theta_1}^{\theta_2} \frac{\alpha(r)}{(\theta_2 - \theta_1)r} r dr d\theta = 1 \Rightarrow \int_0^{r_{max}} \alpha(r) = 1. \quad (6)$$

A. Instantiation of the model

In a real life scenario, the exact node locations determined through some location distribution will be given as an input and the user should be able to generate data values at these nodes using the model. In this section we describe how to generate the data using an instantiation of the model.

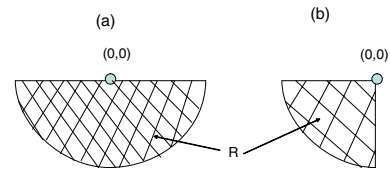


Fig. 3. Two methods to populate data. (a) Semi Circular Dependence: The data value at node $(0,0)$ can be directly derived from any node lying in the semi circular region. (b) Quarter Circular Dependence: The data value at node $(0,0)$ can be directly derived from any node lying in the quarter circular region.

Before we proceed, we look at how the values of θ_1 and θ_2 effect the population of data. A couple of examples are given in Figure 3. The first method corresponds to population of data using a semi circular data dependence while the second method corresponds to a quarter circular data dependence. Quarter circular data dependence implies that a node's data value can be directly derived from only those nodes which lie

in the shaded region R which is quarter of a circle centered at the node. The values of θ_1 and θ_2 are π and $\frac{3\pi}{2}$ for quarter circular dependence and π and 2π respectively for semi circular dependence. Which method to choose will depend on the physical phenomenon being modeled. The default data population method in the rest of the paper is going to be the quarter circular data dependence ($\theta_1 = \pi$ and $\theta_2 = \frac{3\pi}{2}$).

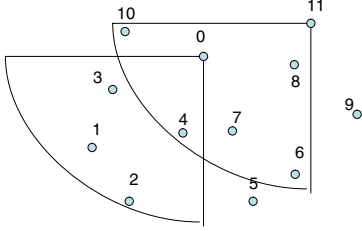


Fig. 4. An example topology.

As an example, consider the node locations given by Figure 4. Let the node location of node i be (r_i, θ_i) , $0 \leq i \leq 11$, and let the data values at these nodes be denoted by $V(r_i, \theta_i)$. The instantiation of the model for node $(0, 0)$ for a quarter circular data dependence is as follows:

$$V(0, 0) = \begin{cases} V(r_1, \theta_1) + Z & \text{with probability } c \frac{\alpha(r_{01})}{r_{01}} \\ V(r_2, \theta_2) + Z & \text{w.p. } c \frac{\alpha(r_{02})}{r_{02}} \\ V(r_3, \theta_3) + Z & \text{w.p. } c \frac{\alpha(r_{03})}{r_{03}} \\ V(r_4, \theta_4) + Z & \text{w.p. } c \frac{\alpha(r_{04})}{r_{04}} \\ Y & \text{w.p. } c\beta \end{cases} \quad (7)$$

Equation (7) is very similar to the model for grid topologies in spirit. Both Equations (7) and (3) are instantiations of the model described by Equation (4). However, there are some differences that it is worth mentioning: First, $\alpha(r)$ in Equation (7) is no longer a discrete function of distance. Second, since the number of nodes at a distance r is proportional to r , instead of $N(r)$ (see Equation (3)) we have r in the denominator of the terms of Equation (7). Third, note the presence of the scaling constant c in Equation (7) which is present to make the sum of probabilities go to one. (c will depend on the exact node locations and will change when these change, but the model parameters remain the same.)

Equation 7 assumes that the data values at nodes lying in the data dependence region of $V(0, 0)$ have already been populated. Thus, an order of populating data has implicitly been assumed. A valid ordering to populate data will ensure that when a data value at a node is populated, the data value at all the nodes lying in its data dependence region have already been populated. Also, before starting to populate the data we randomly initialize the values that are within the data dependence region of the first node we populate.

B. How the model parameters affect correlation

The presence of many parameters in the model gives us great flexibility to model processes having different correlation

properties. In this section, we study how different parameters affect the correlation properties of the generated data.

We use the simple linear topology shown in Figure 5. Synthetic traces are generated using the model under a 20000 node scenario. We assume $Y \sim N(0, 10)$ and $Z \sim N(0, \sigma_z)$.

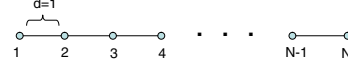


Fig. 5. Simple linear topology used to study the effect of model parameters on the correlation in data.

1) *Effect of β* : Since β governs the probability with which a node will choose a random value independent of every other node, it is expected that a lower value of β will lead to a higher value of correlation. Also, a variation in β will change the distribution of V . The exact relationship between the two will be derived in Section V-A.

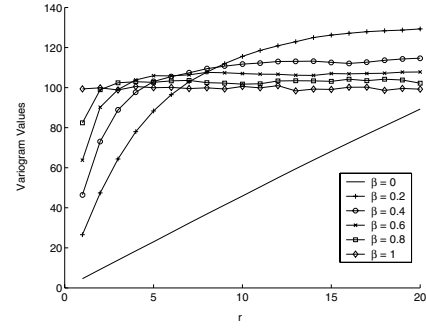


Fig. 6. Effect on the correlation structure of the data when β is varied keeping all the other parameters constant.

Figure 6 plots the variogram for traces generated using different values of β . The other parameters are: $r_{max} = 2$, $\alpha(r) = \lambda 2^{-r}$ for $0 < r < r_{max}$ and 0 otherwise, and $\sigma_z = 3$. The plots show that as the value of β decreases, not only does the distance at which the variogram levels off (the distance beyond which the nodes are uncorrelated) increase, but also the y-value to which it levels off increases.

2) *Effect of r_{max}* : If the distance between the nodes is more than r_{max} , then they cannot be directly derived from each other. Hence, we expect that increasing r_{max} will increase the distance at which the variogram levels off.

Figure 7 plots the variogram for traces generated using different values of r_{max} . The other parameters are: $\alpha(r) = \lambda 2^{-r}$ for $0 < r < r_{max}$ and 0 otherwise, $\beta = 0.4$ and $\sigma_z = 4$. A look at the variograms tells us that correlation between the data values is independent of the value of r_{max} .

This observation is contrary to our initial intuition and hence requires a more detailed explanation. We take this opportunity to highlight a key characteristic of our model. If node 2 is derived from node 1, and node 3 is derived from node 2, then node 1 and node 3 will show a strong correlation too. So, even if r_{max} is small, when β is small, nodes having distances much larger than r_{max} will have high correlation.

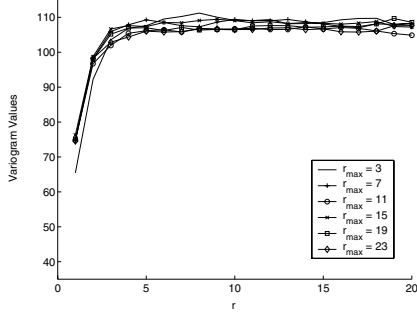


Fig. 7. Effect on the correlation structure of the data when r_{max} is varied keeping all the other parameters constant.

Thus, we infer that the distance at which the variogram levels off depends primarily on β .

3) *Effect of σ_z* : Finally, we study whether changing $f_Z(z)$ will effect the correlation in data. We had assumed $f_Z(z)$ to be $N(0, \sigma_z)$. Traces for different values of σ_z are generated and their variograms are plotted in Figure 8. The other parameters are: $r_{max} = 2, \alpha(r) = \lambda 2^{-r}$ for $0 < r < r_{max}$ and 0 otherwise, and $\beta = 0.4$.

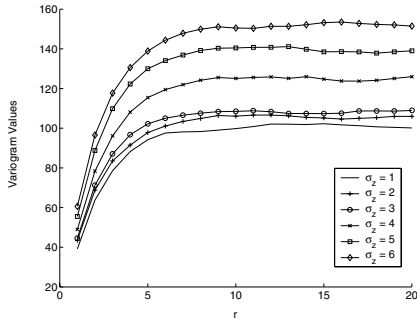


Fig. 8. Effect on the correlation structure of the data when σ_z is varied keeping all the other parameters constant.

It can be easily seen from the plots that the σ_z does not effect the correlation structure of the data, though it has a significant effect on the distribution of V . The value at which the variogram saturates to, which is the variance of V , increases as σ_z increases.

Remark: While evaluating the performance of algorithms for different correlation structures, it is useful to have a single tunable parameter whose value determines the level of correlation in data. For our model, this tunable parameter is β . Traces with different correlation structures can be generated by tuning β from 0 to 1 and the performance of the algorithm can be plotted against β .

V. INFERRING MODEL PARAMETERS

In this section, we present techniques for inferring model parameters from real traces.

The model parameters to be inferred are $\alpha(r)$, β , $f_Y(y)$, r_{max} and $f_Z(z)$. Without loss of generality, from now onwards we will assume that Z is a normal random variable with zero mean and standard deviation $\sigma = \sigma_z$. Note that the distribution

of Z need not necessarily be Gaussian; any other distribution will not effect the model, though the analysis presented in this section will be modified.

First, in Section V-A we derive the relationship between $f_V(v)$, $f_Y(y)$ and σ_z . Note that $f_V(v)$ can be easily estimated by its empirical distribution. Then, in Section V-B we present a rigorous procedure to infer $\alpha(r)$, β , r_{max} and σ_z from a real trace. But this procedure involves solving integral equations [18], [19] and hence, it is not always possible to obtain a closed form expression for the model parameters. So, in Section V-C, we present a simple, practical method that uses both the discrete and the continuous models.

A. Relationship between the distributions of V , Y and Z

In this section, we derive the relationship between the distributions of V , Y and Z .

From Equation (5) we get,

$$f_V(v) = (1 - \beta)f_{V+Z}(v) + \beta f_Y(v). \quad (8)$$

Using characteristic functions and since V and Z are independent, Equation (8) can be rewritten as,

$$\Phi_V(j\omega) = (1 - \beta)\Phi_V(j\omega)\Phi_Z(j\omega) + \beta\Phi_Y(j\omega). \quad (9)$$

Since Z is a zero mean normal random variable, its characteristic function is given by $e^{[-\frac{\sigma_z^2 \omega^2}{2}]}$. Equation (9) finally reduces to

$$\Phi_V(j\omega) = \frac{\beta}{1 - (1 - \beta)e^{[-\frac{\sigma_z^2 \omega^2}{2}]}} \Phi_Y(j\omega). \quad (10)$$

For mathematical convenience, we define a new random variable having a characteristic function given by

$$\Phi_L(j\omega) = \frac{\beta}{1 - (1 - \beta)e^{[-\frac{\sigma_z^2 \omega^2}{2}]}}.$$

Equation (10) can now be rewritten as

$$V = Y + L. \quad (11)$$

B. A rigorous procedure to infer the model parameters

In this section, we present a rigorous method to infer the model parameters. First we compute the variogram $\gamma(r)$ using the model and then equate it with its estimate $\hat{\gamma}(r)$ obtained from the real trace.

Using Equation (1),

$$\begin{aligned} \gamma(r) &= \frac{1}{2} \int_0^{2\pi} \frac{1}{2\pi} E[(V(0,0) - V(r,\theta))^2] d\theta \\ &= \frac{1}{2}(1 - \beta) \int_0^{2\pi} \frac{1}{2\pi} \int_0^{r_{max}} \int_{\theta_1}^{\theta_2} \\ &E[(V(r',\theta') + Z - V(r,\theta))^2] \frac{\alpha(r')}{\theta_2 - \theta_1} dr' d\theta' d\theta \\ &+ \frac{1}{2} \int_0^{2\pi} \frac{\beta}{2\pi} E[(Y - V(r,\theta))^2] d\theta. \end{aligned} \quad (12)$$

The term $E[(V(r', \theta') + Z - V(r, \theta))^2]$ in the above equation can be expanded as,

$$E[(V(r', \theta') + Z - V(r, \theta))^2] = E[(V(r', \theta') - V(r, \theta))^2] + E[Z^2] = 2\gamma \left(\sqrt{r^2 + r'^2 - 2rr' \cos(\theta - \theta')} \right) + \sigma_z^2.$$

The second term in Equation (12) $E[(Y - V(r, \theta))^2]$ is equal to $E[L^2]$. Using Equation (11), $E[L^2]$ is evaluated to be $\frac{(1-\beta)\sigma_z^2}{\beta}$.

Substituting all of the above in Equation (12),

$$\gamma(r) = (1 - \beta)\sigma_z^2 + (1 - \beta) \int_0^{r_{max}} \int_{\theta_1}^{\theta_2} \int_0^{2\pi} \frac{1}{2\pi} \frac{\alpha(r')}{\theta_2 - \theta_1} \gamma \left(\sqrt{r^2 + r'^2 - 2rr' \cos(\theta - \theta')} \right) d\theta d\theta' dr'. \quad (13)$$

Equation (13) gives the relationship between the variogram and the model parameters $\alpha(r)$, β , σ_z and r_{max} . Substituting $\gamma(r)$ with its estimation $\hat{\gamma}(r)$ in Equation (13) gives us an integral equation of the first kind [18], [19], which along with the conditions $\int_0^{r_{max}} \alpha(r) dr = 1$ and $\alpha(r_{max}) = 0$ form a system of equations with one unknown function $\alpha(r)$ and three unknown constants β , σ_z and r_{max} . Solving these equations will give us the model parameters. After obtaining σ_z and β , $f_Y(y)$ is obtained through Equation (10).

In Equation (13), the unknown function $\alpha(r)$ is inside an integral. In general, it is not possible to find closed form solutions for $\alpha(r)$ for every variogram function. In the next section, we assume a specific variogram function that corresponds to a covariance function commonly used in the sensor networks literature, and solve for $\alpha(r)$.

1) *Case Study:* In this section, we will find model parameters for a trace having the following variogram,

$$\gamma(r) = c(1 - e^{-r^2}) \quad 0 < r < 0.5. \quad (14)$$

The corresponding covariance is $C(r) = ce^{-r^2}$ which is a very commonly assumed correlation structure for spatially correlated data in the sensor networks literature, see, for example, [17], [20]. Note that these papers also assume the data to be jointly Gaussian, whereas we don't make any such assumption here. Actually, the jointly Gaussian scenario is a subclass of our model, as discussed in Section VIII.

To find the model parameters, we have to solve the following integral equation:

$$c(1 - e^{-r^2}) = (1 - \beta) \int_0^{r_{max}} \int_{\theta_1}^{\theta_2} \int_0^{2\pi} \frac{1}{2\pi} \frac{\alpha(r')}{\theta_2 - \theta_1} \left(1 - e^{-(r^2 + r'^2 - 2rr' \cos(\theta - \theta'))} \right) d\theta d\theta' dr' + (1 - \beta)\sigma_z^2. \quad (15)$$

Before venturing into the solution of the above equation, we first integrate out θ and θ' ,

$$\int_{\theta_1}^{\theta_2} \int_0^{2\pi} \frac{1}{2\pi} \frac{1}{\theta_2 - \theta_1} \left(1 - e^{-r^2} e^{-r'^2} e^{2rr' \cos(\theta - \theta')} \right) d\theta d\theta'. \quad (16)$$

Since $0 < r, r' < 0.5 \Rightarrow 2rr' \cos(\theta - \theta') < 1$, by neglecting the square terms and beyond, the last term in the above equation can be approximated by,

$$e^{2rr' \cos(\theta - \theta')} = 1 + 2rr' \cos(\theta - \theta').$$

Note that the assumption on the size of the sample area $0 < r < 0.5$ has been made to enable the above approximation, otherwise it is not possible to find a closed form solution for $\alpha(r)$.

We assume the semi circular data dependence to populate data, hence $\theta_1 = \pi$ and $\theta_2 = 2\pi$. With the above approximation, Equation (16) reduces to $c(1 - e^{-r^2} e^{-r'^2})$.

Substituting in Equation (15),

$$c(1 - e^{-r^2}) = c(1 - \beta) \int_0^{r_{max}} \alpha(r') \left(1 - e^{-r^2} e^{-r'^2} \right) dr' + (1 - \beta)\sigma_z^2. \quad (17)$$

Using the method described in [18] to solve for integral equations, we determine that $\alpha(r)$ has the form $a + be^{-r^2}$ where a and b are constants to be determined by the boundary conditions $\int_0^{r_{max}} \alpha(r) dr = 1$ and $\alpha(r_{max}) = 0$. Solving them yields $b = \left(\frac{\sqrt{\pi} \text{Erf}(r_{max})}{2} - r_{max} e^{-r_{max}^2} \right)^{-1}$ and $a = -be^{-r_{max}^2}$, where $\text{Erf}(x)$ is the error function defined as $\text{Erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$.

Now substituting $\alpha(r) = a + be^{-r^2}$ in Equation (17) gives $\beta = 1 - \frac{4}{\sqrt{\pi}} (2a \text{Erf}(r_{max}) + \sqrt{2} b \text{Erf}(\sqrt{2} r_{max}))^{-1}$ and $\sigma_z^2 = \frac{c\beta}{1-\beta}$.

We still need to determine the value of r_{max} . Any value of r_{max} would do, as long as the resulting β is between 0 and 1 -since it is a probability-, and the resulting $\alpha(r)$ is positive for all r -since it is a pdf-. In this example, we choose the largest r_{max} value that satisfies both constraints. In particular, we start with $r_{max} = 0.5$, and keep on reducing its value till we obtain a positive value of β . ($\alpha(r)$ is positive for all r for this value of r_{max} .) The model parameters turn out to be, $r_{max} = 1 \times 10^{-5}$, $\beta = 2.3 \times 10^{-6}$ and $\sigma_z^2 = 0.0023$.

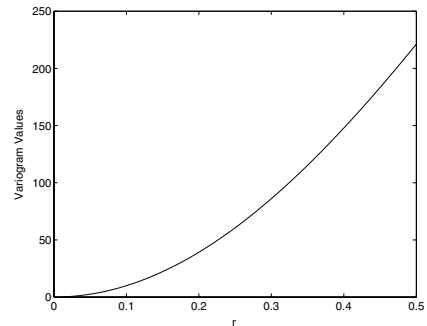


Fig. 9. The given variogram $\gamma(r) = 1000(1 - e^{-r^2})$.

Lets see if these values make sense. To do so, first we plot the variogram of Equation (14) in Figure 9. Drawing from the discussion in Section IV-B, we can easily argue that for the

given correlation structure, the values of β and r_{max} should be very small which is consistent with the values we obtain.

C. A simple method to infer the model parameters

In this section we present a simpler procedure to infer the model parameters. Several numerical techniques to solve integral equations exist in the literature. But integral equations of the first kind are *inherently ill posed problems* [21] and hence, their solutions are generally unstable. This ill-posedness makes numerical solutions very difficult, as a small error can lead to an unbounded error.

So, we present a practical simpler method which uses the discrete model [1] to infer the model parameters and the continuous model to populate the data and generate traces.

The first step is to use the method described in Section II to obtain a discretized variogram, which corresponds to the continuous variogram sampled at multiples of the average nearest neighbor distance. The second step is to use the method described in [1] to obtain a discrete version $\alpha[r]$ of $\alpha(r)$, which corresponds to the continuous $\alpha(r)$ sampled at multiples of the average neighbor distance. Note that the method in [1] is similar to the method described in Section V-B. Except now, instead of integrating over infinitesimally small areas each having at most one node in them, like area A in Figure 2, we sum over square regions as shown in Figure 10, assuming one node resides in the center of each square.

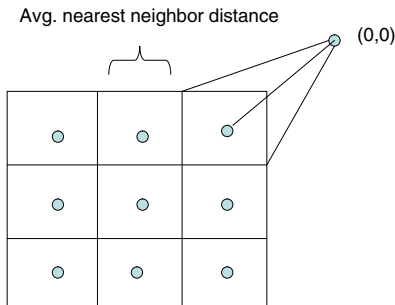


Fig. 10. In the approximate method, instead of integrating over the region A, we sum over the square regions.

Since the square area is no longer infinitesimally small, the integrals in Equation (13) are replaced by sums. The resultant system of linear equations can be easily solved to obtain the model parameters, β , σ_z and $\alpha[r]$.

Finally, the $\alpha[r]$'s are interpolated or curve fitted to obtain the continuous $\alpha(r)$. After obtaining the model parameters, we use the model described in Section IV to generate synthetic traces.

This procedure formulates the problem in continuous domain, converts it to the discrete domain by sampling, solves it in the discrete domain and transforms the solution back to the continuous domain by interpolation. Intuitively, this procedure is very similar to several signal processing techniques, for

example using the FFT to find the fourier transform of a continuous signal. Obviously, as in the signal processing techniques, the distance between the two neighboring samples (which is the average nearest neighbor distance for the given procedure) has an important role to play. The larger the number of samples in an area, the smaller the average nearest-neighbor distance, and the more accurate is the estimation of the model parameters.

VI. MODEL VERIFICATION AND VALIDATION

In this section, the model parameters for experimental traces are inferred using the method described in Section V-C. Then these model parameters are used to generate synthetic traces. We verify our model by comparing the variograms of the original experimental traces and the corresponding synthetic traces, and then we validate it by comparing the performance of an algorithm which exploits spatial correlation, against both the traces.

A. Data Set Description

We need actual sensor network traces to be able to verify and validate our model. In this section, we describe the traces which we have used for verification and validation purposes.

1) *SHM Trace*: One of the real world experiments where real sensor network traces have been collected after deploying a sensor network is reported in [22]. A 14 MicaZ node sensor network was deployed in a large seismic test structure used by civil engineers to study structural health monitoring (SHM). Accelerometers on the sensors collected vibration samples from the structure and send them to a base station using a data acquisition system called Wisden. We use a time snapshot of this trace to verify and validate our model.

We are not aware of other similar sensor network traces. So, we collected our own traces using MICA2 motes with MTS310CA sensor boards attached to them. We used the light sensors on the sensor board to take light intensity measurements. Two traces in two differently lighted environments were collected using these motes.

2) *Trace 1*: 44 sensor nodes are deployed in a 34×64 feet square area. The location of each node is randomly chosen according to a uniform location distribution. We use a master mote to send a message to every mote. When a sensor node receives the message, it samples the light intensity of the environment. Thus all sensors take the readings at the same time. Thus, we get a spatially correlated trace of 44 samples.

The experiment is performed in an outdoor environment under strong sunlight with a few nodes in a shaded area caused by the presence of trees in the environment. Thus, the readings of the sensors will be close to each other, but the readings from the sensors in the shaded area will be lower than those in direct sunlight.

3) *Trace 2*: 30 sensor nodes are deployed in a 4×21 feet square area. The location of each node is randomly chosen according to a uniform location distribution. As before, all sensors take readings at the same time when they receive a message from the master mote.

The experiment is performed in an indoor environment with just one light source. This corresponds to a single source scenario where the readings go on decreasing as the distance from the light source increases. So, the sensors far away from the light source have much lower readings than the sensors closer to the light source. This generates a strongly correlated data trace.

Note that all the above traces are not very big, which is a result of the difficulty in deploying very large sensor networks. Due to statistical considerations, we want to verify and validate our model against a trace having thousands of spatial samples. So, we use the S-Pol Radar Data Trace ², which was obtained from remote sensing studies and has been used in the sensor networks literature as an experimental trace, see, for example [12]. Though the distance between the sensing nodes for this trace is hundreds of metres which is not representative of actual sensor networks in which the inter sensor distance is a few metres, we use it for verification because of the absence of large sensor network traces.

4) *S-Pol Radar Data Set*: The resampled S-Pol radar data, provided by NCAR, records the intensity of reflectivity of atmosphere in dBZ, where Z is proportional to the returned power for a particular radar and a particular range. The original data were recorded in the polar coordinate system. Samples were taken at every 0.7 degrees in azimuth and 1008 sample locations (approximately 150 meters between neighboring samples) in range, resulting in a total of 500×1008 samples for each 360 degree azimuthal sweep. They were converted to the cartesian grid using the nearest neighboring resampling method [23]. In this paper, we have selected a 64×64 spatial subset of the original data (4096 spatial samples) and 259 time snapshots across 2 days in May 2002.

B. Model Verification

1) *Trace 1*: The method described in Section V-C is used to infer the model parameters. But before, applying the method, we have to estimate the variogram from the given trace. We use the second method described in Section II to estimate the variogram. We fitted several standard variograms on to it [15], [16] and retained the one which had the minimum square error.

For the first trace, the spherical variogram given by Equation (18) was the best fit amongst all of them. For the given trace, $c_0 = 90$, $c = 170$ and $a = 9$.

$$\gamma(r) = \begin{cases} c_0 + c\left(\frac{3}{2}\frac{r}{a} - \frac{1}{2}\left(\frac{r}{a}\right)^3\right) & , 0 \leq r \leq a \\ c_0 + c & , a \leq r \end{cases} \quad (18)$$

After inferring the model parameters from the estimated variogram, we generate a synthetic trace on the same sensor node locations as the original trace. We compare the distribution of the traces in Figure 11 and their variograms in Figure 12. Both the distribution and the variograms match closely.

²S-Pol radar data were collected during the IHOP 2002 project (http://www.atd.ucar.edu/rtf/projects/ihop_2002/spol/). S-Pol is fielded by the Atmospheric Technology division of the National Center for Atmospheric Research. We acknowledge NCAR and its sponsor, the National Science Foundation, for provision of the S-Pol data set.

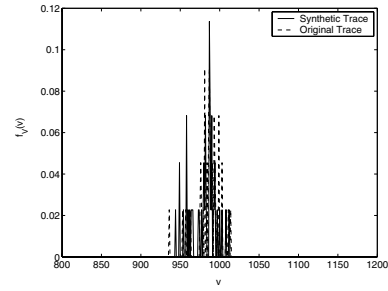


Fig. 11. Trace 1: Comparison of the distribution of the original and the synthetic traces.

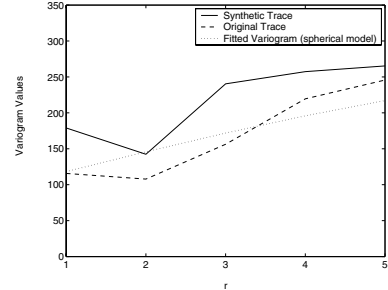


Fig. 12. Trace 1: Comparison of the variogram of the original and the synthetic traces.

2) *Trace 2*: The variogram of the second trace is best estimated by the power semi variogram model (Equation (19)) with parameters $c_0 = 14500$ and $c = 450$.

$$\gamma(r) = c_0 + cr^2. \quad (19)$$

We use the estimated variogram to obtain the model parameters and then generate a synthetic trace on the same sensor node locations as the original trace. We plot the variograms of both the traces in Figure 13. Once again, the variograms match closely.

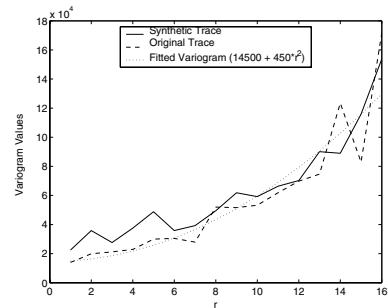


Fig. 13. Trace 2: Comparison of the variogram of the original and the synthetic traces.

3) *SHM Trace*: The variogram of the SHM trace is best estimated by the spherical semi variogram model (Equation (18)) with parameters $c_0 = 6000$, $c = 10000$ and $a = 2.7$. We use the estimated variogram to obtain the model parameters and then generate a synthetic trace on the same sensor node locations as the original trace. We plot the variograms of both

the traces in Figure 14. Once again, the variograms match closely.

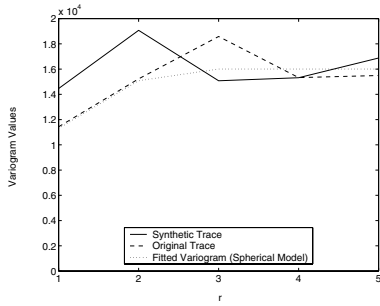


Fig. 14. SHM Trace: Comparison of the variogram of the original and the synthetic traces.

4) *S-Pol Radar data set*: We choose a snapshot in time of the S-Pol Radar data as the experimental data trace. Figure 15 shows the comparison of the distribution of the synthetic and original traces and Figure 16 shows their respective variograms. Both the distribution and the variogram of the two traces match closely.

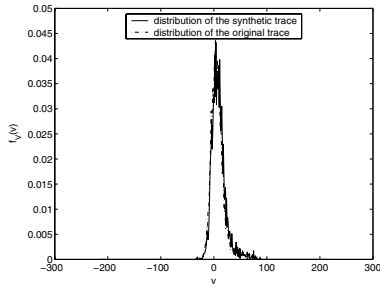


Fig. 15. S-Pol Radar data trace: Comparison of the distribution of the original and the synthetic traces.

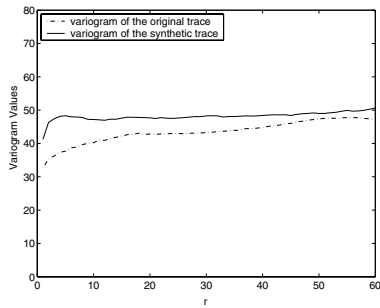


Fig. 16. S-Pol Radar data trace: Comparison of the variogram of the original and the synthetic traces.

C. Model Validation

The proposed model will be used to compare and evaluate different algorithms which exploit the presence of spatial correlation in data. To validate that our model can be used to evaluate the performance of different algorithms, we run

one of these algorithms on both the original and the synthetic traces and compare its performance.

Amongst the many such algorithms mentioned in the introduction, we selected CMAC [9] to evaluate our model. The underlying idea behind CMAC is that due to the presence of spatial correlation between sensor observations, it is not necessary for every node to transmit its data. Amongst a cluster of sensor nodes, one of them can act as a representative of all other nodes. We refer to the node that sends information to the sink as the *representative node* of the cluster. Thus, a smaller number of sensor measurements are adequate to communicate the event features to the sink within a certain acceptable error.

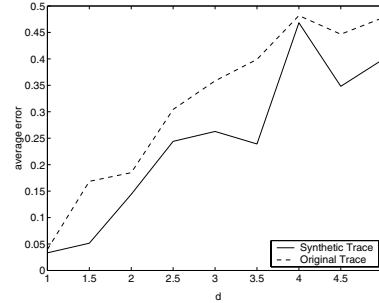


Fig. 17. Trace 1: Comparison of the performance of CMAC on original and synthetic trace: Variation of error with d .

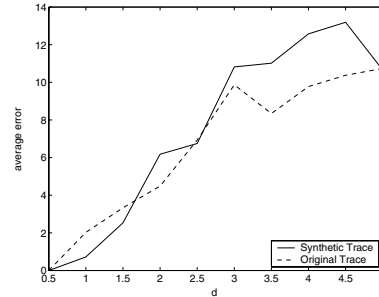


Fig. 18. Trace 2: Comparison of the performance of CMAC on original and synthetic trace: Variation of error with d .

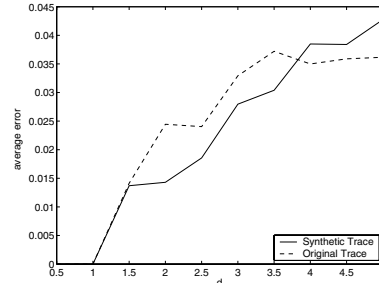


Fig. 19. SHM Trace: Comparison of the performance of CMAC on original and synthetic trace: Variation of error with d .

In our simulations, we assumed the cluster structure to be a square of side d . Amongst all the nodes within this square, the representative node is selected randomly. Only one node

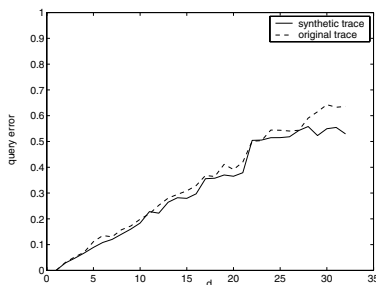


Fig. 20. S-Pol Radar Data Trace: Comparison of the performance of CMAC on original and synthetic trace: Variation of error with d .

in the cluster (the representative node) will transmit its data to the sink. The larger the value of d , the smaller is the number of nodes transmitting data to the sink, and hence larger is the error. We plot the average error against the value of d for the original as well as the synthetic trace for Trace 1, Trace 2, the SHM trace and the S-Pol Radar Data Trace in Figures 17, 18, 19 20 respectively. It is easy to see from the plots that the performance of the algorithm for both the traces is similar as the plots match closely.

From the above plots, we conclude that the proposed model is able to capture the spatial correlation in sensor network data.

VII. TOOLS TO GENERATE LARGE SYNTHETIC TRACES

In this section we describe two tools which we have created to help researchers generate synthetic traces of any size and degree of correlation. These tools can be downloaded from <http://www-scf.usc.edu/~apoorvaj>.

- *generateLargeTraceFromIrregular* will create large synthetic traces having the same correlation structure as the input real data trace. It takes the estimated variogram of the real trace as its input. It also requires the user to specify the data dependence pattern. The user can choose either of the methods described in Section IV-A.
- *generateSyntheticTracesOnIrregular* will create large synthetic traces representing a wide range of conditions by tweaking the model parameters. It takes the model parameters, r_{max} , $\alpha(r)$, β , σ_z and $f_V(v)$, the location of the nodes and the data dependence pattern as its input.

Data collected from a testbed having a few sensor nodes is not sufficient to evaluate protocols. The first tool can generate a large trace having similar correlation properties as the real trace, and hence, help researchers to evaluate protocols with real data. The second tool will enable researchers to evaluate their protocols with data having varied correlation structures. Hence, these two tools will help researchers to evaluate their protocols with data representing wide range of realistic conditions without the need of actual dense deployment of sensor nodes.

VIII. CONCLUSIONS AND DISCUSSION

In this paper, we proposed a mathematical model to capture spatial correlation in the data of sensor networks with irregular

topologies. The model can generate synthetic traces representing a wide range of conditions and exhibiting any degree of correlation for any arbitrary topology. We also described a rigorous mathematical procedure and a simple, practical method to infer the model parameters from a real trace. Finally, we verified the correctness of the model, and validated its ability to accurately capture correlations by comparing the performance of CMAC, an algorithm whose behavior depends on the degree of spatial correlation in data, under real and synthetic traces.

To model spatial correlation in data, most researchers assume the data to be jointly Gaussian [9], [17], [20], [24]. The primary reasons for this choice is ease of use and analytical tractability, rather than accuracy [25].

Our model is more general and more realistic and hence more complex than the jointly Gaussian model. Actually, it is easy to argue that the jointly Gaussian model is a special case of the proposed model. The pdf of jointly Gaussian random variables is completely defined by the covariance matrix. Each covariance matrix corresponds to a unique variogram and each variogram corresponds to a unique $\alpha(r)$, r_{max} , σ_z and β . $f_V(v)$ is Gaussian and $f_Y(y)$ can be inferred from Equation (10).

The chief limitation of the jointly Gaussian model is that it forces the joint pdf's of the data values to be jointly Gaussian, while in most of the experimental traces, even the first order distribution is not Gaussian. The proposed model has no such restriction and through a proper choice of $f_Y(y)$ and $f_Z(z)$, any distribution function of data values can be modeled.

To summarize, the jointly Gaussian model should be used to predict trends and get some quick intuition into the behavior of an algorithm. But to do a more thorough study, through analysis or simulations, a more realistic model such as the proposed model should be used. It is important to point out that trace-driven simulations are the best choice for simulating an algorithm, but in the absence of large sensor network data traces, this model can act as a close substitute.

ACKNOWLEDGMENT

The authors would like to thank Dr. Bhaskar Krishnamachari for helpful suggestions during the course of the work and for providing us with the motes for collecting traces. (The motes were bought under his grant CNS-0347621.)

REFERENCES

- [1] A. Jindal and K. Psounis, "Modeling spatially-correlated data sensor network data," in *Proceedings of the IEEE International Conference on Sensor and Ad hoc Communications and Networks*, Oct. 2004.
- [2] A. Goel and D. Estrin, "Simultaneous optimization for concave costs: single sink aggregation or single source buy-at-bulk," in *SODA*, 2003, pp. 499–505.
- [3] S. Patten, B. Krishnamachari, and R. Govindan, "The impact of spatial correlation on routing with compression in wireless sensor networks," in *Symposium on Information Processing in Sensor Networks (IPSN)*, Apr. 2004.
- [4] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, "Impact of network density on data aggregation in wireless sensor networks," in *ICDCS*, 2002.

- [5] B. Krishnamachari, D. Estrin, and S. B. Wicker, "The impact of data aggregation in wireless sensor networks," in *ICDCS Workshop on Distributed Event-based Systems (DEBS)*, 2002.
- [6] D. Ganesan, D. Estrin, and J. Heidemann, "Dimensions: Why do we need a new data handling architecture for sensor networks?" in *First Workshop on Hot Topics in Networks (Hotnets-I)*, Oct. 2002.
- [7] D. Ganesan, B. Greenstein, D. Perelyubskiy, D. Estrin, and J. Heidemann, "An evaluation of multi-resolution storage for sensor networks," in *SenSys'03*, Nov. 2003.
- [8] J. Faruque and A. Helmy, "Rugged: Routing on fingerprint gradients in sensor networks," in *IEEE International Conference on Pervasive Services (ICPS'2004)*, July 2004.
- [9] M. C. Vuran and I. F. Akyildiz, "Spatial correlation-based collaborative medium access control in wireless sensor networks," *submitted for publication, 2004*.
- [10] J. Chou, D. Petrovic, and K. Ramchandran, "Tracking and exploiting correlations in dense sensor networks," in *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002*, Nov. 2002.
- [11] K. Whitehouse and D. Culler, "Calibration as parameter estimation in sensor networks," in *Proceedings of WSNA'02*, Sept. 2002.
- [12] Y. Yu, D. Ganesan, L. Girod, D. Estrin, and R. Govindan, "Synthetic data generation to support irregular sampling in sensor networks," in *Geo Sensor Networks 2003*, Oct. 2003.
- [13] M. Rahimi, R. Pon, W. J. Kaiser, G. S. Sukhatme, D. Estrin, and M. Srivastava, "Adaptive sampling for environmental robotics," in *IEEE International Conference on Robotics and Automation*, Apr. 2004.
- [14] H. Kargupta, A. Joshi, K. Sivakumar, and Y. Yesha, *Data Mining: Next Generation Challenges and Future Directions*. AAAI/MIT Press, 2003.
- [15] R. A. Olea, *Geostatistics for engineers and earth scientists*. Kluwer Academic Publishers, 1999.
- [16] P. Goovaerts, *Geostatistics for natural resources evaluation*. Oxford University Press, 1997.
- [17] R. Cristescu and M. Vetterli, "On the optimal density for real-time data gathering of spatio-temporal processes in sensor networks," in *Proceedings of IPSN'05*, Apr. 2005.
- [18] R. P. Kanwal, *Linear Integral Equations: Theory and Technique*. Birkhauser Boston Academic Press, 1997.
- [19] D. Porter and D. S. Stirling, *Integral Equations: A practical treatment. from spectral theory to applications*. Cambridge University Press, 1990.
- [20] D. Marco, E. Duarte-Melo, M. Liu, and D. Neuhoff, "On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data," in *Proceedings of IPSN'03*, Apr. 2003.
- [21] P. K. Kythe and P. Puri, *Computation Methods for Linear Integral Equations*. Birkhauser Boston Academic Press, 2002.
- [22] J. Paek, K. Chintalapudi, J. Caffrey, R. Govindan, and S. Masri, "A wireless sensor network for structural health monitoring: performance and experience," in *Second IEEE Workshop on Embedded Networked Sensors*, May 2005.
- [23] W. Venables and B. Ripley, *Modern applied statistics with S*, 4th ed. Springer, 2002.
- [24] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On network correlated data gathering," in *Proceedings of the IEEE Infocom'04*, Mar. 2004.
- [25] Y. Yu, D. Estrin, R. Govindan, and M. Rahimi, "Using more realistic data models to evaluate sensor network data processing algorithms," in *First IEEE Workshop on Embedded Networked Sensors*, Nov. 2004.