Local Linear Suppression for Wireless Sensor Network Data

Kristian Lum* Alan E. Gelfand †

Abstract

With wireless sensor networks, preserving battery life is critical. For such sensors, data collection is relatively cheap while data transmission is relatively expensive. For such networks in ecological settings, certain processes are sufficiently predictable so that transmission of data at a particular time can be suppressed if it does not differ from what is expected at that time. That is, there will not be much loss of *information* with regard to inference. More precisely, there is a presumed model to explain the measurements collected at the sensors, which provides insight into what is expected at a given node, at a given time. Under the suppression, inference objectives include both estimation of the process parameters as well as reconstruction of the entire time series at each of the nodes.

In this paper, we build on the existing literature that has offered ways in which one can use suppression in wireless sensor networks to limit the number of transmissions. We introduce a new, computationally cheap, locally linear suppression scheme based upon process knowledge and compare it to the commonly used "constant" suppression scheme. Maintaining the same suppression threshold, we demonstrate decreased transmission rates under the new scheme while producing comparable posterior inference relative to constant suppression scheme. That is, the untransmitted readings are bounded to within an interval of the same length under both schemes, but the linear suppression scheme will transmit less data.

We implement this scheme for a synthetic dataset produced under the assumption of a diffusion model and show that even under high suppression rates, we are able to recover simulation parameters. We also implement linear suppression on data collected from a real wireless sensor network that measures the amount of light filtering through the forest canopy at a set of locations in the Duke Forest. We show that the in-sample

*K Lum (kristian@dme.ufrj.br) is a postdoctoral researcher in the Departamento de Métodos Estatísticos, Instituto de Matemática, Universidade Federal de Rio de Janeiro.

†AE Gelfand (alan@stat.duke.edu) is a professor in the Department of Statistical Science, Duke University, Durham NC 27707 USA

predictive sum of squared errors from the suppressed data is only a bit larger than that from the full dataset.

Bayesian model, constant suppression, dynamic model, Euler discretization, Markov chain Monte Carlo, Orenstein-Uhlenbeck process, time series

The authors thank Jun Yang, Yi Zhang, Paul Flikkema, and Jim Clark for helpful discussion. This work was supported in part by NSF DDDAS 054034L and 050414NL, and CAPES/FAPERJ.

1 Introduction

Sensor networks are able to extract spatially referenced data in novel ways to learn about processes ranging from the social patterns of zebras (Zhang et al. (2004)) to the forest dynamics of redwood trees (Tolle et al. (2005)). Such data enables interesting new models but also introduces new challenges to the data collection process. Of interest here are wireless sensor networks, small nodes that collect and transmit data, relying solely upon their individual batteries. Replacement of these batteries may be difficult, perhaps impossible. So, attention to energy savings is crucial to the viability of the network. For such sensors, in terms of battery use, data collection is cheap while data transmission is expensive. Hence, attention to ways in which the sensor can minimize data transmission are of particular importance.

Suppression is a commonly employed way in which sensors can reduce transmission. By suppressing the transmission of data that is similar to, i.e., within some *distance* of recently transmitted data, not only is less data sent, but we also have information about the data that was not sent. That is, because we know the mechanism by which the data was suppressed, we can bound the possible range of values of each suppressed piece of data. These bounds are only applicable if we are confident that all of the transmissions the node intended to send, in fact, arrived successfully at the base station. Unfortunately, with the current state of technology, this is often not the case. Silberstein et al. (2007) and Puggioni and Gelfand (2010) treat this transmission failure by appending a record of the time stamp of the last r transmissions. Any failures are then incorporated into the model.

Sensor networks monitor dynamic processes resulting in a time series at each sensor location. The contribution of this paper is to introduce a more informative but locally cheap suppression scheme that anticipates dynamic behavior in the *mean* of the process being observed. Rather than using a suppression scheme that only takes the last transmitted value into account, we suppress according to a locally linear trend in the mean. We implement this scheme within a fully model-based setting. Evidently, there is loss of information in suppression. Assessment of performance focuses on both estimation of model parameters and on predictive performance (reconstruction) of the full time series from the partially transmitted one.

Sensor networks are an increasingly common data collection mechanism across a variety of fields. Indeed the Association for Computing Machines (ACM) has published a quarterly journal, Transactions on Sensor Networks, since August 2005. Selected applications include tracking (Juang et al. (2002)), monitoring volcanoes (Werner-Allen et al. (2006)), and forest dynamics (Szewczyk et al.,2004). Such networks are developed to infer about a process over a region which the sensors span. However, they move beyond "data loggers", where data is collected locally and retrieved locally. In a network, the sensors can communicate with each other as well as with a "gateway" or base station. In some designed fashion, the sensors transmit data to the gateway which serves as a repository for the data.

Recent advances in wireless sensor technology have expanded the possibilities of environmental modeling. In particular, continuous collection of data has become feasible at temporal and spatial scales that were unattainable in the past. Furthermore, wireless sensors can be placed in locations where measurement would otherwise be very costly (requiring specialized technicians), or cumbersome (because of landscape and climatic limitations). Examples of data that are suited to collection with a wireless sensor network include soil moisture, light availability, temperature, and atmospheric CO_2.

Suppression introduces missingness that is a generalization of more familiar censoring (see, e.g., Sun (2006) and further references therein). With censoring, an observation is restricted to a specified (possibly random) set. With suppression, an observation is restricted to a set determined by the previous observations. In other words, suppression is "informed" missingness. It is not sampling at coarser temporal resolution. We note that data suppression is a very broad term in the literature, applied to contexts such as filtering, cleaning, acquisition, confidentiality, and misrepresentation. In the setting of sensor networks we note the recent work of Chu, et al. (2006), Silberstein, et al. (2006) and Silberstein, et al. (2007).

With wireless sensors, we envision high levels of suppression - potentially 70 % or more of the time. Such levels of missingness are much higher than we work with in customary statistical inference settings but, with processes that are highly predictable, such suppression need not cost much in terms of inference performance regarding the process. We note that our goal here is not network design or communication. We are not seeking optimal placement of sensors, optimal specification of sensors, optimal collection rates, optimal communication between sensors, etc. Rather, under a given network, we are focused on the impact of a novel suppression scheme on our ability to learn about the process of interest.

We apply this linear (first order approximation) suppression scheme to two illustrative simulated data examples and compare it to the "comparison with last transmitted" suppression scheme which implicitly assumes a constant mean. The simulated data comes from a stochastic differential equation model. As a real example, we apply linear suppression to a data set of readings of light availability from the Duke Forest in North Carolina. In this setting, we show that even in cases with greater than half of the data untransmitted, the mean posterior sum of squared errors only increases by about 3% over that of the model fitted with the full data series.

Hence, the format of the paper is as follows. In Section 2 we briefly review the "last transmitted suppression" scheme. Section 3 presents a *linear* suppression scheme which can be applied using local linearization of the mean. Section 4 describes a simulation example driven by an Orenstein-Uhlenbeck process. Section 5 investigates the Duke Forest data. Section 6 concludes with a brief summary and possible future work.

2 Last transmitted suppression

Data collection is assumed over a discretized time scale. A very simple suppression algorithm, used in, e.g., Silberstein et al. (2007) and Puggioni and Gelfand (2010), is the *constant* suppression scheme: transmit a new value if it is sufficiently different from the last transmitted value. That is, without loss of generality, let the node begin recording values at time $t = t_0$, at which the first reading, Y_{t_0}, is transmitted. Let t_l be the time stamp of the most recent transmission (which the algorithm resets to $t_0 = t_l$). At each subsequent time point, $t = t_l + i$, transmit Y_{t_l+i} if $|Y_{t_l} - Y_{t_l+i}| > \epsilon$ for some pre-selected threshold, ϵ. If Y_{t_l+i} is transmitted, set $t_0 = t_l + i$ and continue the algorithm, transmitting the next value that differs from the last transmitted value by more than ϵ. Under this algorithm, with no transmission failure, it is clear that each of the missing readings can be bounded to be within an interval of length 2ϵ.

This scheme is suited for a situation in which $E[Y_t|Y_{t_l}] = Y_{t_l}$ for $t > t_l$. Hence, $E[Y_t]$ is constant; there is no drift in the mean. For instance, Figure 1 shows this suppression scheme applied to a Gaussian random walk. The trajectory given by the solid line is observed. The full circles, along with the unfilled ones reveal the full dataset.

3 Linear Suppression

The constant suppression scheme fails to take advantage of possible trend in the incoming data. By ignoring the trajectory of the data collected at a sensor, an opportunity is missed for decreased transmission at that sensor under the same threshold as the constant suppression scheme. We propose a linear suppression scheme instead. Consider a simple dynamic model (West and Harrison (1999)) with the assumption of an *observational* linear trend, $E[Y_t] = a + bX_t$, where X_t evolves dynamically. For convenience, in the sequel we set $X_t = t$ but the scheme is applicable to general X_t. We use the most recent

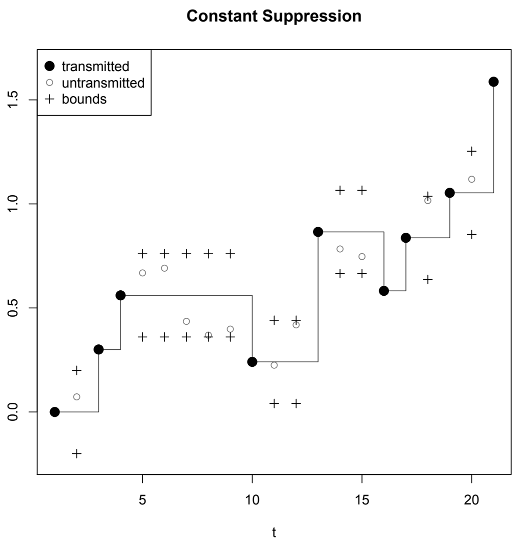

Figure 1: Data generated from a Gaussian random walk (circles). Each of the filled circles represents a transmitted value with the +s showing the ϵ bounds. (See text for details.)

transmission and the value immediately following it to calculate current estimates of a and b in order to inform suppression decisions.

As a first version, let t_0 be the time at which a sensor begins taking readings, and let t_1 be the following reading. Initialize $t_{l_0} = t_0$ and $t_{l_1} = t_1$ to be the two most recently transmitted readings. At each time $t > t_{l_1}$, use t_{l_0} and t_{l_1} to calculate current $\hat{a}$ and $\hat{b}$, the coefficients that connect a line between $(t_{l_0}, Y_{t_{l_0}})$ and $(t_{l_1}, Y_{t_{l_1}})$, which can be found using simple algebra. If $|Y_t - \hat{a} - \hat{b}t)| > \epsilon$, transmit Y_t and set $t_{l_0} = t_{l_1}$ and $t_{l_1} = t$. If the newest reading, Y_t falls within the ϵ-bound of the linear predictor, $\hat{Y}_t = \hat{a} + \hat{b}t$, do not transmit Y_t. In the constant suppression case we do not need to *know* the mean; here, we do not need to *know* the trend.

A second algorithm which allows faster adaptation to a quickly changing linear trend would use only the most recent transmission. Under this algorithm, we can suppress by assuming that the points since the last transmission approximately follow the line implied by Y_{t_l} and Y_{t_l+1}, where t_l was the time of the transmission. The predicted value at t_l+i for $i > 1$ is then $\hat{a} + \hat{b}(t_l + i)$, where $\hat{a}$ and $\hat{b}$ are calculated such that the line passes through Y_{t_l} and Y_{t_l+1}. In this case, the transmission rule is different. Again, for $i > 1$, if $|Y_{t_l+i} - \hat{a} - \hat{b}(t_l + i)| < \epsilon$, do not transmit. If $|Y_{t_l+i} - \hat{a} - \hat{b}(t_l + i)| > \epsilon$, transmit Y_{t_l+i}. Then, we know that $Y_{t_l} + jb_{\min} - \epsilon \leq Y_{t_l+j} \leq Y_{t_l+j} + jb_{\max} + \epsilon$ for $1 < j < i-1$, where $b_{\min} = \frac{(Y_{t_l+i-1}-\epsilon)-Y_{t_l}}{i-1} \leq b \leq \frac{(Y_{t_l+i-1}+\epsilon)-Y_{t_l}}{i-1} = b_{\max}$. Note that, because we use adjacent time points to create the local linear predictor, the estimated lines may have high variability. Also, due to the uncertainty about the $\hat{b}$ that was used for suppression, the bounds become wider as we move away in time from the last transmission. Still, because it adapts rapidly, we employ this scheme in the sequel.

Figure 2 shows an illustration of the second linear suppression scheme. The dotted line, which passes through the last transmitted value and the following value, shows the line to which each subsequent transmission is compared. Once a reading deviates from the dotted line by more than ϵ, the previous value (dot) is transmitted. Figure 3 shows a periodic time series without noise (top) and with random deviations (bottom) suppressed by both linear and constant suppression with the same threshold. It reveals that the linear algorithm can achieve faster adaptation to a quickly changing locally linear trend relative to the constant suppression scheme, particularly when the approximately linear trend is strong relative to the noise.

More elaborate local suppression schemes can be developed, such as second order approximation. However, we suspect that, in many cases, this would be locally very unstable. Moreover, we focus on first order approximation as an approach that requires negligible additional local computation compared with the constant suppression scheme. We note that if $EY_t = a\exp(-bt)$, we can take logs and then implement linear suppression, similarly for any *linearizable* form, i.e., a mean allowing a one-to-one transformation to linearity. More generally, we might devise linear suppression as a first order

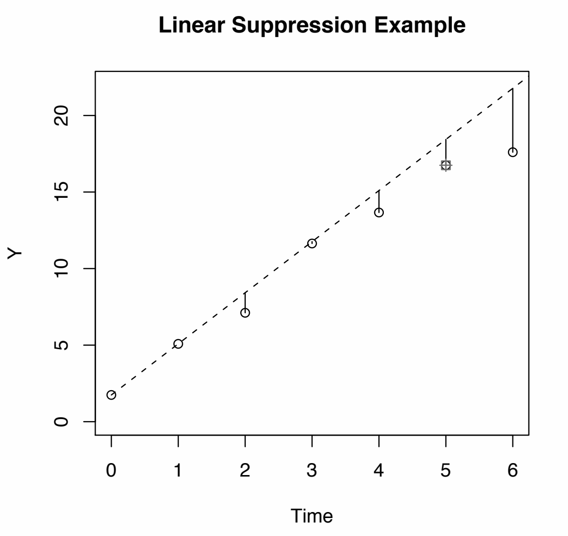

Figure 2: A cartoon illustration of the second linear suppression scheme. The dotted line provides comparison to determine subsequent transmission.

approximation from a more complex mean evolution. That is, if $EY_t = g(t)$ where g is differentiable, then $EY_t \approx g(t_0) + g^{'}(t_0)(t - t_0)$ for t near t_0. Such local linearity supports the use of a linear suppression scheme. Below, we work with a stochastic differential equation to describe the evolution of Y_t. Using customary Euler discretization leads to local linearity in the conditional mean, again encouraging linear suppression. Lastly, all of the above applies if we replace t with X_t, as long as X_t is observed locally with Y_t.

4 Examples using a diffusion model

We turn to a simulated application of model-based suppression, using the second suppression algorithm described above. This example is motivated by a model that has been used to characterize soil moisture, as in Puggioni (2008) and references therein. We apply a suppression scheme tailored to the data generating model which arises from a stochastic differential equation, in fact, a simulated classic Orenstein-Uhlenbeck (O-U) process (see Uhlenbeck and Ornstein (1930)) with fixed parameters. We then demon-

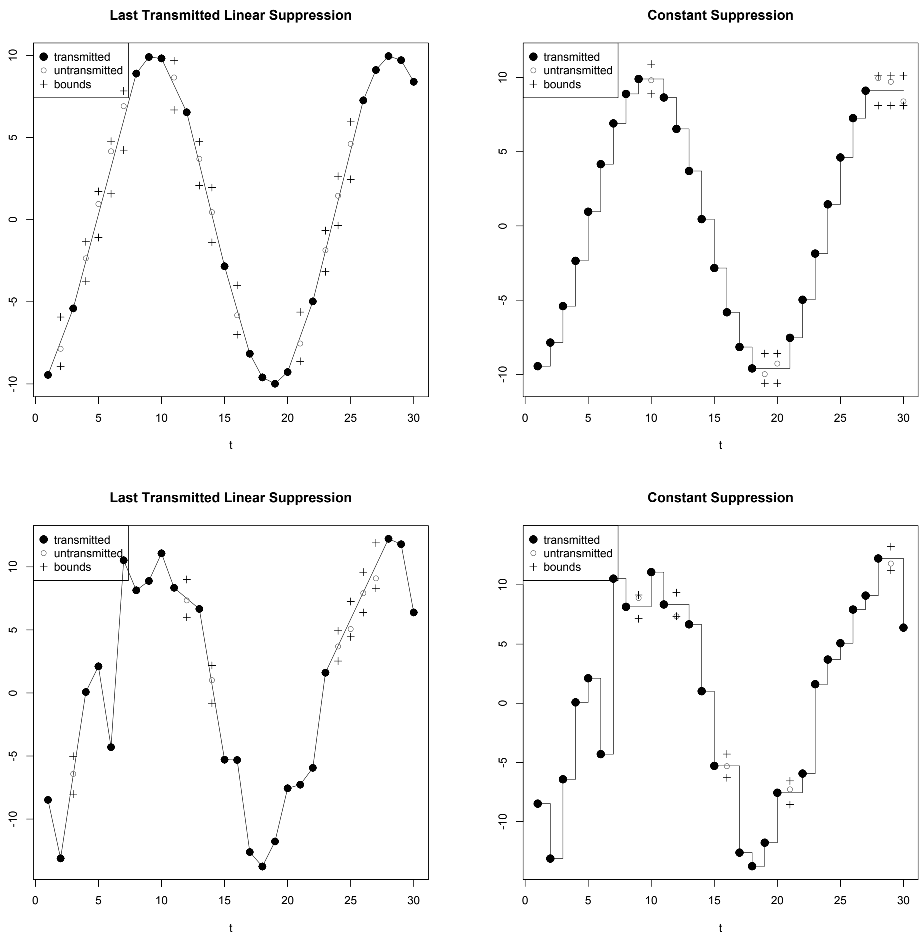

Figure 3: Data generated from $10\cos(t)$, for 30 values of t, and suppressed using both linear (left) and constant (right) suppression, both with $\epsilon = 1$. The suppression rate was $0.4\bar{3}$ for linear suppression and and 0.2 for constant suppression for the top "noise-less" row.

strate our approach with data generated from a logistic growth model, again a stochastic differential equation model, with dynamic parameters that themselves are an O-U processes.

We simulate an O-U process $dY_t = (\theta_1 + \theta_2 Y_t)dt + dW_t$ using the Euler discretization $Y_t - Y_{t-1} = (\theta_1 + \theta_2 X_{t-1})\Delta_t + \theta_3 N(0, \sqrt{\Delta_t})$ (see Iacus (2008), Elerian et al. (2001), and Eraker (1998)). We simulate the series on the process on the interval $[0, 50]$ at a resolution

100 times higher than the data that we use for the analysis (i.e., we sampled 5000 observations in the interval but viewed the series as sampled at t=1,2,...,50. The higher resolution is intended to provide a discrete sample that better resembles a continuous trajectory. Figure 4 shows both the original series and the series used for the analysis.

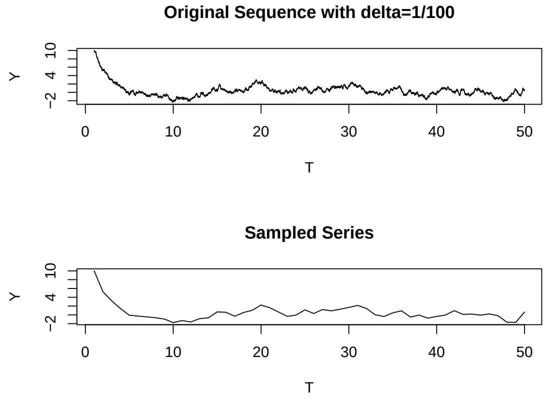

Figure 4: Simulated O-U process realization with parameters $\theta_1 = 0$, $\theta_2 = -.7$, $\theta_3 = 1$.

In order to suppress transmission from this series, we use a scheme tailored to the process. We assume $\theta_1 = 0$ for simplicity, though this can be relaxed. We then suppress according to the following scheme:

1. Transmit first two readings. ($t = 1$, $t' = 2$, $\hat{Y}_t = Y_1$, $\hat{Y}_{t'} = Y_2$, $i = t'$)

2. Calculate current estimate of θ_2 from these two values as $\hat{\theta}_2 = (Y_{t'} - Y_t)/(\Delta_t Y_t)$.

3. Forward simulate with 0 variance from $\hat{Y}_i$ to form predictions of the coming values. Thus the prediction for Y_{i+1} is $\hat{Y}_{i+1} = \hat{Y}_i + \hat{Y}_i \hat{\theta}_2 \Delta_t$.

4. If $|\hat{Y}_{i+1} - Y_{i+1}| > \epsilon$, transmit Y_{i+1}. Set $t = t'$, $t' = i + 1$, $i = 1$, $\hat{Y}_{i+1} = Y_{i+1}$ and go to (2). Else $i = i + 1$ and go to (3).

The dynamic parameter estimates are calculated very similarly to the linear suppression scheme, though prediction is done according to the SDE.

In order to infer about the model parameters, we use the method of Eraker (1998) to estimate the parameters. We use a Metropolis-Hastings step rather than the more complicated rejection sample hybrid Metropolis-Hastings for this simple univariate series. This method requires inserting several latent variables between each of the sampled time points, even in the case of a unsuppressed series. We include four latent variables between each integer time point, and we sample each of the suppressed time points within their known bounds. For further discussion on estimation of stochastic differential equations, see Elerian et al. (2001) or Durham and Gallant (2002).

In an illustrative simulation with parameters $\theta_1 = 0$, $\theta_2 = -.7$, $\theta_3 = 1$, we set ϵ such that the suppression rate was 38%. This resulted in posterior means and 95% credible intervals of $\hat{\theta_1} = .02$ with interval $[-.24, .29]$, $\hat{\theta_2} = -.54$ with interval $[-.72, -.38]$ $\hat{\theta_3} = .82$ with interval $[.77, 1.10]$. For each of the parameters, despite 38% suppression, the credible intervals always contain the true simulated parameters. We note that the maximum likelihood estimates from the full data set were $\theta_2 = -.57$, $\theta_3 = .87$.

The above suppression scheme updates θ_2 dynamically based on the last two transmissions. The data, however, were generated from a fixed θ_2. A more interesting case is when the model parameters are themselves dynamic. For this, we turn to a new model in which Y_t is a logistic stochastic differential equation process with carrying capacity K, which is fixed at one in this example, and time-varying rate parameter r_t governed by an O-U process:

$$\begin{aligned} dY_t &= r_t(1 - Y_t/K)Y_t dt + \sigma dW_t \\ dr_t &= (\theta_1 + \theta_2 r_t)dt + \theta_3 dW_t. \end{aligned}$$

The parameters of the logistic process are themselves governed by a stochastic differential equation, allowing for movement of Y_t according to the sign of r_t, as seen in Figure 5, which is an example realization from this process. A version of this model was considered recently in Duan et al. (2010).

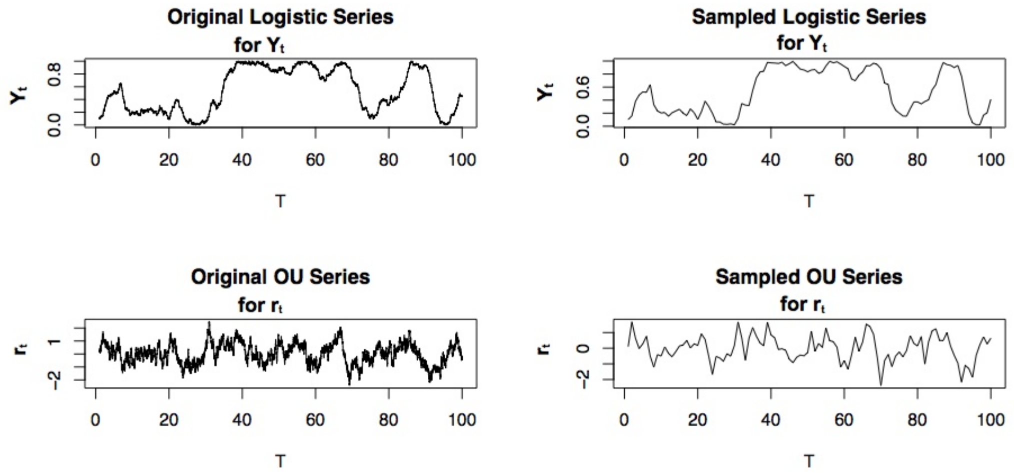

Figure 5: (left) Logistic series simulated at 100 times the resolution sensed by nodes. (right) Series sensed by the node. The parameters for this simulation were $\theta_1 = 0$, $\theta_2 = -.8$, $\theta_3 = 1$, $\sigma = .05$

In order to illustrate, we assume that both series, $\{Y_t\}$ and $\{r_t\}$ are sensed by the node, though each series is suppressed completely independently of the other. The O-U process in r_t is suppressed as above, and the logistic series is transmitted using a suppression scheme in accord with the logistic model. At each time point, we use the last two transmissions, t and t' to estimate $\hat{r} = \frac{Y_{t'} - Y_t}{(t'-t)Y_t(1-Y_t/K)}$. As only the unsuppressed values of the r_t and Y_t series are ever seen by the base station, we must infer both the missing values of each series and the model parameters governing both processes. Table

1 shows posterior mean estimates for each of the parameters averaged over ten series of length 100, suppressed at the rates indicated.

5 Duke Forest light availability data

The Duke Forest in Durham, North Carolina has deployed a sensor network to study various aspects related to the health of the forest. One of the variables collected is a reading of the amount of light each node senses during the course of a day (Clark et al. (2011)). We work with a dataset consisting of five nodes, measured across twelve days, with 72 measurements taken each day (20 minutes apart). After discussion with ecologists, the following model for light filtration through a canopy was proposed:

$$
\begin{aligned}
-log(\frac{I_{itd}}{I_{0td}}) &= F_{it} + G_d + \epsilon_{itd} \\
F_{it} &\sim N_+(\mu_i, \tau_i^2) \\
\mu_i &\sim N_+(\theta_\mu, \tau_\mu^2) \\
G_d &\sim N_+(\mu_G, \tau_G^2) \\
\epsilon_{itd} &\sim N_+(0, \sigma^2).
\end{aligned}
$$

In this model, I_{itd} is the reading of the ith node at time t of day d with I_{0td} the associated above-canopy reading. The F_{it} are intended to capture the local fluctuations in light at each time of day at each node. We also include G_d as a daily average to capture the relative cloudiness of each day. Lastly, we include i.i.d. errors, ϵ_{itd}. This simplified model reflects the ecologists' belief that the shadows which pass over the node throughout the day are highly local, so much so that incorporating information from neighboring nodes would likely not improve explanation. Thus, this model is completely non-spatial. It is essentially a random effects model, where each node has its own random effect for

			Suppression rate for r				
			.05	.25	.45	.65	.85
Suppression rate for Y	.05	$\theta_1 = 0$	-.022	-.023	-.015	-.024	-.001
	.25		-.022	-.016	-.013	-.018	0
	.45		-.019	-.018	-.015	-.026	-.004
	.65		-.019	-.015	-.011	-.032	0
	.85		-.022	-.020	-.020	-.032	-.005
	.05	$\theta_2 = -0.50$	-.497	-.418	-.512	-.506	-.613
	.25		-.514	-.503	-.513	-.519	-.583
	.45		-.495	-.527	-.492	-.511	-.558
	.65		-.483	-.494	-.495	-.512	-.656
	.85		-.559	-.607	-.609	- .624	-.942
	.05	$\theta_3 = 1$	.947	.957	.952	.942	.999
	.25		.956	.947	.944	.944	.973
	.45		.940	.961	.924	.931	.951
	.65		.918	.929	.916	.933	1.00
	.85		.962	.986	.969	.964	1.05
	.05	$\sigma = .05$	.044	.044	.044	.043	.043
	.25		.044	.044	.042	.043	.044
	.45		.045	.043	.045	.043	.041
	.65		.046	.045	.045	.047	.043
	.85		.078	.077	.077	.079	.071

Table 1: Posterior inference for the logistic/O-U model under varying suppression rates. (See text for details.)

each time of day. These are tied together in the hierarchy by μ_i, the node-level mean. Evidently, the first stage is loglinear in the ratio, $\log(\frac{I_{itd}}{I_{0td}})$. We assume that the above canopy reading is known without error because this can be read at the base station directly. Also, we know that if there is no light above the canopy, there must also be no light below. So, we assume that each of the zero readings are also known. Figure 6 shows an example of light readings from one series, the log ratio $\log(\frac{I_{itd}}{I_{0td}})$, and the suppressed data. To complete a Bayesian specification, we adopt flat priors for the mean parameters with inverse Gamma priors on each of the variance components/parameters.

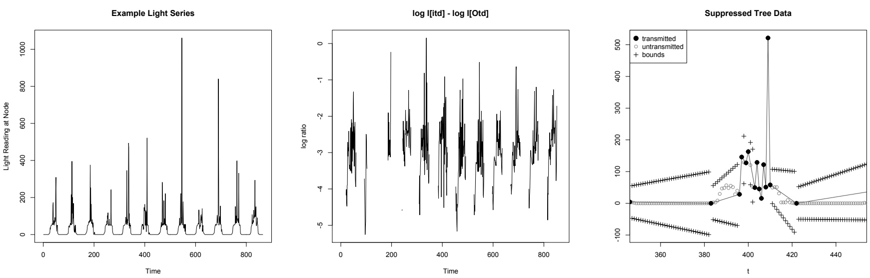

Figure 6: (left) Light readings from a selected node. (center) The log ratio, $\log(\frac{I_{itd}}{I_{0td}})$ for this node to the above canopy reading. (left) A portion of the suppressed series.

We fit the model using several suppression thresholds, resulting in suppression levels up to 62%. Table 2 shows the fitted sum of squared errors (observed-predicted) for each of the models. Note that, even in the case of highest suppression, there is only a 3% increase in the sum of squared errors. Figure 7 shows a comparison between each of the F_{it} fitted from the full data set to each of the suppressed data sets. We find that, in general, again we are able to recover approximately the same parameters, even from the most suppressed data set.

Threshold	Suppression Level	SSE
0	0	268
10	0.39	271
20	0.47	271
30	0.53	271
40	0.57	274
50	0.62	275

Table 2: The sum of squared error of the light model applied to each data sets with varying suppression levels.

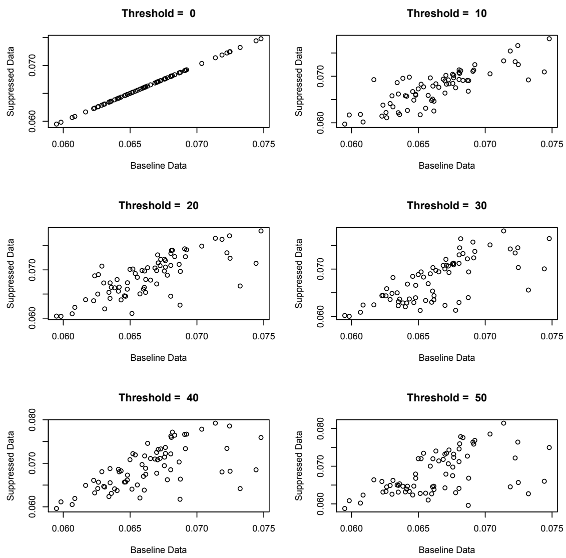

Figure 7: Each panel represents the average fitted value versus the actual data for a different suppression threshold applied to the light data.

6 Summary and future work

We have shown the benefits of working with locally linear suppression schemes. In particular, by allowing the data model to inform about the nature of trend and then adapting the suppression scheme accordingly, under fairly high levels of suppression, we can achieve comparable inference performance, in terms of parameter estimation and prediction, to using the full dataset. We have shown this in a simple simulation example, a more complicated diffusion model simulation example, and with a real dataset.

Future work in this area includes exploring suppression schemes incorporating spatial dependence. By incorporating both the temporal dependence as we have done in the suppression scheme presented here as well as spatial dependence with neighboring nodes, potentially less data might be transmitted. Cascaded suppression is a spatial suppression scheme in which nodes are clustered together depending on location and a "head node" is selected for each cluster. The head node then takes the temporally suppressed readings from each node in its cluster and decides which, if any, values to forward to the base station. In this way, the cluster head will forward representative readings from the cluster so that all readings not sent can be bounded in a way similar to purely temporal suppression. Another research avenue would examine suppression for time series that are non-Gaussian, or even discrete (e.g., binary or count data).

References

Clark, J., Agarwal, P., Bell, D., Ellis, C., Flikkema, P., Gelfand, A., Katul, G., Munagala, K., Puggioni, G., Silberstein, A., and Yang, J. "Getting What We Need from Wireless Sensor Networks: A Role for Inferential Ecosystem Models." (2011). (submitted).

Duan, J., Gelfand, A., and Sirmans, C. "Space-time Point Process Models Using Differential Equations with Application to Urban Development." *Bayesian Analysis*, 4(4):733–758 (2010).

Durham, G. and Gallant, R. "Numerical Techniques for Maximum-likelihood Estimation of Continuous-time Diffusion Processes." *Journal of Business and Economic Statistics*, 279–316 (2002).

Elerian, O., Chib, S., and Shephard, N. "Likelihood Inference for Discretely Observed Nonlinear Diffusions." *Econometrica*, 69(4):959–993 (2001).

Eraker, B. "MCMC Analysis of Diffusion Models with Application to Finance." *Journal of Business and Economic Statistics*, 19:177–191 (1998).

Iacus, S. M. *Simulation and Inference for Stochastic Differential Equations: With R Examples (Springer Series in Statistics)*. Springer, 1 edition (2008).

Juang, P., Oki, H., Wang, Y., Martinosi, M., Peh, L., and Rubenstein, D. "Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet." In *Proceedings of the 10th international Conference on Architectural Support for Programming Languages and Operating Systems*. San Jose, California (2002).

Puggioni, G. "Using Data Augmentation and Stochastic Differential Equations in Spatio Temporal Modeling." Ph.D. thesis, Duke University (2008).

Puggioni, G. and Gelfand, A. "Analyzing Space-time Sensor Network Data under

Suppression and Failure in Transmission." *Statistics and Computing*, 20(4):409–419 (2010).

Silberstein, A., Puggioni, G., Gelfand, A., Munagala, K., and Yang, J. "Suppression and Failures in Sensor Networks: A Bayesian Approach." *Proceedings of the 2007 International Conference on Very Large Data Bases (VLDB '07), Vienna, Austria*, 842–853 (2007).

Sun, J. *The Statistical Analysis of Interval-Censored Failure Time Data*. Springer, New York (2006).

Tolle, G., Culler, D., Dawson, T., Tu, K., Hong, W., Szewczyk, R., Polastre, J., Burgess, S., Turner, N., Gay, D., and Buonadonna, P. "A Macroscope in the Redwoods." In *Publication of the 3rd ACM Conference on Embedded Networked Sensor Systems*. SenSys 2005 (2005).

Uhlenbeck, G. E. and Ornstein, L. S. "On the Theory of the Brownian Motion." *Phys. Rev.*, 36(5):823–841 (1930).

Werner-Allen, G., Lorincz, K., Welsh, M., Marcillo, O., Johnson, J., Ruiz, M., and Lees, J. "Deploying a Wireless Sensor Network on an Active Volcano." In *EEE Internet Computing*, volume 10, 18–25 (2006).

West, M. and Harrison, J. *Bayesian Forecasting and Dynamic Models*. Springer Series in Statistics. Springer, 2 edition (1999).

Zhang, P., Sadler, C. M., Lyon, S. A., and Martonosi, M. "Hardware Design Experiences in ZebraNet." In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, SenSys '04, 227–238. New York, NY, USA: ACM (2004).