Modeling Spatially Correlated Data in Sensor Networks

Apoorva Jindal and Konstantinos Psounis
University of Southern California

The physical phenomena monitored by sensor networks, e.g. forest temperature, water contamination, usually yield sensed data that are strongly correlated in space. With this in mind, researchers have designed a large number of sensor network protocols and algorithms that attempt to exploit such correlations.

To carefully study the performance of these algorithms, there is an increasing need to synthetically generate large traces of spatially correlated data representing a wide range of conditions. Further, a mathematical model for generating synthetic traces would provide guidelines for designing more efficient algorithms. These reasons motivate us to obtain a simple and accurate model of spatially correlated sensor network data.

The proposed model is Markovian in nature and can capture correlation in data irrespective of the node density, the number of source nodes or the topology. 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 on a given topology using these parameters. 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 remote sensing data, publicly available sensor data, and sensor networks that we deploy. We show that the proposed model is more general and accurate than the commonly used jointly Gaussian model. Finally, we create tools that can be easily used by researchers to synthetically generate traces of any size and degree of correlation.

Categories and Subject Descriptors: C.4 [**Performance of Systems**]: Modeling Techniques; I.6.5 [**Simulation and Modeling**]: Model Development-Modeling Methodologies

General Terms: Performance
Additional Key Words and Phrases: Spatial Correlation, Modeling of Physical Environment, Wireless Sensor Networks, Generating Synthetic Data

1. 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

A preliminary version of this paper has appeared in the Proceedings of the IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON'05), Sep. 2005.

Author's Address: A. Jindal and K. Psounis, Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089; email: apoorvaj, kpsounis@usc.edu.

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 [Goel and Estrin 2003], [Pattem et al. 2004], [Intanagonwiwat et al. 2002], [Krishnamachari et al. 2002], data storage and querying [Deshpande et al. 2004], [Ganesan et al. 2002], [Ganesan et al. 2003], [Faruque and Helmy 2004], sensor selection [L.Doherty and K.Pister 2004], [Cristescu and Vetterli 2005], MAC protocol design [Vuran and Akyildiz 2006], data compression and encoding [Chou et al. 2002], and calibration [Whitehouse and Culler 2002].

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 [Deshpande et al. 2004], [Cristescu and Vetterli 2005], [Vuran and Akyildiz 2006], some assume that the data follows the diffusion property [Faruque and Helmy 2004], and some assume a function for the joint entropy of the data [Pattem et al. 2004]. Other researchers propose the use of environmental monitoring data obtained from remote sensing [Ganesan et al. 2002], however the granularity of these data sets do not match the expected granularity of sensor networks' data.

Another model which can be used to model data dependencies in sensor networks data is the Markov random field (MRF) model [Li 2001]. MRF's were proposed in image processing to model dependent random variables such as image pixels and correlated features. But the MRF model requires a description of the joint statistics of the data. Using such a model in its generality is quite cumbersome and, in practice, very hard to track analytically.

The goal of this paper 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. We want the model to be simpler than existing complicated models for two dimensional correlations, like the MRF model, and yet to accurately represent reality. To keep it simple, we want our model parameters to depend only on the first order statistic and the second order moments. It should be noted that a jointly Gaussian model is completely defined by the first and the second order moments and hence is pretty tractable. But, it is not a very accurate model as shown in [Yu et al. 2004]. On the other hand, we will show that our model, inspite of being defined by the first order statistic and second order moments, captures the spatial correlation characteristics of sensor network data.

There are many benefits from such a model. First, the model will provide a procedure to synthetically generate sensor networks data without having to collect real traces first. By varying the model parameters, one can create traces with various degrees of correlation, thus 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, since the model is analytically tractable, it can be used to analyze and bound the performance of algorithms. Thus, it can provide guidelines for designing optimal algorithms. Fourth, it can be used to generate large synthetic traces having the same correlation structure as an input real trace.

The model proposed in this paper is a special case of Markov random fields but is much simpler and yet pretty accurate. It is similar in flavor to the model proposed in [Psounis et al. 2004] (which proposed a model to capture temporal correlation in web traces). A rigorous mathematical procedure and a simple practical method to extract the model parameters from real traces is provided. A method to efficiently generate synthetic traces on a given topology using these parameters is also described, and publicly available trace generation tools are created. Further, it is shown that the jointly Gaussian model, which is commonly used for spatially correlated data [Vuran and Akyildiz 2006], [Cristescu and Vetterli 2005], [Marco et al. 2003], [Deshpande et al. 2004], is a subcase of our more general and more accurate model.

The model is verified by comparing the statistics of the real traces and the corresponding synthetic traces. Since the proposed model will be used to evaluate and compare different algorithms which exploit spatial correlation in data, the model is validated by comparing the performance of such algorithms. We use two well known algorithms, DIMENSIONS [Ganesan et al. 2002] and CC-MAC [Vuran and Akyildiz 2006] for this purpose. We use publicly available remote sensing traces, publicly available sensor network traces, and traces collected from sensor networks that we deploy. The inter-node distance for remote sensing data is hundred of meters while for the traces collected using a sensor network, this distance is of the order of a few meters. These traces verify that the proposed model is valid irrespective of the granularity of data.

The paper is organized as follows. Section 2 introduces the variogram, which is a handy metric to characterize spatial correlation in data. Section 2.2 studies the correlation structure of a real trace using variograms to come up with an intuition about the structure of the model. The model is formally presented in Section 3, followed by a mathematically rigorous procedure, and a simple, practical method to infer the model parameters in Section 4. The correctness of the model is verified by comparing the statistics of the original and synthetic traces in Section 5.2. In Section 5.3, the accuracy of the model is validated by comparing the performance of various algorithms against real and the corresponding synthetic traces. Section 6 discusses the related work to put our contributions in context. In this section, we also show that our model is more general than the jointly Gaussian model (which is the most popular model in the sensor network community). In Section 7, we revisit our chief assumptions in an effort to understand how general is the proposed model. Finally, Section 8 describes the trace-generation tools and Section 9 concludes the work.

2. VARIOGRAM: A STATISTIC TO MEASURE CORRELATION IN DATA

A statistic often used to characterize spatial correlation in data is the variogram [Yu et al. 2003], [Rahimi et al. 2004], [Kargupta et al. 2003]. Given a two dimensional stationary process $V(x, y)$, the variogram (also called semivariance) is defined as

$$\gamma(r_1, r_2) = \frac{1}{2} E[(V(x, y) - V(x + r_1, y + r_2))^2]. \quad (1)$$

For isotropic random processes [Olea 1999] the variogram depends only on the distance $r = \sqrt{r_1^2 + r_2^2}$ [1] between two nodes (as opposed to anisotropic processes in which the variogram depends on both distance and direction). In this case, if (x_r, y_r) denotes a point which is r distance away from (x, y),

$$\gamma(r) = \frac{1}{2} E[(V(x, y) - V(x_r, y_r))^2], \quad (2)$$

where $(x - x_r)^2 + (y - y_r)^2 = r^2$.

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

$$\overset{\wedge}{\gamma}(r) = \frac{1}{2m(r)} \sum_{1}^{m(r)} [v(x_i, y_i) - v(x_j, y_j)]^2, \quad (3)$$

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

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 [Olea 1999], [Goovaerts 1997], [Cressie 1993] 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 used in Geostatistics. [2]

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

2.1 Relationship between the variogram and the covariance

Another very commonly used statistic to measure correlation in data is the covariance [Cristescu and Vetterli 2005], [Marco et al. 2003], [Cristescu et al. 2004]. For a two dimensional isotropic stationary process $V(x, y)$, the covariance is defined as

$$C(r) = E\left[(V(x, y) - \mu)(V(x_r, y_r) - \mu)\right],$$

[1] Unless otherwise stated, we will use the Euclidean distance to measure distances between two points.

[2] Appendix A.2 presents the commonly used standard variogram models.

where (x_r, y_r) is denotes a point r distance away from (x, y) and $\mu = E[V(x, y)]$.

Since both the variogram and the covariance are measures of correlation in data, we derive the relationship between them and verify that both of them can be used interchangeably. From Equation (2),

$$\gamma(r) = \frac{1}{2}E\left[(V(x,y) - V(x_r,y_r))^2\right] = \frac{1}{2}E\left[((V(x,y)-\mu) - (V(x_r,y_r)-\mu))^2\right]$$

$$\Rightarrow \gamma(r) = \sigma_V^2 - C(r), \qquad (4)$$

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

Equation (4) 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.

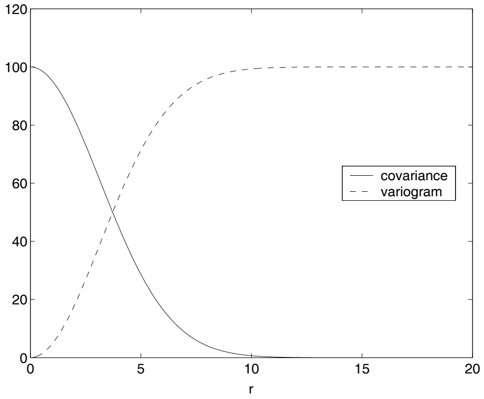


Fig. 1. Variogram and Covariance plots for a trace generated by assuming jointly Gaussian model for the spatial data.

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.

2.2 Analysis of Data using Variograms

In this section, we analyze the correlation structure of different spatial processes and propose a simple model for each of them. More specifically, we present a model for independent data and for data following the diffusion law. We then look at the correlation structure of a real experimental trace and propose a model to capture the spatial correlation in this data. This model combines the two previous models. Using variograms, we show that the proposed model is able to capture the correlation in data.

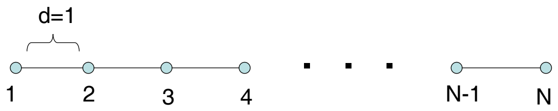
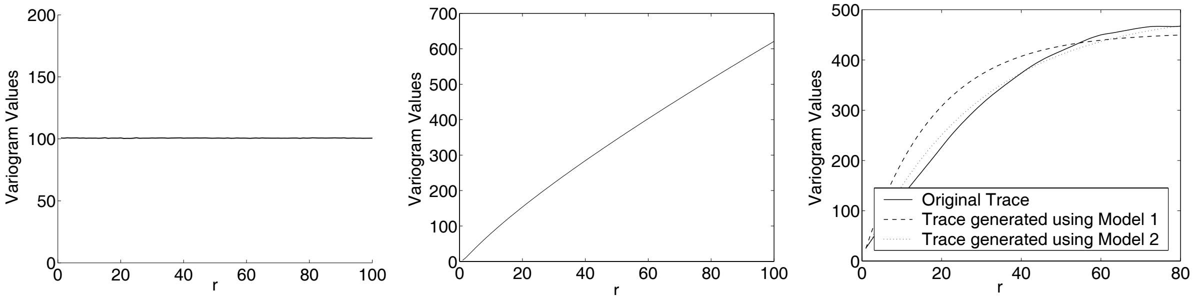


Fig. 2. Linear Topology. The data value at node i is given by V_i



(a)

(b)

(c)

Fig. 3. (a) Variogram for an iid process. (b) Variogram for a process which follows the diffusion law ($\lambda = 0.1$). (c) Variogram of the experimental data at a time snapshot. The x-axis is in units of distance.

The models in this section assume a linear topology as shown in Figure 2. The data value at node i is given by V_i.

2.2.1 *Independent Data.* If a process is independent and identically distributed (iid), its variance will not change with distance and the variogram should be a straight line parallel to the x-axis. Figure 3(a) shows the variogram for an iid process with the underlying random variable being Gaussian with mean 0 and standard deviation equal to 10.

A model for the data values which captures the statistical properties of independent data is given by,

$$V_i = Y,$$

where Y is a normal random variable with mean 0 and standard deviation equal to 10. The variogram for this model can be easily evaluated as follows,

$$\gamma(r) = \frac{1}{2}E[(V_r - V_0)^2] = \sigma_Y^2 = 100,$$

which is in accordance to Figure 3(a).

2.2.2 *Diffusion Model.* When the phenomenon under observation is being emitted from a single source it usually follows the diffusion property with distance, i.e. $f(r) \propto \frac{1}{r^\lambda}$ where $f(r)$ is the magnitude of the event's effect at a distance r

from the source and λ is the diffusion parameter that depends on the physical phenomenon. Figure 3(b) shows the variogram for a process following the diffusion law with $\lambda = 0.1$.

For small λ ($\lambda \sim 0.1$), a model to populate the data values is given by,

$$V_i = V_{i-1} + Z,$$

where Z is a random variable with mean 0 and variance σ_z^2. The variogram for this model can be evaluated as,

$$\gamma(r) = \frac{1}{2}E[(V_r - V_0)^2] = \frac{1}{2}E[(V_{r-1} - V_0 + Z)^2]$$
$$= \frac{1}{2}E[(V_{r-2} - V_0 + Z + Z)^2] = \frac{1}{2}E[(V_0 - V_0 + Z + Z \ldots + Z)^2] = \frac{1}{2}r\sigma_z^2.$$

which is in accordance to Figure 3(b) for $\sigma_z^2 = 12$. (The slope of the linear variogram, with respect to r, from the model is $\frac{1}{2}\sigma_z^2$ and equating it with the slope in Figure 3(b) yields $\sigma_z^2 = 12$.)

2.2.3 *A real data trace.* The process under observation seldom has a single source and the presence of multiple sources will require us to calculate a phasor sum of data values at a node. For atmospheric data such as temperature, precipitation and humidity, it is not even possible to define a source. The data values at nodes close to each other will be correlated, while for large r the process will start looking like an iid process. As an example, the variogram at a time snapshot of the S-Pol radar data is shown in Figure 3(c). The S-Pol radar data trace is a humidity data trace obtained from remote sensing studies. A full description of the trace is provided in Section 5.1

It is observed from the plot that as the distance grows from zero the spatial correlation decreases. Also, for distances larger than 60, the correlation is quite small. The correlation structure looks like that of the diffusion model for smaller distances while it looks like that of independent data for large values of distance. Hence, we propose a model for this data which combines both the previous models. The data value at a node V_i is derived either from V_{i-1} or from a random variable Y.

$$V_i = \begin{cases} V_{i-1} + Z & \text{with probability } \alpha \\ Y & \text{w.p. } 1 - \alpha \end{cases}.$$

We refer to this model as Model 1.

After some simple calculations similar to the ones above, the variogram of Model 1 can be expressed by the following recursive equation,

$$\gamma(r) = \alpha\gamma(r-1) + 2\alpha\sigma_z^2,$$

and $\gamma(1) = 2\alpha\sigma_z^2$. The variogram of Model 1 with $\alpha = 0.945$ and $\sigma_z^2 = 26.4$ is shown in Figure 3(c). Note that the variogram does not depend on the statistics of Y. Y only effects the first order statistics of V_i's and does not effect the correlation structure.

Though Model 1 is able to capture the trends, it is not a very good match. So, we increase the depth of data dependency in Model 1 to come up with the following model,

$$V_i = \begin{cases} V_{i-1} + Z & \text{with probability } \alpha_1 \\ V_{i-2} + Z & \text{w. p. } \alpha_2 \\ Y & \text{w.p. } 1 - \alpha_1 - \alpha_2 \end{cases} .$$

Now, the data value at node V_i is derived either from V_{i-1} or from V_{i-2} or from a random variable Y. We refer to this model as Model 2. The variogram of Model 2 for $\alpha_1 = 0.48, \alpha_2 = 0.47$ and $\sigma_z^2 = 26.3$ is plotted in Figure 3(c).

Figure 3(c) shows that Model 2 is able to capture the correlation characteristics of the data.

Remark: The reason why Model 2 is accurate in capturing the characteristics of the S-Pol radar data trace is intuitively the following: In real traces, the data values of nearby nodes are usually correlated and close to each other, but not identical, whereas the data values of far-away nodes are independent and can differ a lot. The dependence of V_i over V_{i-1} and V_{i-2} captures the correlation between the values of nearby nodes. The random variable Z introduces small deviations between the values of nearby nodes, since they are close to each other but not identical. And, the random variable Y introduces in an independent manner new data values that can differ a lot from prior values.

In this section, we proposed simple models for very specific correlation structures (independent data, diffusion law with a small λ and one snapshot of the S-Pol radar data trace). In the next section, we will generalize the simple models used for the S-Pol radar data trace, so that any given correlation structure can be modeled. Our general model will follow the same principles as Model 2. A node will either derive its data value from one of the nearby nodes plus a small deviation, or from an independent random variable. This approach, as we will show, will accurately capture the correlation characteristics of a wide range of spatially correlated data.

3. 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 disk of unit radius. Let $V(r, \theta)$ be the data value at node (r, θ) inside the unit radius disk. 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:

$$\begin{aligned} V(0,0) = I_{(U=T)}Y + I_{(U=H)} \int_\theta \int_r (V(r,\theta) + Z) \\ \delta(R = r) dr \delta(\Theta = \theta \mid R = r) r d\theta, \end{aligned} \tag{5}$$

where:

a) 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

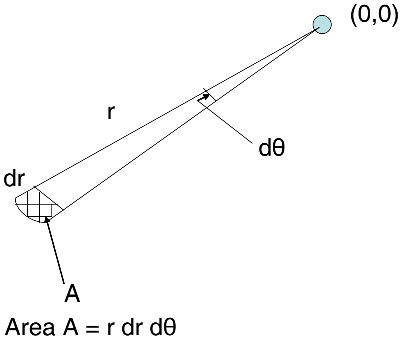


Fig. 4. The probability that the data value at $(0, 0)$ is derived from a node in region A is $\frac{\alpha(r)}{(\theta_2-\theta_1)r}$.

an indicator function that equals one when event A occurs and equals zero otherwise.)

b) 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 distribution of V $(f_V(v))$.

c) R is a random variable with pdf $\alpha(r)$. When $R = r$, the origin's data value is obtained from locations at distance r from the origin. $\alpha(r)dr$ is the probability of this event. $\alpha(r)$ is a parameter of our model. ($\delta(R = r)$ denotes a δ-function of R that is non-zero when $R = r$.)

d) Θ is a random variable with pdf $f_\Theta(\theta)$. When $\Theta = \theta \mid R = r$, the origin's data value is obtained from locations at angle θ given that their distance from the origin is r. $f_{\Theta|R}(\theta \mid r)rd\theta$ is the probability of this event. We assume that θ is uniformly distributed between angles θ_1 and θ_2. Thus,

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

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

$$P(V(0,0) \leq v) = \beta P(Y \leq v) + (1-\beta)$$
$$\int_\theta \int_r P(V(r,\theta) + Z \leq v) \frac{\alpha(r)}{(\theta_2-\theta_1)r} r dr d\theta. \qquad (6)$$

Equation (5) and (6) simply say that the probability that the data value at a node is directly derived from a node lying in the shaded region A in Figure 4 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 $\frac{rd\theta}{(\theta_2-\theta_1)r}$.

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 3.1.

Since correlation is a function of distance only (as the process is isotropic), α is a function of only r and not θ. $\alpha(r)$ 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,

$$\int_0^{r_{max}} \int_{\theta_1}^{\theta_2} \frac{\alpha(r)}{(\theta_2 - \theta_1)r} r dr d\theta = 1 \Rightarrow \int_0^{r_{max}} \alpha(r) = 1. \tag{7}$$

3.1 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.

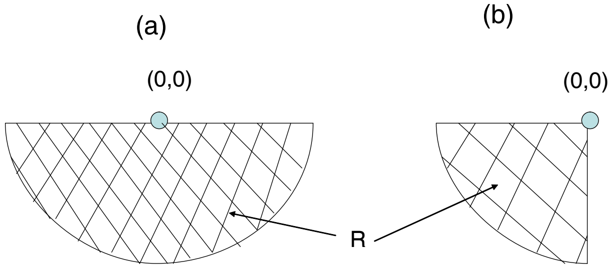


Fig. 5. Two methods to populate data. (a) Semi Circular Dependence: The data value at node $(0, 0)$ can be directly derived from any node lying in the semi circular region. (b) Quarter Circular Dependence: The data value at node $(0, 0)$ can be directly derived from any node lying in the quarter circular 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 5. 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 $\theta_2 = \frac{3\pi}{2}$).

As an example, consider the node locations given by Figure 6. Let the node location of node i be $(r_i, \theta_i), 0 \leq i \leq 11$, and let the data values at these nodes be

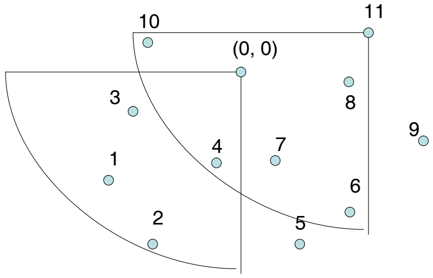


Fig. 6. An example topology.

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:

$$V(0,0) = \begin{cases} V(r_1,\theta_1) + Z & \text{with probability } c\frac{\alpha(r_{01})}{r_{01}} \\ V(r_2,\theta_2) + Z & \text{w.p. } c\frac{\alpha(r_{02})}{r_{02}} \\ V(r_3,\theta_3) + Z & \text{w.p. } c\frac{\alpha(r_{03})}{r_{03}} \\ V(r_4,\theta_4) + Z & \text{w.p. } c\frac{\alpha(r_{04})}{r_{04}} \\ Y & \text{w.p. } c\beta \end{cases}, \tag{8}$$

where r_{0j} denotes the distance between nodes $(0,0)$ and (r_j, θ_j) and c is a scaling constant which is present to make the sum of probabilities go to one.

Equation 8 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. Note that this implies that a full circular data dependence ($\theta_1 = 0$ and $\theta_2 = 2\pi$) cannot be used for populating data. 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.

Remark: Note that Equation (8) implies that the proposed model is Markovian. To see this, assume a valid order for populating the data. Then, if the state is defined to be a vector of data values at nodes which lie in the dependence area of any node whose value has not yet been populated, the Markovian property holds as we populate the node values one by one because the next state depends only on the previous one. The dependence between the two states is characterized by the model parameters.

3.2 Instantiation of the model on a grid topology

Though the instantiation on a grid topology can be constructed in a manner similar to the previous section, the inherent regularity in the topology allows us to simplify the exposition. So, we devote this section to construct the model on grid topologies. A more comprehensive treatment of the model on grid topologies can be found in [Jindal and Psounis 2004].

For ease of notation we use Cartesian coordinates to define node locations. [3] First we describe why a grid topology simplifies the exposition. Since the distance to the nearest node is the same for every node and is equal to the size of the 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 $r = \mid x_1 - x_2 \mid + \mid y_1 - y_2 \mid$. Thus, the distances between nodes on a grid are in multiples of the grid size. This simplifies the model structure as the variogram as well as $\alpha(r)$ can now be viewed as discrete functions of distance. In this paper, we denote a discrete function f as $f[x]$ and a continuous function as $f(x)$.

Let the data value at node (x, y) be given by $V(x, y)$. Let $N[r]$ denote the number of nodes at a distance r from (x, y). Let V_r denote the data value at a node which is r distance away from (x, y), and V_r^k denote the data value at the k^{th} node $(1 \leq k \leq N[r])$ at a distance r from (x, y). Following is the model for generating the data values,

$$V(x,y) = \begin{cases} V_1^1 + Z & \text{with probability } \frac{c\alpha[1]}{N[1]} \\ \vdots & \\ V_1^{N[1]} + Z & \text{w.p. } \frac{c\alpha[1]}{N[1]} \\ V_2^1 + Z & \text{w.p. } \frac{c\alpha[2]}{N[2]} \\ \vdots & \\ V_2^{N[2]} + Z & \text{w.p. } \frac{c\alpha[2]}{N[2]} \\ \vdots & \\ V_{r_{max}-1}^1 + Z & \text{w.p. } \frac{c\alpha[r_{max}-1]}{N[r_{max}-1]} \\ \vdots & \\ V_{r_{max}-1}^{N[r_{max}-1]} + Z & \text{w.p. } \frac{c\alpha[r_{max}-1]}{N[r_{max}-1]} \\ Y & \text{w.p. } c\beta \end{cases}, \qquad (9)$$

where c is a scaling constant present to make the probabilities sum to one.

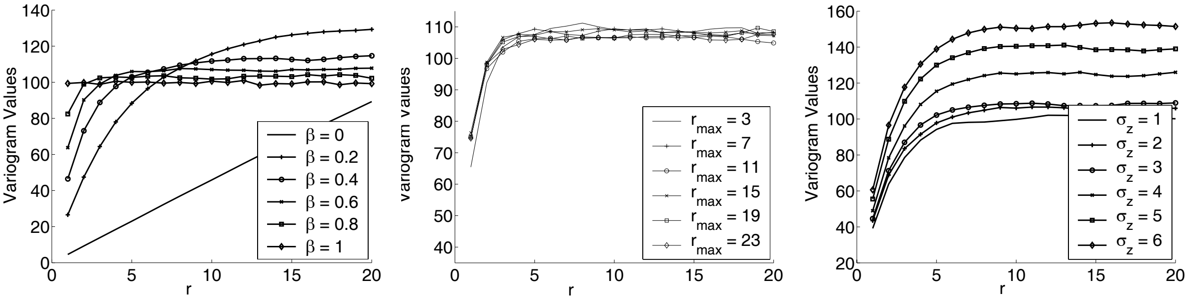
The above equation simply says that the probability that $V(x, y)$ is derived from the value of any node which is r distance away from (x, y) is $\alpha[r]$. Further, the probability that $V(x, y)$ is derived from the value of a particular such node is $\frac{\alpha[r]}{N[r]}$. The value of $N[r]$ will depend on whether the data is populated using a semi circular dependence $(N[r] = 2r)$ or a quarter circular dependence $(N[r] = r + 1)$.

3.3 How the model parameters affect correlation

The presence of many parameters in the model gives us great flexibility to model different spatial processes. 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 2. Synthetic traces are generated using the model under a 20000 node scenario. We assume $Y \sim N(0, 10)$ and $Z \sim N(0, \sigma_z)$.

[3] For ease of notation, whenever we are dealing with irregular grids, we will assume polar coordinate system while whenever we deal with grid topologies, we assume Cartesian coordinates.



(a) (b) (c)

Fig. 7. (a) Effect on the correlation structure of the data when β is varied keeping all the other parameters constant. (b) Effect on the correlation structure of the data when r_{max} is varied keeping all the other parameters constant. (c) Effect on the correlation structure of the data when σ_z is varied keeping all the other parameters constant. The x-axis is in units of distance

3.3.1 *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 is derived in Appendix A.3.

Figure 7(a) 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 $\sigma_z = 3$. Any decreasing function of r can serve as $\alpha(r)$. We choose one of these for these case studies.

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.

Figure 8 shows the actual data values for a sample of the topology for two values of β. For $\beta = 0.95$, the data values look pretty random implying low spatial correlation in data while for $\beta = 0.05$, the data values at close by nodes show high correlation.

3.3.2 *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(b) 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

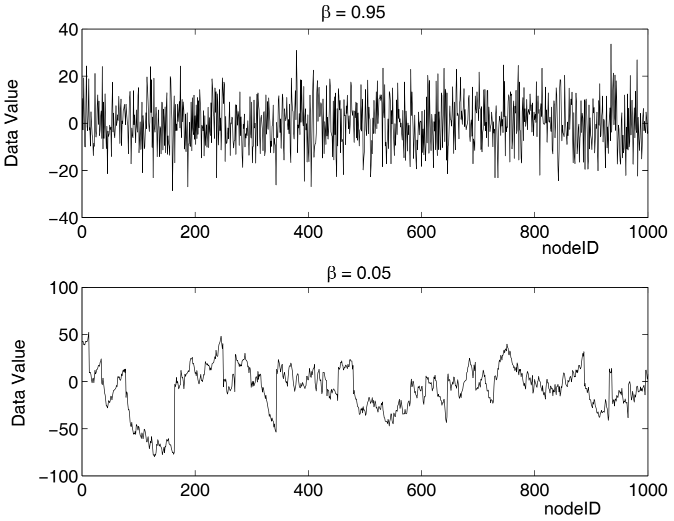


Fig. 8. The actual data values for a sample of the topology for two values of β. $\beta = 0.95$ corresponds to very low correlation while $\beta = 0.05$ corresponds to very high correlation.

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

3.3.3 *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 7(c). The other parameters are: $r_{max} = 2, \alpha(r) = \lambda 2^{-r}$ for $0 < r < r_{max}$ and 0 otherwise, and $\beta = 0.4$.

It can be easily seen from the plots that 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: When one generates trace and uses them to evaluate the performance of an algorithm 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.

4. INFERRING MODEL PARAMETERS

In this section, we present techniques for inferring model parameters from real traces. This section shows that the model parameters can be inferred from the first order statistics and second order moments of the trace.

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

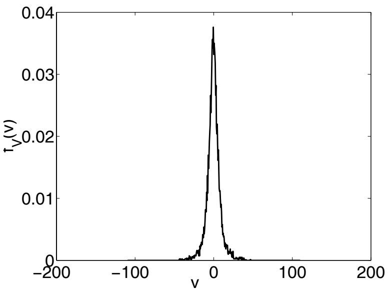


Fig. 9. Distribution of $V - V_1$ for samples from a time snapshot of the S-Pol radar data where V_1 are the sample values at nodes at a unit distance away from V.

variable with zero mean and standard deviation $\sigma = \sigma_z$. The S-Pol radar data justifies our assumption. In particular, Figure 9 shows the distribution of $V - V_1$, where V_1 represents the sample values at nodes at a unit distance away from V and, as it is evident from the plot, this distribution can be very closely approximated by a Gaussian distribution. 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.

Remark: If Z is not Gaussian, then the model parameters cannot be determined from just the first order statistics and the second order moments; they will depend on higher order statistics too. Since all the real traces we studied were accurately modelled with a Gaussian Z, we do not discuss the model for a non-Gaussian Z in this paper.

In Section 4.1, we state 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 4.2 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 of the first kind [Kanwal 1997], [Porter and Stirling 1990] and hence, it is not always possible to obtain a closed form expression for the model parameters. Further, even though several numerical techniques exist in the literature to solve integral equations with no closed form solutions, integral equations of the first kind are *inherently ill posed problems* [Kythe and Puri 2002]. As a result, their solutions are generally unstable and prone to large errors. Motivated by this, in Section 4.3 we describe a procedure to infer the model parameters for data values on nodes on a grid topology, and then, in Section 4.4 we present a simple, practical method which uses the grid topology solution to infer the model parameters on any topology.

4.1 Relationship between the distributions of V, Y and Z

LEMMA 4.1. *$V = Y + L$ where L is a random variable with a characteristic function given by*

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

Proof: See Appendix A.3. □

The distribution of Y, $f_Y(y)$, can be inferred from Lemma 4.1. The following subsections present procedures to infer the rest of the model parameters.

4.2 A rigorous procedure to infer the model parameters

In this section, we present a rigorous method to infer the rest of the model parameters, $\alpha(r)$, β, σ_z and r_{max}. To infer these parameters, we first compute the variogram $\gamma(r)$ using the model and then equate it with its estimate $\overset{\wedge}{\gamma}(r)$ obtained from the real trace.

Using Equation (2),

$$\begin{aligned}\gamma(r) = \frac{1}{2}\int_0^{2\pi} \frac{1}{2\pi} E\left[(V(0,0) - V(r,\theta))^2\right] d\theta = \frac{1}{2}(1-\beta)\\ \int_0^{2\pi} \frac{1}{2\pi} \int_0^{r_{max}} \int_{\theta_1}^{\theta_2} E\left[(V(r',\theta') + Z - V(r,\theta))^2\right] \frac{\alpha(r')}{\theta_2 - \theta_1} dr' d\theta' d\theta\\ + \frac{1}{2}\int_0^{2\pi} \frac{\beta}{2\pi} E[(Y - V(r,\theta))^2] d\theta. \qquad (11)\end{aligned}$$

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

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

The second term in Equation (11) $E[(Y - V(r,\theta))^2]$ is equal to $E[L^2]$. Using Equation (10), $E[L^2]$ is evaluated to be $\frac{(1-\beta)\sigma_z^2}{\beta}$.

Substituting all of the above in Equation (11),

$$\begin{aligned}\gamma(r) = (1-\beta)\sigma_z^2 + (1-\beta)\int_0^{r_{max}} \int_{\theta_1}^{\theta_2} \int_0^{2\pi} \frac{1}{2\pi} \frac{\alpha(r')}{\theta_2 - \theta_1}\\ \gamma\left(\sqrt{r^2 + r'^2 - 2rr'cos(\theta - \theta')}\right) d\theta d\theta' dr'. \qquad (12)\end{aligned}$$

Equation (12) gives the relationship between the variogram and the model parameters $\alpha(r)$, β, σ_z and r_{max}. Substituting $\gamma(r)$ with its estimate $\overset{\wedge}{\gamma}(r)$ in Equation (12) gives us an integral equation of the first kind [Kanwal 1997], [Porter and Stirling 1990], which along with the boundary 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 gives us the model parameters. After obtaining σ_z and β, $f_Y(y)$ is obtained through Equation (25).

In Equation (12), 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 $\alpha(r)$.

4.2.1 *Case Study.* In this section, we will the find model parameters for a trace having the following variogram,

$$\gamma(r) = c(1 - e^{-\lambda r^2}) \;\; 0 < r < R' , \qquad (13)$$

where R' is determined by the area in which the nodes are distributed and λ is a parameter which governs how fast the correlation decays.

The corresponding covariance is $C(r) = ce^{-\lambda r^2}$ which is a very commonly assumed correlation structure for spatially correlated data in the sensor networks literature, see, for example, [Cristescu and Vetterli 2005], [Marco et al. 2003]. 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 subcase of our model, as discussed in Section 6.

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

$$c(1-e^{-\lambda r^2}) = (1-\beta)\int_0^{r_{max}}\int_{\theta_1}^{\theta_2}\int_0^{2\pi}\frac{1}{2\pi}\frac{\alpha(r')}{\theta_2-\theta_1}$$
$$c\left(1-e^{-\lambda(r^2+r'^2-2rr'cos(\theta-\theta'))}\right)d\theta d\theta' dr' + (1-\beta)\sigma_z^2. \qquad (14)$$

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

$$\int_{\theta_1}^{\theta_2}\int_0^{2\pi}\frac{1}{2\pi}\frac{1}{\theta_2-\theta_1}c\left(1-e^{-\lambda r^2}e^{-\lambda r'^2}e^{2\lambda rr'cos(\theta-\theta'))}\right)d\theta d\theta'. \qquad (15)$$

To obtain a closed form approximation for the model parameters, we assume that $2\lambda rr' < 1$ and hence, by neglecting the square terms and beyond, the last term in the above equation can be approximated by,

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

We assume the semi circular data dependence to populate data, hence $\theta_1 = \pi$ and $\theta_2 = 2\pi$. With the above approximation, Equation (15) reduces to $c\left(1-e^{-\lambda r^2}e^{-\lambda r'^2}\right)$.

Substituting in Equation (14),

$$c(1-e^{-\lambda r^2}) = c(1-\beta)\int_0^{r_{max}}\alpha(r')\left(1-e^{-\lambda r^2}e^{-\lambda r'^2}\right)dr' + (1-\beta){\sigma_z}^2. \qquad (16)$$

Using the method described in [Kanwal 1997] to solve for integral equations, we determine that $\alpha(r)$ has the form $a + be^{-\lambda r^2}$ 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 $b = \left(\frac{\sqrt{\pi}Erf(\sqrt{\lambda}r_{max})}{2\sqrt{\lambda}} - r_{max}e^{-\lambda r_{max}^2}\right)^{-1}$ and $a = -be^{-\lambda r_{max}^2}$, where $Erf(x)$ is the error function defined as $Erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$.

Now substituting $\alpha(r) = a+be^{-\lambda r^2}$ in Equation (16) gives $\beta = 1-\frac{4\sqrt{\lambda}}{\sqrt{\pi}}\,(2aErf(\sqrt{\lambda}r_{max})+\sqrt{2}b\ Erf(\sqrt{2\lambda}r_{max}))^{-1}$ 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 $r_{max} = R'$, and keep on reducing its value till we obtain a positive value of β.

For $R' = 10$, $\lambda = \frac{1}{200}$ and $c = 10$, the model parameters turn out to be, $r_{max} = 2$, $\beta = 0.0152$ and $\sigma_z^2 = 0.4$. The corresponding $\alpha(r)$ is plotted in Figure 10(a).

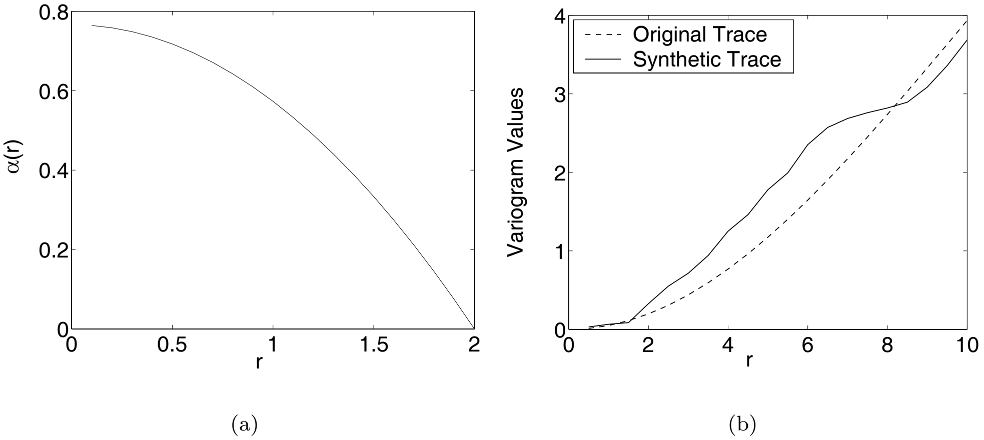


Fig. 10. (a) $\alpha(r)$ obtained after solving the Integral Equation 14. (b) The given variogram $\gamma(r) = 10(1 - e^{-\lambda r^2})$ and the variogram of a synthetic trace generated using the parameters derived in Section 4.2.1
.

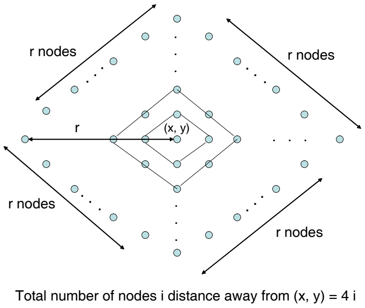


Fig. 11. Grid Topology.

To verify that these parameters capture the correlation characteristics, we plot the variogram of Equation (13) and the variogram of a synthetic trace generated using these parameters in Figure 10(b). Both the curves match closely.

4.3 Inferring model parameters for the grid topology

As discussed in Section 3.2, the grid topology simplifies the exposition because both the variogram and the model parameters are now discrete functions of distance. The mathematical procedure to infer model parameters on a grid is similar to the one described in Section 4.2, but now since all the functions are discrete instead of continuous, the integrals will be replaced by sums.

For a grid topology, Equation (11) is rewritten as,

$$\gamma[r] = \frac{1}{2}\sum_{i=1}^{4r}\frac{1}{4r}E\left[(X - X_r^i)^2\right] = \frac{1}{2}\sum_{i=1}^{4r}\frac{1}{4r}\sum_{j=1}^{r_{max}-1}\sum_{k=1}^{N[j]}$$

$$\frac{\alpha[j]}{N[j]}E\left[(X_j^k + Z - X_r^i)^2\right] + \frac{1}{2}\sum_{i=1}^{4r}\frac{1}{4r}\beta E\left[(Y - X_r^i)^2\right], \tag{17}$$

where, because $\alpha[r_{max}] = 0$, the second sum is up to $r_{max} - 1$ only.

As seen in Figure 11, the number of nodes at a distance r away are equal to $4r$, and the first sum is to be taken over all nodes at a distance r away, so the first sum is over $4r$ nodes.

Using similar expansions as in Section 4.2, Equation (17) reduces to,

$$\gamma[r] = (1 - \beta)\sigma_z^2 + \frac{1}{4r}\sum_{i=1}^{4r}\sum_{j=1}^{r_{max}-1}\sum_{k=1}^{N[j]}\frac{\alpha[j]}{N[j]}\gamma[d_{r_ij_k}] \tag{18}$$

where $d_{r_ij_k}$ denotes the distance between the nodes X_r^i and X_j^k.

Equating $\overset{\wedge}{\gamma}[r] = \gamma[r]$ for $1 \leq i \leq r_{max}$ gives r_{max} equations. These equations along with the equation $\beta + \sum_{i=1}^{r_{max}-1}\alpha[i] = 1$ form a system of $r_{max} + 1$ non linear equations with $r_{max} + 1$ unknowns, the $\alpha[r]$'s, β and σ_z^2.

The above non linear system can be easily converted to a linear system by a change of variables.

1. Substitute $c_0 = (1 - \beta)\sigma_z^2$ in Equation (18) to get a system of r_{max} linear equations with r_{max} variables, the $\alpha[r]$'s and c_0.
2. After solving this linear system, Equation $\beta + \sum_{i=1}^{r_{max}-1}\alpha[i] = 1$ is used to obtain β.
3. Given the value of c_0 and β, get the value of $\sigma_z^2 = \frac{c_0}{1-\beta}$.

After solving the above system, $f_Y(y)$ can be obtained through Equation (25). This procedure implicitly assumes that the value of r_{max} is known. Thus we need a method to determine the value of r_{max}.

As a starting point, we choose a very large value for r_{max}. In theory, overestimating r_{max}, which results in a larger system, would still find the correct parameters. However, in practice, larger r_{max} values leads to more rounding and statistical errors, hence to small negative $\alpha[r]$'s in the solution of the non linear system. A solution to this is to start from an overestimated r_{max}, and lower its value until all the $\alpha[r]$'s are positive.

We illustrate the procedure through an example in Section 5.

4.4 A simple method to infer the model parameters for an irregular topology

In this section we present a simpler procedure than the one presented in Section 4.2 to infer the model parameters for any given topology. The rigorous procedure requires to solve integral equations of the first kind and hence, it is not always possible to get a closed form expression for the model parameter. So, we present a practical simpler method which uses the discrete model to infer the model parameters and the continuous model to populate the data and generate traces.

1. The first step is to obtain a discretized variogram, which corresponds to the continuous variogram sampled at multiples of the average nearest neighbor distance. The second method described in Section 2 is used to obtain an estimate of the continuous variogram. Recall that one of the standard variogram models is fitted to the variogram samples to get the continuous variogram. This variogram is then sampled at multiples of the average nearest neighbor distance to get the discretized variogram.
2. The second step is to use the method described in Section 4.3 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.
3. Finally, the $\alpha[r]$'s are interpolated or curve fitted and then scaled to obtain the continuous $\alpha(r)$. The scaling is carried out to ensure $\int_0^{r_{max}} \alpha(r)dr = 1$.
4. After obtaining the model parameters, we use the model described in Section 3 to generate synthetic traces.

We illustrate the procedure using an example in Section 5.

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.

5. MODEL VERIFICATION AND VALIDATION

In this section, the model parameters for experimental traces are inferred using the method described in Section 4. 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 algorithms which exploit spatial correlation, against both the traces.

5.1 Data Set Description

In this section, we describe the different experimental traces we use to verify and validate our model. The first two traces we use, the Precipitation Data Set [Widmann and Bretherton] and the S-Pol Radar Data Set [4] were obtained from remote

[4] S-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.

sensing studies and have been used in the sensor networks literature as experimental traces to evaluate algorithm performance, see, for example [Yu et al. 2003], [Pattem et al. 2004], [Ganesan et al. 2002].

5.1.1 *Precipitation Data Set.* This data set consists of the daily rainfall precipitation for the Pacific Northwest from 1949-1994. The final measurement points in the data set formed a regular grid of 50 km ×50 km regions over the region under study. We select a subset of data that has no missing values. Specifically, each snapshot of data is a 8×8 spatial grid data with a 50 km resolution.

5.1.2 *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 [Venables and Ripley 2002]. In this paper, we have selected a 64×64 spatial subset of the original data and 259 time snapshots across 2 days in May 2002.

The distance between the sensing nodes for these traces is hundreds of meters which is not representative of actual sensor networks in which the inter sensor distance is a few meters. The only publicly available sensor network traces which the authors are aware of are the SHM Trace [Paek et al. 2005] and Intel Lab Data [int 2004].

5.1.3 *SHM Trace.* One of the real world experiments where real sensor network traces have been collected after deploying a sensor network is reported in [Paek et al. 2005]. 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.

5.1.4 *Intel Lab Data.* Another real world experiment where real sensor network traces have been collected was performed in Intel Berkeley Research Lab [int 2004]. 54 sensors measuring temperature were deployed in a lab. We use a time snapshot of this trace to verify and validate our model.

Due to the lack of the publicly available traces, 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.

5.1.5 *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.

5.1.6 *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.

These 6 traces allow us to validate our model for different network densities, from an average nearest sensor-distance of tens of kilometers to hundreds of meters to a few meters. Note that the S-Pol radar data trace is the only trace which has 1000's of spatial samples. The rest of the traces are not very big, due to the difficulty in deploying very large sensor networks. Hence, the results of the S-Pol radar data trace are particularly important due to their high statistical significance.

5.2 Model Verification

5.2.1 *Precipitation data set.* We choose a snapshot in time of the precipitation data as the experimental data trace. The estimated variogram is plotted in Figure 12(b). We use this trace as an example to illustrate the procedure of Section 4.3.

First, assume the value of r_{max} to be 2. The corresponding system of equations are,

$$\gamma[1] = \sigma_z^2(1-\beta) + \frac{1}{4}\left[\frac{\alpha[1]}{2}\left(2\gamma[2] + 2\gamma[2] + 2\gamma[2]\right)\right]$$

$$\gamma[2] = \sigma_z^2(1-\beta) + \frac{1}{8}\left[\frac{\alpha[1]}{2}\left((2\gamma[1]) + 4(\gamma[1] + \gamma[3]) + 3(2\gamma[3])\right)\right].$$

Substituting $c_0 = \sigma_z^2(1-\beta)$ reduces the above system to the following linear system,

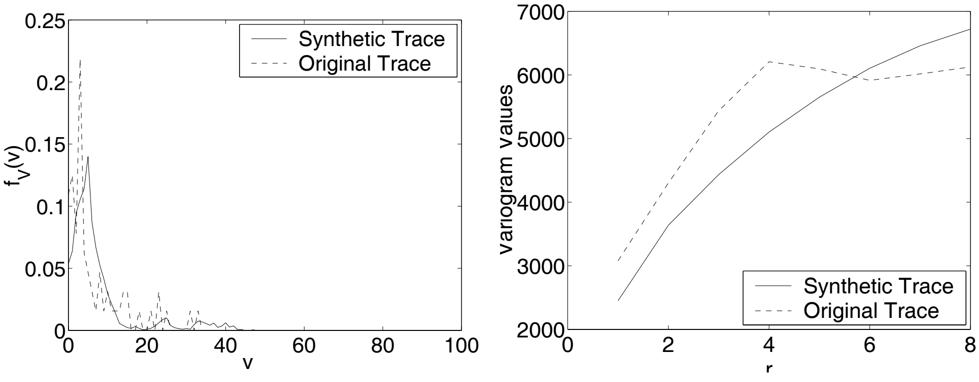
$$\gamma[1] = c_0 + \frac{1}{4}3\alpha[1]\gamma[2]$$

$$\gamma[2] = c_0 + \frac{1}{8}\alpha[1](5\gamma[3] + 3\gamma[1]).$$

Solving the above system yields $\alpha[1] = 0.9245$, $\beta = 0.0755$ and $\sigma_z = 10.08$. It can be verified that choosing $r_{max} = 3$ will result in some $\alpha[r]$'s being negative.

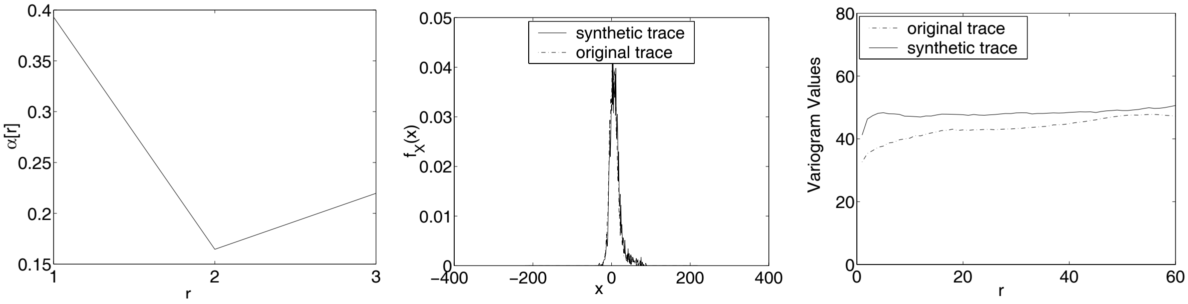
Using these parameters we generate a synthetic counterpart of the original trace. Then we compare the statistics of both the traces. Figure 12(a) shows the distribution of the two traces and Figure 12(b) shows their respective variograms. Both the distribution and variograms match closely.

5.2.2 *S-Pol Radar data set.* We choose a snapshot in time of the S-Pol Radar data as the experimental data trace. Since the underlying topology is a grid, the



(a) (b)

Fig. 12. Precipitation data trace: (a) Comparison of the distribution of the original and synthetic traces. (b) Comparison of the variogram of the original and synthetic traces.

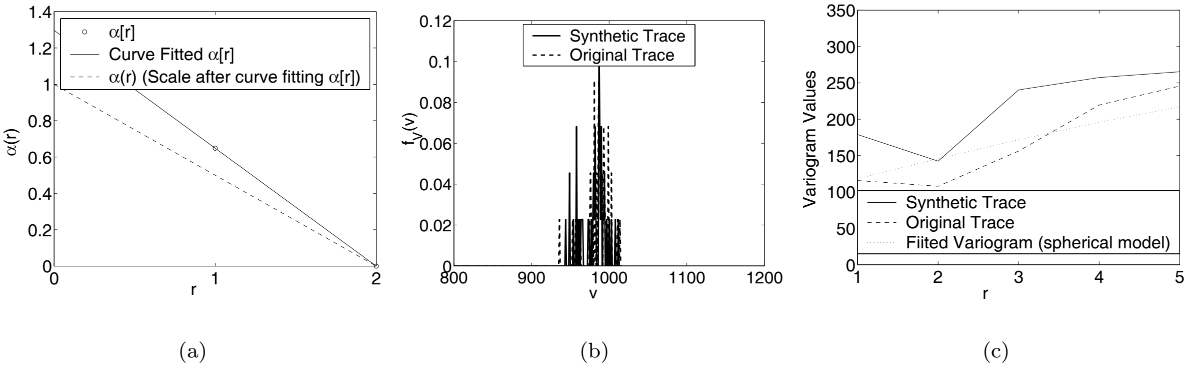


(a) (b) (c)

Fig. 13. S-Pol Radar data trace: (a) $\alpha[r]$ for the trace ($r_{max} = 4$, $\beta = 0.22$ and $\sigma_z = 3.29$). (b) Comparison of the distribution of the original and synthetic traces. (c) Comparison of the variogram of the original and synthetic traces.

parameters of the model for the trace are inferred using the method described in Section 4.3. Figure 13(a) presents the values of $\alpha[r]$, β and σ_z inferred from the trace. Using these parameters, we generate a synthetic counterpart of the original trace. Then, we compare the statistics of both the traces. Figure 13(b) shows the distribution of the two traces and Figure 13(c) shows their respective variograms. Both the distribution and the variogram of the two traces match closely.

5.2.3 *Trace 1.* Since the underlying topology for the rest of the three traces is not a grid topology, the method described in Section 4.4 will be used to infer the



(a) (b) (c)

Fig. 14. Trace 1: (a) $\alpha(r)$ which was obtained by curve fitting $\alpha[r]$'s. (b) Comparison of the distribution of the original and synthetic traces. (c) Comparison of the variogram of the original and synthetic traces.

model parameters. We illustrate the procedure step by step by using Trace 1 as an example.

1. First, the variogram has to be estimated from the given trace. The second method described in Section 2 is used to estimate the variogram. We fitted several standard variograms on to it [Olea 1999], [Goovaerts 1997] and retained the following spherical variogram since it yields the minimum square error amongst all of them:

$$\gamma(r) = \begin{cases} c_0 + c(\frac{3}{2}\frac{r}{a} - \frac{1}{2}(\frac{r}{a})^3) & ,0 \leq r \leq a \\ c_0 + c & ,a \leq r \end{cases}, \tag{19}$$

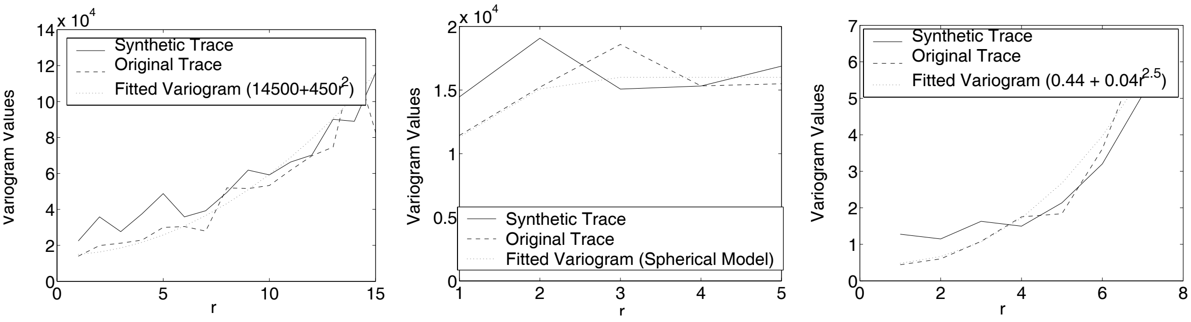
   where $c_0 = 90, c = 170$ and $a = 9$.
2. The next step is to infer the model parameters using the discrete model. The model parameters obtained are $r_{max} = 2, \alpha[1] = 0.6484, \beta = 0.3516$ and $\sigma_z = 8.5451$.
3. The discrete $\alpha[r]$'s are curve fitted and then scaled to obtain $\alpha(r)$. The resulting $\alpha(r)$ is plotted in Figure 14(a). We fit a linear curve onto $\alpha[r]$'s because we have only two known values of $\alpha(r)$.

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 14(b) and their variograms in Figure 14(c). Both the distribution and the variograms match closely.

5.2.4 *Trace 2.* The variogram of the second trace is best estimated by the power semi variogram model (Equation (20)) with parameters $c_0 = 14500$, $\omega = 2$ and $c = 450$.

$$\gamma(r) = c_0 + cr^{\omega}. \tag{20}$$

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



(a) (b) (c)

Fig. 15. Comparison of the variogram of the original and the synthetic traces: (a) Trace 2. (b) SHM Trace. (c) Intel Lab Data

the variograms of both the traces in Figure 15(a). Once again, the variograms match closely.

5.2.5 *SHM Trace.* The variogram of the SHM trace is best estimated by the spherical semi variogram model (Equation (19)) 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 15(b). Once again, the variograms match closely.

5.2.6 *Intel Lab Data.* The variogram of the Intel Lab Data is best estimated by the power semi variogram model (Equation (20)) with parameters $c_0 = 0.44, \omega = 2.5$ and $c = 0.04$. 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 15(c). Once again, the variograms match closely.

5.3 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 two of these algorithms on both the original and the synthetic traces and compare the performance.

Amongst the many such algorithms mentioned in the introduction, we selected DIMENSIONS [Ganesan et al. 2002] and CC-MAC [Vuran and Akyildiz 2006] to evaluate our model. We did not choose any of the algorithms which use entropies because we do not have enough time snapshots of sensor data traces to calculate the joint entropies. We went for algorithms whose evaluation metrics depended chiefly on the nature of correlation in data. The algorithms along with the comparisons are discussed below.

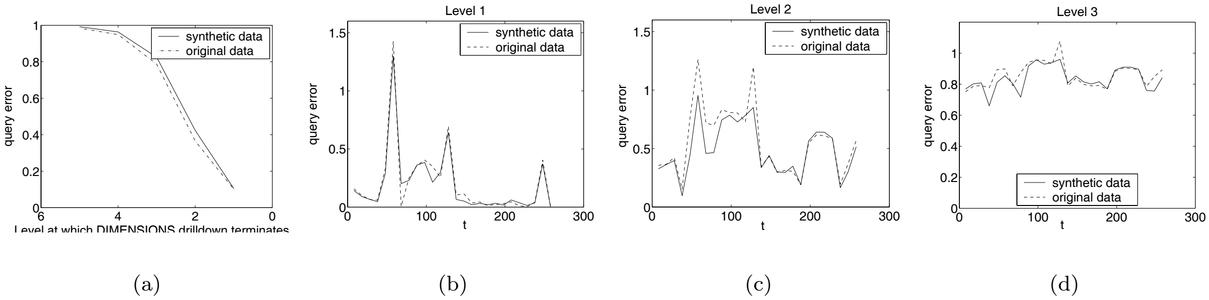
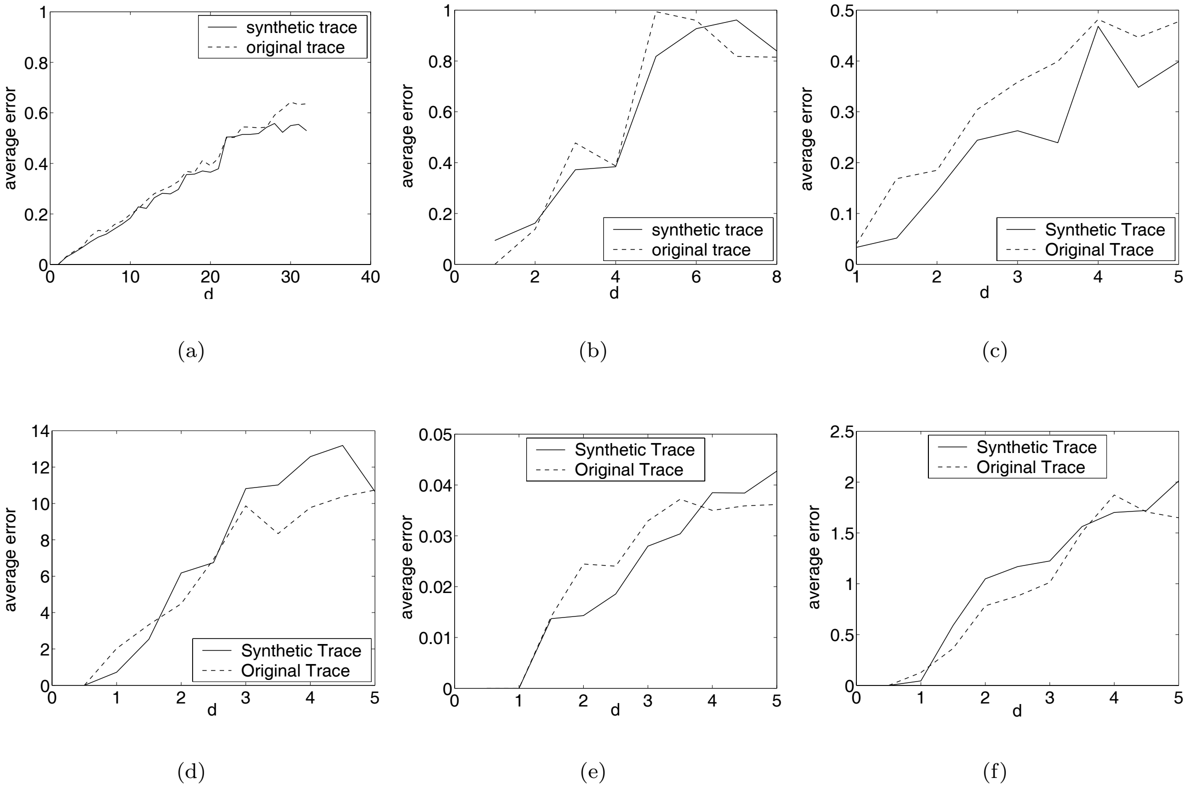


Fig. 16. Comparison of the performance of DIMENSIONS on original and synthetic trace. (a) Error vs Query Termination Level. (b) Error at query termination level 1 at different snapshots. (c) Error at query termination level 2 at different snapshots. (d) Error at query termination level 3 at different snapshots

5.3.1 *DIMENSIONS.* This is a data storage and querying algorithm. It proposes wavelet based multi-resolution summarization and drill down querying. Summaries are generated in a multi-resolution manner, corresponding to different spatial scales. Queries on such data are posed in a drill down manner, that is, they are first processed on coarse, highly compressed summaries corresponding to larger spatial volumes, and the approximate results obtained are used to focus on regions in the network that are most likely to contain relevant information. A variety of queries can be posed on the data set; we present the performance results for the query $average(X)$. The evaluation metric used is the query error which is defined as $QueryError = (QueryResponseOverDimensions - ActualValue)/ActualValue$. In the DIMENSIONS hierarchy, each lower level stores twice the amount of data as the higher level. Therefore, as the query processing proceeds down the hierarchy gaining access to more detailed information, the query error should drop down gradually.

We run DIMENSIONS on the S-Pol radar data trace only because the rest of the traces do not have sufficient spatial samples. We first choose a snapshot in time of the S-Pol radar data as the experimental data trace. After inferring the parameters of the model for the original trace, we generate a synthetic counterpart of the original trace. Figure 16(a) shows the result of running DIMENSIONS on both the traces. It is evident that the two plots match very well. To confirm the observation, we then infer the model parameters for different snapshots in time and run DIMENSIONS on both the original and synthetic traces. Figures 16(b)-16(d) show the comparison for different query termination levels. It is obvious that the performance of the algorithm for both the traces is similar. Thus we conclude that the synthetic data is able to capture the spatial correlation in the original data.

5.3.2 *CC-MAC.* The underlying idea behind CC-MAC 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



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 17. Comparison of the performance of CC-MAC on original and synthetic trace: Variation of error with d. (a) S-Pol Radar Data Trace. (b) Precipitation Data Trace. (c) Trace 1. (d) Trace 2. (e) SHM Trace (f) Intel Lab Data

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.

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 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 S-Pol Radar data trace, Precipitation Data Trace, Trace 1, Trace 2, the SHM trace and the Intel Lab Data in Figures 17(a) - 17(f) 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.

6. RELATED WORK

6.1 Interpolation Techniques

[Yu et al. 2003] proposed the use of interpolation techniques to estimate data at unmonitored locations and then use this estimated data to generate data at irregular topologies. These techniques find application in converting a real trace to a more fine grained trace (having more sensor nodes in the same area). On the other hand, in this work we propose a mathematical model that can be used to generate synthetic traces without a real trace. Modifying the parameter β allows a user to generate traces having different correlations. It can also be used to convert a real trace to a more fine grained trace as well as converting a real trace to bigger trace while retaining its granularity (having more nodes in a bigger area while preserving the density).

6.2 Markov Random Fields (MRF's)

Markov random fields [Li 2001] are used in image processing and computer vision to model dependent random variables such as image pixels and correlated features. The Markov random field model is defined using the joint statistics of the correlated variables. Appendix A.1 provides a brief description of the MRF model. The MRF model provides an alternative but complicated model for sensor network data.

The objective of a modeller is to come up with the simplest possible model which is an accurate representation of reality. Using the most general model is too cumbersome in practice, and it is a bad idea to make a model more complicated without adding any more accuracy.

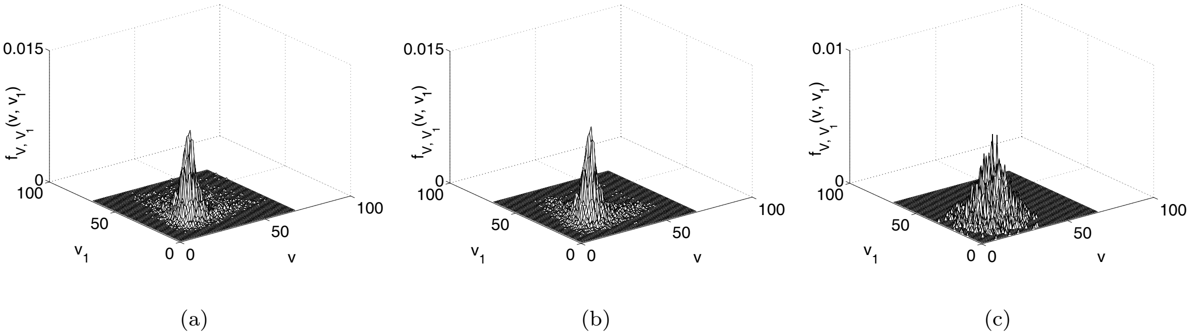
While our model was driven by the intuition obtained after studying the correlation characteristics of real data without using the general MRF theory, it is a special case of MRF with much fewer parameters than the general MRF model. Moreover, it is completely defined by the first order statistic and the second order moments of the process. Hence, unlike the MRF model, it is easily tractable by analysis.

6.3 Jointly Gaussian Model

The most commonly used model for sensor network data is the jointly Gaussian model [Vuran and Akyildiz 2006], [Cristescu and Vetterli 2005], [Marco et al. 2003], [Deshpande et al. 2004]. The primary reasons for this choice is ease of use and analytical tractability, rather than accuracy [Yu et al. 2004].

Our model is more general and more realistic 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 joint 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 (25).

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 discussed in Section 5, even the first order distribution is not Gaussian. (The S-Pol Radar data trace is an exception, as seen in Figure 13(b).) The proposed model has no such restrictions: through a proper choice of $f_Y(y)$ and $f_Z(z)$, any first order



(a)

(b)

(c)

Fig. 18. $F_{V,V_1}(v, v_1)$ for the (a) Original Trace. (b) Synthetic trace generated using the proposed model. (c) Synthetic trace generated using the jointly Gaussian model.

distribution function of data values can be modeled, and through a proper choice of $\alpha(r)$ and β, any correlation structure can be modeled.

We now evaluate how appropriate is the jointly Gaussian model for the S-Pol radar data trace whose first order distribution is Gaussian. To determine the accuracy of the model, we plot the joint pdf V and V_1, where V_1 represents the data value at nodes at a unit distance away from V. Figures 18(a) - 18(c) show $f_{V,V_1}(v, v_1)$ for the original trace, a synthetic trace generated using the proposed model and a synthetic trace generated assuming data to be jointly Gaussian. The covariance matrix of the jointly Gaussian trace is constructed such that the variogram of the synthetic trace is the same as that of the original trace. The joint pdf's of the original trace and the trace generated using the proposed model are similar, but they are very different from the joint pdf of the trace generated using the jointly Gaussian model. We use the Jensen-Shannon divergence [5] [Lee 1999] to quantitatively measure how close two distributions are. The smaller is its value, the more similar are the two distributions. The Jensen-Shannon divergence between the joint pdf of the original trace and the synthetic trace generated using the proposed model is 0.04 while, between the joint pdf of the original trace and the trace generated using the jointly Gaussian model, it is equal to 0.15.

While the proposed model is more accurate than the jointly Gaussian one, it is clearly more complicated.

Based on previous discussion, since the jointly Gaussian model is more tractable, it should be used to predict trends and get some quick intuition into the behavior of an algorithm. But to do a more thorough and realistic 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, the proposed model can act as a close substitute.

[5] See Appendix A.4 for a detailed description of Jensen-Shannon divergence

7. DISCUSSION: CHIEF ASSUMPTIONS OF THE PROPOSED MODEL AND THEIR IMPLICATIONS

In this section, we discuss how accurate and general is the proposed model. We restate our chief assumptions and discuss their physical meaning allowing us to comment on how general is the model.

The first and the most important assumption we make is the Markovian nature of the model (see remark in Section 3.1). This makes sense intuitively since it implies that the data value at a node is derived from other correlated nodes whose data values have already been derived. But it is not always the case that a given spatial process will be Markovian. For example, processes which are governed by diffusion equations that use fractional derivatives [Liu et al. 2004] are non Markovian. For such a scenario, even a much more complicated model like Markov random field will fail to capture the correlation characteristics of the data.

It is to be noted that there is no such assumption that only nodes which are close to each other will be correlated. If some strange application results in nodes which are close to each other to be uncorrelated but nodes far away to be correlated, then the model will not fail. For this application, $\alpha(r)$ will not be a decreasing function but will reach a maximum around the distance at which the nodes show maximum correlation.

The second assumption we make is that the correlation depends only on the distance between the nodes, not the direction. The good fit between the statistics of the original and the synthetic traces validates this assumption. Note that throwing away this assumption only makes the model more complicated. Now, α will no longer be just a function of the distance, it will depend on the direction too.

The third assumption we make is that the data values are derived from a stationary process. If some application results in two or more regions separated by boundaries, and within each region, the data values follow a different process, then our model will not be able to capture these boundaries, though it will model the correlation in data within each region. A Markov random field model can be used for this scenario as it can specify different conditional probabilities for each region.

The model makes no assumption on the granularity of the data, network density, the topology and the number of source nodes.

8. TOOLS TO GENERATE LARGE SYNTHETIC TRACES

In this section we describe four 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.

—*generateLargeTraceFromSmall and generateLargeTraceFromIrregular* will create large synthetic traces having the same correlation structure as the input real data trace on a grid topology and irregular topologies respectively. It takes the real data trace and the dimensions of the output synthetic 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 3.1.

—*generateSyntheticTraces and generateSyntheticTracesOnIrregular* will create large synthetic traces representing a wide range of conditions by tweaking the model parameters on a grid topology and irregular topologies respectively. 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 two tools can generate a large trace having similar correlation properties as the real trace, and hence, help researchers to evaluate protocols with real data. The last two tools will enable researchers to evaluate their protocols with data having varied correlation structures. Hence, these four 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.

9. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a mathematical model to capture the spatial correlation in sensor network data. This model can generate synthetic traces representing a wide range of conditions and exhibiting any degree of correlation. We also described a mathematical procedure and a simple, practical method to infer the model parameters from a real trace. These model parameters are then used to generate synthetic traces having similar correlation properties as the real trace.

We verified our model by showing that the statistics of the synthetic trace is similar to the real trace. We validated our model by showing that the performance of sensor network algorithms exploiting spatial correlation is similar for both the traces. For this purpose, we used the sensor network data storage and querying algorithm DIMENSIONS and the Spatial Correlation based Collaborative Medium Access Control algorithm CC-MAC. The real traces we have used for model verification and validation have network densities ranging from tens of kilometers to a few meters. Finally, we have created four tools to enable researchers to generate data representing real world scenarios and wide range of conditions.

The next step is to extend the framework to model spatio-temporal correlation in data. This is useful since most of the algorithms try to exploit both spatial and temporal correlation in data simultaneously.

REFERENCES

2004. Intel lab data, http://berkeley.intel-research.net/labdata.

Chou, J., Petrovic, D., and Ramchandran, K. 2002. Tracking and exploiting correlations in dense sensor networks. In *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002.*

Cover, T. M. and Thomas, J. A. 1991. *Elements of Information Theory.* John Wiley.

Cressie, N. 1993. *Statistics for Spatial Data.* John Wiley.

Cristescu, R., Beferull-Lozano, B., and Vetterli, M. 2004. On network correlated data gathering. In *Proceedings of the IEEE Infocom'04.*

Cristescu, R. and Vetterli, M. 2005. On the optimal density for real-time data gathering of spatio-temporal processes in sensor networks. In *Proceedings of IPSN'05.*

Deshpande, A., Guestrin, C., Madden, S., Hellerstein, J., and Hong, W. 2004. Model driven data acquisition in sensor networks. In *30th International Conference on Very Large Data Bases (VLDB 2004).*

Faruque, J. and Helmy, A. 2004. Rugged: Routing on fingerprint gradients in sensor networks. In *IEEE International Conference on Pervasive Services (ICPS'2004).*

Ganesan, D., Estrin, D., and Heidemann, J. 2002. Dimensions: Why do we need a new data handling architecture for sensor networks? In *First Workshop on Hot Topics in Networks (Hotnets-I)*.

Ganesan, D., Greenstein, B., Perelyubskiy, D., Estrin, D., and Heidemann, J. 2003. An evaluation of multi-resolution storage for sensor networks. In *SenSys'03*.

Goel, A. and Estrin, D. 2003. Simultaneous optimization for concave costs: single sink aggregation or single source buy-at-bulk. In *SODA*. 499–505.

Goovaerts, P. 1997. *Geostatistics for natural resources evaluation*. Oxford University Press.

Intanagonwiwat, C., Estrin, D., Govindan, R., and Heidemann, J. 2002. Impact of network density on data aggregation in wireless sensor networks. In *ICDCS*.

Jindal, A. and Psounis, K. 2004. Modeling spatially-correlated sensor network data. In *Proceedings of the IEEE International Conference on Sensor and Ad hoc Communications and Networks*.

Kanwal, R. P. 1997. *Linear Integral Equations: Theory and Technique*. Birkhauser Boston Academic Press.

Kargupta, H., Joshi, A., Sivakumar, K., and Yesha, Y. 2003. *Data Mining: Next Generation Challenges and Future Directions*. AAAI/MIT Press.

Krishnamachari, B., Estrin, D., and Wicker, S. B. 2002. The impact of data aggregation in wireless sensor networks. In *ICDCS Workshop on Distributed Event-based Systems (DEBS)*.

Kythe, P. K. and Puri, P. 2002. *Computation Methods for Linear Integral Equations*. Birkhauser Boston Academic Press.

L.Doherty and K.Pister. 2004. Scattered data selection for dense sensor networks. In *Proceedings of IPSN'04*.

Lee, L. 1999. Measures of distributional similarity. In *Proceedings of the 37th ACL*.

Li, S. Z. 2001. *Markov Random Field Modeling in Image Analysis*. Springer.

Liu, F., Shen, S., Anh, V., and Turner, I. 2004. Analysis of a discrete non-markovian random walk approximation for the time fractional diffusion equation. In *Proceedings of the 12th Biennial Computational Techniques and Applications Conference (CTAC'04)*.

Marco, D., Duarte-Melo, E., Liu, M., and Neuhoff, D. 2003. On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data. In *Proceedings of IPSN'03*.

Olea, R. A. 1999. *Geostatistics for engineers and earth scientists*. Kluwer Academic Publishers.

Paek, J., Chintalapudi, K., Caffrey, J., Govindan, R., and Masri, S. 2005. A wireless sensor network for structural health monitoring: performance and experience. In *Second IEEE Workshop on Embedded Networked Sensors*.

Pattem, S., Krishnmachari, B., and Govindan, R. 2004. The impact of spatial correlation on routing with compression in wireless sensor networks. In *Symposium on Information Processing in Sensor Networks (IPSN)*.

Porter, D. and Stirling, D. S. 1990. *Integral Equations: A practical treatment. from spectral theory to applications*. Cambridge University Press.

Psounis, K., Zhu, A., Prabhakar, B., and Motwani, R. 2004. Modeling correlations in web-traces and implications for designing replacement policies. *Computer Networks Journal, Elsevier 45,* 4, 379–398.

Rahimi, M., Pon, R., Kaiser, W. J., Sukhatme, G. S., Estrin, D., and Srivastava, M. 2004. Adaptive sampling for environmental robotics. In *IEEE International Conference on Robotics and Automation*.

Venables, W. and Ripley, B. 2002. *Modern applied statistics with S*, fourth ed. Springer.

Vuran, M. C. and Akyildiz, I. F. 2006. Spatial correlation-based collaborative medium access control in wireless sensor networks. *to appear in IEEE/ACM Transactions on Networking*.

Whitehouse, K. and Culler, D. 2002. Calibration as parameter estimation in sensor networks. In *Proceedings of WSNA'02*.

Widmann, M. and Bretherton, C. 50 km resolution daily precipitation for the pacific northwest, 1949-1994, http://tao.atmos.washington.edu/data-sets/widmann.

Yu, Y., Estrin, D., Govindan, R., and Rahimi, M. 2004. Using more realistic data models to evaluate sensor network data processing algorithms. In *First IEEE Workshop on Embedded Networked Sensors*.

Yu, Y., Ganesan, D., Girod, L., Estrin, D., and Govindan, R. 2003. Synthetic data generation to support irregular sampling in sensor networks. In *Geo Sensor Networks 2003*.

A. APPENDIX

A.1 Markov Random Fields

Markov Random Fields (MRF's) have been used in Image Processing and Computer Vision to model dependencies between random variables [Li 2001]. The concept of MRF's is a generalization of Markov processes which are widely used in sequence analysis. An MRF is defined on a domain of space rather than time. This section gives a brief introduction to the MRF model.

Let $F = F_1, \dots F_m$ be a family of random variables defined on the set $\mathcal{S}$, in which each random variable F_i takes a value f_i in $\mathcal{L}$. The family F is called a random field. For a discrete label set $\mathcal{L}$, the probability that random variable F_i takes the value f_i is denoted by $P(f_i)$ and the joint probability is denoted by $P(f) = P(F_1 = f_1, F_2 = f_2, \dots F_m = f_m)$. For a continuous $\mathcal{L}$, we have probability density functions, $p(f_i)$ and $p(f)$.

F is said to be a Markov random field on $\mathcal{S}$ with respect to a neighborhood system $\mathcal{N}$ if and only if the following two conditions are satisfied:

$$P_F(f) > 0, \ \forall f \in \mathcal{F} \ \text{ positivity} \tag{21}$$

$$P(f_i \mid f_{\mathcal{S}-\{i\}}) = P(f_i \mid \mathcal{N}_i) \ \text{ Markovianity} \tag{22}$$

where $\mathcal{S} - \{i\}$ is the set difference, $f_{\mathcal{S}-\{i\}}$ denotes the set of labels at the sites in $\mathcal{S} - \{i\}$ and $f_{\mathcal{N}_i} = \{f_{i'} \mid i' \in \mathcal{N}_i\}$ stands for the set of labels at the sites neighboring i. The positivity is assumed for some technical reasons and can usually be satisfied in practice. The Markovianity depicts the local characteristics of F. In MRF's, only neighboring labels have direct interactions with each other.

There are two approaches for specifying an MRF, either in terms of the conditional probabilities $P(f_i \mid f_{\mathcal{N}_i})$ or in terms of the joint probability $P(f)$. This modeling approach has the definition of the neighborhood system, $\mathcal{N}$, and the joint pdf's as parameters.

The model proposed in this paper is a special case of a MRF and requires much fewer parameters for a complete description of the model. The parameter r_{max} defines the neighborhood system and the parameters Y, Z, α_i's and β define the conditional probability $P(f_i \mid f_{\mathcal{N}_i})$.

A.2 Standard Variogram Models

When estimating a variogram, it is a common practice to fit one of the standard variogram models to the trace. This section presents the five most common variogram models used in Geostatistics [Olea 1999], [Goovaerts 1997].

(i) Nugget effect model:

$$\gamma(r) = \begin{cases} 0 & r = 0 \\ 1 & \text{otherwise} \end{cases} .$$

(ii) Spherical model with range a:

$$\gamma(r) = \begin{cases} c_0 + c\left(\frac{3r}{2a} - \frac{1}{2}\left(\frac{r}{a}\right)^3\right) & r \leq a \\ c_0 + c & \text{otherwise} \end{cases}.$$

(iii) Exponential model with practical range a:

$$\gamma(r) = c_0 + c\left(1 - e^{\frac{-3r}{a}}\right).$$

(iv) Gaussian model with practical range a:

$$\gamma(r) = c_0 + c\left(1 - e^{\frac{-3r^2}{a^2}}\right).$$

(v) Power model:

$$\gamma(r) = c_0 + cr^{\omega},$$

where ω is a parameter of the model.

A.3 Relationship between the distributions of V, Y and Z

In this section, we prove Lemma 4.1.

From Equation (6) we get,

$$f_V(v) = (1-\beta)f_{V+Z}(v) + \beta f_Y(v). \tag{23}$$

Using characteristic functions and since V and Z are independent, Equation (23) can be rewritten as,

$$\Phi_V(j\omega) = (1-\beta)\Phi_V(j\omega)\Phi_Z(j\omega) + \beta\Phi_Y(j\omega). \tag{24}$$

Since Z is a zero mean normal random variable, its characteristic function is given by $e^{[\frac{-\sigma_z{}^2\omega^2}{2}]}$. Equation (24) finally reduces to

$$\Phi_V(j\omega) = \frac{\beta}{1-(1-\beta)e^{[\frac{-\sigma_z{}^2\omega^2}{2}]}}\Phi_Y(j\omega). \tag{25}$$

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 (25) can now be rewritten as

$$V = Y + L.$$

A.4 Jensen-Shannon divergence: A metric to measure how similar two distributions are

Jensen-Shannon divergence [Lee 1999] is a useful measure of the distance between two distributions. Let $q_V(v)$ and $r_V(v)$ denote two probability density functions of the random variable V. Let $avg_V^{q,r}(v) = \frac{1}{2}\left(q_V(v) + r_V(v)\right)$ denote the average distribution of q and r. Then the Jensen Shannon divergence, $JS(q,r)$ is defined as

$$JS(q,r) = \frac{1}{2}\left[D\left(q\|avg^{q,r}\right) + D\left(r\|avg^{q,r}\right)\right].$$

The function D is the KL divergence [Cover and Thomas 1991] which measures the average inefficiency in using one distribution to code for another and is defined as

$$D\left(q(v)\|avg^{q,r}(v)\right) = \sum_{v} q(v)log\left(\frac{q(v)}{avg^{q,r}(v)}\right).$$

...