This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

Accuracy-aware Interference Modeling and Measurement in Wireless Sensor Networks

Xiangmao Chang[1,2]; Jun Huang[2]; Shucheng Liu[3,5]; Guoliang Xing[2]; Hongwei Zhang[4]; Jianping Wang[3]; Liusheng Huang[5]; Yi Zhuang[1]
[1]Nanjing University of Aeronautics and Astronautics, China,
[2]Michigan State University, USA;
[3]City University of Hong Kong, HKSAR;
[4]Wayne State University, USA;
[5]University of Science and Technology of China;

Abstract—Wireless sensor networks (WSNs) are increasingly deployed for mission-critical applications such as emergency management and health care, which impose stringent requirements on the communication performance of WSNs. To support these applications, it is crucial to model and measure the effect of wireless interference, which is the major factor that limits WSN performance. Accurate modeling and measurement of interference faces two key challenges. First, as shown in our experimental results, interference yields considerable spatial and temporal variations of WSN performance, which poses a major challenge for measurement at runtime. Second, in the unlicensed band, the communication of WSN is interfered by coexisting wireless devices such as smartphones and laptops equipped with 802.11 radios, which lead to cross-technology interference that are difficult to characterize due to the heterogeneous PHY. To tackle these challenges, this paper presents a novel *accuracy-aware* approach to interference modeling and measurement for WSNs. First, we propose a new regression-based interference model and analytically characterize its accuracy based on statistics theory. Second, we develop a novel protocol called *accuracy-aware interference measurement* (AIM) for measuring the proposed interference model with *assured accuracy* at run time. Third, building on interference modeling, we propose an algorithm that accurately forecasts the performance of WSNs in the presence of cross-technology interference. Our extensive experiments on a testbed of 17 TelosB motes show that the proposed approaches achieve high accuracy of interference modeling and WSN performance forecasting with significantly lower overhead than state-of-the-art approaches.

1 Introduction

In recent years, Wireless Sensor Networks (WSNs) have been increasingly deployed for mission-critical applications such as emergency management, civil infrastructure monitoring, and health care. These applications impose stringent requirements on the communication performance of WSNs. For instance, in the scenario of wireless elderly and patient monitoring, ECG body sensors must report the cardiac rhythm data within bounded delay for real-time diagnosis. Audio sensor networks deployed for disaster response must stream survivors' voice (20-50 Kbps/stream) to rescuers with assured Quality of Service (QoS) [18].

To support these applications, it is crucial to understand the effect of wireless interference, which is the major factor that limits the communication performance of wireless networks. The situation is even worsened for WSNs due to their already limited bandwidth. When multiple sensor nodes are event-triggered, simultaneous transmissions of sensor nodes lead to significant interference, resulting in lower throughput and higher packet delivery delay. Moreover, operating in the unlicensed ISM band, WSNs suffer *cross-technology interference* (CTI) from coexisting wireless devices such as smartphones and laptops equipped with 802.11 radio. The transmission power of 802.11 is orders of magnitude higher than low-power sensor nodes, leading to frequent communication outages in WSNs [12] [19] [31]. Understanding interference is thus crucial to support mission-critical applications in WSNs. Early work on wireless interference modeling has widely assumed simple abstract models (e.g., the protocol model [10]). Unfortunately, empirical studies on both WSNs [22] [28] showed that these models are largely inaccurate. It is suggested that the packet-level physical interference model, also referred to as the SINR versus packet reception ratio (PRR) model or the SINR-PRR model, offers significantly improved realism. Taking advantage of the SINR-PRR model, several recent efforts have been made to improve the performance of link scheduling, topology control, medium access control (MAC), and directional antenna based protocols [9] [22] [27] [14] [29] [4].

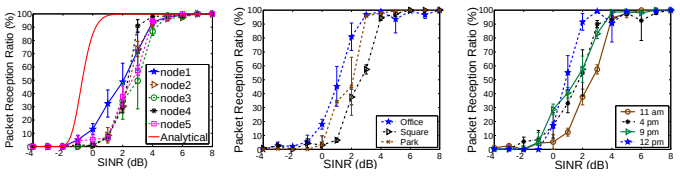
Our experimental results based on real WSN platforms show that the SINR-PRR model yields significant spatial and temporal variations and hence requires accurate measurement at *run time*. In particular, the packet reception performance of a radio must be carefully profiled under different SINRs within the *transitional region* where the PRR varies from zero to 100%. Recently, several interference measurement methods have been proposed [15] [21] [25] [26]. However, they either require nodes

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

to transmit/receive a large number of measurement packets or must sample extensive statistics of data packets, which leads to high measurement overhead. For instance, the methods proposed in [22] [26] [28] need to be periodically seeded by at least $\mathcal{O}(N)$ trials in an N-node network where each node transmits a large number of measurement packets in turn. In addition, none of existing methods is designed to measure interference models with *assured accuracy*, where the error of generated SINR-PRR model is guaranteed to within a given range. As a result, the errors of these models may cause the upper-layer protocols (e.g., link scheduling MAC protocols [6] [22] [27]) built upon them to yield unpredictable performance. Moreover, none of existing methods consider the effect of CTI on WSN performance.

In this paper, we propose a novel *accuracy-aware* approach to interference modeling and measurement for WSNs. Our approach offers several key advantages. First, it employs a regression-based SINR-PRR model, which significantly simplifies the complexity of measurement at run time. Second, it leverages statistical tools to characterize the accuracy of SINR-PRR model, which allows a WSN node to achieve the desired measurement accuracy with minimum overhead. Moreover, building on interference modeling and measurement, we propose a forecasting algorithm for WSNs to predict packet delivery under CTI. Our major contributions are summarized as follows.

1) We present a comprehensive measurement-based study on the SINR-PRR model using TelosB motes. Our results show that the SINR-PRR model yields significant spatial and temporal variations. In particular, the PRR measured on a WSN node may vary up to 50% under the same level of SINR in different environments and at different times, which demonstrates the necessity of accurate measurement at run time.
2) We propose a regression-based model to characterize the relation between SINR and PRR. Our approach features a linear transformation of the theoretical SINR-PRR model, which allows us to use regression analysis in interference measurement. Based on this result, we develop a novel analytic framework to rigorously control the trade-off between measurement overhead and accuracy. Our analysis correlates the error of generated SINR-PRR models with key measurement parameters including the selection of SINR values and the size of samples., which allows us to limit measurement error within a given range by sampling "just enough statistics of data packets.
3) We develop the *Accuracy-aware Interference Measurement* (AIM) protocol for measuring the SINR-PRR model at run time. AIM consists of several novel mechanisms that substantially reduce the overhead of online measurement, including lightweight clock calibration to reduce the overhead of synchronous timestamping, localized interference statistic aggregation, and accuracy control which allows a WSN node to achieve the desired accuracy of interference measurement by sampling "just enough" statistics of data packets.
4) Building on interference modeling and measurement, we develop an algorithm to forecast the PRR of WSNs under CTI. Through extensive analysis of CTI traces captured in real environments, we show that PRR forecasting under CTI is feasible because in most scenarios the CTI is a time-dependent process. Motivated by this observation, we develop an algorithm to forecast the SINR of WSNs, which enables accurate PRR prediction using the SINR-PRR model calibrated at run time. Our PRR forecasting algorithm can be utilized in various ways to improve the performance of WSNs under CTI. For instance, once PRR degradation is forecasted, WSN can switch to another channel or proactively enable error protection schemes to reduce packet loss.
5) We implemented the AIM protocol and the performance forecasting algorithm in TinyOS and conducted extensive experiments on a testbed of 17 TelosB motes to evaluate their performance. Our results show that the proposed approaches achieve accurate SINR-PRR modeling and PRR forecasting with significantly lower communication overhead compared with state-of-the-art approaches.



(a) The SINR-PRR models measured on different nodes. (b) The SINR-PRR models measured in different environments. (c) The SINR-PRR models measured at different time.

Fig. 1. The measured SINR-PRR model at different time in different environments.

2 Related Work

Early work [32] showed that the communication links of WSNs are often lossy and asymmetric. Several recent studies are focused on experimental characterization of the SINR-PRR model. Son et al. [28] studied the SINR-PRR model of CC1000 radios. In [22], several interference models including the SINR-PRR model, disc model, and the thresholded physical interference model, are studied for their accuracies. Interference and packet delivery models are proposed by Reis et al. [26], which can be instantiated by packet transmission traces. Qiu et al. [25] proposed a general interference model to characterize the interference among arbitrary number of 802.11 senders. In [15], a measurement-based approach is proposed to model the interference and link capacity in 802.11 networks. All the above studies employed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

the active approach to measure the interference model, which requires nodes to periodically transmit/receive extensive measurement beacons. To accurately measure interference, a calibration mechanism is presented in [7] to remove the artifacts in the raw RSSI measurements. the errors of raw RSSI readings are consistent across radio platforms of the same model. Therefore, accurate RSSI calibration can be achieved based on off-line measurement. In comparison, AIM is designed to deal with the tempospatial variation of SINR-PRR model, which must be captured at run-time.

In our previous work, we proposed a protocol called passive interference measurement (PIM) [21]. PIM samples the statistics of timestamps and received signal strength (RSS) of data packets and use them to derive the nodes' interference relationship and build their SINR-PRR models. Without generating any measurement packets, PIM reduces the overhead of active approaches [22] [26] [28]. However, it suffers from the following issues. First, PIM adopts a centralized tree-based statistics collection algorithm which incurs high overhead in large networks. Second, PIM requires frequent time synchronization and constant overhearing, which leads to unnecessary energy consumption of idle listening. Third, like all existing solutions, PIM has no accuracy control over the measured interference model. As a result, it may over/under-sample the data traffic leading to long measurement delay, high messaging overhead, or poor measurement accuracy.

The problem of link scheduling under the physical interference model has received significant attention. It is shown in [9] that the problem of finding a minimum-length collision-free schedule is NP-complete. A computationally efficient heuristic with provable performance bound is proposed in [6]. The complexity of scheduling a set of communication requests is also studied in [24]. In [27], a new MAC protocol called C-MAC is developed to maximize the aggregate throughput of a wireless cell based on the empirical SINR-PRR model. All the above works require accurate interference models.

3 Understanding the SINR-PRR Model

In this section, we first provide the theoretical background on the SINR-PRR model. We then present several findings from real testbed experiments. Finally, we present a regression-based SINR-PRR model.

3.1 Background on the Theoretical SINR-PRR Model

According to communication theory, the bit error rate (BER) , i.e., the probability that a receiver fails to receive an incoming bit, is a function of SINR at the receiver, which can be computed as follows,

$$\beta = 1 - Q(\sqrt{\frac{2\varepsilon B_N}{R}}) \tag{1}$$

where $Q(\cdot)$ is the tail probability of the standard normal distribution. ε is the SINR at the receiver. R is the modulation rate. B_N is the noise bandwidth. Because a packet is successfully received when all bits are decoded correctly, the packet reception ratio (PRR) can be computed as,

$$\rho = (1-\beta)^{\lambda}, \tag{2}$$

where λ is the number of bits in the packet. Let $\mathcal{P}(\cdot)$ be the SINR-PRR model. Based on Eq. (1) and Eq. (2), the SINR-PRR model can be expressed as follows,

$$\mathcal{P}(\varepsilon) = \left(\frac{1}{2} + \frac{1}{2} \times \text{erf}(\sqrt{\frac{\varepsilon B_N}{R}})\right)^{\lambda}. \tag{3}$$

3.2 Empirical Observation

In this subsection, we provide an empirical study on the SINR-PRR model. Our objective is two-fold. First, we compare the real-world SINR-PRR measurements with the theoretical SINR-PRR model discussed in Section 3.1. Second, we investigate the SINR-PRR relationship under different spatial and temporal settings. We show that the model yields significant spatial and temporal variation, which poses a major challenge for interference measurement and modeling on low-power wireless sensor nodes.

Our experiments are conducted on TelosB motes equipped with 802.15.4-compliant CC2420 radios. The SINR-PRR curves are measured using an existing method [22] [27] [28]. Specifically, a large number of packets are sent by two interfering links with different transmission power levels, and the SINR and PRR statistics of these transmissions are collected at the receiver to generate the model. The SINR of a packet reception is computed using the received signal strength (RSS) of the incoming packets, interfering packets, and the noise power. The RSS of a packet can be obtained by reading the RSS indicator register of the radio. The noise power can also be obtained from the RSS indicator when there is no packet transmission. We note that the SINR is always integers as the RSS precision level of CC2420 radio is 1 dB. More details about the measurement method can be found in [22] [27] [28]. The experiments last about 12 hours and are conducted in three different environments: an office, the square between two academic buildings, and a small park. Each data point (PRR,SINR) is measured with 12000 packets of 128 bytes and a confidence interval of 90% is computed. Fig. 1(a) depicts the SINR-PRR model measured on different nodes in an office in the morning. Fig. 1(b) and 1(c) show the model on the same node measured at different time in different environments. From the results, we have the following observations.

First, we observe that each SINR-PRR curve has a *transitional region* of about 10 dB wide, in which the PRR grows from zero to one. This result is consistent with the findings reported by recent empirical studies [27] except the slight variation in the width of the transitional region.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

Second, we observe that the SINR-PRR model is significantly influenced by the spatial and temporal factors. Fig. 1(a) shows that the PRR under the same SINR may vary as much as 50%, when the model is measured on different nodes. From Fig. 1(b) and Fig. 1(c), we also observe significant variations when the model is measured at different time, or the node is placed in different environments. For example, when the SINR is 2 dB at the receiver, the PRR measured in office is 45% higher than the PRR in the square; and the PRR measured at 11am is 55% lower than 12pm.

Third, the theoretical model given in Eq. (3) fails to capture the significant temporal and spatial variations observed in reality. For instance, Fig. 1(a) shows that under the same SINR, the difference between theoretical and measured model can be as much as 85%. This is partially due to the fact that the theoretical model is instantiated by the parameters defined on ideal channel, which do not account for the diversity of environments. Although it is possible to improve the accuracy of the theoretical model by carefully measuring some parameters (e.g., the bandwidth and power spectral density of noise) in a particular environment, doing so is time-consuming and particularly difficult at run time. We also note that the bit-level SINR required by the theoretical model cannot be measured on commodity radios. The results in Fig. 1 clearly demonstrate the need of measuring the SINR-PRR model in an on-line manner at run time.

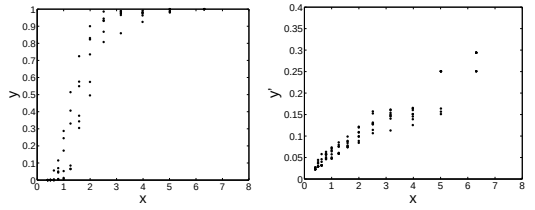
3.3 The SINR-PRR Regression Model

As discussed in the previous subsection, the accuracy of the theoretical SINR-PRR model is poor due to the spatial and temporal variation of SINR-PRR relation in reality. To obtain high accuracy, we may directly measure the SINR-PRR relation for every SINR point to construct a discrete SINR-PRR model, as shown in Fig. 1. However, all (PRR, SINR) pairs in the transitional region of every node must be continuously measured due to the spatiotemporal variation. Therefore, such an approach poses significant message overhead.

In this paper, we propose to build a regression model for characterizing the SINR-PRR relation. Our approach has two major advantages. First, the measurement overhead is significantly reduced because only a subset of (PRR, SINR) samples in the transitional region are needed to build the model. Second, the model allows us to leverage the existing theory in statistics to formally characterize the modeling accuracy and hence provide key guidance on minimizing the overhead in interference measurement.

The regression model is built as follows. First, to approximate the actual SINR-PRR model, we scale and shift the theoretical model by transforming the predictor variable x through a linear transformation as follows:

$$x' = \alpha_1 x + \alpha_2.$$



(a) (x, y) relation (b) (x, y') relation

Fig. 2. The original and linearly transformed PRR-SINR models.

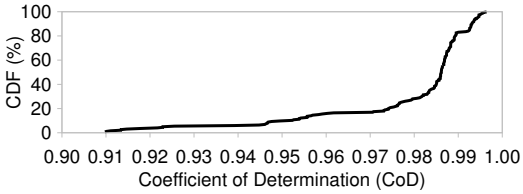


Fig. 3. The CDF of CoD of 208 regression-based PRR-SINR models.

According to the theoretical model given in Eq. (3), we can use the following parameterized model to characterize the SINR-PRR relation in practice:

$$y = \left(\frac{1}{2} + \frac{1}{2} \times \text{erf}(\sqrt{\frac{B_N(\alpha_1 x + \alpha_2)}{R}})\right)^{\lambda}, \quad (4)$$

where λ is the packet length, α_1 and α_2 are the parameters to be estimated in real-world settings. Since the (x, y) relation as described by Eq. (4) is non-linear, we cannot directly apply the model in regression analysis. To address this issue, we first approximate the function $\text{erf}(\cdot)$ according to an existing formula Eq. (5), so that we can derive closed-form solution to the (x, y) relation:

$$\text{erf}(x) \approx \sqrt{1 - e^{-(2x/\sqrt{\pi})^2}} \quad (5)$$

Then, we derive the following based on Eq. (4):

$$-ln(1 - (2y^{\frac{1}{\lambda}} - 1)^2)\frac{\pi R}{4B_N} = \alpha_1 x + \alpha_2. \quad (6)$$

Now, we let y' to represent the left side of Eq. (6),

$$y' = -ln(1 - (2y^{\frac{1}{\lambda}} - 1)^2)\frac{\pi R}{4B_N}, \quad (7)$$

Equaiton 6 is now transformed into a linear model,

$$y' = \alpha_1 x + \alpha_2. \quad (8)$$

Based on this linear model, we can conduct regression analysis to derive α_1 and α_2 using on-line measurements. That is, for each sample point (x, y) where x is a SINR value and y is the corresponding PRR, we can derive a transformed sample point (x, y') using Eq. (7); using the set of transformed sample points and Eq. (8), we can derive the values for parameters α_1 and α_2 through regression analysis; finally, plugging the derived α_1 and α_2 back into Eq. (4) gives the SINR-PRR model.

We now evaluate the accuracy of the above regression-based model. First, Fig. 2 shows the (x, y) and (x, y') relations without and with the transformation (as shown in Eq. (7)) for three sets of measurements. The x-axis

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

in Fig. 2(a) is the SINR in the unit of ratio of received powers. Note that it is not converted to decibel. Fig. 2(b) shows that, after transformation, the values of y' are highly linear with respect to x for each set of PRR-SINR measurement, which demonstrates the validity of our linear transformation. Second, we evaluate the error of regression with respect to real measurement based on commonly used metric called Coefficient of Determination (CoD) [16], which measures how well the data observed in reality are replicated by a statistical model. The model perfectly fits the data if CoD equals one. Fig. 3 shows the CDF of CoD of 208 sets of SINR-PRR curves measured in the experiments described earlier. Our result shows that in more than 80% of sets, the CoD lies above 97%, which demonstrates the accuracy of our regression-based modeling approach.

4 Model-based Accuracy Control

In this section, we analyze the accuracy of regression-based SINR-PRR model and the overhead of measuring it at run time. Given the significant temporal variation of the SINR-PRR relation shown in Section 3.2, the SINR-PRR model needs to be updated via regression of run-time measurements to obtain high accuracy. Therefore, it is critical to understand the impact of measurement overhead on the model's accuracy. There are two major factors that affect the overhead in the SINR-PRR regression modeling: 1) the number of SINR points for which we collect sample PRR values and 2) the number of PRR samples we collect for each individual SINR point. Besides affecting modeling overhead, these two factors directly affect the accuracy of regression modeling. Therefore, our approach is to control the overhead in regression modeling while achieving the desired modeling accuracy. Our approach consists of two key techniques: *principal SINR selection* and *sampling size selection*, which are discussed in the following.

Given the parameterized theoretical model Eq. (4) and its transformed linear model Eq. (8), we first identify a set of *principal* SINR points for which we will measure the corresponding PRRs. We refer to the SINR points whose PRRs will not be measured the *secondary* SINR points; the PRR corresponding to a secondary SINR point is predicted using the regression model Eq. (8), where the regression model is built based on measurement data for the principal SINR points. Based on the selected principal SINR points, we then decide the number of PRR samples to collect for each principal SINR point according to the requirements on the measurement accuracy of the principal SINR point and the prediction accuracy of the secondary SINR points. In what follows, we elaborate on our method of selecting principal SINR points and determining the sample size for each principal SINR point. These results will be used in Section 5 to develop the accuracy-aware interference measurement (AIM) protocol.

4.1 Selection of Principal SINR Points

The objective of choosing the principal SINR points is such that we can achieve the desired regression modeling accuracy while using the minimum number of principal SINR points. The regression modeling accuracy can be characterized by the mean-squared-errors (MSE) in linear regression [13]. Thus the goal is to choose the minimum number of principal SINR points to satisfy the maximum tolerable MSE. Given that the total number of integer SINR points is not very large (e.g., about 10 for a typical transitional region), we perform exhaustive search to identify the minimum set of principal SINR points, and we have found this approach to be effective in practice and affordable for WSN platforms. To identify the principal SINR points, we need to take a small number of measurements for every SINR point (including the secondary SINR points); once the principal SINR points have been identified, they will be used to build the SINR-PRR model until the accuracy of the model exceeds the required bound due to the temporal variation. We will discuss more details about the implementation of principal SINR points selection in Section 5.4.

4.2 Sample Size Selection for Principal SINR Points

Given a selected set of principal SINR points, the overall accuracy of the regression-based modeling can be ensured by controlling the measurement accuracy of the PRRs corresponding to the principal SINR points and by controlling the prediction accuracy for the PRRs corresponding to the secondary SINR points. To minimize the overhead in the regression analysis, we need to control the sampling process to minimize the number of samples taken while ensuring the required measurement accuracy and prediction accuracy. For each principal SINR point x_i, we first compute the minimum number of PRR samples, denoted by m_{x_i}, that we need to ensure the measurement accuracy for the PRR corresponding to x_i; then we compute the minimum number of PRR samples, denoted by n_{x_i}, that we need for each principal SINR point to ensure the required prediction accuracy for the PRRs corresponding to secondary SINR points; finally, we compute the number of PRR samples for a principal SINR point x_i as $\max\{m_{x_i}, n_{x_i}\}$. In what follows, we elaborate on our method of sample size computation for ensuring the required measurement accuracy and prediction accuracy.

4.2.1 Sample size for ensuring measurement accuracy

Asssume that we need to control the measurement accuracy for a principal SINR point x_i. Let y_i be the measured PRR. Based on the Central Limit Theorem, y_i is approximately normally distributed when there is a sufficient number of measurement samples. Note that, in this case, each sample reflects the status (i.e., success or failure) of a packet transmission whose corresponding receiver-side SINR is x_i. In order to limit the measurement error within $r_m\%$ of the mean PRR at the $100(1-\alpha)\%$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

confidence level, the minimum number of samples to ensure the required accuracy, denoted by m_{x_i}, can be derived as follows.

First, given a set of m_{x_i} samples, the $100(1-\alpha)\%$ confidence interval for the mean PRR is

$$y_i \mp z_{1-\alpha/2}\sqrt{\frac{y_i(1-y_i)}{m_{x_i}}}, \tag{9}$$

where $z_{1-\alpha/2}$ is the $(1-\alpha/2)$-quantile of the standard normal variate [13]. To satisfy the required accuracy, the following should hold

$$z_{1-\alpha/2}\sqrt{\frac{y_i(1-y_i)}{m_{x_i}}} \leq y_i\frac{r_m}{100}. \tag{10}$$

From (10), we have the following on the required sample size m_{x_i}:

$$m_{x_i} \geq \frac{10000 z_{1-\alpha/2}^2 y_i(1-y_i)}{r_m^2}. \tag{11}$$

Therefore, $\frac{10000 z_{1-\alpha/2}^2 y_i(1-y_i)}{r_m^2}$ samples of packet transmission status is enough to ensure the required accuracy of $(100-r_m)\%$ for the SINR point x_i.

4.2.2 Sample size for ensuring prediction accuracy

Given n pairs of measurement data, denoted by (x_i, y_i) $(i = 1...n)$ on principal SINR points, where x_i is SINR and y_i is a sample of the corresponding PRR, we can derive the regression model $y' = \alpha_1 x + \alpha_2$ (see Eq. 8) with the corresponding standard deviation of errors $s_e = \sqrt{\frac{SSE}{n-2}}$. When predicting y'_j for an secondary SINR point x_j, the mean value of the predicted y'_j is

$$\widehat{y'_j} = \alpha_1 x_j + \alpha_2, \tag{12}$$

and the standard deviation of y'_j is

$$s_{\widehat{y'_j}} = s_e\left[\frac{1}{n} + \frac{(x_j-\overline{x})^2}{\sum_{i=1}^n x_i^2 - n\overline{x}^2}\right]^{1/2} \tag{13}$$

where $\overline{x} = \frac{\sum_{i=1}^n x_i}{n}$ [13]. Then the $100(1-\alpha)\%$ confidence interval for y'_j is $\widehat{y'_j} \mp s_{\widehat{y'_j}} t_{[1-\alpha/2,n-2]}$, where $t_{[1-\alpha/2,n-2]}$ is the $(1-\alpha/2)$-quantile of a t-variate with $n-2$ degrees of freedom. Assume that the prediction error is required to be within $r_p\%$ of the mean value at the $100(1-\alpha)\%$ confidence level, then the following should hold

$$s_{\widehat{y'_j}} t_{[1-\alpha/2,n-1]} \leq \widehat{y'_j}\frac{r_p}{100}. \tag{14}$$

From (14) and (13), we have the following on the required sample size n:

$$n \geq \frac{2x_j\sum_{i=1}^n x_i - Y(\sum_{i=1}^n x_i)^2 - \sum_{i=1}^n x_i^2}{x_j^2 - Y\sum_{i=1}^n x_i^2}, \tag{15}$$

where $Y = \frac{\widehat{y'_j}^2 r_p^2}{10000 s_e^2 t^2_{[1-\alpha/2,n-2]}}$. Let

$$n_{x_j} = \frac{2x_j\sum_{i=1}^n x_i - Y(\sum_{i=1}^n x_i)^2 - \sum_{i=1}^n x_i^2}{x_j^2 - Y\sum_{i=1}^n x_i^2}.$$

Then the minimum required sample size

$$n = \max_{x_j \in X_a} n_{x_j} \tag{16}$$

where X_a is the set of secondary SINR points whose corresponding PRRs need to be predicted.

To simplify the implementation of measurement protocol, we evenly distribute the n samples to the K principal SINR points. Therefore, to ensure prediction accuracy for secondary SINR points, the number of samples for the principal SINR point x_i must satisfy,

$$n_{x_i} \geq \lceil \frac{n}{K} \rceil. \tag{17}$$

5 The Design of AIM

This section presents the design of AIM. We first give an overview of our approach, and then discuss each component in details.

5.1 Overview

AIM is a novel protocol for accurately measuring the SINR-PRR model with minimum overhead. The system architecture of AIM is illustrated in Fig. 4. AIM consists of several novel mechanisms. (1) *Interference statistics collection*. AIM collects interference statistics from normal network traffic, and piggybacks the statistics in data packets that are forwarded to the base station of the network. For each node whose interference model is to be measured, an aggregator is chosen on the routing tree to generate the SINR-PRR model using collected interference statistics. (2) *Regression-based model generation*. With collected statistics, each aggregator infers the packet-level interference, correlates PRR and SINR values, and generates the regression-based SINR-PRR model in real time. (3) *Accuracy control*. AIM controls the accuracy of measured models to a specified bound based on the rigorous error analysis described in Section 5.4, while minimizing the overall measurement overhead. AIM also tracks the temporal variations of measured models and dynamically adjusts measurement parameters like principal SINR values. (4) *Lightweight clock calibration*. AIM calibrates local clocks on each nodes for accurate interference detection. The clock calibration mechanism is developed based on the empirical observation of clock drift on real sensor hardware.

5.2 Interference Statistics Collection

We now discuss how AIM collects data packet statistics that are needed to build the regression SINR-PRR model. We assume that network has a tree-based topology in which all nodes send their data to the base station. Tree-based topologies have been commonly used in WSN

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

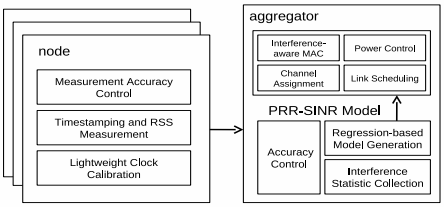


Fig. 4. Architecture of the AIM protocol.

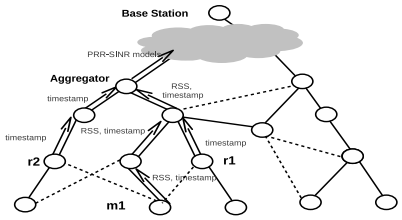


Fig. 5. An example of network topology where node $m1$'s model is generated at the aggregator.

timestamp(P12) = timestamp(P22) : collision but received
SINR (P12) = RSS(P11) – RSS(P21) – RSS(noise) ≈ 2 dB

node id	Packet id	Tx/Rx	Timestamp (ms)	RSS (dBm)
m1	P11	Rx	0.052	-55
m1	P21	Rx	5.058	-57
m1	P12	Rx	10.112	-51
r1	P11	Tx	0.038	-55
r1	P12	Tx	10.038	-55
r2	P21	Tx	5.045	-57
r2	P22	Tx	10.045	-57

Fig. 6. The collected statistics at aggregator.

applications. The design of AIM is not dependent on any particular MAC, although it relies on collecting the statistics of packet events (transmissions, receptions and collisions) for generating SINR-PRR models. We note that significant contention and packet collisions may exist even when a network adopts interference mitigation techniques such as CSMA and TDMA. In particular, the TDMA schedule constructed based on simplistic interference models often cannot effectively avoid packet collisions because these models do not account for the spatiotemporal dynamics of interference.

In contrast to most existing methods that rely on extensive measurement packets [27] [28], AIM adopts a passive scheme [21] that infers the SINR-PRR model purely from the statistics of data packets. We now briefly discuss the basic idea of passive interference measurement. We denote the nodes whose SINR-PRR models are to be measured as *m-nodes*. For a given m-node, a set of *reference nodes*, referred as *r-nodes*, is selected to help the measurement of the SINR-PRR model. The transmission of *r-nodes* must interfere with the packet reception of m-node, so that the transitional region of the SINR-PRR model can be fully captured. The selection of r-nodes can be optimized using an existing algorithm [21]. The m-nodes/r-nodes timestamp packet reception/transmission events. The precise timestamping is crucial, since the timestamps recorded by different nodes are correlated to infer the packet interference at m-nodes. Each m-node works in promiscuous mode, and records the RSS values and reception/loss statistics of overheard packets. The statistics are then piggybacked in data packets, and transmitted to the base station of the network. Along the routing tree, an *aggregator* node is selected for each m-node for SINR-PRR model generation. With the collected timestamps and RSS statistics, the aggregator can infer the SINR of each packet reception or loss for the m-node. Combined with the packet reception statistics, the SINR-PRR model of the m-node can be accurately generated. The aggregator can send the parameters of generated models to the m-node or base station to support the performance optimization of upper layer protocols.

Due to the piggybacking of interference statistics in data packets, AIM induces additional overhead in multihop forwarding. To address this problem, AIM employs an efficient distributed aggregator selection algorithm. Let T_u be the minimum sub-routing tree that includes an m-node u and all the r-nodes of u. The root of T_u should be selected as the aggregator, as only the interference statistics collected by the m-node and r-nodes will be needed for model generation. The details are omitted here due to space limitation and can be found in [11]. After the aggregator is determined, it starts model generation for m-node u using the collected interference statistics. For each data packet transmission of r-node, a record consisting of transmission power, packet ID and timestamp will be generated. In our implementation on the TelosB platform, the value of transmission power is obtained from an 8-bit register of CC2420. In 802.15.4, the packet ID is described using an 8-bit sequence number. A 16-bit timer of MSP430 is employed for packet timestamping. Therefore, each transmission record of r-node consists of only 4-byte. Similarly, a 4-byte record will be generated at m-node for each received r-node packet. As a result, statistic collection incurs an overhead of at most 8-byte per r-node packet transmission. In comparison, the maximum size of 802.15.4 data packet is 128-byte. We note that data packets of large size are common in the traffic of data-intensive WSN applications like infrastructure and environment monitoring. This is because, when compared with packet of small sizes, large packets are more efficient in amortizing the overhead of protocol header transmission. In these scenarios, the overhead incurred by AIM is extremely low when compared with the volume of data traffic.

5.3 Regression based Model Generation

An aggregator in AIM generates the SINR-PRR model by applying regression on a set of (PRR,SINR) samples. That is, the parameters of the linear model Eq. (8) are computed using curve fitting. We now discuss how an aggregator computes (PRR,SINR) samples for an m-node using collected statistics. By analyzing the timing information, the aggregator can infer that a collision occurred at m-node if the air-time of r-nodes' transmissions overlap with each other. We note that this condition is necessary but not sufficient for a packet collision because two non-interfering nodes may transmit simultaneously. AIM employs an interferer detection algorithm [21] to avoid such false positives. The aggregator then derives the resulting SINR using RSS measurements. Assuming that packet $p_{u,v}$ transmitted by r-node u is received by

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

m-node v. The SINR of $p_{u,v}$, denoted as $\varepsilon_{\eta_{u,v}}$ can be calculated as

$$\varepsilon_{\eta_{u,v}} = \frac{RSS(u,v)}{\sum_{x \in J} RSS(x,v) + \overline{I}} \quad (18)$$

where $RSS(u,v)$ is RSS of the packet transmitted from u to v, and $\overline{I}$ is the noise power. And J is the set of r-nodes whose transmissions collide with $\eta_{u,v}$. The interference caused packet loss can be detected, if packet transmissions of r-nodes are overlapped in time, but the receiver does not record any packet receptions. Finally the aggregator uses the statistics collected at the m-node to compute the reception ratio of all packets that have the same SINR, and finally generates a set of (PRR, SINR) samples for each m-node.

We now use a simple example to illustrate the basic idea of generating a (PRR,SINR) sample. Fig. 5 shows the topology of the network. Communication and interference links are marked as solid and dashed lines, respectively. Two r-nodes, r_1 and r_2 are selected to measure the SINR-PRR model of m-node m_1. The m-node and r-nodes are required to timestamp the transmissions/receptions. In addition, m_1 measures the RSS for each overheard packet. The collected statistics, which are given in Fig. 6, are then forwarded to the aggregator. From collected statistics, the aggregator finds that r_2 has transmitted p_{22}, but m_1 did not receive it. As a result, p_{22} is identified as a lost packet. By analyzing the timestamps, the aggregator infers that packets p_{12} and p_{22} collided at m_1, as they have close timestamps. To compute the SINR, the aggregator uses the measured RSS of non-interfered packets, i.e., p_{11} and p_{21}, to infer the signal power and interference power. In this example, the resulted SINR will be 2 dB. This example illustrates how a single (PRR,SINR) sample is computed. A similar process can obtain a set of measured (PRR,SINR) pairs used to generate the regression based model.

5.4 Accuracy Control

The accuracy of the SINR-PRR model is crucial for the performance of upper-layer protocols. We now discuss how AIM implements the accuracy control scheme described in Section 4. Specifically, AIM employs the following three mechanisms for ensuring an accuracy upper bound specified by users: principal SINR identification, adaptive sampling, and temporal variation correction.

First, we identify the principal SINR points in the transitional region of the SINR-PRR model for regression modeling. To begin with, we take a small number of PRR measurements for every SINR points during the initialization phase of AIM. Then we conduct an optimal exhaustive search to find the required number of principal points. The objective is to use the minimum principal points, while achieving the required modeling accuracy, quantified by the mean-squared-errors (MSE) in linear regression. For each m-node whose model needs to be measured, a principal SINR set is found and then stored on aggregator. Our experiments show that a small number of principal SINR points are enough to achieve satisfactory accuracy (see Section 7).

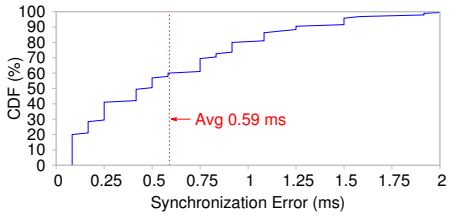


Fig. 7. Error of time synchronization.

Second, at run time, an adaptive sampling algorithm is run by each aggregator to count the number of (PRR,SNR) samples received so far, and notifies the m/r-nodes to stop measurement once the desired number of samples is received. As discussed in Section 4, the overall accuracy of the regression model is decided by 1) the measurement accuracy of PRRs at each principal SINR points, 2) and the prediction accuracy at each secondary SINR points. Specifically, in the accuracy control component of AIM, we use Eq. (11) to control the measurement accuracy, and Eq. (16) for prediction accuracy. The aggregator will notify its m/r-nodes once enough samples have been collected for model generation. Then the m/r-node will stop local measurement and statistic sampling, if they are not involved in model generation of other m-nodes.

The third mechanism used by AIM for accuracy control copes with the temporal variation of the SINR-PRR model. As shown in Section 3.2, the SINR-PRR model may vary with time considerably, which leads to accuracy degradation of measured models. AIM deals with this issue by two solutions: (1) An aggregator tracks the starting and ending SINR points of the measured transitional region and regenerates the principal SINR points once the transitional region has shifted. In such a case, the original principal SINR points no longer guarantee the accuracy of the measured model. As shown by our measurement results in Section 3.2, the shape of SINR-PRR curve remains relatively stable over time. Therefore, a new set of principal SINR points can be generated based on the empirically profiled SINR-PRR model. (2) When the shape of the fitted SINR-PRR curve varies substantially over time (e.g., due to the environmental dynamics), the aggregator asks the m/r-nodes to measure all (PRR, SINR) pairs in the transitional region, and then generates a new set of principal SINR points. Note that an aggregator incurs higher storage overhead in such a case due to the need of buffering more statistics for computing the discrete SINR-PRR model. However, once the new set of principal SINR points is generated, it resumes to normal operation in which only the PRRs corresponding to principal SINR points need to be computed.

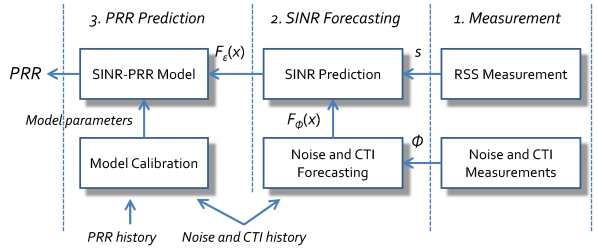


Fig. 8. Overview of PRR forecasting under CTI.

5.5 Lightweight Clock Calibration

In AIM, two packets are deemed to collide at the receiver if their transmission times are overlapped. To accurately detect packet collisions, the local clocks of m-node and r-nodes must be synchronized when timestamping packet transmissions and receptions. For TelosB nodes, the time required to transmit a 100-byte packet is about 4 ms. In our implementation, we find that packet collision can be accurately detected when clock synchronization error is below 1 ms.

AIM adopts a lightweight algorithm to maintain clock synchronization between m-node and r-nodes. Our algorithm is inspired by the Flooding Time Synchronization Protocol (FTSP) [23], which consists of two operations, including (1) *periodic message flooding*, where participating nodes synchronize their clocks based on the timestamps of received messages; and (2) *local calibration*, where each node measures the rate of its clock drift, and performs local drift compensation to maintain clock synchronization during the interval of periodic floodings.

In AIM, the design of FTSP is adapted in two aspects to further reduce synchronization overhead. First, FTSP is designed for global clock synchronization, which requires network-wide message flooding. In comparison, our algorithm is performed on a sub-tree of the network to synchronize the clocks of m-node and r-nodes, which substantially reduces flooding overhead. Second, to detect packet collisions, AIM needs to limit synchronization error below 1 ms. In comparison, FTSP is designed for clock synchronization with error in the micro-second range, which requires frequent flooding [23]. To determine the optimal flooding period for AIM, we conducted a 12-hour empirical measurement on 8 TelosB nodes. We found that a flooding period of 1 hour is enough to meet the requirement of packet collision detection. Specifically, Fig. 7 shows the CDF of errors when flooding period is 1 hour. As shown in the figure, the average error is 0.59 ms, and the error is within 1 ms most of the time.

6 PRR Forecasting Under Cross Technology Interference

Operating in the unlicensed 2.4 GHz frequency bands, WSNs suffer significant cross-technology interference (CTI) from coexisting wireless devices such as smartphones and laptops equipped with 802.11 radios. The transmission power of 802.11 is significantly higher than WSN nodes, leading to frequent communication outages in WSNs. In this section, we propose a forecasting algorithm to predict the PRR of WSN under CTI. Through extensive analysis of CTI traces captured in real environments, we show that in most scenarios the CTI is a time-dependent process. Motivated by this result, we develop an algorithm to forecast the SINR of WSNs, which enables accurate PRR prediction using the SINR-PRR model calibrated at run time. Our algorithm can be integrated with channel assignment mechanisms [17] [30] [33], power control protocols [27] [20], and error protection schemes [19] [8], to avoid communication outage under CTI.

6.1 System Overview

In the unlicensed band, the communication performance of WSNs is affected by noise and CTI, which causes the PRR of WSNs to vary with the traffic dynamics of coexisting wireless devices. The goal of PRR forecasting is to predict the distribution of PRR for WSNs in the presence of time-varying CTI patterns. As shown in Fig. 8, the algorithm consists of three phases, including *signal and interference measurement*, *SINR forecasting* and *PRR prediction*.

In the measurement stage, the receiver measures the signal strength of the sender, denoted by s, using the RSSs of received data packets. The strength of noise and CTI, denoted by ϕ, can be obtained by measuring the noise floor at the receiver when there is no WSN transmission. Specifically, a signal is classified as CTI if its power is higher than the noise floor, but no decodable packet is detected at the WSN node. As an example, Fig. 9 shows the trace of noise and CTI measured using a TelosB node in an office building, where a large-scale 802.11-based WLAN is deployed and heavily used. As shown in the figure, the traffic of WLAN yields frequent spikes during noise floor measurement.

The goal of SINR forecasting is to predict the distribution of SINR under CTI. This is challenging due to the temporal variation of CTI pattern, depending on the traffic dynamics of coexisting wireless devices. Based on statistical analysis, we develop a forecasting algorithm that accurately predicts the CDF of noise and CTI strength using historical observations, and then tranforms the result into the CDF of SINR. Specifically, the SINR of WSN communication can be computed as,

$$\varepsilon = s - \phi \tag{19}$$

where s and ϕ are measured in dBm, and ε is measured in dB. Let $F_\varepsilon(x)$ be the CDF of SINR under CTI. Based on Eq (19), $F_\varepsilon(x)$ can be computed as,

$$F_\varepsilon(x) = 1 - F_\phi(s - x), \tag{20}$$

where $F_\phi(x)$ is the CDF of noise and CTI strength. To forecast the CDF of SINR, the receiver first forecasts

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

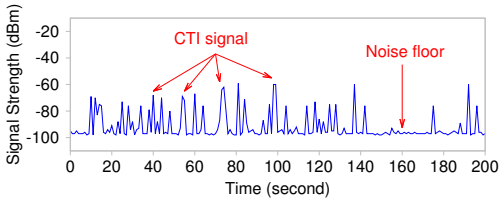


Fig. 9. Trace of noise and CTI strength measured in an office building.

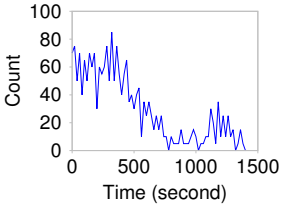


Fig. 10. Count of noise and CTI signals between -60 dBm and -55 dBm measured per second.

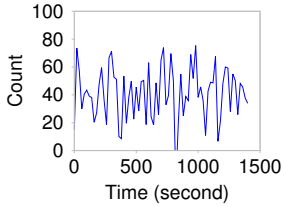


Fig. 11. A stationary Gaussian process whose mean and variance are the same with the trace in Fig. 10.

$F_\phi(x)$ and then computes $F_\varepsilon(x)$ using Eq (20). We will discuss how to forecast $F_\phi(x)$ in Section 6.2.

In the PRR prediction stage, the receiver calibrates the theoretical SINR-PRR model at run-time and then predicts the CDF of PRR using predicted $F_\varepsilon(x)$. Denote the theoretical SINR-PRR model as $\mathcal{P}(x)$, which is given in Eq (3). As we discussed in Section 3.3, to account for the spatiotemporal variations of SINR-PRR model, $\mathcal{P}(x)$ needs to be calibrated through scaling and shifting using a linear transformation, which can be expressed as $\mathcal{P}(\alpha_1 x+\alpha_2)$. As a result, we can compute the CDF of PRR as follows,

$$F_{\mathcal{P}}(x) = F_\varepsilon\left(\frac{\mathcal{P}^{-1}(x)-\alpha_2}{\alpha_1}\right) = 1-F_\phi\left(s-\frac{\mathcal{P}^{-1}(x)-\alpha_2}{\alpha_1}\right). \quad (21)$$

Due to the heterogeneous PHY of coexisting wireless devices, the AIM protocol described in Section 5 cannot be used to calibrate the theoretical SINR-PRR model under CTI. To tackle this problem, we propose a computationally efficient heuristic to estimate the optimal α_1 and α_2 that achieve minimum PRR forecasting error, which is described in detail in Section 6.3.

6.2 Noise and CTI Forecasting

We first discuss how to forecast the distribution of noise and CTI strength. Probability distribution forecasting is known to be computationally expensive [5]. Due to constraints on computation resource in WSNs, we employ a simple discrete heuristic to approach this problem. Specifically, we divide the domain of signal strength into small segments of d dB. Let n_i be the number of signal measurements whose strengths fall into the i-th segment. The CDF of noise and CTI strength can be approximated as follows,

$$F_\phi(x) = \frac{\sum_{i=1}^{\lceil x/d \rceil} n_i}{\sum_i n_i}. \quad (22)$$

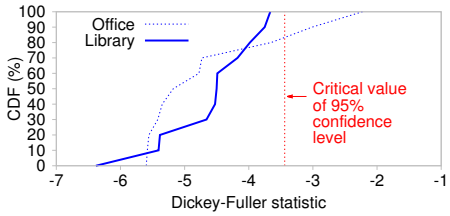


Fig. 12. Results of Augmented Dickey-Fuller test on two noise and CTI traces collected in an office building and a university library. The more negative the Dickey-Fuller statistic is, the stronger the non-stationarity is. Overall, in 91% of cases, the trace show non-stationarity with a confidence level of 95%.

The accuracy of Eq. (22) depends on the segment size d. Small d leads to fine-grained CDF prediction but incurs high computational overhead. In our implementation on TelosB motes, d is empirically set to 5 dB. On other platforms of different computing capability, d can be tuned to achieve the desired trade-off between prediction accuracy and computational cost.

We next discuss how to forecast n_i. Specifically, We divide the time into windows of equal size. Let $n_{i,j}$ be the number of signal measurements falling into the i-th segment during the j-th time window. $n_{i,1}, n_{i,2}, ...$ can be considered as a time series, which can be predicted using time series forecasting algorithms [5] based on the statistic characteristic of n_i. To understand the statistical behavior of n_i, we conducted a measurement-based study using two traces collected in an office building and a university library. The traces contain 269,614 measurements of noise and CTI signal strength, covering a range of 100dB in signal strength domain. Fig. 10 shows the trace of n_i for the segment from -60dBm to -55dBm. We observe that the trace exhibits strong *non-stationarity*, where n_i changes over time following a specific trend decided by the traffic pattern of the coexisting WLAN. In comparison, Fig. 11 plots the trace of a stationary Gaussian process with the same mean and variance, whose behavior is significantly different from n_i. To formally test the non-stationarity of n_i, we perform the Augmented Dickey-Fuller (ADF) test, which computes the Dickey-Fuller statistic. The more negative the statistic is, the stronger the non-stationarity is. As shown in Fig. 12, data collected in 91% of segments show non-stationarity with a confidence level of 95%. Based on this result, we forecast n_i using the Autoregressive integrated moving average (ARIMA) model, which is widely used for forecasting non-stationary time series. Given a time series $n_i = \{n_{i,1}, n_{i,2}, ...\}$, the ARIMA$(u, v, z)$ model can be expressed as follows,

$$(1-\sum_{k=1}^{u}\varphi_k L^k)(1-L)^z n_{i,j} = (1+\sum_{k=1}^{v}\theta_k L^k)\mu \quad (23)$$

where u, v and z are the orders of the autoregressive, moving average, and integrated parts of the model. φ_k and θ_k are the parameters of the autoregressive and moving average parts, respectively. μ is a white noise error term. L is the delay operator, and $L^k n_{i,j} = n_{i,j-k}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

6.3 SINR-PRR Model Calibration

We now discuss how to calibrate the theoretical SINR-PRR model $\mathcal{P}(x)$ under CTI. Our goal is to compute the optimal α_1 and α_2 such that the prediction error is minimized when predicting PRR using Eq (21). In the rest of this section, we first formulate this problem and then present the solution.

As discussed in Section 6.2, we divide the time into windows of ts. During a specific time window j, the real PRR, denoted by p_j, can be obtained at the receiver based on the sequence number of received data packets. Denote the predicted PRR by $\hat{p}_j$. By dividing the domain of signal strength into segments of d dB, $\hat{p}_j$ can be computed using the following discrete approximation,

$$\hat{p}_j = \frac{\sum_i \mathcal{P}(\alpha_1 \times d \times i + \alpha_2) \times n_{i,j}}{\sum_i n_{i,j}} \tag{24}$$

where $n_{i,j}$ is the number of noise and CTI signals in the i-th segment measured during the j-th time window; $\mathcal{P}(\alpha_1 \times d \times i + \alpha 2)$ computes the PRR when the strength of noise and CTI falls into the i-th segment. To improve the accuracy, we calibrate the SINR-PRR model based on the historical observations in the past N time windows. Specifically, the goal of model calibration can be formulated as,

$$\arg\min_{\alpha_1,\alpha_2} \sum_{j=1}^{N} \left| p_j - \frac{\sum_i \mathcal{P}(\alpha_1 \times d \times i + \alpha_2) \times n_{i,j}}{\sum_i n_{i,j}} \right| \tag{25}$$

Large t and N improve the confidence of parameter estimation, but reduces the responsiveness when the SINR-PRR model experiences temporal variation. Based on empirical observations, we set $t = 20$s and $N = 3$ in our implementation.

Due to stringent constraints of computation resource on WSN nodes, we employ a simple heuristic to solve the problem formulated in Eq (25). Specifically, we perform exhaustive search in the two-dimensional space defined by α_1 and α_2, to find the optimal linear transformation that achieves the minimum forecasting error in the past N time windows. The search is empirically confined to a 5×10 space. Since we set $t = 20$s and $N = 3$, model calibration only requires $3 \times 50 = 150$ computations of Eq (24) every 20s.

7 Experimentation

We experimentally analyze the performance of AIM, and compare AIM with other state-of-the-art interference measurement methods. We first discuss the experimental methodology and then the experimental results.

7.1 Experimental Methodology

We have implemented AIM in TinyOS-2.0.2. We use a testbed of 17 TelosB motes [2] for the evaluation. The motes are organized such that 16 of them generate data packets that will be delivered to the 17th mote which serves as the sink. The interference models of all the nodes except the sink are measured. To obtain accurate signal strength measurement on TelosB nodes, we calibrate the raw RSSI readings based on CC2420 datasheet [1]. As discussed in Section 5, an aggregator node is chosen for measuring the SINR-PRR model of each node. Note that a m-node may serve as the r-node and/or aggregator for other m-nodes. The collection-tree-protocol (CTP) [3] is adopted as the routing protocol. The 16 source nodes are deployed in a 4×4 grid in an office, with the distance between two closest grid-points being 10 feet. We then tune the transmission power of nodes to generate different multi-hop topologies up to 4 hops. We conduct an experiment of 2 hours for each topology. We note that AIM does not assume any prior knowledge of network topology. In addition, the grid-based testbed deployment will not affect the fairness of performance comparison between AIM and other measurement protocols. In our experiments, each source continuously generates data packets of 128 bytes at an average frequency of 10 packets per second.

In the following, we first evaluate the performance of accuracy control in AIM, then comparatively study AIM with respect to two baseline interference measurement methods: 1) ACTIVE: a representative *active* method [27] [28] which generates the SINR-PRR models using measurement packets. The number of measurement packets is varied in different experiments; 2) PIM [21]: a state-of-the-art *passive* method which builds the SINR-PRR model using interference statistics collected in normal network traffic. In contrast to AIM, both methods are designed with no accuracy control over the generated models.

7.2 Performance of Accuracy Control

We now evaluate the performance of accuracy control, which includes the control of measurement accuracy, regression accuracy and prediction accuracy. The (PRR, SINR) measurements were collected from aggregators and then used to generate SINR-PRR models according to various settings. At the same time, the active method [22] [27] [28] is periodically executed to measure the SINR-PRR models. Each model is measured with about 8000 measurement packets. It was shown [22] that such an approach can achieve high accuracy at the price of high measurement overhead. These models are used as "ground truth" to evaluate the accuracy of the models generated by AIM.

We first analyze measurement accuracy control, whose purpose is to ensure the accuracy of measuring PRR for each single principle SINR value. We generate a different SINR-PRR model using every m samples where each sample is the status (i.e., success or failure) of a packet transmission. Fig. 13 shows the CDF of average measurement errors with respect to the "ground truth" profiled using 8000 measurement packets. It can be seen that the number of transmission status samples required

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

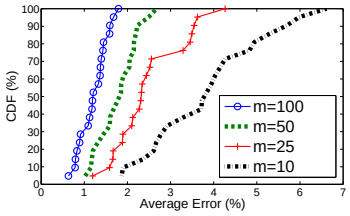


Fig. 13. The CDF of average measurement errors

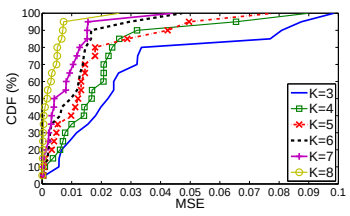


Fig. 14. The CDF of MSE for different number K of principal SINR points.

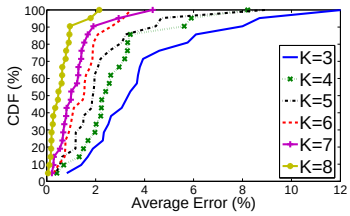


Fig. 15. The CDF of prediction errors for different number K of principal SINR points.

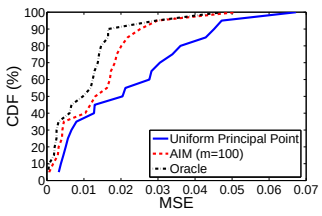


Fig. 16. The CDF of MSE for different strategies of principal-SINR selection.

is small. For instance, the average error is less than 7% even with a sample size of 10. As the number of samples collected increases, the measurement error further decreases. For instance, with 100 samples, the error is usually less than 1.5%. From this, we see that accuracy-aware adaptive sampling can save significant sampling overhead since it only takes very few samples in general.

The regression accuracy depends on the number of principal SINR points and the principal selection strategy. Fig. 14 shows the impact of the number of principal-SINR-points on the accuracy of the regression modeling, which is measured by the mean-squared-errors (MSE) in linear regression as defined in Section 5.4. The MSE quantifies the error, when the regression SINR-PRR model generated from principal SINR points is used to predict the PRRs of secondary SINR points. We see that a small number of principal points is enough to ensure small MSE. For instance, the MSE is less than 10% even if we only use 3 principal points. As the number of principal points increases, the MSE quickly decreases. For instance, the MSE is less than 3% when the number of principal points is 8. Fig. 15 shows the CDF of the average prediction errors for different number of principal points. We see that, similar to the regression error MSE, the prediction error tends to be small and decreases quickly as the number of principal points increases.

We now evaluate the impact of principal selection strategy on the regression accuracy of SINR-PRR modeling. Fig. 16 shows the CDF of MSEs of all measured models when the number of principal SINR points is 6. Three principal selection strategies are evaluated. The "Uniform-Principal-Point" strategy selects the principal SINR points for an arbitrary node in the network using the AIM method and then uses these principal SINR points for every node in the network. The "Oracle" strategy continuously recomputes the optimal principal points based on the current network condition. Specifically, it selects the principal SINR points by analyzing all SINR measurements during a given period of time while a practical method like AIM only has access to the historical data before starting a new round of measurement and model generation. The Oracle strategy serves as a reference on the optimal performance that could be achieved. We see that the uniform-principal-point strategy performs the worst as it neglects the diversity among nodes. The gap between AIM and the oracle strategy is small, and the MSE is less than 4% most of the time, which is accurate enough for model generation. This result demonstrates the effectiveness of principal selection in AIM.

7.3 Accuracy and Overhead

Here we comparatively study AIM, PIM, and ACTIVE in terms of their accuracy and overhead. We first study the accuracy of AIM. For AIM, we use 6 measured (PRR,SINR) pairs for regression model generation, and collect 100 samples for each (PRR,SINR) pair of principal SINR point. That is, the measurement overhead of each model is 600 samples. Fig. 17 shows the PRR modeling error in AIM and ACTIVE, where the error is defined as the absolute difference between the actual PRR and the modeled PRR for each SINR point. For ACTIVE, we vary the number of samples (e.g., measurement packets) used for each model to evaluate its effect on the performance. We see that the accuracy of ACTIVE increases with number of samples used. We also observe that the AIM with 600 samples achieves an accuracy similar to that in ACTIVE with 4096 samples. We observed that this is caused by the diversity of radio performance at different SINRs. Due to unpredictable environmental factors and hardware biases, the PRR of radio at each SINR has different variance. However, ACTIVE cannot choose the right number of samples for measuring each individual PRR. In contrast, based on the accuracy control mechanism, AIM controls the number of samples for each SINR point in the model measurement, which leads to significantly lower overhead.

We now evaluate the communication overhead of different methods. For a fair comparison, we measure the overhead when they achieve the same error of 5%, where the overhead is measured as the additional bytes used to generate the SINR-PRR models. For AIM, we set the accuracy bound to 5%. Each node in the network runs the accuracy control component to guide the SINR-PRR model generation. To understand the impact of adaptive sampling in AIM, we also study AIM-CS that is a variant of AIM without the adaptive sampling mechanism described in Section 4.2, (i.e., continuously sample the principal SINR points). We see that ACTIVE is the most costly approach with the overhead growing linearly with the number of m-nodes. AIM-CS, PIM and AIM perform significantly better than ACTIVE, and their overhead do

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

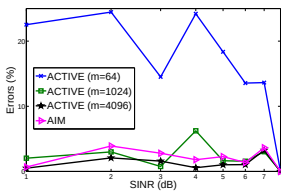


Fig. 17. PRR-SINR models generated by ACTIVE and AIM.

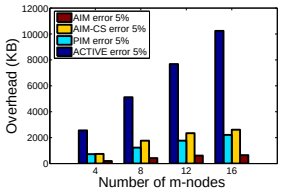


Fig. 18. Overhead in AIM, AIM-CS, PIM, and ACTIVE.

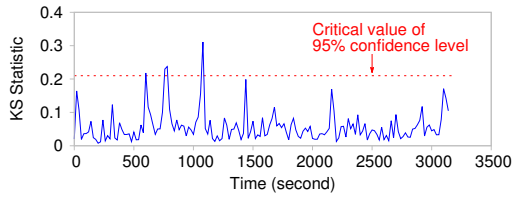
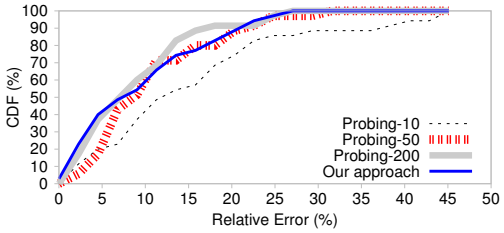
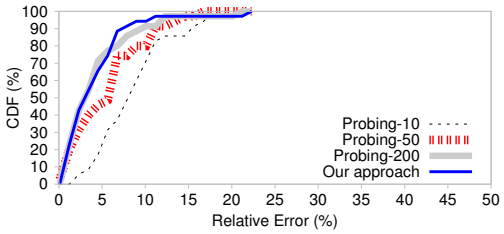


Fig. 19. Accuracy of noise and CTI forecasting. The figure shows the Kolmogorov-Smirnov (KS) statistic computed using the predicted distribution and the distribution observed in reality. The dashed line gives the critical value of 95% confidence level, above which the two distributions are considered different.



(a) Library experiment.



(b) Office experiment.

Fig. 20. Relative error of PRR forecasting in two traces using different approaches.

not increase quickly with the number of m-nodes. This is because they employ a passive approach for model generation where interference samples are collected from normal network traffic and forwarded to the aggregator by piggybacking data packets. We also observe that AIM introduces the lowest overhead. Compared with AIM-CS and PIM, AIM stops sampling when the required modeling accuracy is achieved. This result demonstrates the effectiveness of accuracy-aware sampling in AIM.

7.4 Performance of PRR Forecasting

To evaluate the PRR forecasting algorithm proposed in Section 6, we deploy low-power wireless links composed of telosB motes in two different environments, including an office building and a university library. In both experiments, the communication of motes is interfered by the large-scale coexisting WLANs, as well as nearby Bluetooth devices such as wireless headsets.

We first evaluate the performance of noise and CTI forecasting algorithm described in Section 6.2. Specifically, we compute the Kolmogorov-Smirnov (KS) statistic to compare the predicted distribution of noise and CTI strength with the distribution observed in reality. Fig. 19 shows the result obtained in the library experiment. The dashed line gives the critical value of 95% confidence level, above which the predicted distribution is considered different with the groundtruth. As shown in the figure, the proposed algorithm accurately forecasts the distribution of noise and CTI strength in 97.5% of experiments. The small prediction error could be caused by the discrete approximation when transforming CTI signals into time series. We observed similar results in the office experiment.

To evaluate the forecasting accuracy, we compare the proposed algorithm with a baseline called *probing-γ*, which transmits γ probing packets every time window to measure PRR, and then forecasts PRR based on measurement history. In comparison, our approach forecasts PRR by learning the pattern of noise and CTI strength, which can be done by measuring the noise floor locally when there is no WSN transmission, thereby avoiding communication overhead.

Fig. 20 compares the relative error of our approach with three variations of the probing baseline, including probing-10, probing-50 and probing-200, which forecast PRR based on the PRR history measured using 10, 50 and 200 packets per time window, respectively. We find that the accuracy of probing-based approach increases when more packets are transmitted to measure the PRR. However, the improved accuracy can be easily offset by the increased communication overhead. In comparison, our approach achieves similar accuracy with probing-200, without extra communications. We observe that the average forecasting error of our approach is 3% in the office experiment. The forecasting error is higher in the library, which is most likely caused by the frequent movement of people who carry wireless devices. People movement not only changes the signal propagation environment, but also induces dynamic network traffic, leading to fast variation of CTI patterns. As shown in the figure, our approach is able to maintain a low forecasting error around 15% in the library experiment despite the challenging environment.

8 Conclusion

This paper presents an *accuracy-aware* approach to interference modeling and measurement for WSNs. First, we propose a new regression-based SINR-PRR model whose accuracy is analytically characterized based on statistics theory. Second, we develop a novel protocol called *accuracy-aware interference measurement* (AIM) for measuring the proposed SINR-PRR model with *assured*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2015.2416182, IEEE Transactions on Mobile Computing

accuracy. AIM enforces a specified accuracy bound by adaptively controlling the measurement process at run time. AIM also adopts new clock calibration and in-network aggregation techniques to reduce the overhead of interference measurement. Moreover, building on interference modeling, we propose an efficient algorithm for forecasting the PRR of WSN in the presence of cross-technology interference. We conduct extensive experiments on a testbed of 17 TelosB motes to evaluate the performance of AIM and the PRR forecasting algorithm. Our results show that the proposed approaches achieve high accuracy of SINR-PRR modeling and PRR forecasting with significantly lower overhead than state of the art approaches. Due to spatiotemporal variations, SINR-PRR model needs to be periodically re-measured to maintain satisfactory accuracy. In this work, we focus on reducing the per-round communication overhead of interference measurement. In future work, we plan to develop adaptive controller to optimize the period of measurement at run-time. Such design will further reduce the communication overhead of interference measurement.

ACKNOWLEDGMENT

This work is supported, in part, by the National Science Foundation under grants CNS-0916576, CNS-0954039 (CAREER) and CNS-1054634 (CAREER); the National Nature Science Foundation of China (No. 61202350) and the National Science Foundation of China for Postdoctoral (No. 2012M521080).

REFERENCES

[1] Cc2420 data sheet. In *http://www-inst.eecs.berkeley.edu/ cs150/Documents/CC2420.pdf*.
[2] TelosB sensor node. http://www.crossbow.com/.
[3] TinyOS collection tree protocol (CTP). http://www.tinyos.net/tinyos-2.x/doc/html/tep123.html.
[4] N. Ahmed, S. Kanhere, and S. Jha. Mitigating the effect of interference in wireless sensor networks. In *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, LCN'10, pages 160–167, 2010.
[5] G. E. Box, J. S. Hunter, and W. G. Hunter. *Statistics for experimenters: design, innovation, and discovery*, volume 2. Wiley Online Library, 2005.
[6] G. Brar, D. M.Blough, and P. Santi. Computationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks. In *MobiCom*, 2006.
[7] Y. Chen and A. Terzis. On the mechanisms and effects of calibrating rssi measurements for 802.15.4 radios. In *Proceedings of the 7th European Conference on Wireless Sensor Networks*, EWSN'10, 2010.
[8] H. Dubois-Ferrière, D. Estrin, and M. Vetterli. Packet combining in sensor networks. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, SenSys '05, 2005.
[9] O. Goussevskaia, Y. A. Oswald, and R. Wattenhofer. Complexity in geometric sinr. In *MobiHoc*, 2007.
[10] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46:388–404, 2000.
[11] J. Huang, S. Liu, G. Xing, H. Zhang, J. Wang, and L. Huang. Accuracy-aware interference modeling and measurement in wireless sensor networks. Technical Report MSU-CSE-11-3, Michigan State University, 2011.
[12] J. Huang, G. Xing, G. Zhou, and R. Zhou. Beyond co-existence: Exploiting wifi white space for zigbee performance assurance. In *Proceedings of the The 18th IEEE International Conference on Network Protocols*, ICNP '10, 2010.
[13] R. Jain. *The Art of Computer Systems Performance Analysis*. John Wiley & Sons, Inc., 1991.
[14] H. Jasani and K. Yang. Performance improvement using directional antennas in ad hoc networks. *International Journal of Computer Science and Network Security*, 2006.
[15] A. Kashyap, S. Ganguly, and S. R. Das. A measurement-based approach to modeling link capacity in 802.11-based wireless networks. In *MobiCom*, 2007.
[16] D. G. Kleinbaum, L. L. Kupper, and K. E. Muller, editors. *Applied Regression Analysis and Other Multivariable Methods*. PWS Publishing Co., Boston, MA, USA, 1988.
[17] H. K. Le, D. Henriksson, and T. Abdelzaher. A practical multi-channel media access control protocol for wireless sensor networks. In *IPSN*, 2008.
[18] L. Li, G. Xing, L. Sun, and Y. Liu. QVS: quality-aware voice streaming for wireless sensor networks. In *ICDCS*, 2009.
[19] C.-J. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis. Surviving wi-fi interference in low power zigbee networks. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, 2010.
[20] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He. Atpc: Adaptive transmission power control for wireless sensor networks. In *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, SenSys '06, 2006.
[21] S. Liu, G. Xing, H. Zhang, and J. Wang. Passive interference measurement in wireless sensor networks. In *IEEE ICNP*, 2010.
[22] R. Maheshwari, S. Jain, and S. R. Das. A measurement study of interference modeling and scheduling in low-power wireless networks. In *SenSys*, 2008.
[23] M. Maróti, B. Kusy, G. Simon, and A. Lédeczi. The flooding time synchronization protocol. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, 2004.
[24] T. Moscibroda and R. Wattenhofer. The complexity of connectivity in wireless networks. In *IEEE INFOCOM*, 2006.
[25] L. Qiu, Y. Zhang, F. Wang, M. K. Han, and R. Mahajan. A general model of wireless interference. In *MobiCom*, 2007.
[26] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan. Measurement-based models of delivery and interference in static wireless networks. In *SIGCOMM*, 2006.
[27] M. Sha, G. Xing, G. Zhou, S. Liu, and X. Wang. C-mac: Model-driven concurrent medium access control for wireless sensor networks. In *Infocom*, 2009.
[28] D. Son, B. Krishnamachari, and J. Heidemann. Experimental study of concurrent transmission in wireless sensor networks. In *Sensys*, 2006.
[29] K. Staniec and G. Debita. Interference mitigation in wsn by means of directional antennas and duty cycle control. *Wirel. Commun. Mob. Comput.*, 2012.
[30] Y. Wu, J. A. Stankovic, T. He, and S. Lin. Realistic and efficient multi-channel communications in wireless sensor networks. In *Infocom*, 2008.
[31] X. Zhang and K. G. Shin. Enabling coexistence of heterogeneous wireless systems: Case for zigbee and wifi. In *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '11, 2011.
[32] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *Sensys*, Los Angeles, CA, November 2003.
[33] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic, and T. F. Abdelzaher. Mmsn: Multi-frequency media access control for wireless sensor networks. In *Infocom*, 2006.