

# Exploring Spatial Correlation for Link Quality Estimation in Wireless Sensor Networks

Yingqi Xu    Wang-Chien Lee

The Pennsylvania State University, University Park, PA 16802, USA.

{yixu, wlee}@cse.psu.edu

## Abstract

*The irregularity in quality of wireless communication links poses significant research challenges in wireless sensor network design. Dynamic network conditions and environmental factors make an on-line, self-adapted link quality estimation mechanism within sensor nodes a necessity for making routing decisions and improving network performance. In this paper, we present a weighted regression algorithm for efficient and accurate estimation of link quality in wireless sensor networks. This algorithm captures the spatial correlation in quality of links between a sensor node and its neighbor nodes, such that the quality of a link to a neighbor node can be estimated based on the quality of links to other nodes geographically close. We evaluate the proposed algorithm using a trace-based simulator which takes into account the variances of link quality over time and spatial locations. The experimental results show that the weighted regression algorithm is able to achieve more accurate estimates than WMEWMA, a state-of-the-art link quality estimator, at a much lower communication cost.*

## I. Introduction

Recent studies have shown that the quality of communication links has a significant impact on the performance of wireless sensor networks, including network lifetime, network throughput, resource usage and reliability [16], [18], [23], [25]<sup>1</sup>. Due to a number of factors (e.g., the varying environments where sensor networks are deployed, unreliable wireless communications, and irregular radio patterns), link quality among sensor nodes varies dramatically over time and location. Existing protocols proposed for wireless sensor networks (e.g., [2], [10], [11], [19], [21], [22]) are mostly designed based on an ideal spherical pattern of wireless links, which are not close to reality and hence result in non-ideal performance and less accurate research results. The performance of many routing protocols may be improved by excluding low-quality wireless links in their routing. However, sensor nodes need to know about the quality and availability of communication links in order to make

<sup>1</sup>For the rest of this paper, we refer to quality of communication links as *link quality* in short.

right routing decisions. [16] has shown that link quality should be considered as a critical dimension in the design space for wireless sensor networks.

Although there exists some similarity between mobile ad hoc networks (MANETs) and wireless sensor networks, the existing approaches proposed for evaluating the link quality of nodes in MANETs are not applicable to wireless sensor networks due to different operational conditions such as network density, radio components, power supplies, and etc. Moreover, techniques developed for sensor networks typically have to be transmission and energy efficient, since most of the wireless sensor nodes have a very strict energy budget. Furthermore, with a wide range of deployment environments, different operational conditions, and various types of radio frequency and modulation schemes, a fixed model of link quality may not be applicable to all kinds of sensor networks. Thus, an on-line, self-configuring mechanism within sensor nodes to dynamically estimate the quality and the pattern of their links is necessary.

There is an increasing amount of research effort on measuring wireless link quality and patterns [3], [4], [26] and on exploiting the impact of link quality on the performance of sensor networks [6], [23], [25]. These research results show that the communication range of a sensor node varies temporally and spatially. These variations have a major impact on the data acquisition, packet delivery and reliability of the network infrastructure, and hence affect the network performance significantly. In order to incorporate the awareness of link quality in the design and operation of wireless sensor network, research projects such as [17] and [18] have modeled the pattern of link quality changes over *time*. They observe that the link quality between a pair of sensor nodes has a statistical relationship to time. By tracking the packets heard/overheard, a sensor node is able to extract the trend of link quality changes over time and use it to estimate future link quality. Since estimators in these works measure the link quality purely based on the history of packets received in a node, they require very frequent updates from all neighbor sensor nodes to ensure the agility of the estimators to the variances in link quality. For instance, the temporal-based link quality estimators in [17] and [18] require a message exchange

every 0.5 minute. This results in noticeably heavy traffic and tremendous overhead on energy consumption. Moreover, this situation quickly worsens when network density increases and may lead to other networking problems, such as transmission collisions, network congestion, low throughput, and high packet transmission delay. This is a crucial issue since high node density widely exists in many sensor networks and applications. Therefore, we see a need to develop a new link quality estimator that meets the goal of estimating quality of links to other sensor nodes without degrading network performance.

This paper studies the link quality of wireless sensor nodes by exploiting the *spatial correlation* in links. The intuition behind spatial correlation is that sensor nodes geographically close to each other may have correlated link quality (more detail about spatial correlation is presented shortly in Section III-A.). The spatial correlation in link quality is observed in [23], but has not been well exploited. We show that the spatial correlation in link quality of neighbor sensor nodes can be captured to estimate the link quality with substantially less transmission cost than the link quality estimators based on temporal correlation. The history information of link quality for one node may be used for estimating not only its own link quality but also that of other neighbor sensor nodes geographically close.

In this paper, we propose a novel approach, based on a *weighted regression algorithm*, to allow each sensor node to capture the spatial correlation in the quality of its links. By categorizing the links into classes in accordance with their quality ranges and then employing a separate regression model for each class, the link quality at a given geographical point can be very accurately estimated. This research work presents several important contributions:

- This is the first study, to the best of the authors' knowledge, proposing to exploit the spatial correlation among wireless links in estimating the link quality in wireless sensor networks. This research result has a significant impact on the design of wireless sensor networks and their routing protocols.
- A weighted regression algorithm is developed to model the spatial distribution of link quality for a sensor node. The resilience of the proposed algorithm to the irregularity in link quality is improved by categorizing link quality for a sensor node into classes, and applying a regression model for each class.
- Determining the weight of regression models from different classes for estimating the link quality at a given location is a non-trivial research issues. An efficient solution is developed by transforming the sensor nodes in a two-dimensional space into a new feature space (i.e., a three-dimensional space) in which classes become linearly separable.
- A trace-based simulator is developed for the evaluation of our proposal and for a comparison with WMEWMA [17] which is the best of the state-of-the-art techniques for link quality estimation.

An extensive evaluation shows that the proposed weighted regression algorithm is efficient, accurate, and adaptive to the link quality variances over time and location.

The remainder of the paper is organized as follows. An overview of related work is provided in Section II. The details of the proposed weighted regression model are presented in Section III. A performance evaluation is provided in Section IV. Finally, we conclude our work and discuss future directions in Section V.

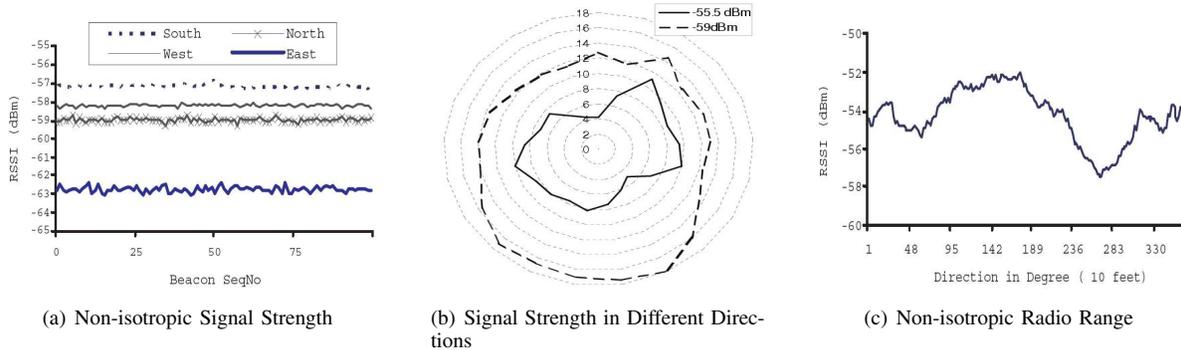
## II. Related Work

Our work is inspired and motivated by a number of research efforts. Due to the space limitation, we only briefly review the work closely related to our proposal.

In networks with abundant resources, such as the Internet, link quality is usually estimated by sending passive probing packets using Internal Gateway Routing Protocol. Link quality is measured as the ratio of arrived packets to the expected packets. However, in such networks, each packet is transmitted along a particular link, which is known a priori. Packets in wireless networks, sent via a broadcast medium, can easily get lost due to different environmental and network factors. Thus, the pattern of link quality in wireless sensor networks is expected to be very different from the Internet.

The wireless local area network (e.g., IEEE 802.11) and sensor networks share similar qualitative communication patterns [1]. The link qualities of wireless ad hoc and local networks are studied in [5], [14], which demonstrate that shortest path routing may not yield a satisfactory performance, due to the variance of communication links. However, these work have not explored the research issues in estimating link quality that varies over time and locations, especially given the unique characteristics of sensor networks (e.g., high node deployment density) and restrictions on energy.

Links with high packet loss and link asymmetry have been studied in [6], [24], which focus on the link layer and the MAC layer. The pattern of link quality and behavior of several link quality estimators for sensor networks are studied in [17], [18]. These works explore the temporal correlation in the link quality to each individual neighbor sensor node based on a sequence of packets that a sensor node hears/overhears from the monitored sensor node over time. In their study, link quality is defined as: packets received in  $t$  / max(packets expected to be received in  $t$ , packets received in  $t$ ), where  $t$  is a time window. Thus, for an estimator to measure the link quality, a minimum rate for message exchange between neighbor sensor nodes is required (e.g., [17], [18] require a message exchange every 30 seconds). However, the number of messages generated for link quality estimation has a direct impact on the expected lifetime of the networks. For instance, a typical sensor network may last for about one month if each sensor node broadcasts a beacon message every 30 seconds, but the network lifetime may be increased to more than ten months if each sensor node broadcasts a



**Fig. 1. Empirical Study of Link Quality [25]**

beacon every 10 minutes [8]. Moreover, frequent message exchanges may cause an upsurge in network traffic when node density is increased, which may further result in a number of problems in the network, e.g., lower transmission throughput, network delay, unreliability, and etc.

Predictors or estimators are widely used in various systems and applications, including wireless sensor networks. [15] points out that sensor networks exhibit significant correlated variations in computing and communication workloads. Thus, the energy usage can be optimized by setting the voltage accordingly, if the workload of individual tasks can be predicted. [7], [20] predict the future movements of mobile objects, to reduce the network communications for reporting sensor readings to a base station. [9] proposes a localization algorithm for mobile sensor networks, which predicts the location of mobile sensor nodes based on their previous locations and the maximum velocity.

### III. The Link Quality Estimator

We first introduce the assumptions used by our work in Section III-A. A regression model for capturing the spatial correlation in link quality and the weighted regression algorithm are presented in Section III-B and Section III-C, respectively.

#### A. Preliminaries

Prior studies on link quality and radio irregularity of sensor nodes motivate our work [25]. These empirical studies, conducted on the Berkeley motes platform, show that the transmission range, instead of being an ideal spherical pattern, varies in different directions (shown by Figure 1(a)). More importantly, we observe that received signal strength exhibits a continuous change with incremental changes in direction (shown by Figure 1(b) and Figure 1(c)). This can be explained from two aspects: 1) Radio strength is irregular in nature. As shown in Figure 1(a), the RSSI (*received signal strength indicator*) in the South is much higher than that in the East, although receivers have the same distance from the sender. 2) Environmental/network factors usually affect the link quality in a geographical *region* rather than at individual geographical points. For instance, when an obstacle moves into the communication range of a sensor node, called the *source node*, it's very likely that all the links that have the obstacle on their way to the source

node yield highly correlated link quality because they are weakened at the same time. This spatial correlation is also observed by [18], [23].

For clarity of the presentation, this paper considers a sensor network in a two-dimensional space only. However the proposed approach can be easily extended to three- (or even higher) dimensional space with minor changes. Sensor nodes are assumed to be location-aware. We do not constrain the link quality to a particular definition, such that it can refer to the link connectivity, the received signal strength, or other metrics, but assume that these metrics are numerically measurable. Before discussing the detailed design of our proposal, we want to point out that our approach can be used independently or can be used together with the existing temporal-based link quality estimators for further optimizing network performance.

#### B. Regression Model

Regression is a classic technique widely used for pattern recognition, classification and prediction. A basic functionality of regression techniques is to model the relationship between an *input variable* and a *response variable*, such that the future response variable can be estimated by a regression model and the corresponding input variable. It is reasonable to apply regression techniques in our problem to capture the relationship between the link quality and geographical locations, i.e., the spatial correlation in link quality. As such, the link quality at a given location can be estimated based on the established regression model.

A source node collects link quality  $p_1, p_2, \dots, p_m$  (i.e., the response variables) from a set of sensor nodes that are located at  $s_1, s_2, \dots, s_m$ , to derive a model  $f(\mathbf{s})$ <sup>2</sup>. This regression model captures the relationship between the link quality  $\mathbf{p} = (p_1, p_2, \dots, p_m)$  and the spatial location  $\mathbf{s} = (s_1, s_2, \dots, s_m)$ , and thus can be used to estimate link quality  $\hat{p}_i$  at a given spatial location  $s_i$ . In general, a regression function can be represented as  $\hat{\mathbf{p}} = f(\mathbf{s}) = \beta_0\Phi_0(\mathbf{s}) + \beta_1\Phi_1(\mathbf{s}) + \dots + \beta_k\Phi_k(\mathbf{s})$ , where  $\beta_0, \beta_1, \dots, \beta_k$  are regression coefficients and  $\Phi_0(\mathbf{s}), \Phi_1(\mathbf{s}), \dots, \Phi_k(\mathbf{s})$  are functions of  $\mathbf{s}$  that do not contain linear coefficients. For instance, consider a univariate regression model for  $\mathbf{s}$  in a one-dimensional space. Let  $\Phi_0(\mathbf{s}) = 1, \Phi_1(\mathbf{s}) = \mathbf{s}, \dots, \Phi_k(\mathbf{s}) = \mathbf{s}^k$ . Then,  $\hat{\mathbf{p}} = f(\mathbf{s}) = \beta_0 + \beta_1\mathbf{s} + \dots + \beta_k\mathbf{s}^k$ . By sampling  $m$  responses (i.e., readings  $p_1, p_2, \dots, p_m$ ) at

<sup>2</sup>The regression model can be expressed as a function

$s_1, s_2, \dots, s_m$  and minimizing the error between estimated responses (i.e.,  $\hat{p}_1, \hat{p}_2, \dots, \hat{p}_m$ ) and the real readings, we can derive the regression coefficients  $\beta_0, \beta_1, \dots, \beta_k$ . The regression model will be dynamically adapted as more responses/readings are collected to maintain its freshness in capturing the relationship between input and response variables.

For our application, considering the two-dimensional space where a sensor network is deployed:  $\mathbf{s} = (\langle \mathbf{x}, \mathbf{y} \rangle) = (\langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle, \dots, \langle x_m, y_m \rangle)$ ,  $\hat{\mathbf{p}} = f(\mathbf{x}, \mathbf{y}) = \sum_{i=0}^k \beta_i \Phi_i(\mathbf{x}, \mathbf{y})$ . In order to evaluate accuracy of the estimation  $\hat{\mathbf{p}} = f(\mathbf{x}, \mathbf{y})$ , a metric for errors is needed. We choose *root mean squared error* (RMSE) because it is more sensitive to the changes in the relationship between the input and response variables than the other metrics (e.g., the mean absolute error (MAE), or the mean absolute percentage error (MAPE)). RMSE is defined as:

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m (p_i - \hat{p}_i)^2} = \sqrt{\frac{1}{m} \sum_{i=1}^m \left( p_i - \sum_{j=0}^k \beta_j \Phi_j(\mathbf{s}_i) \right)^2}$$

Thus, the problem we have (i.e., derive the regression model) has been reduced to a problem of deriving appropriate regression coefficients,  $\beta = (\beta_0, \beta_1, \dots, \beta_k)$  to minimize the RMSE. This problem can be easily described and solved by matrix manipulation. The regression function can be written more compactly as

$$\hat{\mathbf{p}} = \Phi(\mathbf{s})\beta^T = \Phi(\mathbf{x}, \mathbf{y})\beta^T$$

where  $\Phi(\mathbf{s}) = \Phi(\mathbf{x}, \mathbf{y})$  is a  $m \times (k+1)$  matrix:

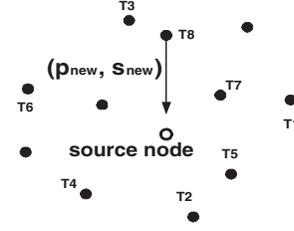
$$\begin{pmatrix} \Phi_0(x_1, y_1) & \Phi_1(x_1, y_1) & \Phi_2(x_1, y_1) & \dots & \Phi_k(x_1, y_1) \\ \Phi_0(x_2, y_2) & \Phi_1(x_2, y_2) & \Phi_2(x_2, y_2) & \dots & \Phi_k(x_2, y_2) \\ \dots & \dots & \dots & \dots & \dots \\ \Phi_0(x_m, y_m) & \Phi_1(x_m, y_m) & \Phi_2(x_m, y_m) & \dots & \Phi_k(x_m, y_m) \end{pmatrix}$$

This minimization requires no gradient descent, and thus can be obtained directly by

$$\begin{aligned} \beta &= \left( \Phi(\mathbf{s})^T \Phi(\mathbf{s}) \right)^{-1} \Phi(\mathbf{s})^T \mathbf{p} \\ &= \left( \Phi(\mathbf{x}, \mathbf{y})^T \Phi(\mathbf{x}, \mathbf{y}) \right)^{-1} \Phi(\mathbf{x}, \mathbf{y})^T \mathbf{p} \end{aligned} \quad (1)$$

where  $\Phi(\mathbf{x}, \mathbf{y})^T \Phi(\mathbf{x}, \mathbf{y})$  is a  $(k+1) \times (k+1)$  matrix with  $(\Phi(\mathbf{x}, \mathbf{y})^T \Phi(\mathbf{x}, \mathbf{y}))_{ij} = \sum_{l=1}^m \Phi_i(x_l, y_l) \Phi_j(x_l, y_l)$ , and  $\Phi(\mathbf{x}, \mathbf{y})^T \mathbf{p}$  is a  $(k+1) \times 1$  matrix with  $(\Phi(\mathbf{x}, \mathbf{y})^T \mathbf{p})_i = \sum_{l=1}^m \Phi_i(x_l, y_l) p_l$ .

To collect inputs for regression modeling, the source node continues to hear/overhear packets which contain the measurement of communication link quality  $\mathbf{p}$  from neighbor sensor nodes. The new arrival of link quality is dealt with for real-time update of the regression function, while the old link quality may become obsolete and have to be discarded from the model. To ensure that only some most recently occurring data is processed, our work adopt the idea of *sliding window*, which is widely used in data stream analysis [12], [13]. The basic idea of sliding window is that the data falling out of the window should be discarded. In other words, the criteria



**Fig. 2. Criteria Used for Discarding a Data from the Sliding Window**

used for discarding data is defined by the time stamps associated with each data, i.e., data expires after exactly  $t$  time units or after  $v$  new data arrive, where  $t$  or  $v$  is the user-specified window size. However, observing the spatial correlation in the link quality, we argue the criteria used for discarding old data should take both spatial property and temporal property into consideration. Figure 2 shows an example. When a source node at time stamp  $T_8$  receives a new link quality from a neighbor node and window is full, old link quality reported at  $T_1$  should be discarded if the criteria of discard considers only the temporal condition. However, we see that it is more reasonable to replace the old link quality reported at  $T_3$  by a neighbor node that is geographically close to the node that reports at  $T_8$ , instead of the oldest data reported at  $T_1$ , because 1) if the link quality reported at  $T_3$  and  $T_8$  are similar, it is unnecessary to keep two similar link quality located close to each other; 2) if the link quality reported at  $T_3$  and  $T_8$  are different, it is more likely that the link quality in this region where both neighbor nodes reside changes, which implies that the old link quality reported at  $T_3$  becomes obsolete and should be discarded.

Therefore, this criteria is defined as follows. Let  $v$  data denote the window size,  $\mathbf{W}$  denote the set of data falling inside the window, and  $(p_{new}, s_{new})$  denote the newly arrived data  $p_{new}$  at  $s_{new}$ . We look for a  $(p_{old}, s_{old}) \in \mathbf{W}$  which is nearest to  $s_{new}$ , i.e.,  $\{d(s_{new}, s_{old}) \leq d(s_{new}, s_j) \mid \forall j, s_j \in \mathbf{W}\}$ , where  $d(s_i, s_j) (= \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2})$  denotes the distance between  $s_i$  and  $s_j$ . We define a distance threshold  $\xi$ , such that  $(p_{old}, s_{old})$  is discarded when  $d(s_{new}, s_{old}) \leq \xi$ . If no  $(p_{old}, s_{old}) \in \mathbf{W}$  can be found to satisfy the criteria, we can again consider the criteria defined over the temporal conditions, i.e., the  $(p, s) \in \mathbf{W}$  with the smallest time stamp is discarded. As  $\xi$  is the spatial threshold for discarding a data inside the sliding window, the value of  $\xi$  should be determined based on how strong the spatial correlation in link quality of sensor nodes. The strong spatial correlation allows a larger value of  $\xi$ . Otherwise,  $\xi$  should be kept small. The setting and the impact of threshold  $\xi$  will be studied in our future work. Each time,  $\Phi(\mathbf{s})$  and  $\mathbf{p}$  are updated by removing the old data and meanwhile by appending the new data  $(p_{new}, s_{new})$ . With updated  $\Phi(\mathbf{s})$  and  $\mathbf{p}$ , the source node recomputes the regression coefficients  $\beta$  based on Equation (1)<sup>3</sup>.

<sup>3</sup>With the constraint of window size  $v$ , the total number of input variables and of response variables fed into the regression model are  $v$ , instead of  $m$ , i.e.,  $(s_1, s_2, \dots, s_v)$  and  $(p_1, p_2, \dots, p_v)$ . In calculation of regression coefficients  $\beta$ , Equation (1) remains same.

Moreover, to reduce the computation overhead caused by the frequent arrival of new data, the source node can pre-define an update threshold. For instance, only when  $(p_i - \hat{p}_i)^2$  exceeds certain threshold, the source node recomputes  $\beta$ , otherwise, only updates  $\Phi(\mathbf{s})$  and  $\mathbf{p}$ .

### C. Weighted Regression Algorithm

The above section describes how to form a regression model for capturing the spatial correlation in link quality of sensor nodes in wireless sensor networks. However, in our study, we notice that the link quality from neighbor nodes changes continuously, and the irregularity exists in the link quality as well. To precisely model this spatial correlation, a more complicated regression model (e.g., a high-degree regression model) may be required to minimize RMSE. However, a high-degree regression model would increase the computation overhead drastically, and thus is not appropriate for resource-restrained sensor network. To solve this problem, we use multiple low-degree regression models and estimate link quality by giving different weights to the different regression models, because using multiple low-degree regression models only increase the computation overhead linearly. We plan to quantitatively examine the computation cost of different regression models in our future work.

Given all the collected data falling inside the window  $(p_j, \langle x_j, y_j \rangle)$ , where  $1 \leq j \leq v$ , a source node uniformly categorizes these data pairs into  $c$  classes. A class  $C^{(i)}$  ( $1 \leq i \leq c \leq v$ ) is defined by an upper bound (denoted by  $U^{(i)}$ ) and a lower bound (denoted by  $L^{(i)}$ ), such that  $C^{(i)} = \{(p_j, \langle x_j, y_j \rangle) | L^{(i)} \leq p_j < U^{(i)}\}$ . Given all data pairs in class  $C^{(i)}$ , the source node derives the regression coefficients  $\beta^{(i)}$  and the regression function  $f^{(i)}$  for class  $C^{(i)}$ . Now, given a group of regression functions and a location point  $\langle x, y \rangle$ , we study how to estimate the link quality at  $\langle x, y \rangle$ . Due to the irregularity in link quality, it may not be clear which regression function should be used for estimation. In other words, the link quality of a geographical location may be affected by multiple regression functions, which are all close to the point. Thus, we propose a *weighted regression* algorithm that considers the impact of multiple regression functions on a location point. Generally, the link quality  $p$  at location point  $\langle x, y \rangle$  can be estimated by the following equation:

$$\hat{p} = w^{(1)} f^{(1)}(x, y) + w^{(2)} f^{(2)}(x, y) + \dots + w^{(c)} f^{(c)}(x, y)$$

where  $w^{(i)}$  is the weight of the estimate from regression function  $f^{(i)}$ , which will be accounted to the estimated link quality  $\hat{p}$ . Intuitively,  $w^{(i)}$  should reflect the spatial closeness between  $\langle x, y \rangle$  and class  $C^{(i)}$ . However, the measurement of this closeness is not straightforward. More specifically, there are two research challenges: 1) there are more than one data point in each class, such that the distance between  $\langle x, y \rangle$  and any single data point  $s$  in a class cannot represent the closeness of  $\langle x, y \rangle$  to this class; 2) the data points in each class geographically scatter around a source node (shown by Figure 1(b)). Thus, the existing approach of enclosing all geographical

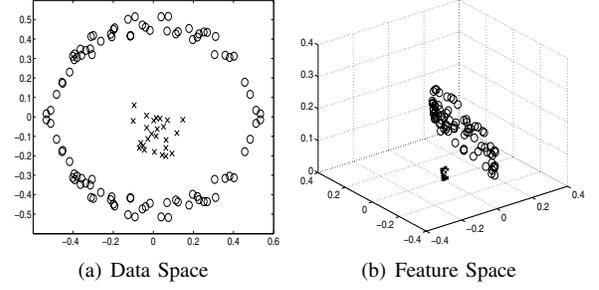


Fig. 3. Feature Space Transformation

points by a geographical region and then measuring the closeness as the distance between  $\langle x, y \rangle$  and the center of this geographical region is not applicable.

In order to solve the above two research challenges, we identify a solution, called the *feature space transformation*, which is widely used in clustering and classifications. The goal of using the feature space transformation is to overcome the second research challenge discussed above that the data points in different classes cannot be separated linearly. Feature space transformation employs a nonlinear mapping  $\Psi$  (shown below) to transform the input variables  $\mathbf{e} = (e_1, e_2, \dots) \in \mathbb{R}^N$ , which are nonlinearly separable in nature, into a feature space  $\mathcal{F}$ , where the data points become linearly separable<sup>4</sup>.

$$\begin{aligned} \Psi : \quad \mathbb{R}^N &\rightarrow \mathcal{F} \\ \mathbf{e} &\mapsto \Psi(\mathbf{e}) \end{aligned}$$

For instance, a set of data points scattered in a two-dimensional space  $\mathbb{R}^2$  (shown by Figure 3(a)) can be mapped into a three-dimensional space  $\mathbb{R}^3$  (shown by Figure 3(b)) using:

$$\begin{aligned} \Psi : \quad \mathbb{R}^2 &\rightarrow \mathbb{R}^3 \\ \langle x, y \rangle &\mapsto \langle z_1, z_2, z_3 \rangle = \langle x^2, \sqrt{2}xy, y^2 \rangle \end{aligned}$$

To measure the closeness of the given geographical point  $\langle x, y \rangle$  to each class in a feature space, we derive a regression surface, denoted by  $g^{(i)}$ , for each class  $C^{(i)}$  in the feature space. Without loss of generality,

$$g^{(i)} = a^{(i)} z_1 + b^{(i)} z_2 + c^{(i)} z_3 + d^{(i)}$$

where  $a^{(i)}$ ,  $b^{(i)}$ ,  $c^{(i)}$  and  $d^{(i)}$  are the regression coefficients of the regression surface for class  $i$  in the feature space. Thus, the distance between  $\langle x, y \rangle$  and the class  $C^{(i)}$  is evaluated as the spatial distance between  $\langle x, y \rangle$  and the regression surface  $g^{(i)}$  that models the transformed data for class  $C^{(i)}$  in feature space

$$\begin{aligned} D^{(i)} &= \left| \frac{a^{(i)} z_1 + b^{(i)} z_2 + c^{(i)} z_3 + d^{(i)}}{\sqrt{a^{(i)2} + b^{(i)2} + c^{(i)2}}} \right| \\ &= \left| \frac{a^{(i)} x^2 + b^{(i)} \sqrt{2}xy + c^{(i)} y^2 + d^{(i)}}{\sqrt{a^{(i)2} + b^{(i)2} + c^{(i)2}}} \right| \end{aligned}$$

where  $z_1 = x^2$ ,  $z_2 = \sqrt{2}xy$ ,  $z_3 = y^2$ . It is reasonable to consider that the class closer to the geographical point <

<sup>4</sup>Other mapping functions can be employed. The choice of mapping functions are not expected to affect the performance of estimation of link quality, as long as the transformed classes are linearly separable.

$x, y >$  should contribute more to the estimate at  $\langle x, y \rangle$ , and thus have a larger weight  $w$ . A Gaussian function is used to calculate the weights for class  $C^{(i)}$  as

$$w^{(i)} = \exp\left(-\frac{D^{(i)2}}{\sigma}\right) \\ = \exp\left(-\frac{(a^{(i)}z_1 + b^{(i)}z_2 + c^{(i)}z_3 + d^{(i)})^2}{(a^{(i)2} + b^{(i)2} + c^{(i)2})\sigma}\right) \quad (2)$$

where parameter  $\sigma$  determines how quickly the weight declines as the distance between  $\langle x, y \rangle$  and the class increases. This Gaussian function decays with respect to the distance  $D^{(i)}$ , a property that our problem requires. Using Equation (2), we define the *normalized weight*  $\hat{w}^{(i)}$  for class  $C^{(i)}$  as

$$\hat{w}^{(i)} = \frac{w^{(i)}}{\sum_{j=1}^c w^{(j)}} \quad (3)$$

With the normalized weight, the communication link quality  $p$  at location  $\langle x, y \rangle$  is calculated by combining the weighted estimate from each class:

$$\hat{p} = \sum_{i=1}^c \hat{w}^{(i)} f^{(i)}(x, y) = \sum_{i=1}^c \hat{w}^{(i)} \sum_{j=0}^k \beta_j^{(i)} \Phi_j^{(i)}(x, y)$$

If the geographical location  $\langle x, y \rangle$  is far from class  $C^{(i)}$ ,  $w^{(i)}$  and  $\hat{w}^{(i)}$  both approach 0 and the estimate  $\hat{p}$  is not influenced by the estimate produced by class  $C^{(i)}$  (i.e.,  $p^{(i)}$ ). If  $\langle x, y \rangle$  is close to several classes, the estimate  $\hat{p}$  is the weighted average of the estimate from these classes<sup>5</sup>. The complete algorithm for modeling the spatial distribution of link quality for a source node is shown by Algorithm 1 and the algorithm for estimating the link quality of a given location using weighted regression algorithm is shown by Algorithm 2.

## IV. Performance Evaluation

In this section, we evaluate the performance of the weighted regression algorithm and compare it with a temporal-based link quality estimator, called *WMEWMA* [17]. Since the impact of link quality on the performance of network routing protocols has been well studied [4], [16], [18], [23], [25], our evaluation focuses on the performance of the proposed weighted regression algorithm in terms of its *accuracy*, *configuration latency*, and *adaptability*.

### A. Experimental Settings

This section lays out the basic experimental settings, including the the link quality dataset and network settings.

In order to evaluate the performance of link quality estimator, we generated a synthetic dataset of link quality based on the empirical traces obtained using MICA II sensor nodes [3]. We employ a simple mean function on the empirical data (as suggested in [18]) to capture the link behavior. More specifically, the collected link quality are sorted with respect to their distances to the source

<sup>5</sup>An error bound can be set for the regression surfaces. As long as the regression error of a class's regression surface is less than the error bound, which implies the distribution of link quality for this class does not change significantly, the source node does not need to regenerate the regression surface for this class.

---

### Algorithm 1 Model Spatial Distribution of Link Quality

---

```

Initialize ()
  v ← size of the sliding window
  c ← number of classes
  L(i) and U(i) ← the lower bound and the upper bound for class i
  for i = 1 to c do
    Φ(i) ← 0, w(i) ← 0, p(i) ← 0, s(i) ← 0
  end for
  for i = 1 to v do
    receive a packet from a sensor node located si = < xi, yi > with
    measurement pi
    for j = 1 to c do
      if L(j) ≤ p < U(j) then
        p(j) ← p(j) + pi
        s(j) ← s(j) + si
        break
      end if
    end for
  end for
  for i = 1 to c do
    β(i) = (Φ(i)T Φ(i))-1 Φ(i)T p(i)
  end for

Update (pnew, snew)
  {Receive a packet from a sensor node located snew = <
  xnew, ynew > with measurement pnew, and the number of data
  points in W = v}
  ξ ← distance threshold
  W ← all data points falling inside the sliding window
  dmin = MIN {d(snew, sj) | ∀ sj ∈ W}
  (pold, sold) = (pj, sj), where d(snew, sj) = dmin
  cold = j, where L(j) ≤ pold < U(j) and 1 ≤ j ≤ c
  if dmin ≤ ξ then
    p(cold) ← p(cold) - pold
    s(cold) ← s(cold) - sold
    β(cold) = (Φ(cold)T Φ(cold))-1 Φ(cold)T p(cold)
  else
    (pold, sold) = the data point with the earliest time stamp in W
    for j = 1 to c do
      if L(j) ≤ pold < U(j) then
        p(j) ← p(j) - pold
        s(j) ← s(j) - sold
        break
      end if
    end for
    β(j) = (Φ(j)T Φ(j))-1 Φ(j)T p(j)
  end if
  for j = 1 to c do
    if L(j) ≤ pnew < U(j) then
      p(j) ← p(j) + pnew
      s(j) ← s(j) + snew
      break
    end if
  end for
  β(j) = (Φ(j)T Φ(j))-1 Φ(j)T p(j)

```

---

node and are aggregated into bins which have a range of 10 meters. The mean and the standard deviation are obtained for each bin. In the simulation, link quality is assumed to follow a normal distribution given the mean and standard deviation extracted from the empirical data. [18] demonstrates that this model, matching well with the empirical observation, allows more exploration of the design space while incorporating the most important variants observed in practice.

We assume the sensor nodes are deployed within a two-dimensional space. The sensor nodes are aware of their locations (i.e.,  $\langle x, y \rangle$ ). Our proposals can be

---

**Algorithm 2** Estimating link quality using Weighted Regression Algorithm
 

---

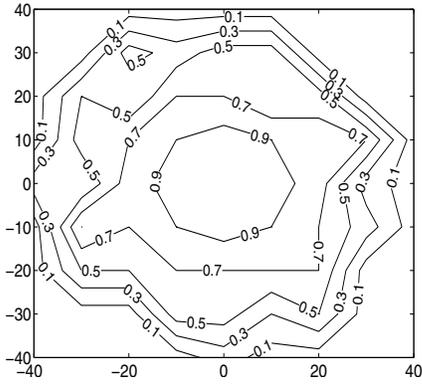
**Estimate** ( $s$ )

```

{Estimate the link quality  $p$  at location  $s = \langle x, y \rangle$ }
 $g^{(i)} = a^{(i)}z_1 + b^{(i)}z_2 + c^{(i)}z_3 + d^{(i)} \leftarrow$  regression surface for
class  $i$ 
 $\sigma \leftarrow$  weight parameter
 $w^{(i)} \leftarrow$  weight for class  $i$ 
 $\hat{w}^{(i)} \leftarrow$  normalized weight for class  $i$ 
for  $i = 1$  to  $c$  do
   $D^{(i)} = \frac{(a^{(i)}x^2 + b^{(i)}\sqrt{2}xy + c^{(i)}y^2 + d^{(i)})}{\sqrt{a^{(i)2} + b^{(i)2} + c^{(i)2}}}$ 
   $w^{(i)} = e^{-\frac{D^{(i)}}{\sigma}}$ 
end for
for  $i = 1$  to  $c$  do
   $\hat{w}^{(i)} = \frac{w^{(i)}}{\sum_{j=1}^c w^{(j)}}$ 
end for
 $p = \sum_{i=1}^c \hat{w}^{(i)} f^{(i)}(x, y)$ 

```

---


**Fig. 4. Empirical Link Quality**

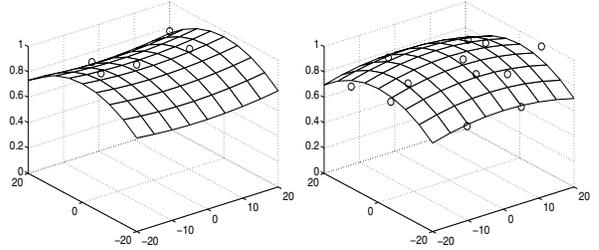
adopted for three or even higher dimensional space with minor changes. We implement all schemes under comparison in MATLAB and assume the size of sliding window in weighted regression algorithm is 40 data points.

Figure 4 shows the link quality of a source node (generated as described above) in a two dimensional space. The source node, located at  $(0,0)$  has a transmission range of 40 meters. In our simulations, the value of link quality varies between 0 and 1, which could be the link connectivity [18], the received radio signal strength [25], or other measurements of communication links. 1 implies the best link quality, while 0 implies the worst link quality. From the figure, we observe the irregularity and the spatial correlation in the link quality. This model of link quality is used in our simulation.

## B. Study of the Basic Algorithm

This section explores insights of the proposed weighted regression algorithm by examining the impact of the number of classes ( $c$ ) and the choices of the candidate regression models for proposed estimator.

We generate two trails of link quality, and derive two different regression functions for the fifth class in these two trails (shown by Figure 5(a) and Figure 5(b)). In both figures,  $x$ - and  $y$ -axis draw the two-dimensional simulation field and  $z$ -axis denotes the link quality. The circles represent the collected link quality. We observe



(a) Regression Function I      (b) Regression Function II

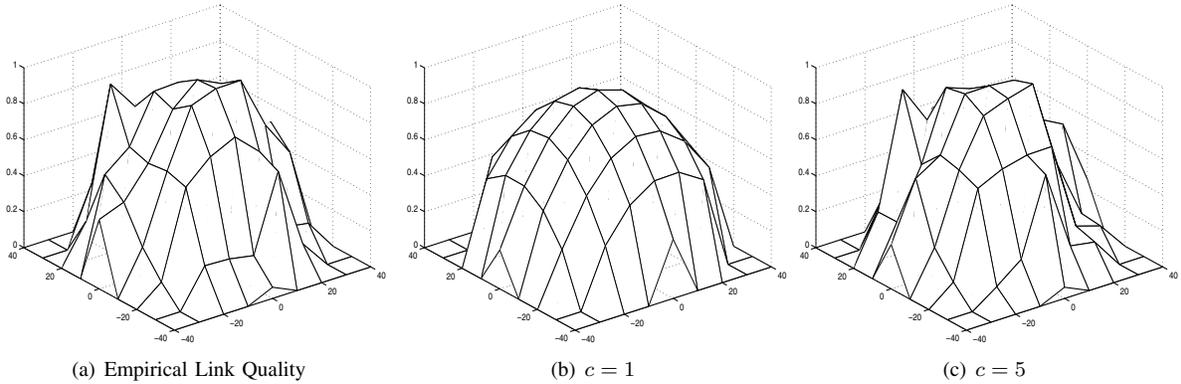
**Fig. 5. Regression Function for the Fifth Class**

that even though we use the same mean and variance in both trails, the generated dataset (i.e., link quality) are not the same.

Figure 6 shows the impact of the number of classes  $c$  on the performance of the estimator. Again in this figure,  $x$ -axis and  $y$ -axis denote the locations of sensor nodes and  $z$ -axis denotes the link quality. The link quality generated by our simulation is shown in Figure 6(a), while the link quality estimated by the weighted regression algorithm with one class and the one with five classes are shown in Figure 6(b) and Figure 6(c), respectively. We can easily see that the estimated link quality with  $c = 5$  is much closer to the link quality in dataset than the one estimated with  $c = 1$ .

To take a closer look at the impact of  $c$  on the performance of estimators, we conduct another experiment by varying  $c$  from 1 to 9 (shown in Figure 7). We consider three representative regression models: *linear regression model*, *factorial regression model* and *response surface regression model*. Linear regression model only consider the *main effect* for input variables. In other words, linear regression model assumes that input variables are independent and raised only to the first power. The linear regression model used in our model is  $p = \beta_0 + \beta_1x + \beta_2y$ . On the other hand, the factorial regression model considers the combination of two or more input variables by forming the products of these input variables. More specifically, in our experiment, the factorial regression model is  $p = \beta_0 + \beta_1x + \beta_2xy + \beta_3y$ . Response surface regression employs high-order terms for input variables by raising the values for the original input variable to a power greater than 1 and by forming products of different input variables. In our experiment, the response surface regression model used is  $p = \beta_0 + \beta_1x + \beta_2x^2 + \beta_3xy + \beta_4y + \beta_5y^2$ . To have a clear idea about how well our estimators can estimate the link quality, we examine root square mean error (RMSE) calculated from the estimated link quality and the empirical link quality. The experimental results are obtained by averaging the results over 25 runs for each estimator.

Figure 7 shows that the RMSE caused by different regression models, i.e., linear regression model, factorial regression model, and response surface regression model with a varying number of classes. We observe that linear regression model has the worst performance when fewer classes are used (i.e., RMSE is higher than 0.15 when  $c \leq 3$ ) due to its linear property in the regression function.



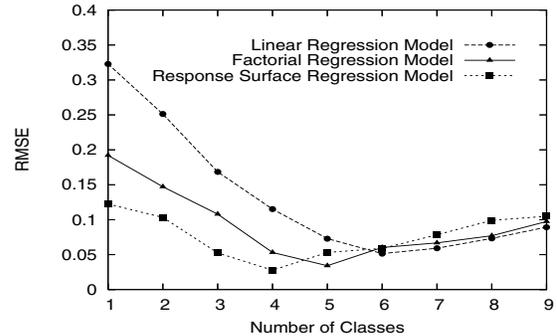
**Fig. 6. Empirical Model and Weighted Regression Models**

The performance of factorial model is closer to that of the response surface model, because the relationship between input variables are considered in both models. As we expected, RMSE for all regression models decreases when more classes are used, since a regression function can fit a class better when a class contains less data. However, this trend stops after these three regression models reach their minimal RMSE. We observe a reverse trend that RMSE for these three models increases slightly when more classes are used. We believe this is because that when more regression models are used, the weighted algorithm *overfits* the data.

In statistics, overfitting takes place when the regression function is adjusted to very specific random training data, that have no relation to the target function. In the process of overfitting, the performance on modeling the collected link quality still improves, while the performance on estimating an unknown link quality at a given location point worsens. The problem of overfitting can be solved by *cross-validation* technique, which is widely used for evaluating how well a regression model predicts. Cross-validation partition the whole input variables into subsets. The regression models are derived based on a single subset of data, while other subsets are used for confirming and validating the performance of obtained regression function. For our problem, cross-validation provides more insights of choosing an appropriate  $c$  to minimize the RMSE in estimation at a higher computation cost. However, once an appropriate  $c$  is chosen, the cross validation needs to be conducted only when the estimation performance of regression function obviously worsens, which implies a new  $c$  should be chosen. We consider applying cross-validation technique to our proposal as one of our major future work.

### C. Study of Configuration Time

In this section we examine how fast the proposed weighted regression algorithm can reach a reasonable RMSE by initializing the regression coefficients to zero. Meanwhile, we consider the same algorithm with different *broadcast interval* (denoted by  $I$ ). A source node is informed about the link quality between itself and a neighbor node every  $I$  time units. Thus, the broadcast



**Fig. 7. Weighted Regression Algorithm**

interval implies the communication cost. The shorter the broadcast interval, the more frequently a node broadcasts and the more communication cost is incurred. On the other hand, one can expect that an estimator predicts better when the interval is short since more information can be collected. In this experiment, we only use factorial regression model with  $c = 5$  classes due to its low estimation RMSE (as shown in Figure 7). We compare the performance of the weighted regression algorithm against a temporal-based link quality estimator, i.e., *window mean with exponentially weighed moving average* (WMEWMA), which has been shown in [17] to yield the best estimation performance among existing techniques.

We adopt a model from [17] to simulate the variances in link quality over time. This model has been shown to capture the essence of the empirical trace of link quality changes, except for a slightly smaller degree of variance. In this model, the link quality at a given time follows a Bernoulli distribution with  $p(t)$  probability of success and  $1 - p(t)$  probability of failure, where probability  $p(t)$  is defined as function of time. More specifically, when  $p(t)$  is fixed for a period of time, link quality is generated as a Bernoulli process with the mean of  $p(t)$ , which simulates a period of stable link quality between two nodes. In this experiment, to evaluate the latency for a regression model to configure its optimal regression coefficients, we consider a scenario with relatively stable link quality by fixing the  $p(t)$ . The adaptiveness of estimators to the varying  $p(t)$  will be studied shortly in Section IV-D.

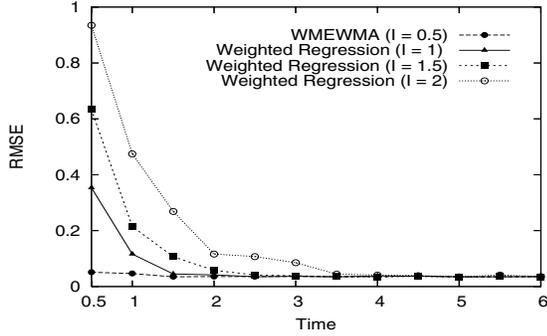


Fig. 8. Configuration Time

Figure 8 shows the configuration time of WMEWMA with  $I = 0.5\text{sec}$  (which is used in [17]) and that of the weighted regression algorithms with  $I = 1\text{sec}$ ,  $1.5\text{sec}$  and  $2\text{sec}$ , respectively. We have two observations from this experiment. First, the weighted regression algorithm demonstrates competitive performance in estimating link quality to that of WMEWMA, while achieving significant savings in communication cost by using a longer broadcaster interval  $I$  (i.e., two, three and four times longer than the broadcast interval of WMEWMA, respectively). Our second observation is that even though with less frequent updates of link quality, the weighted regression algorithm does not spend significantly longer time for configuring its regression function than WMEWMA which uses a shorter  $I$ . For instance, the configuration time of the weighted regression algorithm with  $I = 2\text{sec}$  is less than four times the configuration time of the WMEWMA with  $I = 0.5\text{sec}$ .

#### D. Study of Estimator Adaptability

This section studies the adaptability of the weighted regression algorithm to the changes in link quality over time. Again, we adopt the same link quality model described in [17] but varying value of  $p(t)$  over time. More specifically, we assume an obstacle scenario, in which an obstacle moves across the transmission region of a source node. During the obstacle's movement, all the sensor nodes behind the object suffer a deterioration in their link quality to the source node. Figure 9(a) shows an example of this scenario. The link quality between the source node and the neighbor nodes residing in the area behind the obstacle (i.e., the shaded area in Figure 9(a)) become weaker. Figure 9(b) shows an example of the link quality variances over time for a neighbor node. Therefore, in this experiment, the  $p(t)$  for each sensor node is dynamically determined based on the location of the moving obstacle. If the line between the source node and a neighbor node intersects with the obstacle, the neighbor node, considered as being blocked by the obstacle, experiences a deterioration in its link quality. For simplicity, we assume that the obstacle moves along a straight line. We do not claim that Figure 9(b) is a realistic scenario for link quality changes in the presence of a moving obstacle. However, since we are studying the adaptability of link quality estimators, the above scenario

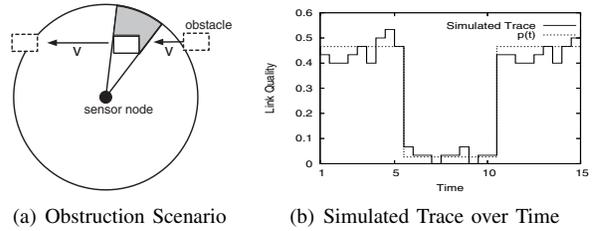


Fig. 9. Variances in Link Quality

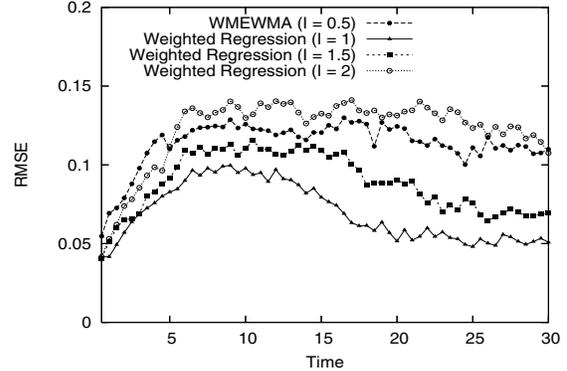


Fig. 10. Adaptability of Link Quality Estimators

satisfies the experiment requirements<sup>6</sup>.

Figure 10 shows the RMSE of WMEWMA with  $I = 0.5\text{sec}$  and RMSE of the weighted regression algorithms with  $I = 1\text{sec}$ ,  $1.5\text{sec}$  and  $2\text{sec}$ . We assume the total time that an obstacle remains inside the communication region of a source node is  $30\text{sec}$  (as shown by x-axis in the figure). We observe a quick increase in RMSE for all estimators at the beginning. WMEWMA has a faster increase trend, which is consistent with the claim in [17], i.e., WMEWMA is not agile with small variance in link quality. The RMSE of WMEWMA remains relatively constant, as it only considers the temporal correlation in the history link quality between one pair of nodes. The proposed weighted regression algorithms with  $I = 1$  and  $I = 1.5$  both outperform WMEWMA, as the estimation of link quality is not only based on the link quality from one node but also from the nodes around. Thus, the changes in link quality for one node is informative for estimating the link qualities for the other nodes around. Therefore, our estimator, even though with less communication overhead (i.e., theoretically 2 times and 3 times less than WMEWMA, as its  $I$  is 2 and 3 times longer than WMEWMA), indeed achieves more accurate estimations than WMEWMA (e.g., 7%). It is a quite significant result. The weighted regression algorithm with  $I = 2$  suffers from low-frequent updates of link quality and yields a slightly higher RSME than that of WMEWMA. However, considering the savings in communication overhead (i.e., theoretically 4 times), this result is still worth consideration.

<sup>6</sup>In fact, the empirical study in [17] shows a smoother change in the link quality in the presence of a moving obstacle, which places less difficulty on an estimator, comparing with the abrupt changes in our experiment.

## V. Conclusion and Future Work

Knowing the quality of links to neighbor nodes can help sensor nodes make the right routing decisions and significantly improve the performance of wireless sensor networks. Existing research has shown that the link quality between a pair of sensor nodes has a statistical relationship to time and used this temporal correlation for link quality estimation. In this paper, we explore the spatial correlation in link quality among a group of sensor nodes that are geographically close. We develop an efficient and accurate on-line link quality estimator based on regression models which capture and model this spatial correlation. To adapt to the irregularity existing in the link quality, we propose a weighted regression algorithm to estimate the link quality of a sensor node at any given location. In determining the weight for each regression model for estimation, the spatial distance between a node to a class of nodes scattered around a source node cannot be measured by the traditional Euclidean distance. Thus, we perform a feature space transformation to map the sensor nodes to a new feature space in order to measure the closeness of nodes to a class of nodes. To evaluate the proposed new method for link quality estimation, we generate a synthetic dataset of link quality based on the empirical traces, and simulate both variances in link quality over time and locations. Our experimental results show that, in all cases the proposed weighted regression algorithm achieves comparably more accurate estimate than WMEWMA, a state-of-the-art estimator based on temporal correlation. More significantly, this improved accuracy is achieved at a much lower communication cost than WMEWMA.

Our research result has shown that the weighted regression algorithm is a promising approach to build a link quality estimator for wireless sensor networks. As a next step, we would like to studying the choice of algorithm parameters, including the number of classes  $c$  and the window size  $v$ , by employing statistic techniques, including cross-validation technique and forward/backward selection technique. Moreover, as the link quality should be taken as a new dimension in sensor network design, we plan to explore the cross-layer design and optimization by studying the impact of the proposed algorithm on different system layers.

## VI. Acknowledgment

The authors would like to thank our shepherd Dr. Andrea Passarella of University of Cambridge, anonymous reviewers, and Dr. Gail Mitchell of BBN Technologies for their valuable comments and constructive inputs which significantly improve the quality of this paper. Wang-Chien Lee and Yingqi Xu were supported in part by National Science Foundation grant IIS-0328881.

## VII. References

- [1] G. Anastasi, E. Borgia, M. Conti, E. Gregori, and A. Passarella. Understanding the real behavior of mote and 802.11 ad hoc networks: an experimental approach. *Pervasive and Mobile Computing*, 1(2):237–256, 2005.
- [2] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 7(5):28–34, Oct. 2000.
- [3] A. Cerpa, N. Busek, and D. Estrin. Scale: A tool for simple connectivity assessment in lossy environments. Technical Report CENS-TR 03-0021, UCLA Computer Science, 2003.
- [4] A. Cerpa, J. L. Wong, M. Potkonjak, and D. Estrin. Temporal properties of low power wireless links: modeling and implications on multi-hop routing. In *Proceedings of the ACM MobiHoc*, Urbana-Champaign, IL, 2005.
- [5] D. De. Couto, D. Aguayo, B. A. Chambers, and R. Morris. Performance of multihop wireless networks: shortest path is not enough. *SIGCOMM Comput. Commun. Rev.*, 33(1):83–88, 2003.
- [6] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker. Complex behavior at scale: An experimental study of low-power wireless sensor networks. Technical Report UCLA/CSD-TR 02-0013, UCLA Computer Science, 2002.
- [7] S. Goel and T. Imielinski. Prediction-based monitoring in sensor networks: taking lessons from MPEG. *ACM Computer Communication Review*, 31(5), October 2001.
- [8] C. Guestrin, P. Bodi, R. Thibau, M. Paski, and S. Madde. Distributed regression: an efficient framework for modeling sensor network data. In *Proceedings of IPSN*, Berkeley, CA, 2004.
- [9] L. Hu and D. Evans. Localization for mobile sensor networks. In *Proceedings of MobiCom*, Philadelphia, PA, 2004.
- [10] B. Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of MobiCom*, pages 243–254, Boston, MA, Aug 2000.
- [11] F. Kuhn, R. Wattenhofer, and A. Zollinger. Worst-case optimal and average-case efficient geometric ad-hoc routing. In *Proceedings of ACM MobiHoc*, pages 267–278, Annapolis, MD, Jun. 2003.
- [12] J. Li, D. Maier, K. Tuft, V. Papadimos, and P. A. Tucker. No pane, no gain: efficient evaluation of sliding-window aggregates over data streams. *SIGMOD Rec.*, 34(1):39–44, 2005.
- [13] J. Li, D. Maier, K. Tuft, V. Papadimos, and P. A. Tucker. Semantics and evaluation techniques for window aggregates in data streams. In *Proceedings of ACM SIGMOD*, pages 311–322, Baltimore, Maryland, 2005.
- [14] D. A. Maltz, J. Broch, and D. B. Johnson. Quantitative lessons from a full-scale multi-hop wireless ad hoc network testbed. In *Proceedings of the IEEE Wireless Communications and Network Conference*, Sept. 2000.
- [15] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava. Energy aware wireless microsensor networks. *IEEE Signal Processing Magazine*, 19(2):40–50, March 2002.
- [16] K. Seada, M. Zuniga, A. Helmy, and B. Krishnamachari. Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks. In *Proceedings of ACM SenSys*, pages 108–121, Baltimore, MD, USA, 2004.
- [17] A. Woo and D. Culler. Evaluation of efficient link reliability estimators for low-power wireless networks. Technical Report UCB//CSE-03-1270, U.C. Berkeley, 2003.
- [18] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of ACM SenSys*, Los Angeles, CA, 2003.
- [19] Y. Xu, W. C. Lee, J. Xu, and G. Mitchell. PSGR: Priority-based stateless geo-routing in wireless sensor networks. In *To appear in IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, Washington, DC, Nov. 2005.
- [20] Y. Xu, J. Winter, and W.-C. Lee. Dual prediction-based reporting for object tracking sensor networks. In *Proceedings of International Conference on Mobile and Ubiquitous Systems: Networking and Services*, Boston, MA, Aug. 2004.
- [21] T. Yan, T. He, and J. A. Stankovic. Differentiated surveillance for sensor networks. In *Proceedings of SenSys*, pages 51–62, Los Angeles, CA, 2003.
- [22] Y. Yu, R. Govindan, and D. Estrin. Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks. Technical Report UCLA/CSD-TR-01-0023, UCLA Computer Science Department, May 2001.
- [23] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *Proceedings of ACM SenSys*, pages 1–13, Los Angeles, CA, Nov. 2003.
- [24] J. Zhao, R. Govindan, and D. Estrin. Computing aggregates for monitoring wireless sensor networks. In *Proceedings of IEEE ICC Workshop on Sensor Network Protocols and Applications*, Anchorage Ak, May 2003.
- [25] G. Zhou, T. He, S. Krishnamurthy, and J. A. Stankovic. Impact of radio irregularity on wireless sensor networks. In *Proceedings of MobiSys*, pages 125–138, Boston, MA, 2004.
- [26] M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *Proceedings of International Conference on Sensor and Ad-hoc Communications and Networks*, Santa Clara, CA, Oct. 2004.