Suppressing temporal data in sensor networks using a scheme robust to aberrant readings

Ilka A. Reis[1,2], Gilberto Câmara[1], Renato Assunção[2] and Antônio Miguel V. Monteiro[1]

[1]National Institute for Space Research (INPE), Brazil
Av. dos Astronautas, 1758
12227-010 São José dos Campos – SP - Brazil
Phone: (55) 12 3945-6522 – Fax: (55) 12 3945-6468
E-mail: ilka@est.ufmg.br,{gilberto,miguel}@dpi.inpe.br

[2]Universidade Federal de Minas Gerais (UFMG), Brazil
Departamento de Estatística – ICEx - Campus Pampulha
31270-901 Belo Horizonte - MG - Brazil
Phone: (55) 31 3499 5920 - Fax: (55) 31 3499 5924
E-mail: assuncao@est.ufmg.br

***Abstract*—**The main goal of a data collection protocol for sensor networks is to keep the network's database updated while saving the nodes' energy as much as possible. To achieve this goal without continuous reporting, data suppression is a key strategy. The basic idea behind data suppression schemes is to send data to the base station only when the nodes' readings are different from what both nodes and base station expect. Data suppression schemes can be sensitive to aberrant readings, since these outlying observations mean a change in the expected behavior for the readings sequence. Transmitting these erroneous readings is a waste of energy. In this paper, we present a temporal suppression scheme that is robust to aberrant readings. We propose to use a technique to detect outliers from a time series. Since outliers can suggest a distribution change-point or an aberrant reading, our proposal classifies the detected outliers as aberrant readings or change-points using a post-monitoring window. This idea is the basis for a temporal suppression scheme named TS-SOUND (*Temporal Suppression by Statistical OUtlier Notice and Detection*). TS-SOUND detects outliers in the sequence of sensor readings and sends data to the base station only when a change-point is detected. Therefore, TS-SOUND filters aberrant readings and, even when this filter fails, TS-SOUND does not send the deviated reading to the base station. Experiments with real and simulated data have shown that TS-SOUND scheme is more robust to aberrant readings than other temporal suppression schemes proposed in the literature (value-based temporal suppression, PAQ and exponential regression). Furthermore, TS-SOUND has got suppression rates comparable or greater than the rates of the cited schemes, in addition to keeping the prediction errors at acceptable levels.

Keywords: temporal data suppression, outlier detection, erroneous readings, time series.

1. Introduction

Sensor networks are a powerful instrument for data collection, especially for applications like habitat and environmental monitoring. These applications often require continuous updates of the database at the network's root. However, sending continuous reports would quickly run out the limited energy of the nodes. A solution for continuous updating without continuous reporting is to use data suppression [1].

To define a data suppression scheme, nodes and base station have to agree on an expected behavior for the nodes' readings. Thus, if nodes' readings fit to the expected behavior, nodes suppress these data. Otherwise, when their sensed values do not fit to the expected behavior, nodes send reports to the base station. These reports are used to predict the suppressed data.

Suppression schemes are an alternative to improve the *reactivity* of a sensor network, which is defined as the ability of a network to react to its environment providing only relevant data [2]. Instead of changing the sampling rates according to the sampled values and sending all collect data to the base station as in [2], a suppression scheme collects data using a constant rate. However, it only sends data if they represent a deviation from the behavior agreed by nodes and base station.

Model-driven data suppression [3] defines the mean of a node's observations as their expected behavior and models this mean using temporal or spatio-temporal correlations.

A temporal data suppression scheme uses the correlation among the readings of a same node to build the expected behavior for the nodes' readings [4]. A spatio-temporal suppression scheme also considers the correlation among the observations of neighboring nodes [1].

Usually, suppression schemes define an absolute error measure to evaluate the deviation between sensed data and their expected behavior. This produces data collection schemes that are sensitive to aberrant readings. These outlying values can be the result of a temporarily malfunctioning of a particular sensor or due to some intervention on the environment on which the network is operating and it does not have any relation with the monitored variables. Sometimes, aberrant readings can be the

result of an expected change in the sensed values. For instance, solar radiation measurements often suffer the effect of temporary clouds. In this case, a reduction in the radiation values is expected and, perhaps, non-interesting to the network user.

Sensors measuring environmental variables can produce such erroneous or nonsense readings [5-10], particularly in outdoor applications [11, 12]. In monitoring networks with low energy constrains, such as the regular weather stations, the nodes transmit or record the aberrant readings, which are identified and deleted in the base station. However, for a sensor network, transmitting nonsense values means to waste valuable resources.

In this paper, we propose a temporal suppression scheme that is robust to aberrant readings. Our proposal is based on the detection of outliers and their posterior classification into change-points or aberrant readings. We consider the sequence of data collected by a node as observations of a temporal process. The probabilistic distribution of this process at each time period is used to infer about the expected behavior of the observations. An outlier is an observation that presents a small probability to belong to the distribution at the current time period. An outlier reading may suggest a change in the expected value for the time series or it may be an aberrant reading.

To detect outliers from a time series, we have adapted the proposal in [13]. We have inserted our version as part of a suppression scheme for data collection in sensor networks, the TS-SOUND scheme (*Temporal Suppression by Statistical OUtlier Notice and Detection*). After detecting an outlier, TS-SOUND classifies it into a change-point or an aberrant reading. In the former case, the node sends data to the base station. Otherwise, the node suppresses its data.

We have designed TS-SOUND for applications that are not interested in aberrant readings, since they represent a failure in data sensing or processing. Usually, these erroneous measurements occur at random, isolated or clustered. If they remain, this means malfunctioning and suggests a non reliable node.

TS-SOUND scheme adopts a procedure to avoid detecting an aberrant reading as a change-point. Furthermore, even if this misdetection occurs, TS-SOUND does not send the aberrant reading

to the base station.

In this paper, we claim and demonstrate that our proposed scheme for temporal suppression data is robust to aberrant readings. Furthermore, considering the trade-off between energy consumption and data quality, TS-SOUND has outperformed the model-based suppression schemes we have considered in this paper (PAQ [4] and exponential regression [1]) and also the simplest data suppression scheme, VB scheme [1]. The prediction error measures the quality of the data sent to the base station. Since the data transmission is the most important energy consumer, we use the suppression rates as a proxy for the energy consumption. To evaluate TS-SOUND scheme, we have run evaluation experiments with real and simulated data.

The remainder of this paper is organized as follows. Section 2 presents a TS-SOUND overview. In section 3, we describe the related work and the framework for suppression schemes proposed in [1]. Section 4 describes SDAR algorithm [13], which allows for the on-line estimation of time series parameters. In addition, it describes the procedure in [13] to detect outliers, how we have adapted it to be part of our proposed suppression scheme and how TS-SOUND deals with classifying the outliers into change-points or aberrant readings. In section 5, we present TS-SOUND protocol and frame it as a suppression scheme according to the proposal in [1]. Section 6 describes the evaluation experiments and section 7 presents their results using real and simulated data. Finally, section 8 discusses the experiments results and section 9 presents some future directions.

2. TS-SOUND overview

Techniques for outlier detection have been proposed in communities as Statistical Process Control (for example [14, 15]), Data Mining, Database and Machine Learning (for example [13, 16-18]).

In Statistical Process Control (SPC), for instance, the goal is to monitor a process initially “in-control” and raise an alarm when this process is considered to be “out-of-control” as soon as possible.

Often, the "in-control" state of the process is a predefined condition: nominal values for the monitored parameters and their tolerance bounds. To raise the alarm, SPC uses procedures to detect outliers.

For TS-SOUND, the "in-control" state is the probabilistic distribution of the monitored variable at the last time period. If the process is "in-control" during a time interval, the sensor readings follow the same probabilistic distribution along this interval and different values are caused by random fluctuation around an expected value. Then, we can suppress these readings. We consider the process is "out-of-control" if the expected value of this distribution changes. After the change, a new "in-control" state is defined. The change's relevancy is a user-defined parameter.

As in the SPC techniques, TS-SOUND uses the outlier occurrence to infer if the process is "out-of-control". To detect outliers, TS-SOUND adapts the technique in [13], which has been proposed to detect outliers from a time series. TS-SOUND employs an algorithm that considers the temporal dependence of the time series to update the parameters of the probability distribution at each new sensor reading (on-line estimation). This algorithm is called SDAR (*Sequentially Discounting Auto-Regressive*) [13]. SDAR combines the last parameters' updates with the new sensor reading to produce the new parameters' updates. SDAR uses a discounting factor to control the weight of the new sensor data in the updates' values. The outliers are detected as deviations from the data distribution.

In a time series, an outlier can suggest a distribution change-point or an aberrant reading. We can distinguish a change-point from an aberrant reading if we compare the time series values before and after the outlier, examining, for instance, the time series plot (Figure 1). The aberrant points appear as the "peaks" or "spikes" of the time series plot. The time series has similar behaviors before and after the occurrence of aberrant readings. On the other hand, after a change-point, the time series changes its behavior. Then, a data suppression scheme must update the database at the base station only when change-points occur.

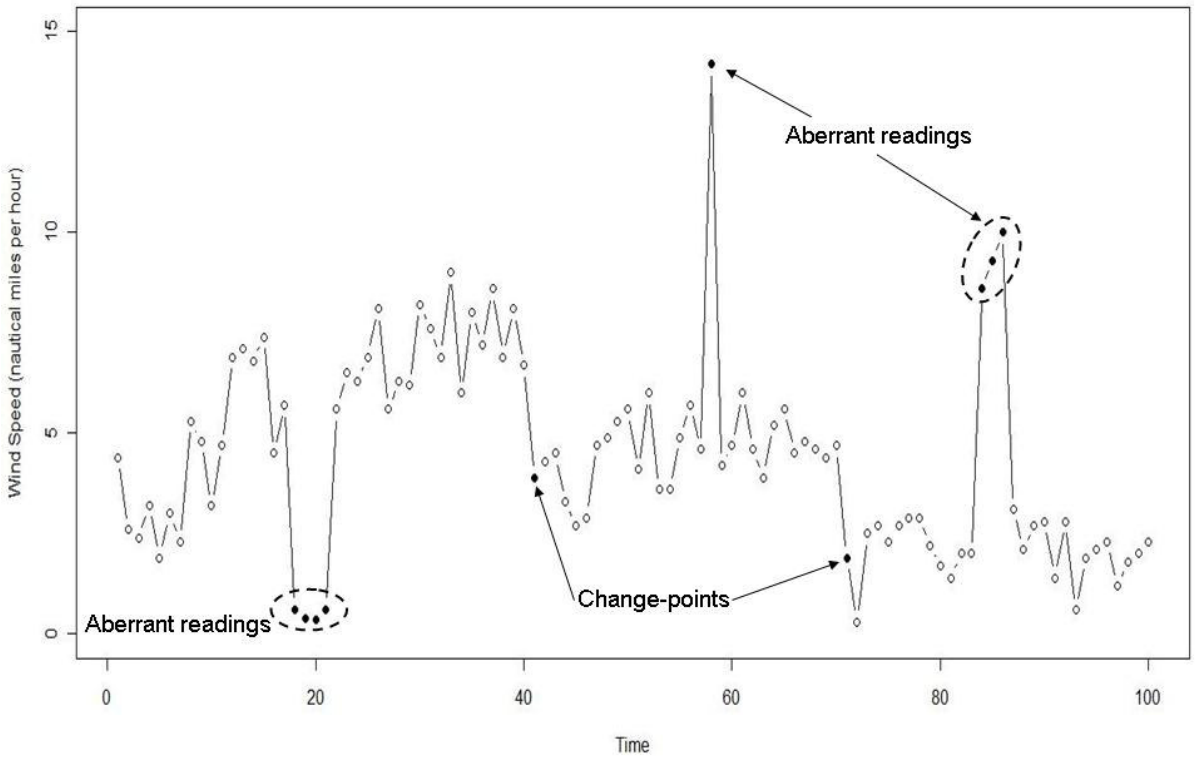


FIGURE 1. Outliers in a wind speed time series (Source: weather station of the University of Washington, USA, October 2006. We have inserted the aberrant readings to produce this figure).

To distinguish change-points from aberrant readings, TS-SOUND opens a post-monitoring window whenever it detects an outlier. During this time interval, the node keeps collecting data and updating the estimated distribution parameters. At the end of this time window, TS-SOUND compares the collected values with the distribution before and after the detected outlier. This outlier is classified as a change-point if the post-monitoring data are considered to be: 1) discrepant readings in relation to the distribution *before* the outlier; 2) non discrepant readings in relation to distribution *after* the outlier. If TS-SOUND classifies the detected outlier as a change-point, it summarizes the data collected during the post-monitoring and sends the result to the base station.

We have adopted a post-monitoring window for two reasons: a) to be able to distinguish change-points from aberrant readings. It avoids sending the latter ones to the base station; b) to allow for capturing the value of the new expected behavior through the summary of the collected values.

The base station uses the last sent data as an estimate for the node's readings until it receives a message with new data. Thus, for each node in the network, the base station stores a sequence of summaries and uses this time series as an estimate for the real node's time series. Section 5 describes TS-SOUND suppression scheme in detail.

3. Related Work

In this section, we describe the work related to our proposal considering two distinct topics: data suppression schemes for sensor networks and outliers detection in sensor networks.

Section 3.1 describes some proposals for temporal data suppression schemes and relates them to our proposal. In section 3.2, we describe a proposal a general framework for data suppression schemes [1]. This framework makes easier the comparisons among data suppression schemes. We use proposal in [1] to frame TS-SOUND as a data suppression scheme in section 5.4.

Since TS-SOUND uses outliers detection as the basis for its suppression scheme, section 3.3 provides a brief review of previous works on detecting outliers in a sensor network.

3.1 – Temporal Data Suppression Schemes

Recently, some protocols for data suppression in sensor networks have proposed to use statistical models to predict the nodes' data at the base station reducing the amount of communication inside the network. This approach to data suppression is called model-driven [3].

The main idea in [3] is to keep synchronized two probabilistic models: one at base station and other at the nodes. The model parameters are estimated in a learning phase. Based on these identical models, nodes and base station make the same predictions on the data to be collected. Then, the node collects the actual data and compares them to its prediction. If the difference between the real and predicted values is greater than a user-defined error bound, the node sends its data to the base station.

Otherwise, the node suppresses the collected data.

A similar idea appears in [4]. The PAQ protocol makes predictions based on a time series model, the third-order autoregressive model, AR(3). Given a time period *t*, the predicted value in *t* is written as a linear combination of the last three observations before *t*. PAQ uses two predefined error bounds to monitor the prediction error, defined as the absolute difference between the real and the predicted value. When the prediction error is greater than ε_υ, PAQ considers the observation as an outlier and sends it to the base station. If the prediction error is smaller than ε_υ but it is greater than ε_δ ($\varepsilon_\delta < \varepsilon_\upsilon$), PAQ opens a monitoring window. During the next A_{PAQ} time periods, the node goes on collecting data, predicting their values and monitoring outliers, sending these last ones to base station. At the end of monitoring window, PAQ counts how many observations have had prediction errors greater than ε_υ or greater than ε_δ but smaller than ε_υ. If this sum is greater than a threshold *a* ($a \leq A_{PAQ}$), PAQ decides to relearn the four model parameters. Then, PAQ calculates their new values and sends them to the base station. A variation of PAQ, called in [1] as exponential regression (EXP), uses the observation in the time period (*t*-1) in a simple linear regression to predict the observation in *t*. Thus, EXP has to estimate two model parameters.

It is worth to mention that, differently from TS-SOUND, neither PAQ nor EXP distinguishes a change-point from an aberrant reading. Once they detect an outlier reading, the node sends the observation to the base station, even if it is an aberration.

3.2 – A Framework for Data Suppression Schemes

According to Silberstein et al. [1], the nodes in the network are classified into *updaters* and *observers*. A *suppression link* describes the suppression/reporting relationship between an updater and its observer. The set of suppression links within the sensor network defines a *suppression scheme*.

In a simple suppression scheme, all the network nodes are updaters. These updaters collect data and decide to send them (or not) to the observer node, which is the base station. To produce a report r_t to its observer, the updater uses an *encoding function* f_{enc}. To decode the updater report, the observer

uses a *decoding function*.

The vector X_t represents the data of the updater node at time period *t* and the vector $\hat{X}_t$ represents the data as calculated by the observer node at same time period. The suppression link maintains X_t and $\hat{X}_t$ synchronized by evaluating a function $g(X_t, \hat{X}_t)$. The function *g* returns the logical TRUE value if $\hat{X}_t$ is within a user-defined error tolerance (ε) of X_t.

In Value-Based (VB) suppression scheme, for instance, the encoding and decoding functions are defined by (1) and (2), respectively,

$$f_{\text{enc}} = \begin{cases} x_t - x_{t'}, & \text{if } |x_t - x_{t'}| > \varepsilon_{VB} \\ \perp, & \text{otherwise} \end{cases} \quad (1)$$

$$\hat{x}_t = \begin{cases} \hat{x}_{(t-1)} + r_t & \text{if } r_t = x_t - x_{t'} \\ \hat{x}_{(t-1)}, & \text{if } r_t = \perp \end{cases} \quad (2)$$

where x_t is a component of the vector X_t, *t'* is the last time the updater sends a message to its observer and the symbol $\perp$ represents data suppression. The value $x_{t'}$ is what the observer knows about its updater at time period *t*. If the relative difference between the current updater value x_t and $x_{t'}$, the *g* function, is greater than error bound ε_{VB}, the updater produces a report $r_t = x_t - x_{t'}$ and sends it to the observer node. Otherwise, no message is sent ($r_t = \perp$). The observer computes its value $\hat{x}_t$ by adding the received report r_t to its old value $\hat{x}_{t-1}$. If the updater does not send a message, the observer updates $\hat{x}_t$ by repeating the old value.

PAQ and exponential regression have also been framed as temporal suppression schemes. Although PAQ also has a proposal for spatio-temporal suppression [4], we just consider its temporal version in this paper. The expressions in (3) and (4) reproduce the encoding functions of PAQ and EXP, respectively,

$$f_{\text{enc}} = \begin{cases} \alpha_t, \beta_t, \gamma_t, \eta_t & \text{if (modelRelearn)} \\ x_t & \text{if (outlier)} \\ \perp & \text{otherwise} \end{cases}, \quad (3)$$

$$f_{\text{enc}} = \begin{cases} \alpha_t, \beta_t & \text{if (modelRelearn)} \\ x_t & \text{if (outlier)} \\ \perp & \text{otherwise} \end{cases} . \quad (4)$$

In (3), α_t , β_t , γ_t and η_t are the coefficients of the AR(3) model adopted by PAQ scheme and, in (4), α_t and β_t are the coefficients of the simple linear regression model adopted by EXP scheme. The functions `modelRelearn` and `outlier` enclose the *g* function of PAQ and EXP schemes. As in VB scheme, it also evaluates the error between real and predicted values.

We classify our TS-SOUND proposal as a model-driven approach for temporal suppression [1]. TS-SOUND models the mean of the monitored variable and uses it to decide if an observation is an outlier of the current data distribution. However, the model runs only at the nodes, not at the base station, being not necessary to keep synchronized models as in the other model-driven proposals. We frame TS-SOUND approach as a temporal suppression scheme in section 5.4.

3.3 – Outliers detection in a sensor network

Recently, the problem of detecting outliers in a sensor network has gained importance [5] and generated works such as [6-10]. The proposal in [6] removes outlier readings from the data aggregation. Differently from TS-SOUND, the proposal in [6] makes the outliers available to the monitoring application. In [7], the authors detect outliers within a sliding window that holds the last W values of the sensor data. To estimate the data distribution, they use nonparametric models. As in [6], they report the outlier readings to the base station. However, this is done through a hierarchical structure, using the union of the outliers coming from multiple sensors. The authors in [8] propose a generic distributed algorithm that accommodates many nonparametric methods to detect outliers such as "distance to the k^{th} nearest neighbor" and "average distance to the *k* nearest neighbors". Nodes use one of these techniques to find out their local outliers and exchange information about them with their neighboring nodes to find out global outliers. In [9], the authors propose to use kernel density estimators to approximate the data distribution at each sensor node. As SDAR algorithm in [13], the kernel density estimation allows for adjusting itself to the input data distribution, as this distribution

changes overtime. The proposal in [9] assumes a heterogeneous sensor network, in which few sensor nodes are more powerful than the other sensors in the network. The detection of outliers is performed by these empowered nodes, which combine the models of two or more sensor nodes in this task. The authors discuss the trade-off among data accuracy, number of updates and the size of estimation models in some application scenarios. However, they do not provide evaluation experiments to show how this would work on real data. In [10], the authors propose to identify local outliers using temporal and spatial autocorrelations among nodes' values. Using the "distance" between its current value and its past values, a node is able to identify a potential (temporal) outlier comparing the "distance" with a learned distance threshold. If a potential outlier is detected, the node uses the distance threshold of its neighbors to finally classify its current value as an outlier (or not). The "distance" measurement can be done using several types of functions. The authors in [10] also propose to classify the detected outliers into error or events, which could be equivalent to what we call aberrant readings and change-points, respectively. If a node observes an outlier due to an event, the authors argue that the most of node's neighborhood should also detect outliers, since the value of neighboring nodes are spatially correlated. Then, summarizing the idea in [10], if a node classifies its current value as an outlier and most of its neighboring nodes also classify their current values as outliers, the node's value is considered to be an event.

Differently from the proposals described above, our proposal to detect outliers does not require communication among sensor nodes, since we have treated only the temporal aspect of the data suppression in this paper. Moreover, except by the proposal in [10], the described proposal are not concerned in classifying the detected outlier into aberrant readings or change-points. However, some described proposals can be an interesting basis for a future spatio-temporal version of TS-SOUND scheme.

An extensive survey of outliers detection techniques for sensor networks is not our aim. For a comprehensive overview on this subject, we refer to the work in [5].

4. Detecting outliers from a time series

In this section, we present the procedure in [13] to detect outliers from a time series and our proposal for adapting it to the constrained environment of a sensor network.

We consider the sequence of the data sensed by a sensor node, $\{X_t, t=1,2,3\ldots\}$, as a time series.

The autoregressive (AR) model is the simplest model to represent the statistical behavior of a time series. In AR(k), the autoregressive model of order k, the observation at time t, X_t, is written as a combination of the last k past observations as following

$$X_t = \mu + \rho_1(X_{t-1} - \mu) + \rho_2(X_{t-2} - \mu) + \ldots + \rho_k(X_{t-k} - \mu) + \varepsilon_t, \quad k=1,2,3,\ldots,t\text{-}1 \tag{5}$$

where μ is the mean of X_t, ρ_k is the autocorrelation of order k and ε_t is a noise term following a Gaussian distribution with zero mean and variance σ_ε^2.

If k=1, for example, we have the AR(1) model and the probability density function of X_t, given X_{t-1}, is

$$p_t\left(X_t \mid X_{t-1}; \theta^t\right) = \frac{1}{\sigma^t\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{X_t - w^t}{\sigma^t}\right)^2\right], \tag{6}$$

where $w^t = \mu^t + \rho_1^t\,(X_{t-1} - \mu^t)$ is the prediction for X_t using the AR(1) model, $\rho_1^t = C_1^t / C_0^t$ is the autocorrelation between X_t and X_{t-1}, $\left(\sigma^t\right)^2 = C_0^t - \rho_1^t C_1^t$, C_0^t is the variance of X_t, C_1^t is the covariance between X_t and X_{t-1} and $\theta^t = \left(\mu^t, \rho_1^t, \sigma^t\right)$ is the parameters vector. In other words, $[X_t|X_{t-1}]$ follows the Gaussian distribution with mean w^t and variance $\left(\sigma^t\right)^2$.

If k>1, the parameters' updating in SDAR algorithm involves matrices. Then, to simplify the calculations in the sensor nodes, we have adopted the AR(1) model. From now on, we use this model to present the approach in [13].

4.1 – The Yamanishi and Takeuchi's proposal to detect outliers

Yamanishi and Takeuchi [13] adopted the AR model to represent the time series.

To estimate the parameters in θ^t and, as a result, the value for $p_t\left(X_t \mid X_{t-1};\theta^t\right)$, the authors in [13] proposed the *Sequentially Discounting AR* (SDAR) algorithm. The goal of SDAR is to learn of the AR model and provide the *on-line* estimation of θ, which is updated at each new observation X_t. A *discounting factor r* controls the weight given to the new observation X_t in the estimation of θ.

SDAR has two main steps: initialization and parameters updating. In the first step, SDAR sets $\mu^0, C_0^0, C_1^0\rho_1^0$ and σ^0, which are the initial values for $\mu^t, C_0^t, C_1^t\rho_1^t$ and σ^t, respectively. The initial values for μ^t, C_0^t and C_1^t can be defined by the user or calculated using a learning sample.

The second step of SDAR is parameters updating. At each time *t*, the node collects a new observation X_t and, for a given value of *r*, $0 \le r \le 1$, the parameters are updated as following:

$$\hat{\mu}^t = (1-r)\hat{\mu}^{t-1} + r X_t \ , \tag{7}$$

$$\hat{C}_j^t = (1-r)\hat{C}_j^{t-1} + r(X_t - \hat{\mu}^t)(X_{t-j} - \hat{\mu}^{t-j}) \ , j = 0,1. \ , \tag{8}$$

$$\hat{\rho}_1^t = \hat{C}_1^t \Big/ \hat{C}_0^t \ , \tag{9}$$

$$\hat{w}^t = \hat{\rho}_1^t(X_{t-1} - \hat{\mu}^t) + \hat{\mu}^t \ , \tag{10}$$

$$\left(\hat{\sigma}^t\right)^2 = (1-r)\left(\hat{\sigma}^{t-1}\right)^2 + r\left(X_t - \hat{w}^t\right)^2 \ . \tag{11}$$

The discounting factor *r* enables SDAR to deal with nonstationary time series.

Since SDAR updates the parameters at each time *t*, it produces a sequence of probability densities $\{p_t, t=1,2,3\ldots\}$, where p_t is the probability density function in (6) specified by the parameters updated by the SDAR algorithm at time *t*.

To detect outliers, the authors in [13] have proposed to evaluate each observation X_t using the sequence $\{p_t, t=1,2,3\ldots\}$ and the score function

$$score(X_t) = -\ln\left[p_{t-1}(X_t)\right] = \frac{1}{2}\left(\frac{X_t - w^{t-1}}{\sigma^{t-1}}\right)^2 - \ln\left[\frac{1}{\sigma^{t-1}\sqrt{2\pi}}\right] \quad (12)$$

Intuitively, this score measures how large the probability density function p_t has moved from p_{t-1} after learning from X_t. A high value for $score(X_t)$ indicates X_t is an outlier with a high probability.

To detect change-points, the authors in [13] have proposed to use the average of the T, T > 1, last values of $score(X_t)$ to construct a time series Y_t. SDAR algorithm is applied on Y_t to construct a sequence of probability densities q_t and $score(Y_t) = -\ln\left[q_{t-1}(Y_t)\right]$ is calculated. Then, they define a function $Score(t)$, which the average of the T' last values of $score(Y_t)$, T'>1, and use $score(Y_t)$ to detect change-points in the time series.

It is worth to note there are many calculations involved in Yamanishi and Takeuchi's proposal [13]. Moreover, they have not made clear how to distinguish aberrant readings from change-points.

4.2 – The outlier detection in the TS-SOUND scheme

TS-SOUND scheme uses the detection of outliers to decide whether a node must suppress its data or it must not. If an outlier is detected, the node opens a post-monitoring window to decide if the outlier is a change-point or an aberrant reading. In the first case, the node sends data to the base station.

The authors in [13] did not considered power limitations in the calculations. Therefore, using a logarithm operator in $score(X_t)$ was not a concern. However, in the constrained environment of a sensor node, using the logarithm function can be a costly operation. Then, to meet the requirements of a scheme for data collection in sensor networks, we have simplified the definition of $score(X_t)$ by evaluating the distance between X_t and w^{t-1} using the function

$$SD_{t-1}(X_t) = \frac{\left|X_t - w^{t-1}\right|}{\hat{\sigma}^{t-1}} \quad , \quad (13)$$

where $\hat{\sigma}^t$ represents the estimate for the standard deviation of X_t.

Note that $SD_{t-1}(X_t)$ is the absolute value of a normalized score. In fact, we can see $SD_{t-1}(X_t)$ as part of the G statistic[1] proposed in [19] to detect outliers in a static dataset. As the original $score(X_t)$ in (12), $SD_{t-1}(X_t)$ evaluates how far X_t is from w^{t-1}, which is the prediction for X_t using the AR(1) model in *t-1*. Then, a high value for $SD_{t-1}(X_t)$ also indicates X_t is an outlier of the distribution in *t-1* with a high probability.

As in [13], we evaluate the $SD_{t-1}(X_t)$ function over a time window composed by the T past time periods, where $T \geq 1$. However, instead of using a T-averaged score, we simplify the calculations and use the sum of the T past values of $SD_{t-1}(X_t)$. Then, at each time period *t*, we calculate the score Z_t as

$$Z_t = \sum_{i=t-T+1}^{t} SD_{i-1}(X_i) = \sum_{i=t-T+1}^{t} \frac{\left| X_i - w^{i-1} \right|}{\hat{\sigma}^{i-1}} \tag{14}$$

The expression for Z_t compares the values of $\{X_i, i=t\text{-}T+1, \ldots, t\}$ with w^{i-1}, which is the predicted value for them if they come from the *p* distribution in *t=i-1*. Large differences indicate the values of $\{X_i, i=t\text{-}T+1, \ldots, t\}$ have a small probability to belong to the *p* distribution in *t=i-1*. The sum over the T past time periods in Z_t allows for capturing smooth changes in the average of the time series.

If the value of Z_t is greater than a pre-defined threshold, X_t is considered to be an outlier. However, X_t can be an aberrant reading or a change-point. To decide this, TS-SOUND scheme opens a post-monitoring window.

[1] The G statistics is defined as the maximum of the absolute value of the normalized scores of observations in a static dataset.

4.2.1 – The threshold for Z_t

Besides simplifying the calculations of Z_t, the scoring function $SD_{t-1}(X_t)$ makes the definition of a threshold for Z_t more intuitive than choosing a threshold to the original *Score*(X_t) in [13]. We have used the theory of statistical significance tests [20] to help us with this choice.

At each time period *t*, we can see the classification of X_t as an outlier of the *p* distribution in *t*-1 as a significance test of the following hypothesis

H_0: the expected value for X_t is w^{t-1} (X_t is not an outlier) versus

H_1: the expected value for X_t is not w^{t-1} (X_t is an outlier).

At a significance level of α, $0 < \alpha < 1$, the null hypothesis H_0 is rejected if $\left|Z_{test}^{t}\right| > z_{\alpha/2}$, where $Z_{test}^{t} = \frac{X_t - w^{t-1}}{\hat{\sigma}^{t-1}}$ is a normalized score and $z_{\alpha/2}$ is the percentile 100(1-α/2) of the standard Gaussian distribution (average and standard deviation equal to 0 and 1, respectively). Here, we assume the estimates for w^{t-1} and $\hat{\sigma}^{t-1}$ carry enough information from the past data to approximate the distribution of Z_{test} by a standard Gaussian distribution.

Since Z_t is the sum of $\left|Z_{test}^{i}\right|$, *i*=*t*-T+1, …, *t*, one can use the Gaussian model with average equals to zero and standard deviation equals to $\sqrt{T}$ to *guide* the choice of the values for z_T^{α}, the threshold for Z_t. For instance, if T = 2 and the significance levels α = (0.20, 0.10, 0.05, 0.025, 0.01), the values for z_T^{α} would be 1.81, 2.32, 2.77, 3.17 and 3.64, respectively. These are the values of the percentiles 100(1-α/2) of a Gaussian distribution with mean and standard deviation equal to 0 and $\sqrt{2}$, respectively.

The value of z_T^{α} depends on two user-defined parameters: the size of the risk of making a mistake when the scheme classifies X_t as an outlier (α) and how much of the past observations should be considered in this classification (T). For a fixed value of T, the smaller the value of α, the more

rigorous the criterion to consider X_t as an outlier of the distribution in *t-1*. Then, decreasing the value of α increases the value of z_T^{α} and, as a result, the data suppression rate increases.

For a fixed value of α, the greater the value of T is, the greater the delay to detect an outlier. On the other hand, increasing the value of T allows for capturing smooth changes in the expected value for the time series. The relevance of the change is a user-defined parameter and also has to do with the value for α: if α is large, the scheme will be able to detect small changes, since the outlier alarm will rise more often.

In our experiments, we have evaluated the values α = (0.25, 0.20, 0.15, 0.10, 0.05, 0.025, 0.01) and T = (2, 4, 6, 8, 10). We discuss these values using a simple case study in section 7.1.

4.2.2 – Detecting change-points

After detecting an outlier at time period *t*, TS-SOUND has to classify it as a change-point or an aberrant reading. To make this decision, the node has to study the time series behavior before and after *t*. Then, if TS-SOUND detects an outlier, it opens a post-monitoring window of size T. From *t*+1 to *t*+T, the node collects data and updates the AR(1) parameters. At the end of post-monitoring window, the node compares the T observations collected during the time window with the *p* distribution *before* and *after* the detected outlier.

As we discussed at section 2, the outlier detected at time period *t* is considered to be a change-point if the observations within the monitoring window are considered to be outliers of the *p* distribution *before t* and non-outliers of the *p* distribution *after t*. In Figure 1, we can visualize the reason for this rule.

To make the "before-comparison", we use the function Z_{t+T}^{B} defined as following

$$Z_{t+T}^{B} = \sum_{i=t+1}^{t+T} \frac{\left| X_i - w^{(i-1)-T} \right|}{\hat{\sigma}^{(i-1)-T}}. \qquad (15)$$

Note that Z^{B}_{t+T} uses the estimates for the AR(1) parameters of time periods from *t*-T to *t*-1, that is, the last T estimates *before* the detected outlier.

The "after-comparison" is made using the function Z^{A}_{t+T} defined as

$$Z^{A}_{t+T} = \sum_{i=t+1}^{t+T} \frac{\left| X_i - w^t \right|}{\hat{\sigma}^t} \quad . \tag{16}$$

The expression for Z^{A}_{t+T} uses the estimates for the AR(1) parameters calculated when the outlier was detected, at time period *t*.

Then, X_t is considered to be a change-point if $Z^{B}_{t+T} \geq Z_T^{c \cdot \alpha}$ and $Z^{A}_{t+T} \leq Z_T^{c \cdot \alpha}$, where $0 < c \leq 1$. If $c < 1$, the rigor to consider the observations after *t* as outliers is greater than the rigor used to detected the outlier in *t*. Actually, we propose to keep the same rigor level for the "before-comparison" (c=1) and increase the rigor for the "after-comparison" (e.g., c=0.05 or c=0.10). This strategy takes into account the values produced immediately after a change-point are possibly accommodating themselves around the new expected value. This can produce values for Z^{A}_{t+T} larger than they should be if a longer time period had been observed. This would lead to the wrong classification of a change-point as an aberrant reading. Then, increasing the rigor in the "after-comparison" decreases the probability of making this mistake. In other words, the smaller the value of *c*, the smaller the probability of mistaking a change-point for an aberrant reading. In our evaluation experiments, we have set c=(0.75, 0.50, 0.25, 0.10, 0.05, 0.01) in the threshold for Z^{A}_{t+T}. In most of the experiments, the value c=0.05 has got the best results.

If the detected outlier is considered to be a change-point, the node updates the database at the base station sending a summary of the observations collected during the post-monitoring window. We have adopted the median to calculate this summary, since the median is more robust to aberrant readings than the average, for instance. This property of the median can be especially useful if TS-SOUND mistakes the beginning of sequence of aberrant readings for a change-point. In this case, the node will send the summary to the base station unnecessarily, which will degrade the suppression rate.

However, the median will suffer less influence of these erroneous readings, especially if the length of the monitoring window is larger than the size of the aberrant sequence. Then, at least the prediction error at base station will be preserved.

It is worth to mention that the length of the post-monitoring window (T) could be different from the number of past observations used in SDAR parameters estimation and in Z_t statistics. However, in our additional experiments to evaluate this possibility, TS-SOUND has got the best results when both time windows have had the same length.

4.3 – Other proposals to detect outliers in a time series

There are other proposals for outliers detection in time series (for example [7, 14, 16, 17, 19] and those described in [18]). However, we have considered the proposal in [13] as the best one to be adapted to a scheme of data suppression in sensor networks. The reasons for this choice have been the following: a) the proposal in [13] considers the temporal autocorrelation of sensor data by adopting a times series model; b) it is adaptative to nonstationary data sources; c) it allows for on-line detection of outliers and d) the calculations can be made simpler.

5. TS-SOUND scheme

The TS-SOUND scheme has two phases: learning and operation. In the learning phase, TS-SOUND estimates the initial values for the SDAR parameters and the first two values for Z_t.

5.1 - Learning phase

Before beginning its operation, the node collects values during a short time window, say, N_{ini} time periods. The values for the initial values μ^0, C_0^0, C_1^0 are calculated as following

$$\mu^{0} = \frac{\sum_{t=1}^{N_{ini}} X_t}{N_{ini}}, \quad C_0^0 = \frac{\sum_{t=1}^{N_{ini}} \left(X_t - \mu^{0}\right)^2}{N_{ini} - 1}, \quad C_1^0 = \frac{\sum_{t=1}^{N_{ini}} \left(X_t - \mu^{0}\right)\left(X_{t-1} - \mu^{0}\right)}{N_{ini} - 1} \; . \tag{17}$$

To calculate the first value for Z_t, the node needs T additional observations. Then, the size of learning sample is $N_{learn} = N_{ini} + T$. Figure 2 presents the pseudo-code for the algorithm running in the learning phase.

```
learning()

Input     r, T, N_ini
Output    initial values for SDAR parameters, μ̂^j, Ĉ_0^j, Ĉ_1^j, ρ̂_1^j, σ̂^2 ^j, and Z_t.
1)  j=1
2)  every t_s time units while j ≤ N_ini do
3)        read x_j;
4)        enqueue X = X ∪ x_j ;
5)        j= j +1 .
6)  calculate OUT_UPPER , OUT_LOWER .
7)  from j=1 to j=N_ini do
8)        if ( OUT_LOWER < X_j < OUT_UPPER ) enqueue X_noOut = X_noOut ∪ X_j .
9)  calculate μ^0, C_0^0, C_1^0, ρ_1^0, and σ^2 ^0 using X_noOut .
10) j = N_ini + 1
11) read x_j;
12) enqueue X = X ∪ x_j ;
13) send x_j;
14) calculate the SDAR parameters μ̂^j, Ĉ_0^j, Ĉ_1^j, ρ̂_1^j, σ̂^2 ^j;
15) j = j+1;
16) every t_s time units while j ≤ N_ini + T do
17)       read x_j;
18)       enqueue X = X ∪ x_j ;
19)       calculate and store the SDAR parameters μ̂^j, Ĉ_0^j, Ĉ_1^j, ρ̂_1^j, σ̂^2 ^j;
20)       j = j+1;
21) calculate the first value of Z_t
22) return  μ̂^j, Ĉ_0^j, Ĉ_1^j, ρ̂_1^j, σ̂^2 ^j  and Z_t.
```

FIGURE 2 - Pseudo-code for the learning phase algorithm.

Until completing N_{ini} observations, the node collects and stores data every t_s time units, which is the user set sampling rate (lines 1-5).

Discrepant values can affect the estimative for the initial values. Then, the learning algorithm filters these outliers before calculating the initial values. The outliers limits (OUT_{UPPER} and OUT_{LOWER}) are calculated according to the rules for building boxplots [21]. First, we calculate P_{25} and P_{75}, which are the 25^{th} and the 75^{th} percentiles of the observations, respectively. To calculate the percentiles, the algorithm has to sort the data, which can be done during the values storage. The difference $IQ=(P_{75}-P_{25})$ is called *interquartile range*. The upper and lower limits are defined as $OUT_{UPPER} = (P_{75} + 1.5\ IQ)$ and $OUT_{LOWER} = (P_{25} - 1.5\ IQ)$. Values outside these limits are considered to be outliers.

After removing the possible outliers (lines 7-8), the algorithm calculates the initial values for SDAR parameters (line 9).

To update the initial values, the node samples T additional observations and sends the first of them to the base station (line 10-13). The SDAR algorithm updates its parameters according to the expressions from (7) to (11) and stores the results (lines 14-15). The node collects the remaining (T-1) values and runs the SDAR algorithm (lines 16-20). Then, the node calculates the first value for Z, Z_T, using the expression in (14).

The learning algorithm returns SDAR parameters and the first value of Z_t.

5.2 – The operation phase

After the learning phase, the node has all the parameters it needs to start the operation phase: the user-set values (r, α and T), the SDAR parameters and the first value for Z_t, $t = N_{ini} + T$. Figures 3A and 3B presents the pseudo-code for TS-SOUND operation phase and post-monitoring algorithm, respectively.

The operation phase continues while the node's battery has a noncritical level of energy, which is evaluated by the function

$$\texttt{energy.OK} = \begin{cases} 1, & \text{if the battery's level is noncritical} \\ 0, & \text{otherwise} \end{cases} . \qquad (18)$$

```
TS-SOUND operation.phase()
Input      r, T, c, Z_T^α , Z_T^{c·α} , μ̂^t, Ĉ_0^t, Ĉ_1^t, ρ̂_1^t, σ̂^{2 t} , Z_t .
Output     values sent to the base station
1)  t = (N_ini + T)+1 ;                    # time counter
2)  every t_s time units while (energy.OK = 1) do
3)      read x_t ;
4)      enqueue X = X ∪ x_t ;
5)      keep the last T values of X ;
6)      calculate and store SDAR parameters μ̂^t, Ĉ_0^t, Ĉ_1^t, ρ̂_1^t, σ̂^{2 t} ;
7)      keep the last (T+1) values of μ̂^t, Ĉ_0^t, Ĉ_1^t, ρ̂_1^t, σ̂^{2 t} ;
8)      calculate Z_t ;
9)      if (Z_t > Z_T^α ) do                # if an outlier is detected...
10)         run monitoring.window() ;       # ... it opens the monitoring window.
11)         calculate Z_{t+T}^B and Z_{t+T}^A ;
12)         if Z_{t+T}^B ≥ Z_T^α and Z_{t+T}^A ≤ Z_T^{c·α} do
13)             calculate x̃ = median[ X_{t+1} ... X_{t+T} ] ;
14)             send x̃ .
15)         else
16)             do [μ̂^j, Ĉ_0^j, Ĉ_1^j, ρ̂_1^j, σ̂^{2 j}]_{j={t,t+1,...,t+T}} = μ̂^{t-1}, Ĉ_0^{t-1}, Ĉ_1^{t-1}, ρ̂_1^{t-1}, σ̂^{2 t-1} .
17)     t = t + 1.
18) send ( x_t , end.flag ) .              # End of node's operation
```

FIGURE 3A – Pseudo-code for the TS-SOUND operation phase algorithm

The node reads the sensed value, stores only the last T sensed values (lines 3-5), runs the SDAR algorithm and stores the T+1 last values of the distribution parameters (lines 6-7), and calculates the value of Z_t (line 8).

If the suppression scheme considers that X_t has a small probability to be generated by the current distribution ($Z_t > Z_T^{\alpha}$), TS-SOUND opens a monitoring window of size T (lines 9-10). During this time interval (Figure 3B), the node collects data, updates the SDAR parameters and keep their (2T+1)

last values. After closing the monitoring window, the node calculates Z^B_{t+T} and Z^A_{t+T} (line 11) and compares their values with their respective thresholds (line 12). If the outlier detected at time period *t* is considered to be a change-point, the node summarizes the values collected inside the post-monitoring window using the median and sends this summary to the base station (lines 13-14). Otherwise, since the detected outlier is classified as an aberrant reading, the updates for the SDAR parameters calculated during the monitoring window are replaced by the updates at *t-1*, the time period before the occurrence of the detected outlier (lines 15-16). This procedure avoids the bad effect of aberrant readings on the estimation of the distribution parameters.

monitoring.window()

Input *r*, T, *the last (T+1) values of* $\hat{\mu}^t, \hat{C}_0^t, \hat{C}_1^t, \hat{\rho}_1^t, \hat{\sigma}^{2\,t}$.

Output X , *the last (2T+1) values of* $\hat{\mu}^t, \hat{C}_0^t, \hat{C}_1^t, \hat{\rho}_1^t, \hat{\sigma}^{2\,t}$

```
1)  j = 1;
2)  every t_s time units while (j <= T) do              # monitoring window
3)      t = t + j ;
4)      read x_t ;
5)      enqueue X = X ∪ x_t ;
6)      keep the last T values of X ;
7)      calculate and store μ̂^t, Ĉ_0^t, Ĉ_1^t, ρ̂_1^t, σ̂^2 t
8)      keep the last (2T+1) values of μ̂^t, Ĉ_0^t, Ĉ_1^t, ρ̂_1^t, σ̂^2 t ;
9)      j = j+1 .
10) return X and the last (2T+1) values of μ̂^t, Ĉ_0^t, Ĉ_1^t, ρ̂_1^t, σ̂^2 t .
```

FIGURE 3B – Pseudo-code for the post-monitoring window algorithm

When the node is running out of energy (`energy.OK=0`), the algorithm transmits the last sensed value and an end flag.

Opening a time window after the outlier detection introduces a delay of T time periods in the base station updating. However, we have three reasons to adopt this post-monitoring window. First, it allows for comparing the time series before and after the detected outlier. Second, it allows for

summarizing the values generated by the new distribution. This summary estimates better the next data to be suppressed than the value that was responsible by the alarm raising. Third, it avoids sending the observation detected as an outlier to the base station, since TS-SOUND may mistake an aberrant point for a change-point.

5.3 – Costs

At the end of the learning phase, the node stores N_{ini} values. After that, at each time period *t*, the node has to store the last T updates for the SDAR parameters (5T values) and the last T sensed values. Besides, the node has to store five user-set parameters. Four of them are permanent (*r*, Z_T^{α}, $Z_T^{c\cdot\alpha}$ and T). The size of the learning sample (N_{ini}) can be deleted after the learning phase, as well as the learning sample. During a monitoring window, the node has to store the last (2T +1) values of the SDAR parameters, that is, 5(2T +1) values. Then, during the operation phase, the node has to store (6T+4) values outside the monitoring window and (10T+9) values during the monitoring window.

TS-SOUND operation phase involves mainly simple calculations, as additions and multiplications. The most costly operation is the square-root in the expression $\hat{\sigma}^t = \sqrt{\hat{\sigma}^{2\,t}}$.

The message sent to the base station contains only the median of the data collected during the post-monitoring window.

5.4 - TS-SOUND as a suppression scheme

In this section, we frame the TS-SOUND protocol as suppression scheme according to framework proposed in [1]. At each time period *t*, the node collects data x_t, updates the SDAR parameters, calculates Z_t and evaluates the function `Z.fcn`, defined as following

$$\texttt{Z.fcn} = \begin{cases} 1, & \text{if } Z_t > Z_T^{\alpha} \\ 0, & \text{otherwise} \end{cases} . \qquad (19)$$

As in PAQ and EXP schemes, `Z.fcn` evaluates the error between real and predicted values. However, in TS-SOUND case, the calculations of the predicted values are based on a time series model updated at each new sensor reading.

If `Z.fcn` = 1, the nodes opens a monitoring window and, for T time periods, sense and store the data. At time period t+T, the node evaluate two functions, `Zb.fcn` and `Za.fcn`, defined as following

$$\texttt{Zb.fcn} = \begin{cases} 1, & \text{if } Z^{B}_{t+T} \geq z^{\alpha}_{T} \\ 0, & \text{otherwise} \end{cases} \quad \text{and} \quad \texttt{Za.fcn} = \begin{cases} 1, & \text{if } Z^{A}_{t+T} \leq z^{c\cdot\alpha}_{T} \\ 0, & \text{otherwise} \end{cases}, \tag{20}$$

where Z^{B}_{t+T} and Z^{A}_{t+T} are defined by the expressions (15) in and (16), respectively. The functions `Z.fcn`, `Zb.fcn` and `Za.fcn` play the role of the *g* function in the data suppression framework in [1].

To decide if a message has to be sent to the base station, the node uses the following encoder function

$$f_{\text{enc}} = \begin{cases} \tilde{x}_T, & \text{if } (\texttt{Zb.fcn} \cap \texttt{Za.fcn}) \\ \perp, & \text{otherwise} \end{cases}, \tag{21}$$

where $\tilde{x}_T$ is the median of the *T* values read inside the post-monitoring window. If *T*=1, $\tilde{x}_1 = x_{t+1}$.

At each time period *t*, the base station waits for the r_t messages from the nodes and uses the following decoding function to update its database

$$\hat{x}_t = \begin{cases} \tilde{x}_T, & \text{if } r_t = \tilde{x}_T \\ \hat{x}_{(t-1)}, & \text{if } r_t = \perp \end{cases}. \tag{22}$$

In case of data suppression ($r_t = \perp$), the base station uses the last sent value, $\hat{x}_{(t-1)}$, as the estimative for the current time period. Then, as we see in equation (22), the base station stores a sequence of median values.

VB and TS-SOUND schemes have similar encoding and decoding functions. They send only one value to the base station.

5.5 – On TS-SOUND's parameters

TS-SOUND scheme is defined by three parameters: the size of the time windows (T); the amount of change in the expected behavior of the monitored variable we want to detect (α) and how much weight the current observation must have in the on-line updating of the distribution parameters (*r*).

As the length of the post-monitoring, the value of T should be as large as the size of the clusters of aberrant readings. On the other hand, we should choose a small value for T to decrease the delays to detect an outlier and to update the base station if a change-point occurs.

As we will discuss in section 7, we do not know how large the clusters of aberrant readings will be. Then, the choice of the value for T must consider TS-SOUND's performance when it is applied on time series with clusters of aberrant readings of several sizes. Then, we have to choose the value of T that produces the most homogeneous performances considering aberrant clusters of different sizes. The experiments results in section 7 will help us to make this choice.

On choosing the value of *r*, we should consider how large the local variation of time series is. For instance, a wind speed time series has a local variation larger than the local variation of an atmospheric pressure time series (Figure 4, section 6). Therefore, the current observation in a wind speed series should have a weight (*r*) larger than the weight of the current observation in an atmospheric pressure series. However, giving larger weights to the observation in the estimation of the distribution parameters makes harder to detect this observation as an outlier. In fact, as we will see in section 7, values for *r* larger than 0.1 have degraded the suppression rates in the evaluation experiments.

The value of α is the probability of making a mistake: detecting a non-outlier as an outlier. If we set a small value for α, we decrease this error probability. However, small values for α make harder the detection of change-points, especially if these points represent a small change in the expected behavior of the time series. On the other hand, if α is large, the scheme will be able to detect small changes, even though false outlier alarms will rise more often. Then, the user has to define what is more important to

her: capturing small changes or avoiding aberrant readings.

6. Evaluation Experiments

In this section, we describe a set of extensive experiments to evaluate the performance of the TS-SOUND suppression scheme.

6.1 – The data

We used real data collected by the weather station of the University of Washington (USA)[2]. Our goal has been to be able to evaluate the performance of TS-SOUND scheme and compare it with other suppression schemes considering data with diverse types of temporal behavior. Then, we have selected time series for wind speed (nautical miles per hour), air temperature (F), air relative humidity (%) and atmospheric pressure (milibars). The temporal resolution is one measurement per minute (average of measurements at each 5 seconds). To account for seasonal variability in the weather data, we have chosen four different months (October'06, January'07, April'07 and July'07). For each month, we have selected the data of the days from 10^{th} to 16^{th}. We have run the experiments using these 28 daily time series (1440 readings per series) for each variable.

Figure 4 presents the typical daily time series for each variable. These time series present different behaviors: from series with large local movements relative to its global variation (wind speed) until series with small local movements relative to its global variation (atmospheric pressure).

[2] http://www-k12.atmos.washington.edu/k12/grayskies/nw_weather.html

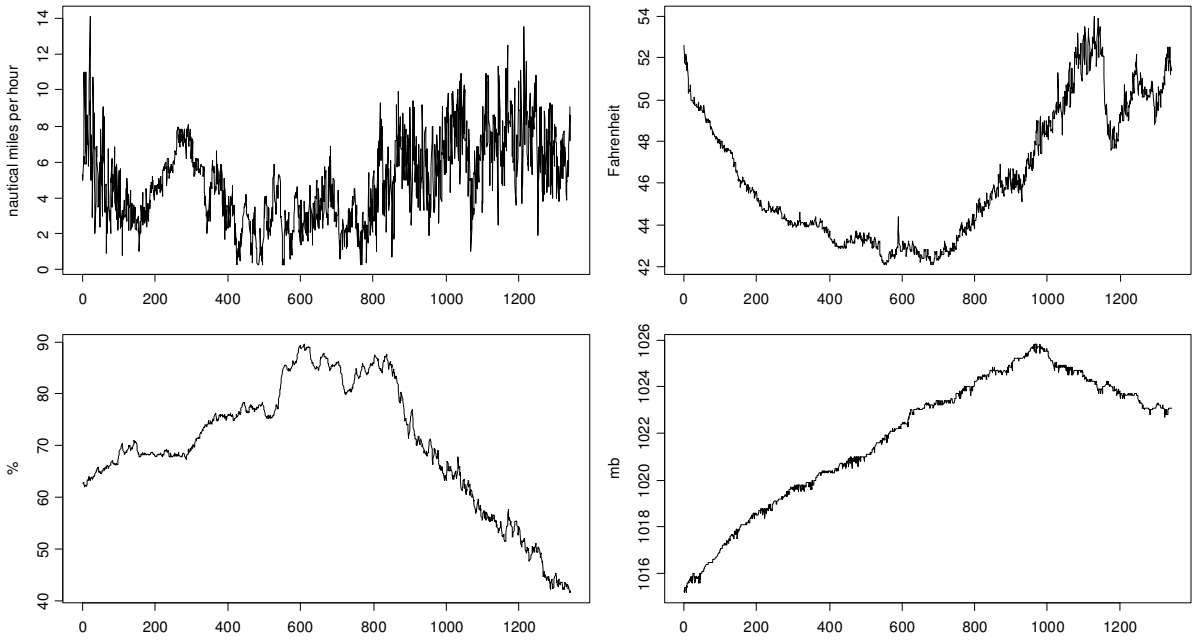


FIGURE 4 – Typical daily time series used in the evaluation experiments. From left to right: wind speed (July'07); air temperature (April'06); air relative humidity (October'06); atmospheric pressure (April'07).

6.2 – The experiments

We have designed the experiments to evaluate the performance of TS-SOUND scheme and compare it with the performance of the following suppression schemes: value-based (VB), exponential regression (EXP) and PAQ.

For the parameters of TS-SOUND scheme, we have set the values r = (0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6), α = (0.25, 0.20, 0.15, 0.10, 0.05, 0.025, 0.01) and T = (2, 4, 6, 8, 10). The value for the threshold $z_T^{c\cdot\alpha}$ corresponds to the percentile 100(1-cα/2) of the Gaussian distribution with mean zero and standard deviation $\sqrt{T}$. As we mentioned in section 4.2.2, we have adopted c=1 in the threshold for Z_{t+T}^{B} and c=0.05 in threshold for Z_{t+T}^{A}. The first (100 + T) values of the time series have composed the learning sample.

Making the TS-SOUND scheme comparable to the other evaluated schemes (PAQ, EXP and VB) is not a trivial task, since they use different criteria to trigger their data sending. The latter schemes use absolute value of the prediction error to decide when the node must send data to the base station,

whereas TS-SOUND uses the detection/classification of outliers. Then, we have had to answer the question: "how to choose values for ε_υ and ε_{VB} (PAQ/EXP and VB error thresholds, respectively) so that we make these schemes comparable to TS-SOUND scheme using the values chosen for α?"

Our solution for this problem has been to use the prediction errors of TS-SOUND scheme to define the values for ε_υ and ε_{VB}. Then, after applying the TS-SOUND scheme to a real time series data using a given value for α, we have calculated the absolute prediction error as following

$$AE_t = \left| x_t - \hat{x}_t \right|, \quad t = 1, 2, ..., N_{TS} \tag{23}$$

where $\hat{x}_t$ is the prediction value for real data x_t and N_{TS} is the size of the time series. To avoid the influence of discrepant values, we have decided to throw out the 10% largest values of AE_t and define the value for ε_υ and ε_{VB} as the percentile 90 of the AE_t values. Therefore, the maximum error of the predictions using PAQ, EXP and VB schemes is the percentile 90 of the prediction error of TS-SOUND schemes. Once the range of the absolute prediction error has been equalized, the distribution of the error within this range will be determined by the performance of the evaluated schemes.

The values for the other parameters of PAQ and EXP have been chosen based on the values cited in [4] as good choices : $\varepsilon_\delta = (1.8/3.0)\varepsilon_\upsilon$, A_{PAQ}=(5, 15) and a=(8/15)A_{PAQ}. The learning sample size (N_{LS}) has been set as the first 100 observations of the time series.

6.2.1 – Evaluating the influence of aberrant readings

We have designed an experiment to evaluate how sensitive to aberrant points are the suppression schemes analyzed in this paper. This experiment has used using the real time series previously described. For each time series, we have inserted aberrant values, isolated or clustered, in randomly chosen time periods. To generate isolated aberrant readings, we have sampled 100 time periods of a given time series to be replaced by an aberrant reading, preserving a minimum interval of 11 time periods between two sequential positions. Then, about 10% of a time series has been composed by aberrant points. To generate the aberrant reading at the selected time period, we have used the interquartile range IQ, defined as IQ = $P_{diff}(75)$ - $P_{diff}(25)$, where $P_{diff}(p)$

is the percentile p of the sequential differences $|X_t - X_{t-1}|$. In a boxplot analysis [21], values smaller than $P_{diff}(25) - 3 \times IQ$ or greater than $P_{diff}(75) + 3 \times IQ$ are considered to be extreme outliers. Then, to generate an aberrant reading, we have added (*sign* x *range* x IQ) to the current value of the candidate time period, where *range* has been randomly chosen inside the interval [3 ; 6] and *sign* has been randomly chosen between -1 and +1. Adopting the boxplot's rule and a random value for *range*, we have expected to decrease our influence on the generation of the aberrant values.

In addition to isolated aberrant readings, we have generated clusters with 2, 3, 4 and 5 aberrant readings. From now on, we will denote the clusters of aberrant readings by *aberrant clusters*. To produce such clusters, we have supposed that all the aberrant readings in a cluster are generated in a same direction, as those ones presented in Figure 1. Given the size of the cluster, we have grouped the initial 100 aberrant readings. For instance, in the experiments with clusters of 4 aberrant points, we have generated 25 clusters. The first reading of the cluster has been inserted in the time series as in the isolated case. To generate the sequential aberrant readings, we have used the same rule to produce the first aberrant reading. However, their *signs* have been constrained to the sign of the first reading in the cluster. We have applied TS-SOUND, PAQ, EXP and VB schemes on these modified time series using as parameters the values described in the previous section.

6.2.2 – Assessing the performance of the suppression schemes

We have evaluated the performance of suppression schemes using the trade-off between two measures: the *suppression rate* and the *prediction error*.

We have adopted the *median absolute error* (MAE) to measure the prediction error. The median absolute error has been calculated as

$$MAE = median_{(t=1,2,\ldots,N_{TS})} \left| x_t - \hat{x}_t \right|, \tag{24}$$

where N_{TS} is the size of the time series.

We can cite some advantages of adopting MAE to assess the prediction error instead of using other error measures such as the mean square error (MSE). First, the absolute difference between predicted and real values is an intuitive measure for the prediction error. Second, the absolute error preserves the original measurements units, which makes easier its interpretation. Finally, the median is more robust to the influence of discrepant values.

The suppression rate (SR) has been calculated as the proportion of suppressed data

$$\text{SR} = 1\text{-}\frac{(\text{number of sent messages})}{N_{TS}}. \tag{25}$$

If a scheme increases its suppression rate, we expect MAE also increases, since the node updates the base station database less often. A suppression scheme S1 can be defined as better than other suppression scheme S2 if, for a given value of prediction error, S1 is able to get suppression rates larger than the suppression rates of S2.

To evaluate the robustness to aberrant readings of TS-SOUND scheme, we have calculated the odds of sending data to the base station provided that an aberrant reading has been detected as

$$Odds_{SENT}^{Aberrant} = \frac{\text{number of detected aberrant readings that have caused data sending}}{\text{number of detected aberrant readings that have not caused data sending}}. \tag{26}$$

A TS-SOUND scheme is considered to be robust to aberrant readings if its $Odds_{SENT}^{Aberrant}$ is smaller than 1. Then, a suppression scheme S1 can be defined as more robust to aberrant readings than a suppression scheme S2 if S1 has got an odds of sending data smaller than S2's odds.

Since PAQ, EXP and VB schemes always send the detected outliers to the base station, their $Odds_{SENT}^{Aberrant}$ are infinite. Then, we have evaluated the robustness to aberrant readings of these schemes by comparing their suppression rates in the time series with and without aberrant readings. For a robust scheme, this ratio is close to 1.

7. The results

In this section, we present the main results of the extensive experiments described in the previous

section. We start our analysis with a simple case study.

7.1 – A simple case study

We have had access to the air temperature and relative humidity data collected by three Tmote Sky sensor nodes[3]. They have collected data at each 30 seconds during 32 hours. Each sensor node has produced about 4000 readings of each variable. The left side of the Figure 5 presents the time plot of the temperature data collected by the sensor node 2.

Since these data have not enough time series to be used in an extensive evaluation, we have used them to perform an initial analysis. Table 1 presents the values for the performance measures of the evaluated schemes using T=2, α=0.15, r=0.1 (TS-SOUND's parameters) and A_{PAQ} = 15 (PAQ and EXP's parameter). The values for ε_{υ} and ε_{VB} were determined as we have described in section 6.2.

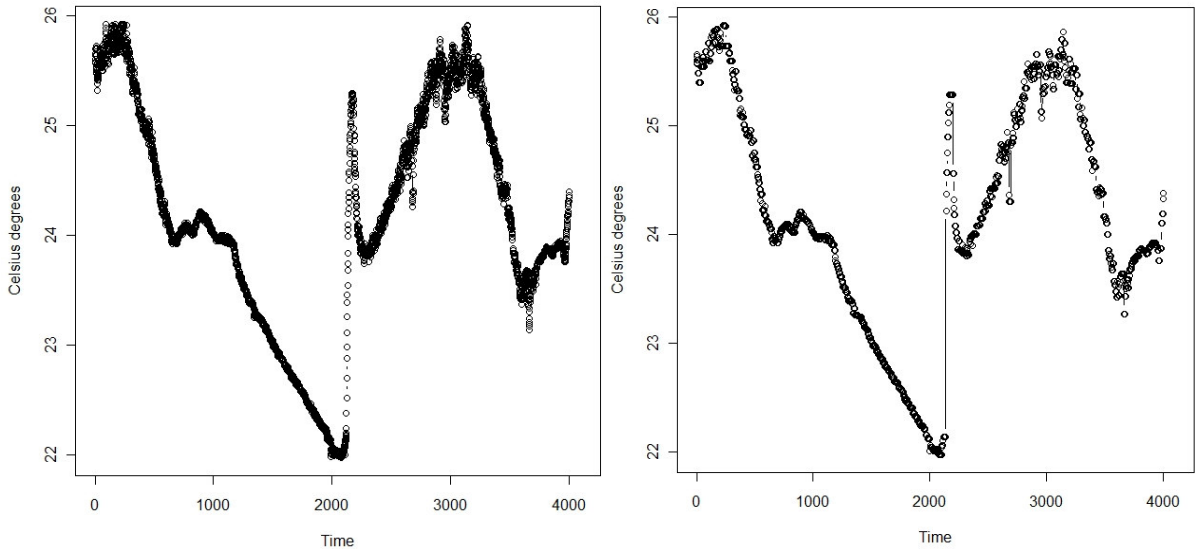


FIGURE 5 – Results of TS-SOUND scheme: time series predicted at the base station (on the right); real air temperature data collected at the sensor node 2 (on the left).

[3] Thanks to the Professor Rone Ilídio da Silva of Universidade Presidente Antônio Carlos (Campus Conselheiro Lafaiete), for making these data available.

TABLE 1 – Results of the evaluation experiments applied to the Air Temperature (^oC) and Relative Humidity (%) data collected by three Tmote Sky sensors: suppression rate and median absolute error (within the parenthesis).

Scheme	Sensor Node 1		Sensor Node 2		Sensor Node 3	
	Temperature ($\varepsilon_\upsilon = \varepsilon_{VB} =$ 0.03 ^oC)	Relative Humidity ($\varepsilon_\upsilon = \varepsilon_{VB} =$ 0.41 %)	Temperature ($\varepsilon_\upsilon = \varepsilon_{VB} =$ 0.08 ^oC)	Relative Humidity ($\varepsilon_\upsilon = \varepsilon_{VB} =$ 0.25%)	Temperature ($\varepsilon_\upsilon = \varepsilon_{VB} =$ 0.03 ^oC)	Relative Humidity ($\varepsilon_\upsilon = \varepsilon_{VB} =$ 0.17%)
TS-SOUND (r=0.1;T=2; α=0.15)	0.823 (0.005 ^oC)	0.857 (0.057 %)	0.858 (0.015 ^oC)	0.877 (0.040 %)	0.865 (0.010 ^oC)	0.883 (0.020 %)
PAQ (A_{PAQ}=15)	0.753 (0.014 ^oC)	0.807 (0.157 %)	0.812 (0.031 ^oC)	0.826 (0.086 %)	0.783 (0.010 ^oC)	0.836 (0.053 %)
EXP (A_{PAQ} = 15)	0.893 (0.010 ^oC)	0.816 (0.146 %)	0.825 (0.028 ^oC)	0.829 (0.082 %)	0.789 (0.009 ^oC)	0.846 (0.051 %)
VB	0.858 (0.010 ^oC)	0.872 (0.124 %)	0.874 (0.020 ^oC)	0.892 (0.086 %)	0.859 (0.010 ^oC)	0.897 (0.041 %)

For both variables, TS-SOUND has got suppression rates similar to the rates of the other schemes, whereas its prediction error has been smaller than the prediction error of the other schemes. The right side of the Figure 5 presents the time series predicted at the base station when the TS-SOUND scheme has been applied to the temperature data collected by the sensor node 2. Comparing the real and predicted series, we have noticed that TS-SOUND avoids reporting the erratic movement of the series as, for instance, in the beginning and final parts of the time series in the Figure 5. On one hand, TS-SOUND delays the notification of fast changes such as the one near the time period 2000. TS-SOUND classifies this behavior as an aberrant one until it notices there is a change. From this moment on, it updates the base station more often. On the other hand, likely clusters of aberrant readings are represented by few updates, as those ones near the time period 3000.

Since no messages can be sent to base station during the TS-SOUND's monitoring window, increasing its size (T) has increased the suppression rates. As a result, the value of the median absolute error has also increased. The parameter α has had a similar effect on the suppression rates and prediction errors: the larger the rigor to consider an observation as an outlier, the larger the chance of suppressing data.

On the value of *r*, our initial experiments have pointed to *r*=0.1 as the value that produces the best trade-off between the suppression rate and the prediction error. This means that we obtain the best performance for TS-SOUND when the on-line estimation of the new values for the distribution parameters sets less weight to the current sensor reading (equations (7) to (11)). TS-SOUND schemes using *r* values smaller than 0.1 have produced results very similar to the results with *r*=0.1. However, increasing the value of *r* up to 0.5 has degraded the suppression rates. In fact, giving larger weights to the observation in the estimation of the distribution parameters makes harder to detect this observation as an outlier.

7.2 – Selecting the best value for the length of the monitoring window

TS-SOUND's strategy to distinguish a change-point from an aberrant reading is to use a post-monitoring window whenever an outlier is detected. This time window works as a filter of aberrant readings and makes TS-SOUND robust to these erroneous data. The success of this filtering strategy is closely related to the length of the monitoring window. We expect large aberrant clusters require large windows to be filtered. However, we do not know how large the clusters of aberrant readings will be.

In this section, we examine the results of experiments with the meteorological data of the University of Washington to answer the following question: "Considering several sizes for the clusters of aberrant readings, which is the minimum value for the length of the monitoring window that leads to TS-SOUND scheme with

a) the largest robustness to aberrant readings and

b) the best trade-off between suppression rate and prediction error ?"

To answer the first part of the question, we have summarized some of the experiments results using plots as the ones in Figure 6. They present the odds of "sending data to the base station provided that an aberrant reading has been detected" as a function of the length of the monitoring window considering aberrant clusters of several sizes. Figure 6 presents the results for the sets of time series that have got the most irregular behaviors: air relative humidity and wind speed measurements. We have looked for the smallest length for the monitoring window that leads to the most similar values for the odds among aberrant clusters of different sizes. For the wind speed time series, the monitoring windows of length 10 and 2 have presented the most similar odds. Then, the chosen length is T=2. For the air relative humidity, the length is also T=2. For air temperature and atmospheric pressure time series, the larger the monitoring window is, the less homogeneous the odds are. Therefore, T=2 is the chosen length.

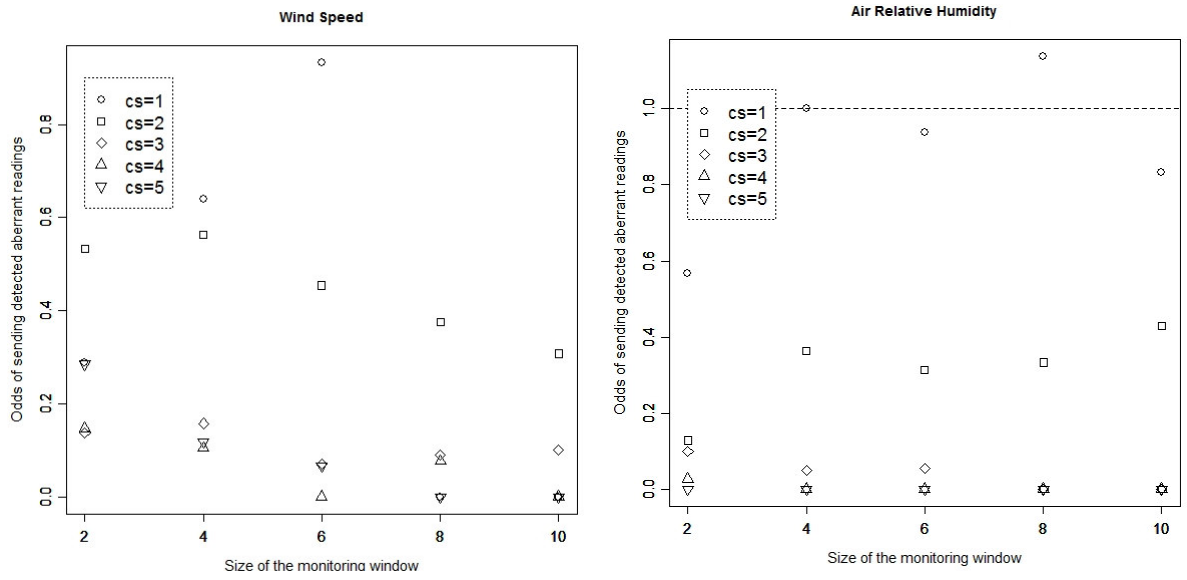


FIGURE 6* – Robustness to aberrant readings of TS-SOUND scheme according to the length of the monitoring window (T) and the size of the aberrant clusters (CS). The other TS-SOUND's parameters have been α=0.15 and *r*=0.1. *There is an enlarged version of these figures in the end of this document.

Increasing the value of α decreases the odds of "sending data to the base station provided that an aberrant reading has been detected", since the rigor to classify an observation as an outlier increases.

We have answered the second part of the question by examining plots as the ones in the Figure 7. They present the trade-off between suppression rate and prediction error for several lengths of the monitoring window and considering aberrant cluster of different sizes. We have looked for the smallest length for the monitoring window that leads to the most similar suppression rates and prediction errors among aberrant clusters of different sizes. For wind speed and air relative humidity time series (Figure 7), we have looked for the group of symbols (*T* values) that are more "clustered". The monitoring windows of length 6 and 4 have presented the most similar suppression rates and prediction errors. Then, the chosen length is T=4. Examining the air temperature and atmospheric pressure time series, we have got the same value for T.

Since we have got different answers for the two parts of the proposed question, we have chosen the best value for T by examining the effect of using the value chosen in part (a) on the context of part (b) and vice versa. Then, we have examined the effect of choosing T=2 on the trade-off between suppression rate and prediction error and the effect of using T=4 on the odds of "sending data to the base station provided that an aberrant reading has been detected". In the former case, exchanging T=4 for T=2 produces a substantial increasing in the heterogeneity of the suppression rates and prediction errors for the wind speed, air temperature and atmospheric pressure time series. In the latter case, the effect of exchanging the values of T (T=2 for T=4) is smaller than in the former case. The worst effect has occurred in the air relative humidity time series (right side of Figure 6). For T=4, the odds of "sending data to the base station provided that an aberrant reading has been detected" is, in median, equal to 1 when isolated aberrant readings (CS=1) occur in the time series. However, the other odds are smaller than 1. Then, considering all evaluated time series and sizes for aberrant clusters, we have chosen the value 4 as the best one for the length of the monitoring window.

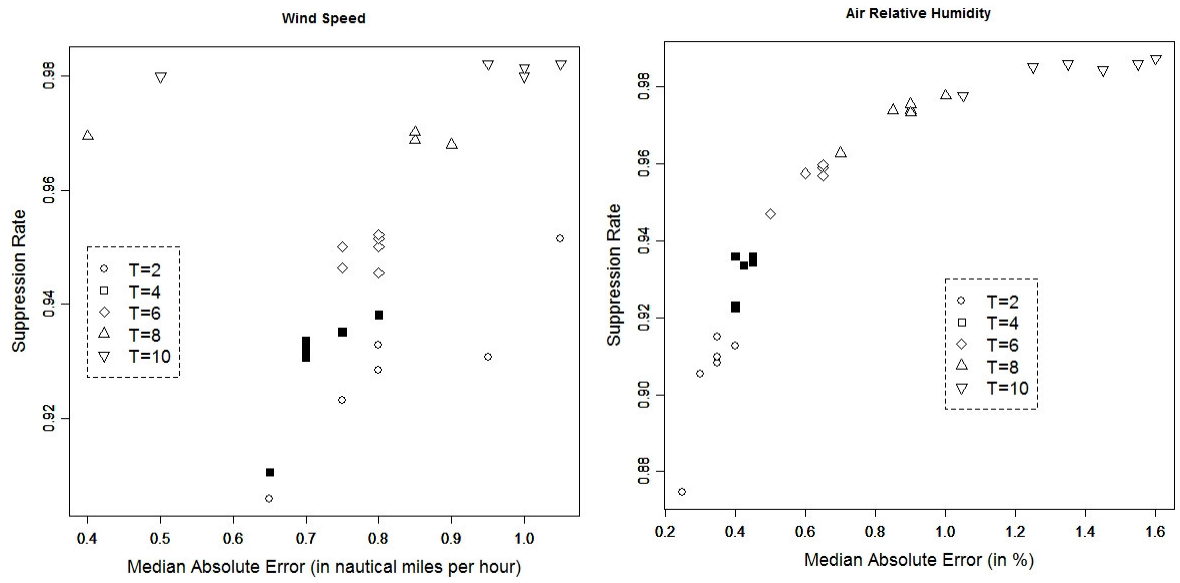


FIGURE 7* – Performance of TS-SOUND scheme applied to wind speed and air relative humidity time series. The parameters have been α=0.15, *r*=0.1 and several values for the length of the monitoring window (T). Each point represents the summary of the results for time series with aberrant clusters of different sizes: 0 (no aberrant readings), 1 (isolated aberrant readings), 2, 3, 4 and 5. *There is an enlarged version of these figures in the end of this document.

7.3 – Evaluating the schemes' performances

In this section, we compare the performance of TS-SOUND scheme using T= 4, selected in previous section, with the performance of PAQ, EXP and VB schemes.

As we have mentioned in section 6, we have used the trade-off between the suppression rate and the prediction error of a scheme as a measure for its performance. We represent graphically this trade-off for each one of the sets of meteorological time series using the scatter plots of the figures from 8A to 8D. Each point of a scheme represents the summary of its performance using a different value for α (0.15, 0.10, 0.05, 0.025, 0.01), in this order, following the increasing of the suppression rates. For PAQ/EXP and VB schemes, the values for the correspondent error thresholds ε_δ and ε_{VB}, respectively, have been defined as described in section 6.2. Points closer to the upper-left corner represent the schemes with the best performances. Since TS-SOUND with T=4 has got its worst results when the times series had isolated aberrant readings (figures 6 and 7), we have

chosen to use this scenario to compare TS-SOUND's performance with the performance of the other evaluated schemes. The upper and bottom subfigures illustrate which data the base station would have if the node applied TS-SOUND and VB schemes, respectively, on the real times series in the middle subfigure. The real times series in the middle subfigures are the original ones in Figure 4 with generated aberrant clusters of size 1 (isolated aberrant readings).

To understand what values we should expect for the prediction errors so that we could consider them acceptable, we have used the size of the sequential changes in the time series as a basis for comparison. Then, we have calculated the sequential absolute differences, $|X_t - X_{t-1}|$, in the series of each variable and summarized the sequential changes (non-zero differences) using the percentiles 5 and 95. Therefore, in the air relative humidity and temperature time series, 90% of the sequential changes are within the interval [0.10 ; 1.0] % and [0.10 ; 1.0] F, respectively. In the atmospheric pressure time series, 90% of the sequential changes are within the interval [0.10 ; 0.40] mb. In the wind speed time series, 90% of the sequential changes are within the interval [0.10 ; 2.1] nautical miles. Analyzing figures from 8A to 8D, we notice all evaluated schemes have got median prediction errors compatible with the expected sequential changes in a given type of meteorological time series. In other words, all evaluated schemes have got acceptable errors on predicting the real time series at base station.

TS-SOUND scheme has got its best performance in air relative humidity and temperature time series (figures 8A and 8B, respectively). In the air relative humidity data, TS-SOUND has been the scheme with the best performance for all values of α, reaching the highest suppression rates and the smallest prediction errors. For the smallest two values of α in the air temperature data and for α=(0.10 , 0.05) in the atmospheric pressure data (Figure 8C), the prediction errors of the TS-SOUND and VB are, in median, the same. However, TS-SOUND has got suppression rates higher than VB's rates.

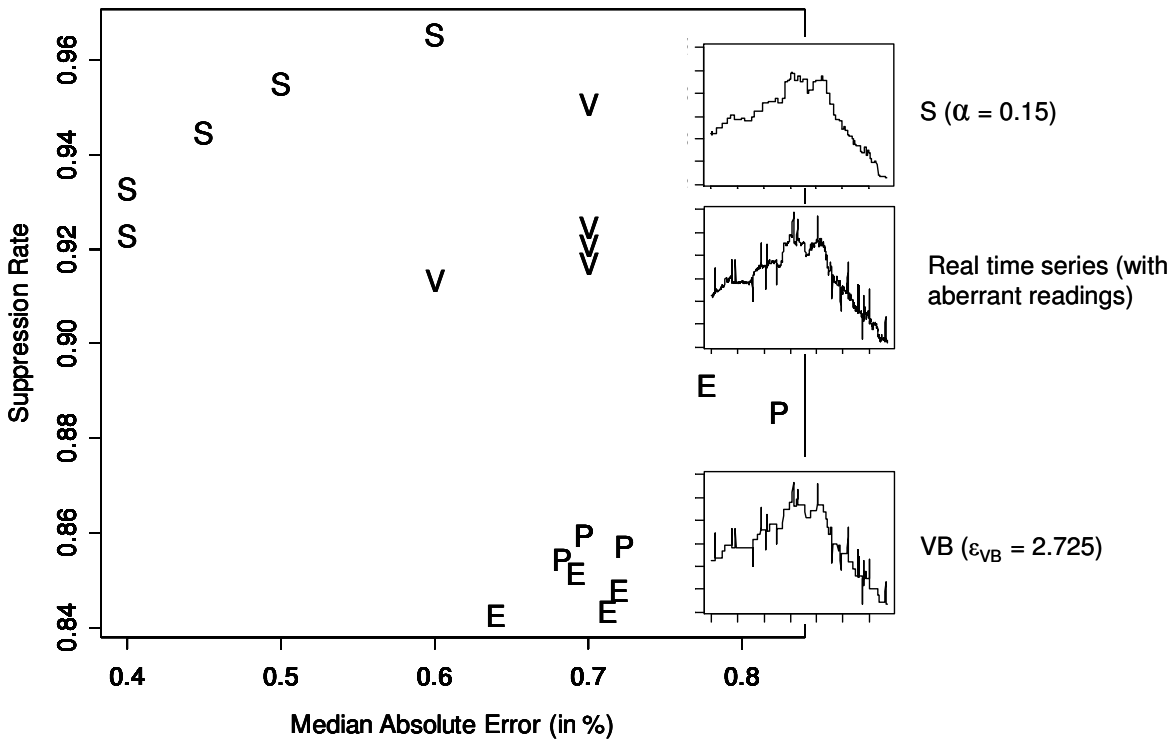


FIGURE 8A. Performance of the evaluated schemes in the air relative humidity times series with isolated aberrant readings. Legend: S for TS-SOUND (*r*=0.1, T = 4), V for value-based, P for PAQ (A_{PAQ}=15) and E for EXP (A_{PAQ}=15). Each point of a scheme represents the summary of its performance using a different value for α (0.15, 0.10, 0.05, 0.025, 0.01), in this order, following the increasing of the suppression rates. For PAQ/EXP and VB schemes, the values for the correspondent error thresholds ε_δ and ε_{VB}, respectively, have been defined as described in section 6.2.

In the wind speed time series, which have a large local variation, TS-SOUND has increased the prediction errors in comparison to the other schemes' errors (Figure 8D). Nevertheless, it has got a higher increase in the suppression rates in relation to maximum possible increasing. As an example, for α=0.15, TS-SOUND has got a median prediction error of 0.8 nautical miles per hour, which has been 14% larger than VB's median prediction error. However, TS-SOUND's suppression rate has been 0.938, whereas VB has got 0.798. Then, TS-SOUND's rate has got an increasing of 69% in relation to maximum increasing in the VB rate (1 – 0.798). For α=0.10, TS-SOUND's error has been 43% larger than VB's error but TS-SOUND's has increased the suppression rate in 77% of the maximum possible increasing. If we compare TS-SOUND with the PAQ and EXP schemes, the gains are higher.

In time series with small local variation, as the atmospheric pressure series, VB scheme has got median prediction errors equal to zero, even suppressing about 77% of the readings (Figure 8C). However, the correspondent TS-SOUND scheme has suppressed about 95% of the readings, in median, at the cost of increasing 0.05 milibars in the prediction error. Since this increasing is among the 5% smallest sequential changes in atmospheric pressure series, we conclude it is worth to adopt TS-SOUND for this type of data, getting a higher suppression rate at the cost of a small increasing in the prediction error.

On choosing the best value for α, we have to consider how large the local variations in the series are. Comparing figures 8C, 8A, 8B and 8D (in this order), we conclude the larger the local variation the larger the best value for α must be. In general, for values of α larger than 0.05, the increasing in the suppression rate does not compensate the increasing in the prediction error.

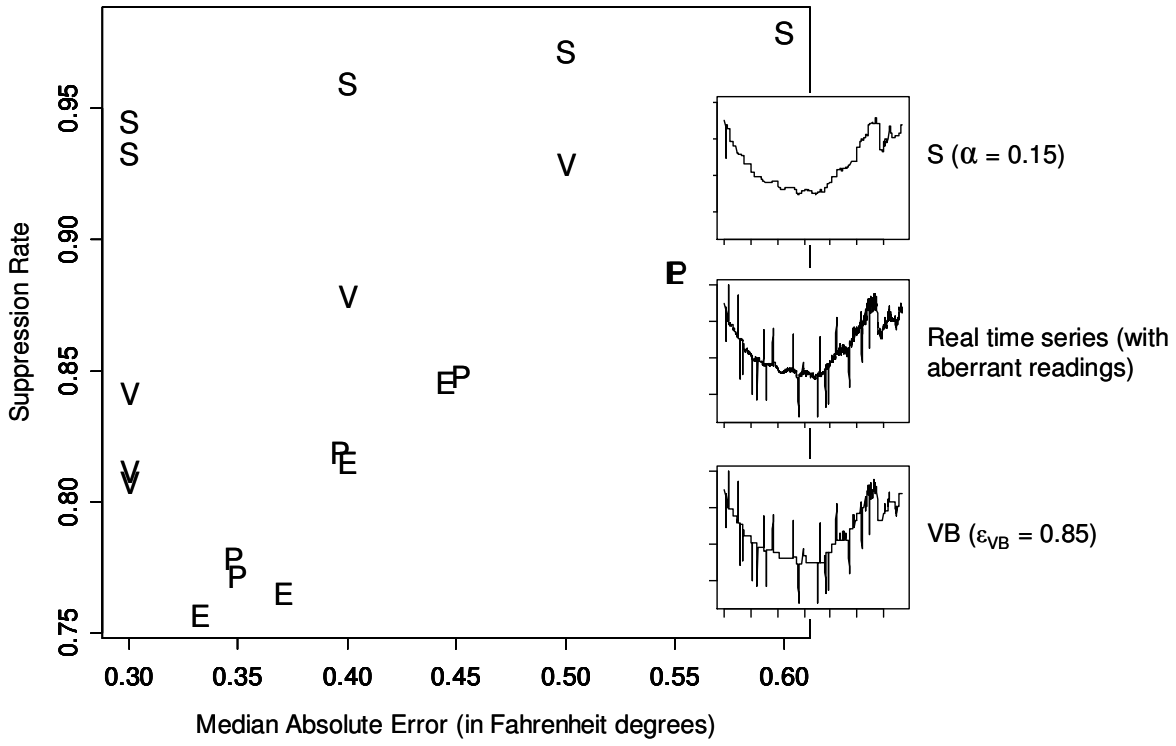


FIGURE 8B. Performance of the evaluated schemes in the air temperature times series with isolated aberrant readings. The legend and other details are in the caption of Figure 8A.

Comparing the predicted time series to the real ones (subfigures), we notice the robustness to the aberrant readings of TS-SOUND scheme, whereas VB suffers a large influence of these erroneous data. VB's predicted series are similar to the series with aberrant readings (middle subfigures), whereas TS-SOUND's predicted series look like the original series, without aberrant readings, in Figure 4.

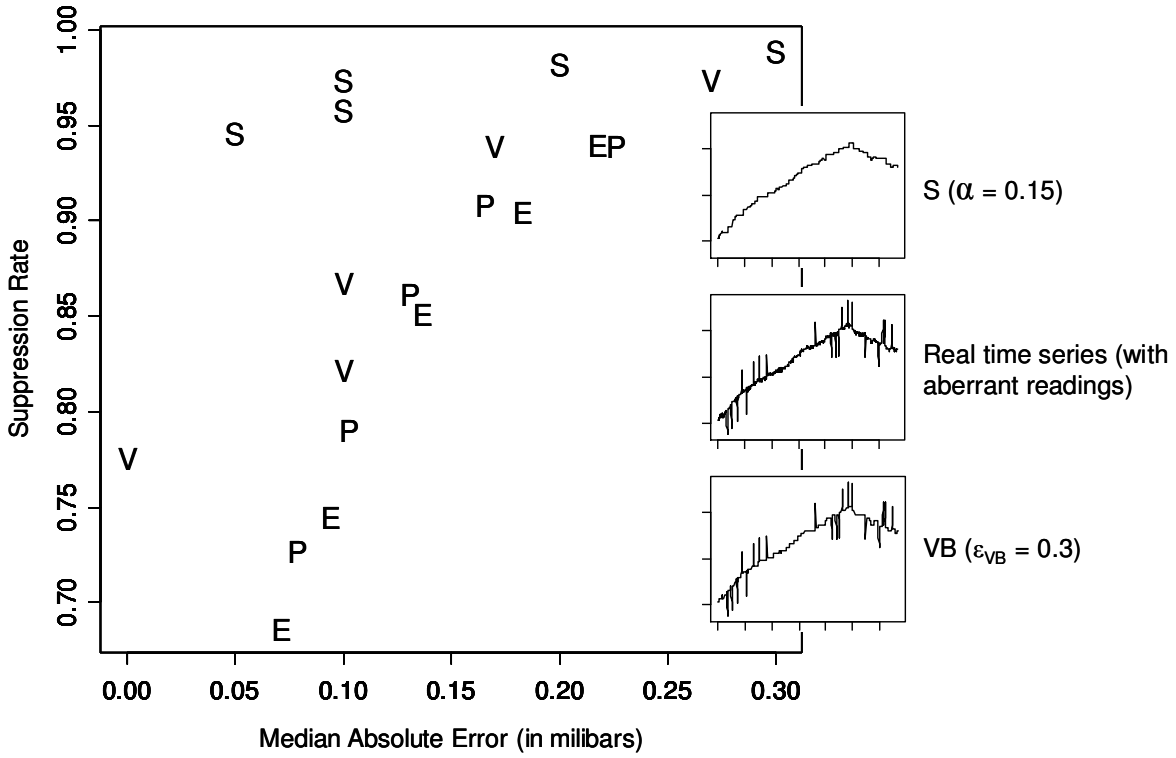


FIGURE 8C. Performance of the evaluated schemes in the atmospheric pressure times series with isolated aberrant readings. The legend and other details are in the caption of Figure 8A.

PAQ and EXP schemes using the largest monitoring window (A_{PAQ}=15) have got suppression rates larger than the rates of those schemes using a smaller window (A_{PAQ}=5). Therefore, PAQ and EXP schemes having a larger period to evaluate the re-estimation of the model parameters have been a better alternative, even if the prediction errors have been slightly larger. Despite of having updated the base station more often than the other schemes, PAQ and

EXP schemes have not got the smallest prediction errors. In other words, using these model-based suppression schemes is not a good strategy if the dataset may have aberrant readings.

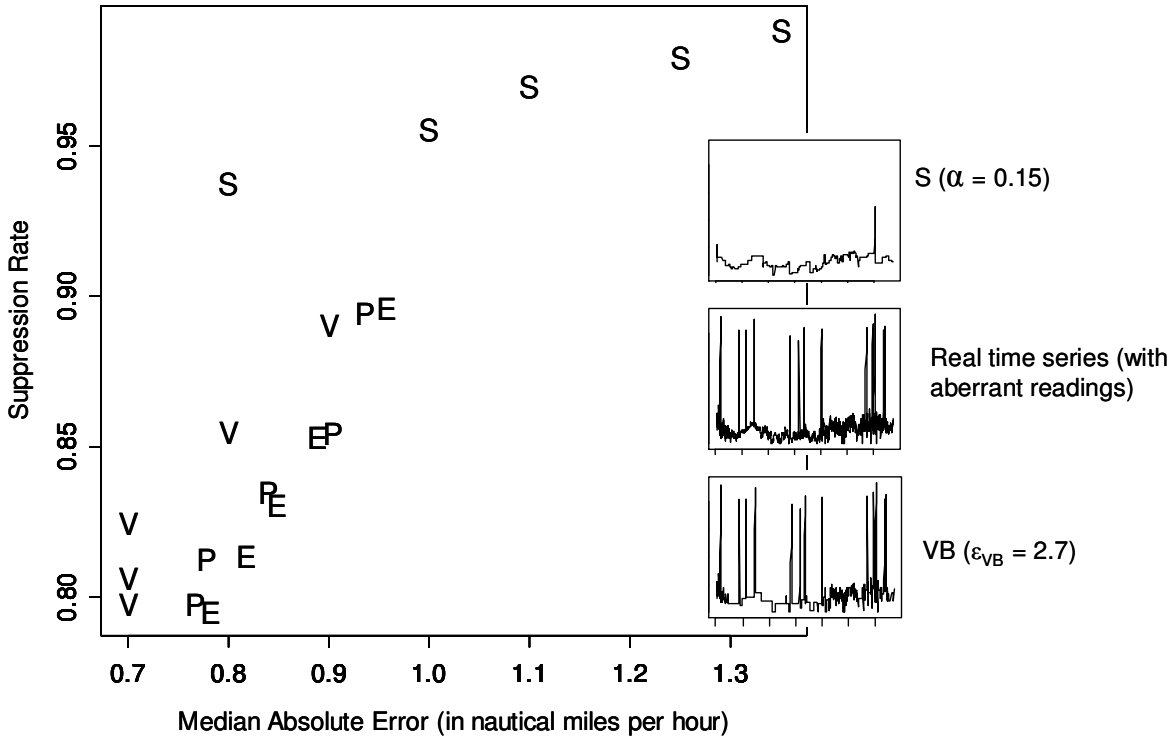


FIGURE 8D. Performance of the evaluated schemes in the wind speed times series with isolated aberrant readings. The legend and other details are in the caption of Figure 8A.

7.4 – Evaluating the schemes' robustness to aberrant clusters

In this section, we compare the robustness to aberrant clusters of the suppression schemes. Since the $Odds_{SENT}^{Aberrant}$ of PAQ, EXP and VB are infinite, we have calculated the ratio between the suppression rates with and without aberrant clusters. A suppression scheme robust to aberrant readings should present this ratio close to 1. For a suppression scheme that suffers the influence of aberrant readings, this ratio is smaller than 1.

Figure 9 presents the ratios for the suppression schemes applied on atmospheric pressure and wind speed time series. In these sets of series, the evaluated schemes have suffered the largest and the smallest influence of aberrant clusters, respectively.

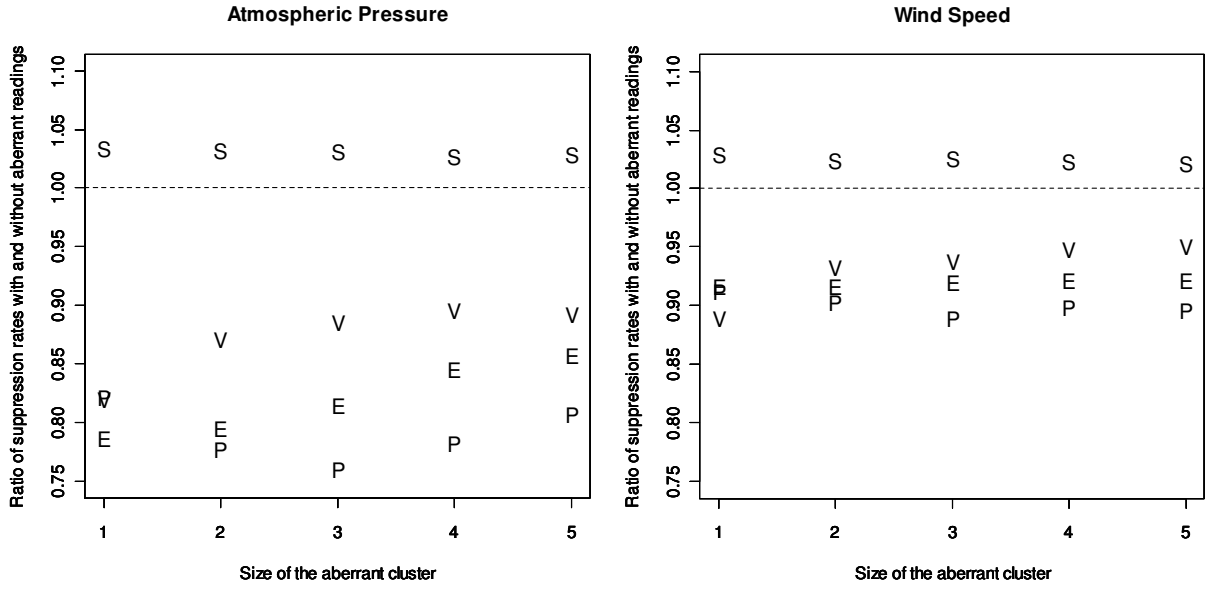


FIGURE 9*. Influence of aberrant readings on the suppression rate of the evaluated schemes applied on atmospheric pressure and wind speed time series. Legend: S for TS-SOUND (*r*=0.1, T = 4, α=0.15), V for value-based, P for PAQ (A=15) and E for EXP (A=15). *There is an enlarged version of these figures in the end of this document.

The suppression rates of TS-SOUND scheme have not presented relevant changes, whereas the suppression rates of the other schemes have decreased, especially for PAQ and EXP schemes. This is because the model-based prediction adopted by PAQ/EXP schemes is quite sensitive to aberrant readings. They decrease PAQ/EXP's suppression rates for two reasons: the node has to send them as detected outliers to the base station and they cause the re-estimation (and sending) of the new model parameters.

For VB scheme, aberrant clusters make nodes send data to the base station at least two times: in the beginning and in the end of the cluster. Inside the cluster, aberrant readings tend to be similar to each other. This reduces data sending. This could explain why the influence of aberrant readings on the suppression rates has been smaller for aberrant clusters than for isolated aberrant readings. Clusters of aberrant readings would tend to amortize the initial and final data sending.

7.4- A note on the order of the AR model

The model-driven approach is an efficient solution to data collection in sensor networks if the monitored variable has a well-known behavior so reliable models can be defined [1]. Then, let us suppose that a sophisticated model is the best representation for the expected behavior of the sensor data. In this case, the simplicity of AR(1) model in the TS-SOUND scheme could degrade its performance if we compare it to the performance of a scheme adopting a more sophisticated model.

To evaluate this hypothesis, we have simulated time series according to the AR(3) model, which is the model that PAQ scheme uses. To generate the model coefficients, we have fit an AR(3) model to the time series in Figure 4, which represents the typical time series for each variable we have considered in the experiments. For each set of coefficients, we have simulated 50 time series with 1440 observations each, which corresponds to 50 days of monitoring with one reading per minute).

The simulated time series have presented different behaviors because the AR(3) coefficients used in the simulations have come from series with different behaviors (Figure 4). Since it is necessary to analyze the schemes' performances in groups of series with similar behaviors, we have had to quantify the differences between the behaviors of the simulated time series. To do this, we have defined the *Relative Lagged Difference* (RLD_l) as

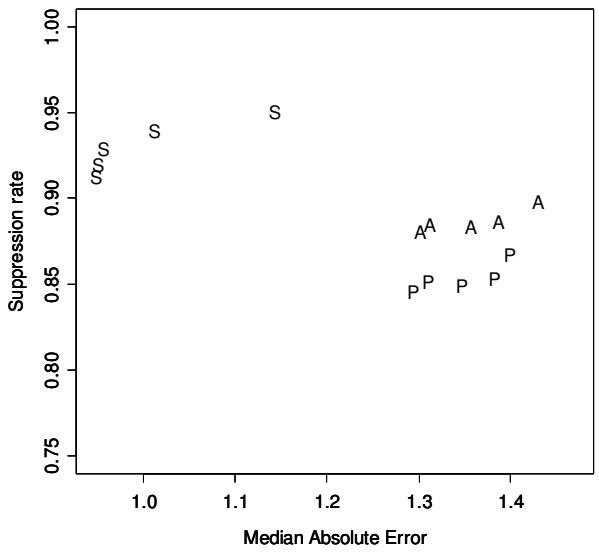
$$RLD_l = \frac{\underset{t=l+1,l+2,\ldots,N}{median}\left(\left|X_t - X_{t-l}\right|\right)}{\underset{t=1,\ldots,N}{max}(X_t) - \underset{t=1,\ldots,N}{min}(X_t)}, \quad l = 1, 2, \ldots, N-1. \tag{27}$$

It compares the typical (median) difference between time periods t and t-l with the total range of the values. The values of RLD_l range from 0 to 1. The lag l indicates how local is the movement we want to capture. Smaller the value of l, the more localized the analysis. For instance, the values of RLD_{10} for the time series in Figure 4 are: 0.0942 (wind speed), 0.0252 (air temperature), 0.0201 (air relative humidity) and 0.0081 (atmospheric pressure). Therefore, time series with smooth changes relative to the total range (e.g., atmospheric pressure) have low values for RLD_l, whereas abrupt changes result in a higher value for RLD_l (e.g., wind speed).

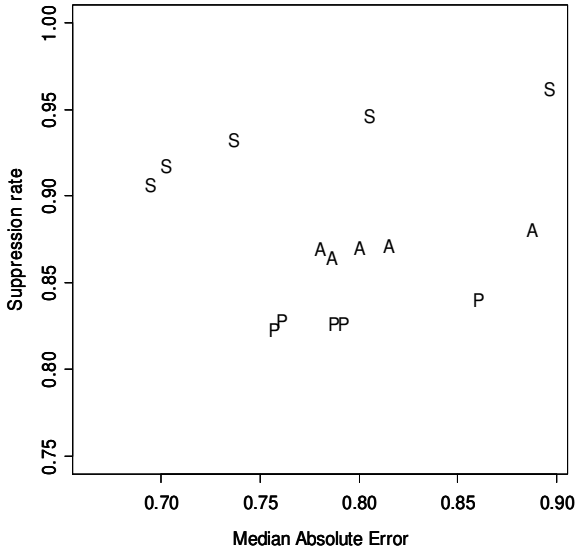
After calculating the RLD_{10} for all 200 times series, we have separated them into three groups according to their RLD_{10} value and applied TS-SOUND and PAQ schemes on the time series of each group. The values for the parameters have been the same of the experiments in section 7.4.

Figure 10 presents the summaries for the performance of both schemes in the three groups of time series. Similarly to the figures of section 7.1, points closer to the upper-left corner represent the schemes with the best performances. As in the experiments with real data, PAQ scheme using the largest post-monitoring window (A_{PAQ}=15) have outperformed the schemes using a smaller time window (A_{PAQ}=5).

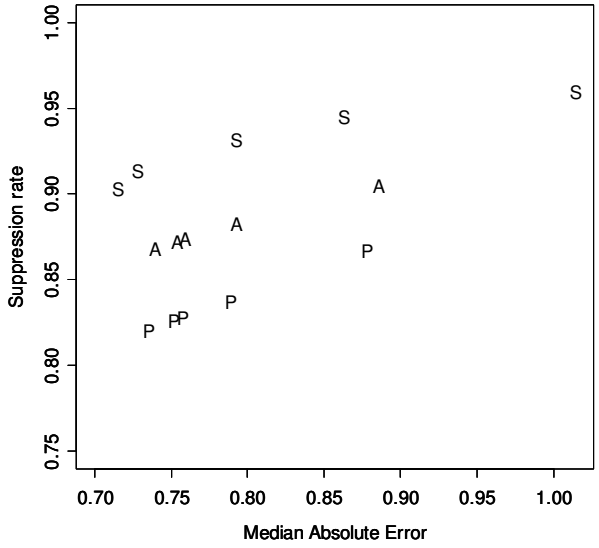
We expected that PAQ scheme could get at least prediction errors smaller than the errors of TS-SOUND. However, even in a scenario clearly favorable to PAQ, the most of TS-SOUND schemes have outperformed their correspondent PAQ schemes. In the time series with smooth changes relative to the total range (Figure 10A), all TS-SOUND schemes have outperformed all PAQ schemes, getting the highest suppression rates and the smallest prediction errors. As the time series have increased their local variation relative to their total range (RLD_{10} increases), PAQ schemes have got prediction errors closer to the errors of TS-SOUND schemes. However, for the first two values of α, TS-SOUND has still outperformed PAQ.



(A) $0 \leq RLD_{10} < 0.025$



(B) $0.025 \leqslant RLD_{10} < 0.050$



(C) $0.05 \leqslant RLD_{10} < 0.075$

*There is an enlarged version of these figures in the end of this document.

FIGURE 10*. Summaries for the performance of TS-SOUND and PAQ schemes in data simulated according to the AR(3) model. Legend: S for TS-SOUND (r=0.1, T = 4), P and A for PAQ with A_{PAQ} = 5 and 15, respectively. Each point of a scheme represents the summary of its performance using a different value for α (0.15, 0.10, 0.05, 0.025, 0.01), in this order, following the increasing of the suppression rates. For PAQ scheme, the values for the correspondent error thresholds, ε_δ, have been defined as described in section 6.2.

8. Discussion

Data suppression schemes are defined by an agreement between sensor nodes and base station about the expected behavior for the sensor readings. To decide when the sensor nodes may suppress their data, the schemes evaluate the prediction error, which is the difference between the value the sensor actually collects and the value predicted according to the expected behavior for the sensor readings. If the collected value fits to the expected behavior, node suppresses its data. Otherwise, it sends data to the base station.

Since the schemes for data suppression look for changes in the expected behavior of the sensor data, they are sensitive to aberrant readings. Transmitting these erroneous data is a waste of energy. In a simple suppression scheme as the Value-based [1], for instance, an aberrant point may produce two unnecessary messages to the base station. That is because the scheme detects two sequential changes of behavior: one when the aberrant readings occur and another when the readings get normal again.

To avoid sending aberrant readings, one can propose to use a fixed threshold: readings smaller or greater a predefined value would be considered as erroneous data. However, that is a naive solution, since what would be aberrant at a time period of the series might not be aberrant at another time period. For instance, a reading of 1026 mb at time period 200 in the atmospheric pressure series (Figure 4) would be considered aberrant. However, this value should not be considered aberrant at time period 1000.

In this paper, we have proposed TS-SOUND, a scheme for temporal data suppression in sensor networks that is robust to aberrant readings. TS-SOUND considers the data collected by a sensor node as a time series and monitors the behavior of this series. It adopts a procedure to detect outliers from a time series and the posterior classification of the detected outlier into

change-points or aberrant readings. In the former case, data are sent to the base station, since it means a change in the expected behavior of the data series. Otherwise, data are suppressed.

Schemes for temporal data suppression proposed in sensor networks literature (PAQ [4], EXP and Value-based [1]) suppress data by comparing the absolute value of the prediction error with a fixed threshold. Using the absolute value of the prediction error allows for controlling its maximum value. However, if the random fluctuations around the expected value (local variations) are larger than the threshold for the absolute error, a large amount of unnecessary data will be sent to the base station and the suppression rates will be small. On the other hand, if the local variations are smaller than the threshold for the absolute error, the suppression scheme will not be able to capture changes in the expected behavior of the monitored data. Then, if the time series has a nonstationary variance, a fixed threshold for the absolute prediction error will not be able to work well during all data collection.

TS-SOUND scheme also uses an error measure to decide if an observation is an outlier. However, it adopts a relative error measure, comparing the absolute error with the data variance, which captures the random fluctuations of the data. As a result, TS-SOUND is able to be adaptable to the local variations of the time series. The suppression rates of TS-SOUND scheme are more robust to the size of the local variations than the other schemes evaluated in this paper.

Besides adopting the relative prediction error, TS-SOUND scheme tries to minimize its sensitivity to aberrant readings using the past data through a moving average. Moreover, even if an aberrant reading raises the outlier alarm, TS-SOUND opens a post-monitoring window to avoid sending this erroneous data to the base station. Although this post-monitoring window introduces a delay in the data delivery, our experiments have shown that a small delay (four time periods) can deal with time series presenting aberrant clusters of several sizes.

Using real data from several sources and presenting different temporal behaviors, we have run experiments to evaluate the suppression rates of TS-SOUND scheme and the prediction errors attached to them. We have used both of these measures to quantify the performance of a

data suppression scheme. We have also evaluated TS-SOUND's robustness to aberrant readings and compared its performance with the performance of PAQ, EXP and VB schemes. The evaluation experiments have shown that TS-SOUND is more robust to aberrant readings than the other schemes considered in this paper. Moreover, TS-SOUND has outperformed the model-based suppression schemes (PAQ and EXP) in all evaluated scenarios and VB scheme in the most of these situations.

The Value-Based is the simplest suppression scheme and has got one of the best performances in our experiments. However, we can list at least three situations in which using TS-SOUND would be better than using Value-Based scheme: a) when the applications is not interested in aberrant readings; b) when the series presents different behaviors along the time, since VB uses a fixed error threshold and TS-SOUND is adaptable to the local variation of the time series; c) when having high suppression rates is more important than having small prediction errors.

To define a TS-SOUND suppression scheme, the user has to choose the values for three parameters: the weight of the last sensed data (r) in the on-line estimation of the distribution parameters, the length of the post-monitoring and past time windows (T) and the rigor to classify an observation as an outlier (α). As we have discussed in section 7, we have found that the value of T has not to be as large as the cluster size. Our experiments have pointed out to 4 as the smallest value for T that leads to homogeneous performances in time series with different behaviors and several sizes of aberrant clusters. On the value of r, our experiments have shown that we obtain the best performance for TS-SOUND when the on-line estimation of the new values for the distribution parameters sets less weight to the current sensor reading. TS-SOUND schemes using r=0.1 have produced the best results and values of r smaller than 0.1 have got very similar results. However, weights larger than 0.1 have degraded the suppression rates.

Since the values for T and r can be constrained to some predefined values, the network user has to choose only the value for α. To do this, it is necessary to define what is more crucial: capturing small changes (large values for α) or avoid aberrant readings (small values for α).

The main contributions of this paper are twofold: a proposal for a data suppression scheme that is robust to aberrant readings and the evaluation of the performance of data suppression schemes considering not only the saved energy but also the quality of the data collected at base station.

9. Future Directions

Sensor networks collect spatially correlated data, which produces areas in the sensors field that are spatially homogeneous. Our future work includes a spatio-temporal version of the TS-SOUND scheme having as its spatial basis the clustering algorithm in [22]. Instead of sending its reports to the base station, the nodes organize themselves into clusters that explore the spatial homogeneity of the data in the sensors field. Besides localizing the most part of the communication among the nodes, such clusters improve the quality of the cluster data summaries to be sent to the base station [23].

The nodes of a sensor network are prone to failures as well as the communication between nodes can be very noisy. Thus, a data collection protocol based on a suppression scheme has to address an important question: how can we distinguish suppressed reports from nodes failures and lack of communication between nodes and base station? Silberstein et al. [24] have proposed interesting alternatives to deal with this problem using Bayesian inference. We study to incorporate the proposed solutions in the spatio-temporal version of TS-SOUND scheme.

9. Acknowledgments

The first author was partially supported by CAPES under PICDT program. The authors thank to the anonymous reviewers for their excellent suggestions, which have contributed to improve this paper.

References

[1] A. Silberstein, R. Braynard, G. Filpus, G. Puggioni, A. Gelfand, K. Munagala, and J. Yang, "DataDriven Processing in Sensor Networks," presented at Biennial Conference on Innovative Data Systems Research, 3., Asilomar (CA), USA., 2007.

[2] R. Cardell-Oliver, K. Smettemy, M. Kranzz, and K. Mayerx, "A Reactive Soil Moisture Sensor Network: Design and Field Evaluation," *International Journal of Distributed Sensor Networks*, vol. 1, pp. 149 -- 162, 2005.

[3] D. Chu, A. Deshpande, J. M. Hellerstein, and W. Hong, "Approximate Data Collection in Sensor Networks using Probabilistic Models," presented at International Conference on Data Engineering (ICDE'06), 22., Atlanta, GA, 2006.

[4] D. Tulone and S. Madden, "PAQ: Time Series Forecasting For Approximate Query Answering In Sensor Networks," *Lecture Notes in Computer Science*, vol. 3868, pp. 21--37, 2006.

[5] Y. Zhang, N. Meratnia, and P. J. M. Havinga, "Outlier Detection Techniques for Wireless Sensor Network: A Survey," University of Twente, Enschede, The Netherlands 2008.

[6] Y. Kotidis, A. Deligiannakis, V. Stoumpos, V. Vassalos, and A. Delis, "Robust Management of Outliers in Sensor Network Aggregate Queries," presented at International ACM Workshop on Data Engineering for Wireless and Mobile Access (MobiDE'07), 6., Beijing, China., 2007.

[7] S. Subramaniam, T. Palpanas, D. Papadopoulos, V. Kalogeraki, and D. Gunopulos, "Online Outlier Detection in Sensor Data Using NonParametric Models," presented at 2nd International Conference on Very Large Data Bases (VLDB'06), Seoul, Korea, 2006.

[8] J. Branch, B. Szymanski, C. Giannella, R. Wolff, and H. Kargupta, "In-network outlier detection in wireless sensor networks," presented at International Conference on Distributed Company Systems (ICDCS), 26. , Lisboa, Portugal, 2006.

[9] T. Palpanas, D. Papadopoulos, V. Kalogeraki, and D. Gunopulos, "Distributed deviation detection in sensor network," *ACM SIGMOD Record*, vol. 32, pp. 77-- 82, 2003.

[10] Y. Zhang, N. Meratnia, and P. J. M. Havinga, "An Online Outlier Detection Technique for Wireless Sensor Networks using Unsupervised Quarter-Sphere Support Vector Machine," presented at Fourth International Conference Series on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2008), Sydney, Australia, 2008.

[11] R. Szewczyk, J. Polastre, A. Mainwaring, and D. Culler, "Lessons From A Sensor Network Expedition," presented at First European Workshop on Sensor Networks (EWSN), Berlin, Germany, 2004.

[12] J. Tateson, C. Roadknight, A. Gonzalez, T. Khan, S. Fitz, I. Henning, N. Boyd, and C. Vincent, "Real World Issues in Deploying a Wireless Sensor Network for Oceanography," presented at Workshop on Real-World Wireless Sensor Networks (REALWSN'05), Stockholm, Sweden 2005.

[13] K. Yamanishi and J.-i. Takeuchi, "A Unifying Framework for Detecting Outliers and Change Points from Non-Stationary Time Series Data," presented at ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 8., Alberta, Canada, 2002.

[14] M. Pollak and D. Siegmund, "Sequential detection of a change in a normal mean when the initial value is unknown," *The Annals of Statistics*, vol. 19, pp. 394--416, 1991.

[15] M. Frisén, "Statistical Surveillance. Optimality and Methods," *International Statistical Review* vol. 71, pp. 403–434, 2003.

[16] S. Muthukrishnan, R. Shah, and J. S. Vitter, "Mining Deviants in Time Series Data Streams," presented at International Conference on Scientific and Statistical Database Management (SSDBM '04), 16., Santorini Island, Greece, 2004.

[17] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient Algorithms for Mining Outliers from Large Data Sets," *ACM SIGMOD Record*, vol. 29, pp. 427 -- 438, 2000.

[18] V. Hodge and J. Austin, "A Survey of Outlier Detection Methodologies," *Artificial Intelligence Review*, vol. 22, pp. 85--126, 2004.

[19] F. E. Grubbs, "Procedures for Detecting Outlying Observations in Samples," *Technometrics*, vol. 11, pp. 1 -- 21, 1969.

[20] E. L. Lehman, *Testing statistical hypothesis*, 2 ed. Berlin: Springer, 1997.

[21] J. W. Tukey, *Exploratory data analysis*, 1 ed. Reading, MA: Addison-Wesley, 1977.

[22] I. A. Reis, G. Câmara, R. M. Assunção, and A. Monteiro, "Distributed Data-Aware Representative Clustering for Geosensor Networks Data Collection," presented at Brazilian Workshop on Real-Time and Embedded Systems (WRT 2008), 10., Rio de Janeiro, RJ, Brazil, 2008.

[23] I. A. Reis, G. Câmara, R. M. Assunção, and A. M. V. Monteiro, "Data-Aware Clustering for Geosensor Networks Data Collection," presented at Brazilian Remote Sensing Symposium, 13., Florianópolis (SC), Brazil, 2007.

[24] A. Silberstein, G. Puggioni, A. Gelfand, K. Munagala, and J. Yang, "Suppression and Failures in Sensor Networks: A Bayesian Approach," presented at Very Large Data Bases (VLDB ‘07), Vienna, Austria, 2007.

FIGURE 6

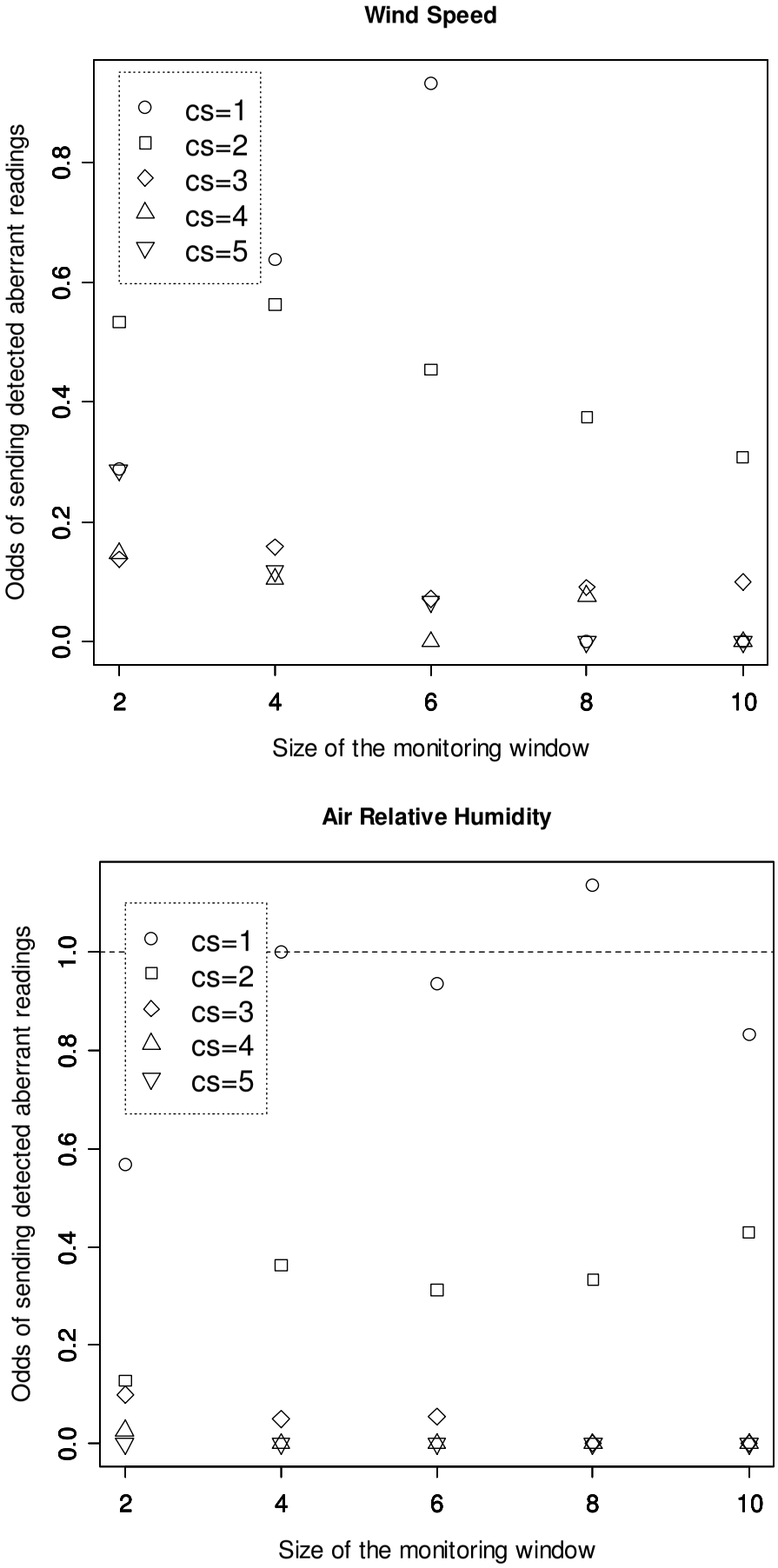
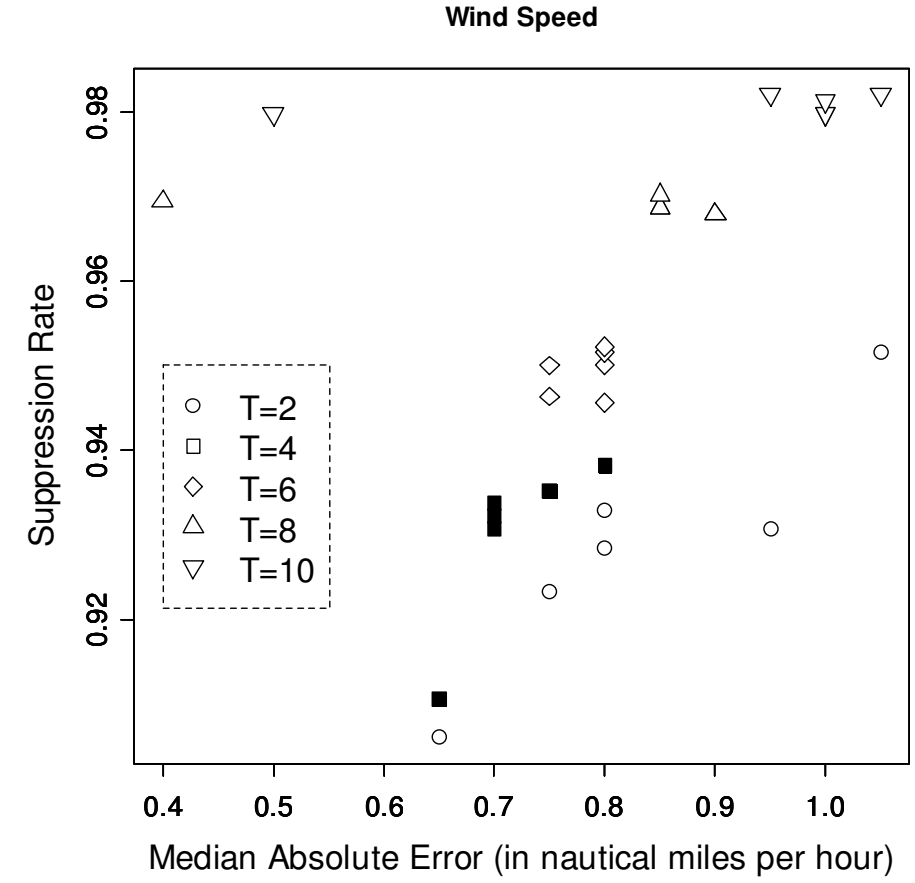

Wind Speed
cs=1
cs=2
cs=3
cs=4
cs=5
Odds of sending detected aberrant readings
0.0
0.2
0.4
0.6
0.8
2
4
6
8
10
Size of the monitoring window
Air Relative Humidity
cs=1
cs=2
cs=3
cs=4
cs=5
Odds of sending detected aberrant readings
0.0
0.2
0.4
0.6
0.8
1.0
2
4
6
8
10
Size of the monitoring window

FIGURE 7


Wind Speed
Suppression Rate
0.92
0.94
0.96
0.98
T=2
T=4
T=6
T=8
T=10
0.4
0.5
0.6
0.7
0.8
0.9
1.0
Median Absolute Error (in nautical miles per hour)

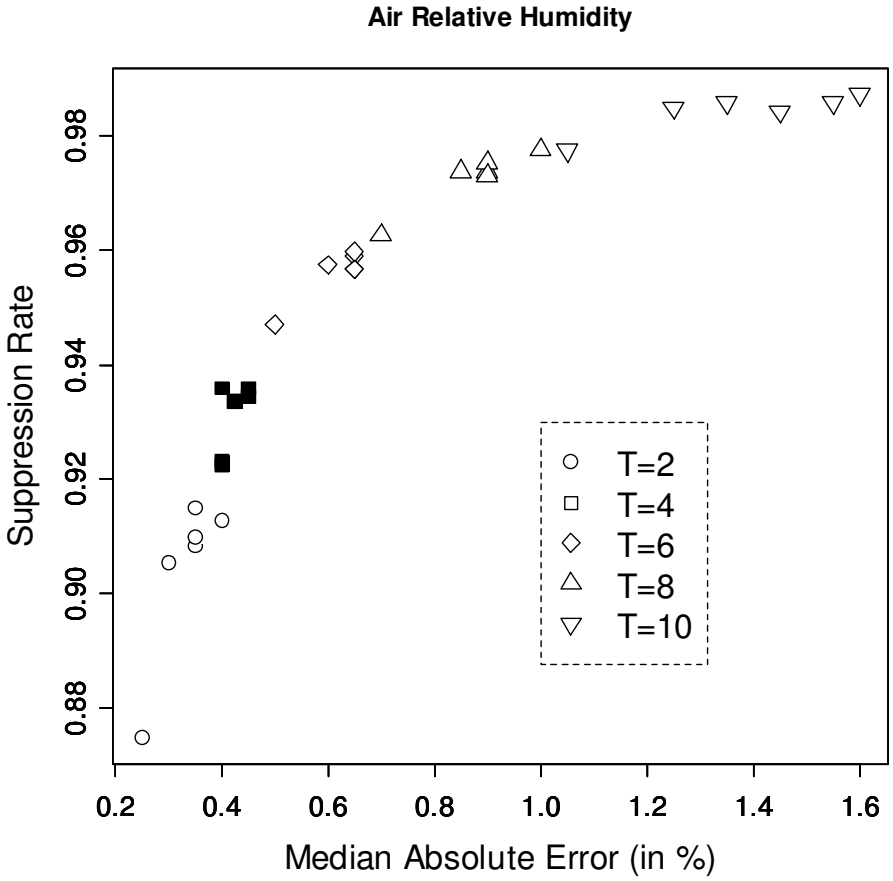
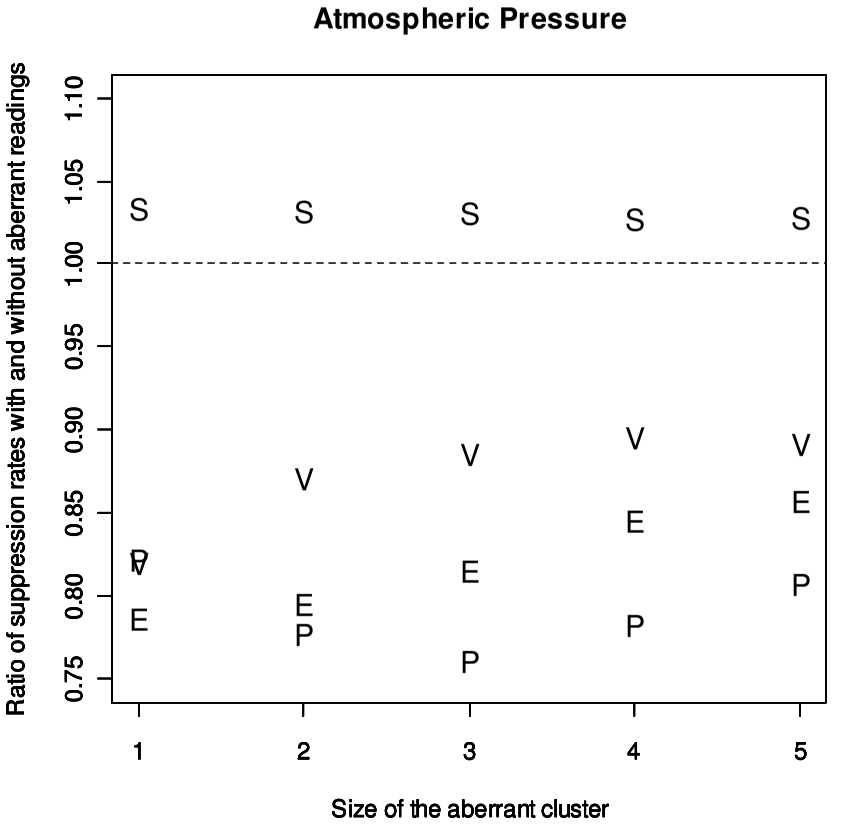

Air Relative Humidity
Suppression Rate
0.88
0.90
0.92
0.94
0.96
0.98
T=2
T=4
T=6
T=8
T=10
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
Median Absolute Error (in %)

FIGURE 9


Atmospheric Pressure
Ratio of suppression rates with and without aberrant readings
0.75
0.80
0.85
0.90
0.95
1.00
1.05
1.10
1
2
3
4
5
Size of the aberrant cluster

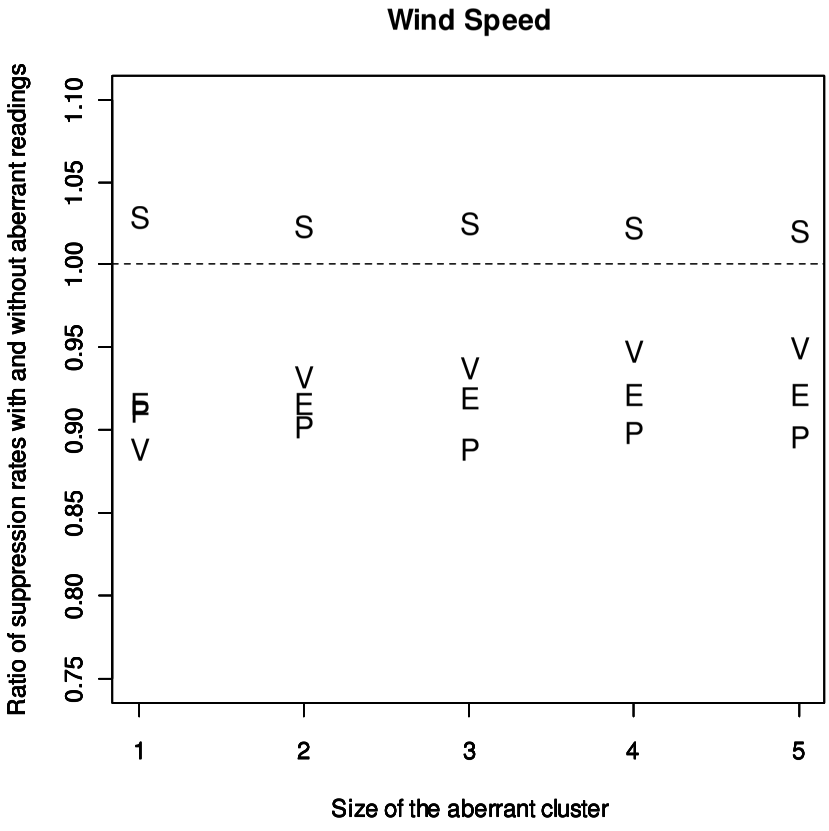
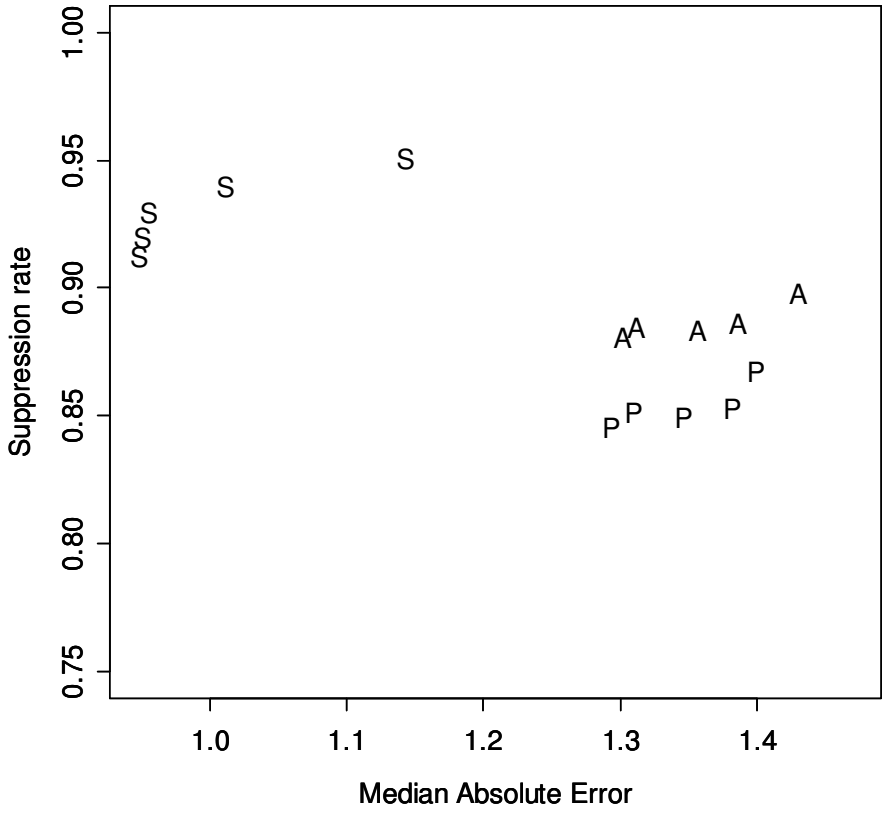
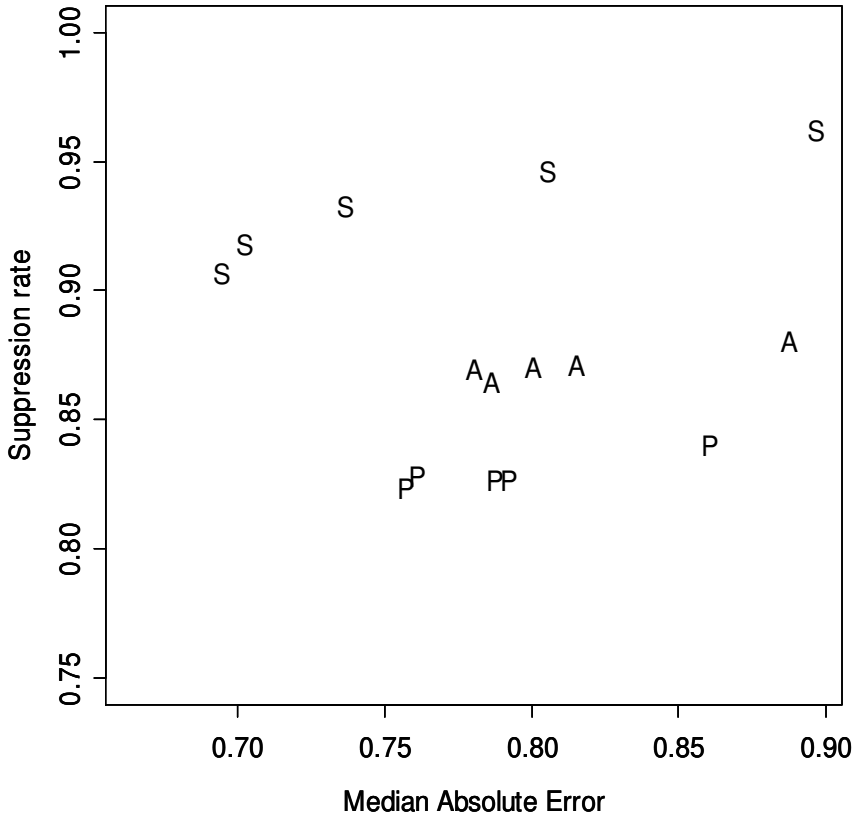

Wind Speed
Ratio of suppression rates with and without aberrant readings
0.75
0.80
0.85
0.90
0.95
1.00
1.05
1.10
1
2
3
4
5
Size of the aberrant cluster

FIGURE 10



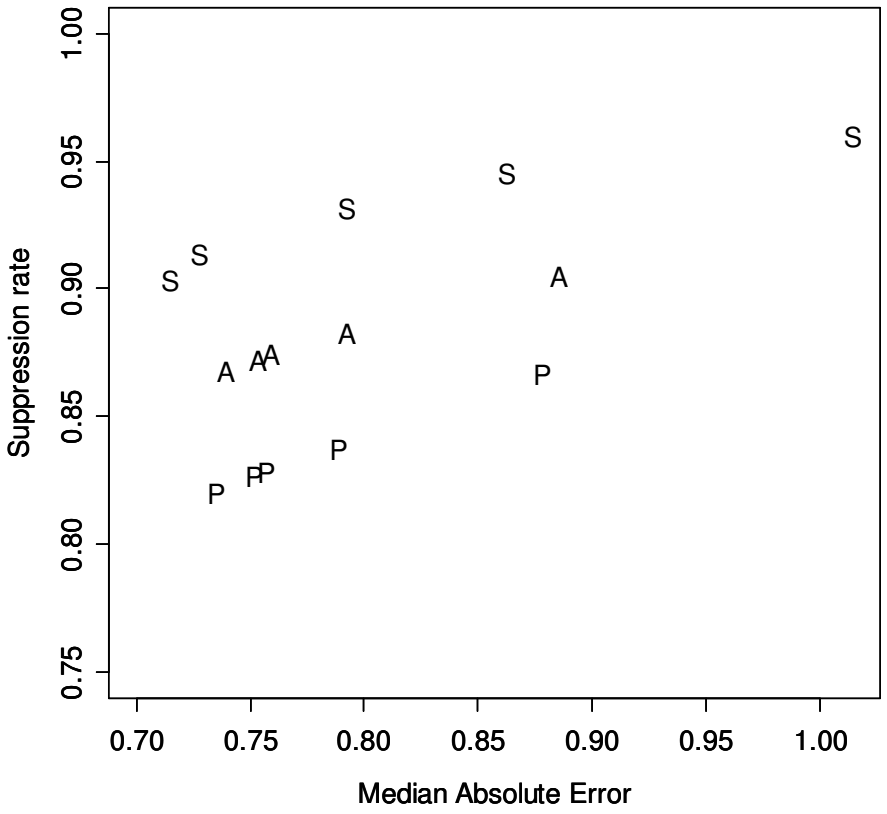
(a) $0 < \mathrm{RLD}_{10} < 0.025$

FIGURE 10



(b) $0.025 \leqslant \mathrm{RLD}_{10} < 0.050$

FIGURE 10



(c) $0.05 \leqslant \mathrm{RLD}_{10} < 0.075$

BIOGRAPHICAL NOTES

Ilka Afonso Reis has a first degree and a MsC. in Statistics from the Universidade Federal de Minas Gerais - UFMG, Brazil. Recently, she has got her doctor degree in Remote Sensing by the Program of the National Institute for Space Research - INPE, Brazil. She also is an Assistant Professor of the Department of Statistics of the Universidade Federal de Minas Gerais -UFMG, Brazil. Her current research focuses on methods for data collection using geosensor networks.

Giberto Câmara is the General Director of the National Institute for Space Research - INPE, Brazil. He teaches and advises students in INPE's graduate programs in Remote Sensing/GIS and Computer Science. His research interests include: Geographic Information Science and Engineering, Spatio-Temporal Databases, Spatial Analysis, Environmental Modelling, and Scientific and Technological Policy. He is one of the leaders of R&D in GIS and Image Processing in Brazil, notably in the development of SPRING, a free object-oriented GIS, and TerraLib, an open source GIS library. He has published more than 120 full papers on refereed journals and scientific conferences. According to Google Scholar, his h-index is 18 and the number of citations is 1985, as of July 2008. He serves on the Scientific Steering Committee of Global Land Project and on the editorial board of the Journal of Earth Science Informatics.

Renato Martins Assunção is an Associate Professor of the Department of Statistics of the Universidade Federal de Minas Gerais - UFMG, Brazil. He is the head of the Laboratory of the Spatial Statistics (LESTE-UFMG) and one of the coordinators of the Center for Crime and Public Safety Studies (CRISP-UFMG).

Antonio Miguel Vieira Monteiro has a first degree in Electrical Engineering from the Federal University of Espirito Santo, a MsC. in Applied Computer Science from the National Institute for Space Research (INPE – Brazil) and a DPhil in Electronic Engineering and Control/Computer Science at the Space Science Centre, School of Engineering and Applied Sciences, University of Sussex at Brighton, obtained in October, 1993. Miguel has been head of INPE´s Image Processing Division since November 1999. He was the manager of the SPRING Project

(www.dpi.inpe.br/spring) during its development (1998-2000) and he is currently the manager of the TerraLib Open Source Project (www.terralib.org). SPRING and TerraLib are a significant effort on Open Source and Public technologic development in Brazil. The SPRING GIS and the TerraLib Geographical Components Library are available on the Internet. Miguel's work aims at building sensitive geographic indicators and discussing the use of geotechnologies and spatial analysis methods applied to urban and public health problems. He has been focusing on matters involving territorial studies of social inequalities and social-spatial segregation on metropolitan areas.