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.2020.2978205, IEEE Transactions on Mobile Computing

Exploiting Interference Fingerprints for Predictable Wireless Concurrency

Meng Jin, *Student Member, IEEE*, Yuan He*, *Member, IEEE*, Xiaolong Zheng, *Member, IEEE*, Dingyi Fang*, *Member, IEEE*, Dan Xu, Tianzhang Xing, Xiaojiang Chen, *Member, IEEE*

Abstract—Operating in unlicensed ISM bands, ZigBee devices often yield poor performance due to the interference from ever increasing wireless devices in the 2.4 GHz band. Our empirical results show that, a specific interference is likely to have different influence on different outbound links of a ZigBee sender, which indicates the chance of *concurrent transmissions*. Based on this insight, we propose Smoggy-Link, a practical protocol to exploit the potential concurrency for adaptive ZigBee transmissions under harsh interference. Smoggy-Link maintains an accurate link model to quantify and trace the relationship between interference and link qualities of the sender's outbound links. With such a link model, Smoggy-Link can translate the low-cost interference information to the fine-grained spatiotemporal link state. The link information is further utilized for adaptive link selection and intelligent transmission schedule. We implement and evaluate a prototype of our approach with TinyOS and TelosB motes. The evaluation results show that Smoggy-Link has consistent improvements in both throughput and packet reception ratio under interference from various interferers.

Index Terms—Cross-Technology Interference, Concurrent Transmission, Interference Identification, Link Estimation.

1 Introduction

The explosive growth of wireless devices boosts the proliferation of heterogeneous network technologies on the 2.4 GHz ISM (Industrial Scientific Medical) band. Typical examples include WiFi, ZigBee, Bluetooth, and etc. While those technologies bring to people convenience and efficiency, a serious problem attracts increasing attention: spectrum sharing among incompatible wireless technologies has led to a severe cross-technology interference problem (CTI) [1–7], especially for the low-power technologies such as ZigBee [4–7].

The existing approaches of interference resolution largely aim at communicating over non-overlapping segments of the spectrum [8–10]. However, the ISM band is becoming increasingly crowded, making it difficult to find an interference-free channel. This leads the researchers to focus on developing interference avoidance solutions in time domain [4, 11, 12]. These time domain solutions usually sacrifice the efficiency of channel utilization for conservative backoff, which limits network throughput. Clearly, there is an inherent conflict between interference avoidance and channel utilization. This motivates us to think about a question: can a ZigBee node transmit packets *concurrently* with the interference?

Our key observations in Section 3 show that a specific interference is likely to have different influence on different outbound links of a ZigBee sender. Even under strong interferences, such as WiFi, there is still certain chance for any one of the receivers to successfully decode the sender's packet. This is due to exposed terminal phenomenon and the DSSS (direct-sequence spread spectrum) modulation scheme used by ZigBee technology. Exploiting such opportunities can significantly improve network throughput.

Co-corresponding authors: Yuan He, E-mail address: he@greenorbs.com, Dingyi Fang, E-mail address: dyf@nwu.edu.cn

However, we may meet two challenges towards the above goal: first, considering the dynamic network environment, the feasible concurrency pattern (which link enables concurrently transmit) seems unpredictable and changeable, thus we need an accurate link model to quantify and trace the relationship between the interference and the quality of each outbound link. Second, to avoid the collision between the ACKs and the interference, which would undermine link estimation, we need to carefully schedule the transmission of ACKs to make them arrive at the sender during the idle spaces between interference frame clusters.

In this paper, we propose Smoggy-Link, a practical protocol to exploit the potential concurrency for adaptive transmission under interference. The idea of Smoggy-Link is based on our observation that *link quality is highly related to the characteristics of the interference*. Therefore, given the feasibility of interference identification according to the featured patterns of the interference signals, we can utilize the low-cost interference information to obtain fine-grained spatiotemporal link state. The link state information can be further utilized for adaptive link selection. In addition, we also observe *predictable patterns* of the data arrival processes of interferences. This enables a node to predict the arriving time of the idle spaces and intelligently schedule the transmissions of data and ACKs to achieve both high channel utilization and low ACK collision probability. The contributions of this paper are summarized as follows:

- Based on the observation of the abundant opportunities for concurrent transmissions, we propose Smoggy-Link, an adaptive transmission protocol that can fully exploit concurrency to maximize the network throughput while achieving predictable packet reception ratio.
- We present a novel link model to accurately characterize the relationship between link quality and the interference. The model is utilized for adaptive link

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

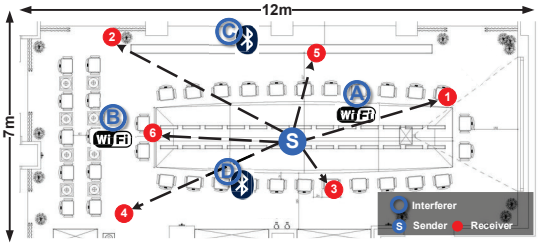


Fig. 1: Layout of the experiment setup.

selection and intelligent transmission schedule.

- We implement and evaluate a prototype of Smoggy-Link with TinyOS and TelosB motes. The evaluation results show that Smoggy-Link has consistent improvements in both throughput and packet reception ratio under interference from various interferers.

Roadmap. Section 2 summarizes the related work. Section 3 presents our key findings. An overview of Smoggy-Link is given in Section 4 and the design of Smoggy-Link is given in Sections 5 and 6. Section 7 discusses practical issues. Section 8 evaluates the performance of Smoggy-Link's and Section 9 concludes the paper.

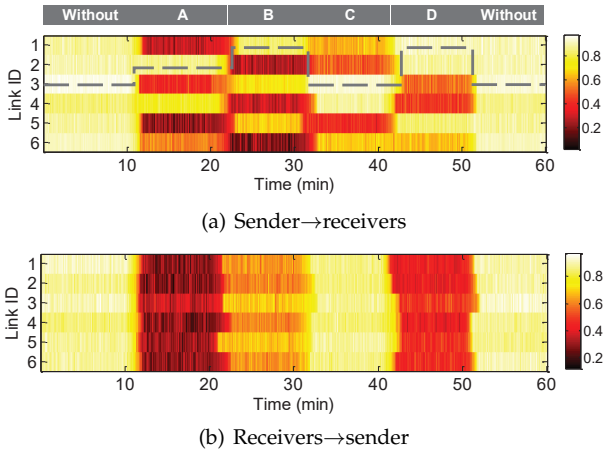
2 Related Work

Wireless links are unreliable due to interference [5–7, 13–15]. Many methods are proposed to mitigate this problem.

Opportunistic communication. Opportunistic communication uses packet replication to improve the reliability when interference occurs. Such methods achieves high throughput but at the same time wastes the channel resources due to the replication. Different from opportunistic communication, Smoggy-Link can accurately identify the interference-free links (if exist) when interference occurs, thus achieves robust transmission without incurring additional cost of network bandwidth. Indeed, Smoggy-Link is complemental with opportunistic communication, it can find the top-ranked links for opportunistic transmission.

Interference cancellation. Many efforts have been made to cancellate the cross technology interferences using MIMO [1, 16]. These methods use signal processing techniques to minimize or completely cancel interference from other links. With such a method, concurrent transmissions from multiple devices are possible. However, although MIMO based method is proved efficient in interference cancellation, it cannot be applied on ZigBee devices where MIMO technology is not supported.

Interference avoidance. Interference avoidance based schemes attempt to eliminate interference by isolating the signals in the time or frequency domain. The common principle of the time domain solutions [11, 17] is to scatter the transmissions in the temporal dimension. The ability of such schemes, however, usually comes at the cost of sacrificing the efficiency of channel utilization. For example, TDMA-like protocols incur non-negligible overhead in coordination and synchronization to schedule channel accesses. CSMA-like schemes always conservatively set the size of the backoff windows, leaving a lot of idle slots unused in the channel. The frequency domain solutions [18, 19] enable the sender to hop to an interference-free channel when



(a) Sender→receivers

(b) Receivers→sender

Fig. 2: The PRR of Link 1-6 under different interferences.

interference occurs. However, coordinating the frequency-hopping behavior of senders and receivers generally means considerable overhead, not to mention the possibility that one may fail to find sufficient spectrum resource for channel hopping, especially in the spectrum-crowded environments.

Interference Tolerance. To achieve higher channel utilization, some schemes [20–23] are proposed to tolerate the interferences. For example, CoCo [23] exploits the capture effect phenomenon to achieve concurrent transmission. However, it just blindly tolerates the interference without knowing the characteristics of the interferences, thus achieves limited improvement in channel utilization. Different from CoCo, Smoggy-Link is able to characterize the interference, based on which it can predict the potential chance and benefit of concurrency. This allows Smoggy-Link to achieve high channel utilization in the CTI scenario.

Other works (such as COF [24] and CMAP [25]) enable concurrent transmissions by exploiting the exposed terminal phenomenon. They rely on information exchange among the interferer(s) and the sender(s) to estimate the chance and benefit of concurrency. However, information exchange is infeasible in CTI scenarios. In comparison, Smoggy-Link is able to obtain fine-grained spatiotemporal link information only based on the PHY information, thus it achieves efficient concurrency under all kinds of interferences.

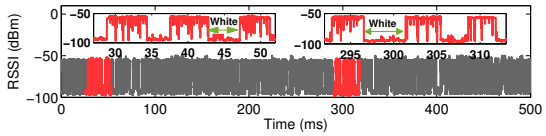
Interference identification. Accurate interference identification is feasible and profitable. For example, ZiSense[26] utilize the featured patterns of different interferences to identify ZigBee signals, which helps to avoid the unnecessary wake-ups. CrossZig[7] features physical layer hints to infer interference patterns and harnesses this to adapt the recovery mechanism. SoNIC[27] proposes a method to classify non-ZigBee interferences, which helps to identify the corrupted bits in a packet. Our work aims to bridge the interference identification and the transmission concurrency. Based on interference identification, we can obtain fine-grained link information which helps to design efficient concurrency approaches.

3 Motivation

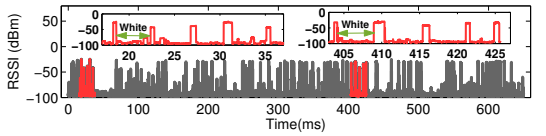
In this section, we introduce the key findings of our empirical studies that motivate our work.

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing



(a) RSSI sequences of interference A



(b) RSSI sequences of interference D

Fig. 3: RSSI sequences of different interference.

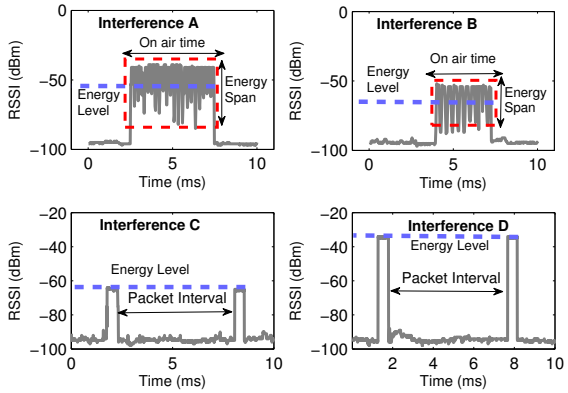


Fig. 4: RSSI patterns of different interference.

3.1 Experiment setup

We randomly deploy seven ZigBee nodes in a room (as shown in Figure 1), where one is the sender and the others are the receivers. Links between the sender and the receivers are denoted as Links 1-6. We also set four interferers in the room, denoted as $A \sim D$, where A and B are WiFi interferers, and C and D are Bluetooth interferers. In the experiment, the sender periodically broadcasts packets at a rate of 100 packets per second. The experiment lasts for 60mins, during which we periodically turn ON each interferer to generate interference signals. The ON/OFF pattern is shown in Figure 2.

3.2 Impact of Interference on Link Quality

Figure 2(a) shows the PRR (packet reception ratio) of each link under different interferences. We can find that a specific interference is likely to have different influences on different links. Thus there is a strong possibility that some of the receivers can successfully decode the sender's packets even under an ongoing interference. Another important observation is that the PRR of each link is not constant, but fluctuates with the interference. The above observation implies that: *if a senders can adaptively select the strongest link based on the ongoing interference, it can get both high throughput and high PRR even under harsh interference.* An example of the selection trace is given by the dotted line in Figure 2(a).

Since the sender also acts as a receiver when receiving the ACKs, we further conduct an experiment to observe qualities of the links on the "receivers→sender" direction. The result is shown in Figure 2(b). By comparing Figures 2(b) and 2(a), we find that links exhibit significant asymmetry: although a sender can transmit concurrently with the interference, ACKs are likely to be corrupted by the ongoing interference. Therefore, ACKs must be scheduled to arrive at the sender during the white spaces (i.e., the idle space) of the interference. This can be achieved only if the distribution of the white space of the interference is predictable.

3.3 Distribution of White/Black Space

In this experiment, we explore the transmission pattern of WiFi and Bluetooth. As an example, Figure 3 shows the collected RSSI sequences of interferences A and D. We find that *signals from both WiFi and Bluetooth exhibit relatively stable black/white alternating pattern*, in a short time duration. To show it more clear, we zoom in two different periods of the RSSI sequence, as shown in the subfigures. Obviously, the white/black spaces are distributed similarly within the two windows. Such property implies that a node can predict the distribution of the white/black spaces in the near future, if it knows who is the ongoing interference.

3.4 Feasibility of Interference Identification

The above discussions tell that a node can get fine-grained spatiotemporal link information if the ongoing interference can be identified. In this subsection, we study characteristics of the interferences. Figures 4(a)-(d) present the RSSI sequences of different interferences sampled by the sender. The figures tell that due to the differences in network standards, hardware specifications and modulation methods, *different interferers generate signals with different RSSI patterns*. There exist several features (in energy and time domain) that can be leveraged to identify different interferences [26]. Table 1 shows a list of selected features. As shown in the table, we use both energy domain features (e.g., the energy level) and time domain features (e.g., the packet interval) for reliable interference identification.

We further evaluate the *stability* of the signal features under different scenarios. In the experiment, we use a ZigBee device (a TelosB node) to sample the signal of a WiFi interferer (a smartphone). Without otherwise specified, the distance between the two devices is 3m. The experiment is conducted under four different scenarios:

- *Static.* This refers to the scenario where both the environment and the locations of the devices are stable. We conduct such an experiment in an empty room.
- *Environmental mobility.* This refers to the scenario where the devices are static but some objects move around it. We conduct this experiment in a cafe during lunch time.
- *Micro mobility.* This refers to the scenario where the WiFi device is moved within a certain range. We conduct such an experiment in an empty room, where a volunteer moves a WiFi device along random trajectory within 1m of its location. This mimics typical usage scenarios of the mobile devices (like in an office) where the device's moving range is usually limited (e.g., on the user's desk).
- *Macro mobility.* This refers to the scenario where the location of the WiFi device changes significantly. In this

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

Feature	Description	Category
(1) Energy span during interference	$range(RSSI_{normalized})$	Short-term features
(2) Energy level during interference	$median(range(RSSI_{normalized}))$	
(3) Energy variance during interference	$variance(RSSI_{normalized})$	
(4) Peak to Average Power Ratio	$\lvert max(RSSI_{normalized}) - mean(RSSI_{normalized})\rvert$	
(5) Average on-air time	Average time span of x consecutive busy sample of RSSI	Long-term features
(6) Average packet interval	Average time span of x consecutive idle sample of RSSI	

TABLE 1: Features utilized by interference identification component.

Feature	Static	Env	Micro	Macro	
Energy level	0.71	2.24	2.52	4.53	(dBm)
Energy span	0.52	0.63	0.67	0.75	
Energy Var	0.09	0.12	0.14	0.18	
PAPR	0.41	0.43	0.47	0.61	
On-air time	0.18	0.21	0.25	0.29	(ms)
Packet interval	0.12	0.13	0.14	0.18	

TABLE 2: Stability of the features under different scenarios.

experiment, the WiFi device is moved around a 5m×10m empty room.

Each experiment lasts for 10min, during which the smartphone generate WiFi signal by transmitting large files (i.e., a movie) and a ZigBee node collects RSSI series of the WiFi signal The received RSSI series are sliced into 5s segments, and signal features showed in Table 1 are calculated for each segment.

Table 2 shows the deviation of each feature under different scenarios. We can see that both environmental mobility and micro movement of the interference source result in only small variation in the signal features. However, a macro movement of the interference can significantly change the features. Fortunately, find that the time domain features, like the packet interval and the on-air time, are almost unaffected by all the influencing factors.

3.5 Summary

Based on the above experiments, we can conclude that:

- *A specific interference is likely to have different influences on different links.* Thus, in comparison to just avoiding the interference, the throughput can be significantly improved if the node can transmit concurrently with the interferences via the strong link.
- *Signals from the interferences exhibit stable black/white alternating pattern.* Thus, nodes can predict the distributions of the white spaces for appropriate transmission schedule.
- *The interferences exhibit distinct and stable RSSI patterns,* which can be used for interference identification.

4 Overview

We in this paper propose Smoggy-Link, a practical approach that exploits concurrency for efficient data transmission under harsh interference. Figure 5 depicts the system architecture, which involves three components: interference identification, link estimation, and data transmission.

Then, we take Figure 6 as an example to show how Smoggy-Link works. As shown in Figure 6(a), the sender S maintains a set of receivers, i.e. Nodes 1-8. When S intends to transmit data, it first conducts channel assessment (CCA)

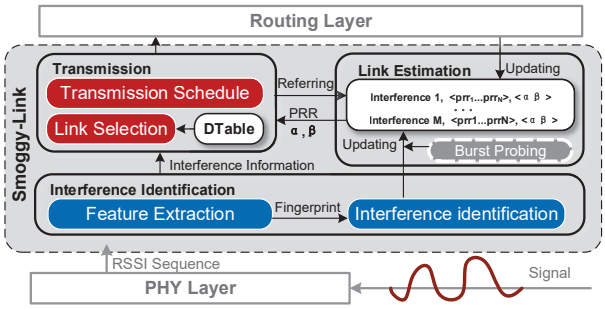
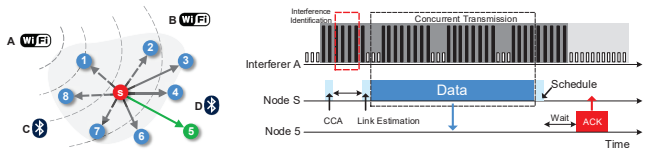


Fig. 5: Framework of Smoggy-Link.



(a) Topology (b) Process of Smoggy-Link

Fig. 6: Overview of Smoggy-Link.

to judge whether the channel is busy. If the channel is busy, S performs channel sampling and extracts features of the sampled RSSI (as shown in Figure 6(b)). Note that it only takes 2∼5 ms for Smoggy-Link to collect enough signal samples for signal feature extraction, so the time overhead for feature extraction is low. Then, based on an *interference identification method*, S knows A is the ongoing interference. According to our *interference-aware link estimation method*, S estimates the influencing pattern of A and infers that Node 5 is the best receiver. It then schedules the transmission of data packets and ACKs on Link 5.

5 Link Estimation

In the link estimation component, the node first identifies the ongoing interference and then translate the interference information to the qualities of the links.

5.1 Interference Identification

As shown in Section 3, almost every radio communication has specific channel features that form a fingerprint for it. A node can distinguish different interferences by using such a fingerprint measured by the built-in RSSI function of ZigBee radios. Formally, the fingerprint of an interference can be represented as a vector $F = \{f^1, ..., f^K\}$, which is calculated from the RSSI segment in a sampling window W. Table 1 lists a set of features that are found feasible to distinguish different interferences [6, 26]. To reduce the time consumption, we try to use only the short-term features,

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

which can be extracted with a short sampling window ($W = 2ms$), for fast interference identification. Here, we assume that the signal in one sampling window comes from only one interferer, since it is unlikely that signals from multiple interferers can coexist and interleave in such a short period of time. In Section 7.3, we will further discuss how Smoggy-Link handle the collision among interferences. Method for extracting the features has been proposed in Zisense [26], and its design detail is therefore omitted here.

After obtaining the fingerprints, we discriminate different interferences based on the *cityblock* distances between their fingerprints. Specifically, we denote the detected interference as I_{det} , and its fingerprint as F_{det}. Suppose there are M interferences in the network environment, denoted as $\mathbf{I} = \{I_1, ..., I_M\}$, and their fingerprints are maintained in a table denoted as $\mathbf{FTable} = \{F_1, ..., F_M\}$. To identify the detected interference, the node first measures the cityblock distance between F_{det} and each fingerprint in **FTable**, and obtains a distance set as $\mathbf{D} = \{d_1, ..., d_m, ..., d_M\}$.

$$d_m = \frac{\sum_{k=1}^{K} |f_{det}^k - f_m^k| / f_m^k}{K}. \quad (1)$$

Then I_{det} will be identified as I_m if the distance between I_{det} and I_m is the minimum. To keep the **FTable** up-to-date, we use moving average to update the **FTable** as:

$$F_m = \lambda \cdot F_m + (1 - \lambda) \cdot F_{det}. \quad (2)$$

Where λ is an adjustable parameter which is set as 0.9 in our implementation. To note that, if i) the **FTable** is currently empty, or ii) $min(\mathbf{D}) > d_{th}$ (where d_{th} is a pre-determined threshold), I_{det} will be treated as a new interference. Then the node would add F_{det} into **FTable**. In the design of Smoggy-Link, we empirically set d_th at 0.1.

Robust Identification. In practice, the energy level of an interference can vary with the change in the environment, which would corrupt the short-term features of a detected interference. Thus the node will incorrectly treat an interference as a new one. To solve this problem, we propose to extend the sampling window to get the long-term features of the interferences (which is proved unaffected by environmental factors in Section 3) if the fast identification fails (i.e., $min(\mathbf{D}) > d_{th}$). The length of the extended sampling window (denoted as W_{ext}) is set as $5ms$, which is long enough to get the long-term features [26]. Then the interference will be re-identified with both the short-term and the long-term features. The flow chart of the interference identification module is illustrated in Figure 7. Note that, the robust identification will incurs 3ms additional delay. However, as shown in [28], one burst of interference signal usually last over tens of milliseconds, so the benefit brought by concurrent transmission apparently outweighs the cost of interference identification.

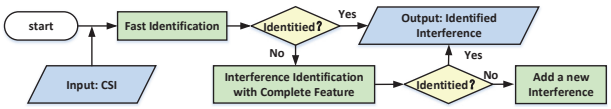


Fig. 7: Flow chart of interference identification module.

5.2 Interference-Aware Link Estimation

The target of the link estimation component is to translate the interference information to: i) PRRs of the outbound links; and ii) the distribution of the white/black spaces.

5.2.1 LinkMap

For computing and updating the PRRs, each sender S constructs a *LinkMap* to maintain PRRs of its outbound links under different interferences, forming as follows:

$$LM_{M\times N} = \begin{bmatrix} prr_{S,L_1}^{I_1} & prr_{S,L_2}^{I_1} & \cdots & prr_{S,L_N}^{I_1} \\ \vdots & \vdots & prr_{S,L_n}^{I_m} & \vdots \\ prr_{S,L_1}^{I_M} & prr_{S,L_2}^{I_M} & \cdots & prr_{S,L_N}^{I_M} \\ prr_{S,L_1}^{\varnothing} & prr_{S,L_2}^{\varnothing} & \cdots & prr_{S,L_N}^{\varnothing} \end{bmatrix} \quad (3)$$

where $\mathbf{L_S} = \{L_1, ..., L_N\}$ is $S's$ outbound links, and $\{I_1, ..., I_M\}$ are the interferences in the environment. $prr_{S,L_n}^{I_m}$ is the PRR of Link L_n under interference I_m. We also maintains an entry (the last entry) for the case that there is no interference. Using the *LinkMap*, a sender S can obtain PRR of the outbound links under a specific interference.

Initially, the *LinkMap* is empty and thus when a new interference is detected, the sender has to rely on burst probing [28] to measure PRR of the outbound links in $\mathbf{L_S}$. Different from the conventional periodic probing where probes are broadcast at a fixed interval, the sender sends a batch of consecutive probes (10 probes in Smoggy-Link) in each burst probing process. The reason to use burst probing is that we need to measure the up-to-date PRRs of links under the current interference, which cannot be captured by periodic probes that utilize history information for PRR estimation. In the meantime, to avoid the collision between the interference and the ACKs, instead of replying the ACKs immediately, the receivers piggyback the ACKs on the future normal data traffic. After receiving the ACKs, the sender can compute the PRRs of the links in $\mathbf{L_S}$ under the new interference, and adds them to the *LinkMap*.

The sender also updates *LinkMap* via burst probing and normal data traffic.

Update via burst probing. Burst probing is triggered by the changes in the features of the interferences. Specifically, qualities of the links are subject to the features of the current interference. Thus, the changes in an interference's features may also change the influencing pattern of the interference. Thus for an interference I_m, if its fingerprint F_m changes, we consider that the PRRs in the LinkMap corresponding to I_m also changes. In this case, the node has to conduct burst probing to update all these PRRs. To implement the above idea, the sender records the fingerprint of I_m after each burst probing as $F_{m\_pre} = \{f_{m\_pre}^1, f_{m\_pre}^2, ..., f_{m\_pre}^K\}$. Each time I_m appears, the sender will calculate the distance between F_m and F_{m_pre}. A distance exceeding a pre-defined threshold will trigger a new round of burst probing.

Updating via normal traffic: If the sender selects Link L_n as the best link and transmits packets through link L_n under a certain interference I_m, it would obtain the PRR for this transmission task denoted as $prr_{S,L_n}^{I_{m\_new}}$. Then $prr_{S,L_n}^{I_m}$ can be updated using weighted moving average as follows:

$$prr_{S,L_n}^{I_m} = \theta \cdot prr_{S,L_n}^{I_m} + (1 - \theta) \cdot prr_{S,L_n}^{I_{m\_new}}. \quad (4)$$

where θ is a tunable parameter which is set as 0.9 in our implementation.

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

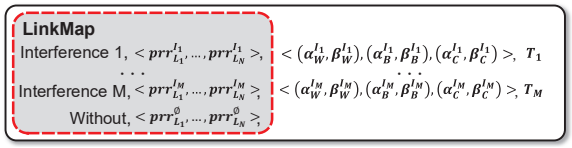


Fig. 8: Interference-link table.

5.2.2 White/black space model

As discussed in Section 3, signals from the interferences exhibit relatively stable black/white alternating pattern. Thus we use the Pareto model to fit the distribution of the white and black spaces as follows:

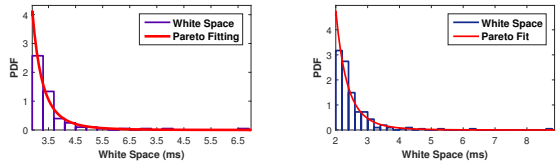
$$P(x > t) = \begin{cases} (\frac{\alpha_W}{t})^{\beta_W} & t > \alpha_W \\ 1 & otherwise \end{cases} \quad (5)$$

where $P(x > t)$ presents the probability that the length of the coming white space is larger than t. α_W and β_W are the scale and shape of the Pareto model, respectively. Specifically, α_W is the minimum length of the white spaces, and β_W is given by $\frac{\lambda_W}{\lambda_W - \alpha_W}$, where λ_W is the average length of the white space. Similarly, we can also model the length of black space (i.e., the white space interval) and the white/black period using Pareto model with parameter (α_B, β_B) and (α_C, β_C).

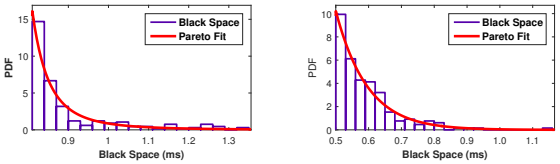
We further conduct a set of experiments to validate our assumption on the Pareto distribution. In the experiments, we collect the RSSI series of WiFi and Bluetooth signals, and extract white and black spaces of the signals. Then we divide the series into equal-sized segments (with a length of 200ms). For each segment, a Pareto distribution is fitted using maximum likelihood estimation (MLE). Figures 9(a)-9(d) show the fitting results for arbitrary segments of WiFi and Bluetooth signals. We can see that the estimated Pareto distribution curves fit well with the measurement results. We further apply the K-S test for each segment to quantify the goodness-of-fit of the estimated Pareto distribution. The test results show that for all the four cases (white and black spaces of both WiFi and Bluetooth signals), there are more than 80% segments pass the K-S test.

Measurement of α and β. When a new interference is detected, the node samples the channel and measures the interval between two interference signals to build the white space model. Since the on-air time of ZigBee ACK is at least $200\mu s$ [4], thus only the interval that is longer than $200\mu s$ can be treated as a sample of white space. The spaces between white spaces are treated as black spaces. Then α and β can be calculated based on the minimum and average length of the white and black spaces. The size of the sampling window is set at $30ms$. Note that the value of α and β is also periodically updated. Specifically, for each interference I_m, if its α and β have not been updated for a certain periods of time (e.g., 10 min in our paper), SmoggyLink will resample the signal of I_m and update α and β.

The values of α and β, together with the *LinkMap*, are stored in a **ILTable** (interference-link table, as shown in Figrue 8) which provides fine-grained to data transmission component for further decision making. We set a timer T_m for each interferer I_m. T_m records the time duration since the last appearance of I_m. If T_m exceeds a predefined value



(a) White space of WiFi (b) White space of Bluetooth



(c) Black space of WiFi (d) Black space of Bluetooth

Fig. 9: Fitting results for WiFi and Bluetooth signals.

(such as half an hour), I_m will be considered as moved away or turned OFF, and thus be deleted from the **ILTable**.

6 Data Transmission

Targets of the data transmission component are two folds: i) selecting the best link for concurrent transmission; and ii) schedule the transmission of data and ACKs.

6.1 Interference-Adaptive Transmission

Before a transmission task, a node should first determine the transmission mode, which refers to: i) the selected best link, denoted as L_{best}; and ii) the option of concurrent transmission (denoted as H_c) or backoff transmission (denoted as H_b) through L_{best}. To this end, we propose an adaptive transmission method, which enables a node to predict the transmission capability and energy efficiency of each potential transmission mode, and then make a decision (i.e., select an appropriate $\langle L, H \rangle$, $L_n \in \mathbf{L_s}$, $H = H_c$ *or* H_b) based on the predict result.

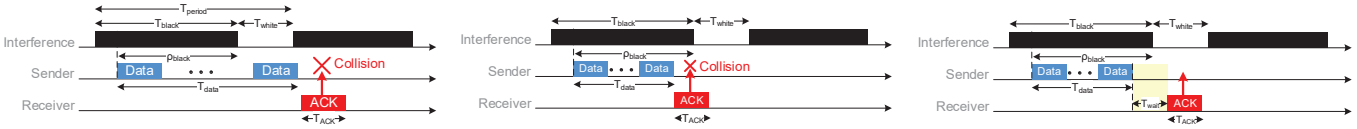
We define the transmission capability as the expected number of successfully transmitted packets in one white/black period. Thus under a specific interference I_m, a sender can estimate the transmission capability of each mode $\langle L_n, H \rangle$ as follows:

$$\begin{aligned} C_c^{(n)} &= \sum_{i=0}^{N_c} N_c \cdot \sum_{k=0}^{i} \binom{\text{N-c-N-b}}{k} \cdot (prr_{S,L_n}^{I_m})^k \cdot (1 - prr_{S,L_n}^{I_m})^{\text{N-c-N-b-k}} \\ &\quad + \sum_{i=0}^{N_b} N_b \cdot \sum_{k=0}^{i} \binom{N_b}{i-k} \cdot (prr_{S,L_n}^{\varnothing})^{i-k} \cdot (1 - prr_{S,L_n}^{\varnothing})^{\text{N-b-(i-k)}}. \\ C_b^{(n)} &= \sum_{i=0}^{N_b} N_b \cdot \binom{N_b}{i} \cdot (prr_{S,L_n}^{\varnothing})^i \cdot (1 - prr_{S,L_n}^{\varnothing})^{N_b - i}. \end{aligned} \quad (6)$$

where $C_c^{(n)}$ and $C_b^{(n)}$ are the espected transmission capabilities of link L_n for concurrent and backoff transmission, respectively. N_c and N_b are the maximum number of the successfully transmitted packets for concurrent and backoff transmission in one white/black period. Assuming that the time for transmitting one packet is T_p, then N_c and N_b can be estimated as $N_c = \frac{T_{black} + T_{white}}{T_p}$ and $N_b = \frac{T_{white}}{T_p}$, respectively. T_{black} and T_{white} can be obtained from the

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing



(a) Example of *Case I* (b) Example of *Case II* (c) Example of data and ACK scheduling

Fig. 10: Two collision cases and the solution.

inverse Pareto function as $T_{black} = P_b^{-1}(p) = \frac{\alpha_B}{\sqrt[\beta_B]{p}}$ and $T_{white} = P_w^{-1}(p) = \frac{\alpha_W}{\sqrt[\beta_W]{p}}$, which give the lengths of black space and white space for a given confidence interval p.

In addition, assuming the energy cost for transmitting one packet is E_{trans}, the energy efficiency of mode $\langle L_n, H_c\rangle$ and $\langle L_n, H_b\rangle$ can be calculated as $E_c^{(n)} = \frac{C_c^{(n)}}{E_{trans}\cdot N_c}$ and $E_b^{(n)} = \frac{C_b^{(n)}}{E_{trans}\cdot N_b}$, respectively.

Then the sender can obtain the transmission capabilities and energy efficiencies of all the transmission modes as:

$$\begin{bmatrix} \langle C_c^{(1)}, E_c^{(1)}\rangle & \langle C_c^{(2)}, E_c^{(2)}\rangle & \cdots & \langle C_c^{(N)}, E_c^{(N)}\rangle \\ \langle C_b^{(1)}, E_b^{(1)}\rangle & \langle C_b^{(2)}, E_b^{(2)}\rangle & \cdots & \langle C_b^{(N)}, E_b^{(N)}\rangle \end{bmatrix} \tag{7}$$

where $\langle C_c^{(N)}, E_c^{(N)}\rangle$ is the capability and energy efficiency of mode $\langle L_N, H_c\rangle$, and $\langle C_b^{(N)}, E_b^{(N)}\rangle$ is those of mode $\langle L_N, H_b\rangle$. Given a predefined energy efficient constraint E_{th}, a sender would determine the transmission mode as follows: i) it first screens out the modes with poor energy efficiency (i.e., with $E < E_{th}$), and then ii) it selects the mode with the highest capability as the best mode. If mode $\langle L_n, H_c\rangle$ is selected, the sender would immediately transmit data packets through link L_n. If mode $\langle L_n, H_b\rangle$ is selected, the sender would makes a backoff.

6.2 Schedule of Data and ACKs

Collisions between the ACK and the interference can be classified into two cases: i) the ACK collides with the black space in the next black/white period, which caused by a long data period T_{data}, as shown in Figure 10(a); and ii) the ACK collides with the black space in the current black/white period, which caused by a short data period T_{data}, as shown in Figure 10(b). To avoid collision, we should first limit the length of T_{data} and then adjust the arrive time (i.e. T_{wait}) of ACKs as shown in Figure 10(c).

6.2.1 Limiting the length of data period

As shown in Figure 10(a), to avoid *Case I*, the length of data period T_{data} must satisfies that:

$$T_{data} < \rho_{black} + T_{white} \tag{8}$$

where ρ_{black} is the remaining duration of T_{black} upon the start of data transmission. Clearly, *Case I* occurs if the first data packet arrives $T_{period} - T_{data}$ later than the start of the current black space (T_{period} is the length of one white/black period as shown in Figure 10(a)). Assume that ρ_{black} is uniformly distributed over the entire black space, the probability of *Case I* can be estimated as $\min\{\frac{T_{data}}{T_{period}}, 1\}$.

Based on the model in Section 5.2, the expected probability of *Case I* is given by

$$P_{C_1}(T_{data}) = 1 - \frac{1}{\beta_C}\left(\frac{\alpha_C}{T_{data}}\right)^{\beta_C - 1} \tag{9}$$

Given a specific probability threshold C_{th}, T_{data} must satisfies:

$$P_{C_1}(T_{data}) < C_{th} \tag{10}$$

By solving Equation (10), we have

$$T_{data} < T_{data}^{(max)} = \frac{\alpha}{(\beta\cdot(1-C_{th}))^{\frac{1}{\beta-1}}} \tag{11}$$

Therefore, before a sender starts a sending task with n data packet, it should first compare the required $T_{data} = n\cdot T_p$ with $T_{data}^{(max)}$. If $T_{data} > T_{data}^{(max)}$, the sender has to reduce the packet number in the current white/black period utile the T_{data} satisfy Equation (11).

6.2.2 Adjusting the arrive time of the ACK

As shown in Figure 10(b), *Case II* occurs if $T_{data} < \rho_{black}$. In this case, the sender has to predict the arrival of the next white space, and notifies the receiver to wait for T_{wait} before replying the ACK. Here T_{wait} should satisfies:

$$T_{data} + T_{wait} > \rho_{black} \tag{12}$$

Thus, the probability of *Case II* can be estimated as $\min\{\frac{T_{black}-(T_{data}+T_{wait})}{T_{black}}, 1\}$. Then the expected collision probability can be given by

$$P_{C_2}(T_{wait}) = \frac{1}{\beta_B}\left(\frac{\alpha_B}{T_{data}+T_{wait}}\right)^{\beta_B - 1} \tag{13}$$

Given a collision probability threshold C_{th}, T_{waite} must satisfies:

$$P_{C_2}(T_{wait}) < C_{th} \tag{14}$$

Thus we set a constrain on T_{wait} as follows:

$$T_{wait} > T_{wait}^{(min)} = \frac{\alpha_B}{(\beta_B\cdot C_{th})^{\frac{1}{\beta_B-1}}} - T_{data} \tag{15}$$

$T_{wait}^{(min)}$ is piggybacked on the last data packet sent to the receiver. If $T_{wait}^{(min)} \le 0$, the receiver can rely the ACKs immediately. Otherwise, it has to wait at least $T_{wait}^{(min)}$ before replying the ACKs.

Smoggy-Link uses one ACK to reply all the packets transmitted in one white/black period. Consider that the minimum packet transmission time of ZigBee is $700\mu s$ and the length of one white/black period is typically shorter than $10ms$, the number of packets transmitted in one period must be less than 15. So we use a 15-bit payload to describe reception of the data packets, where '1' indicates failure and '0' indicates success. To achieve this, each data packet has to piggyback the information about its sequence number and the number of the packets to be transmitted in this period.

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

7 Practical Issues

7.1 Applicability

Smoggy-Link adopts a best effort design. In the absence of interferences, it acts like the conventional ZigBee, but when interferences occur, it tries to transmit concurrently with them. Of course there are some cases that the performance gain of concurrency may not so significantly.

Particularly high density of interference. The particularly high density of the interference may leads to two problems: i) continuous collision among interference, which makes Smoggy-Link unable to identify the interference and thus the concurrency fails; ii) the poor PRR for all the candidate links. In this case, Smoggy-Link can hardly find out a interference-free link for concurrent transmission.

Particularly high mobility of the interference. When an interference exhibit significant mobility, its fingerprint changes and thus Smoggy-Link will treat it as a new interference. So, in the scenarios where all the interferences keep moving, Smoggy-Link will stay in the training phase (i.e., keeps updating the fingerprint table and the LinkMap). This leads to low concurrency efficiency.

Particularly low traffic load of the interference. In the environment where the traffic load of the interferences is low, the channel is idle most of the time. The benefit of concurrency is marginal in this case.

In Section 8, we conduct a series of real-world experiences, which show that Smoggy-Link works well in most typical application scenarios of sensor networks.

7.2 Generalization

Although this paper demonstrates Smoggy-Link's benefits in making the ZigBee survive from the interference of WiFi and Bluetooth, the idea of using exposed terminal phenomenon for wireless concurrency can be generalized to other scenarios since exposed terminal is a common phenomenon in all wireless networks. For example, based on CSI sensing, WiFi device can extract signal features of the interference and infer its transmission pattern. So WiFi device can still use Smoggy-Link's idea to survive in the interference of ZigBee and Bluetooth.

7.3 Collision among Interferences

In some cases, interference signal may collide due to the hidden terminal problem. When a collision is detected, the node have to directly avoid the interference since it cannot identify the interference based on the collision signal. We conduct experiments in three representative real-word environments (i.e., lab, cafe and dorm) to observe the probability of collision between interferences. The result is shown in Table 3. We can see that Smoggy-Link can always find more than 92% chance for concurrency.

TABLE 3: Probability of collision among interferences.

Dorm	Lab	Cafe
3.24%	1.26%	7.58%

We further investigate how a node detects collisions. Figure 11 shows examples of RSSI segments with single and multiple interferences. We can see that signal collision leads to changes in signal patterns. Therefore, a RSSI segment can be seen involving collision if we can find a sudden change in either RSSI variance (such as Case II) or RSSI level (such as Case III) of the busy period of the RSSI samples.

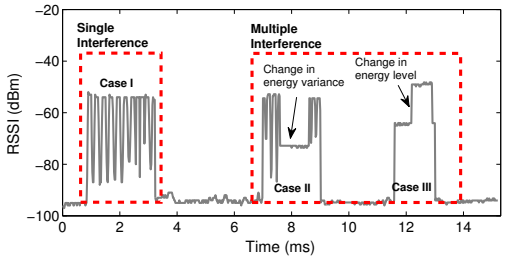


Fig. 11: Example of the collision between interferences.

7.4 Variation in Interference Features

Smoggy-Link is able to handle the variation in the interference features. In this subsection, we discuss how Smoggy-Link handles the following three types of variation.

Slow variations: Slow variations are usually caused by changes in the surrounding environment. This kind of variation can be captured by the moving average method as described in Section 5.1.

Small variations: Small variation is usually caused by the mobility of the interference source within a small area, or the people's movements around the interference source. We will show in Section 8 that the small variations of the signal feature only slightly affect the performance of Smoggy-Link.

Apparent variation: When the location of interference source is significantly changed, the features of the interference will exhibit apparent changes, as discussed in Section 3. In this case, Smoggy-Link will treat the moved interference as a new one, and triggers burst probing for a new round of link estimation. This is reasonable because such significant changes in the interference features will also change the interference's influence on the links' transmissions. Therefore, we should treat the interference as a new one.

7.5 Impact of Smoggy-Link on ongoing transmissions

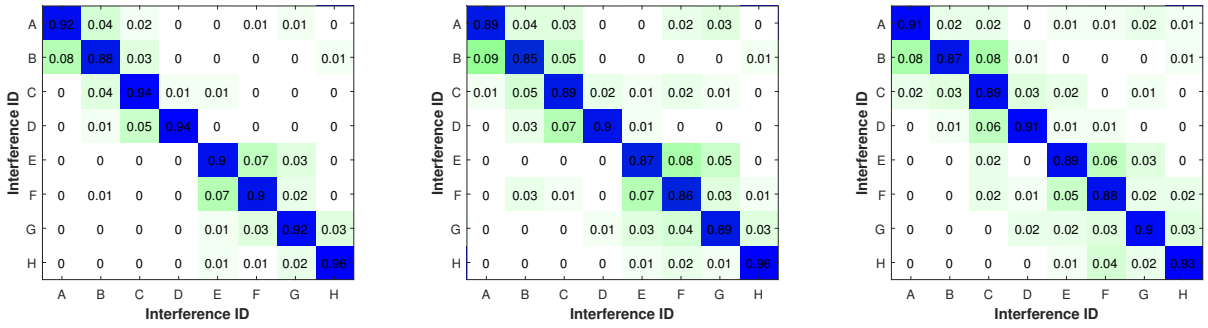
Although Smoggy-Link improves the throughput of ZigBee, it might hurt the packet reception rate of either ZigBee itself or the interfering device. Then we discuss how Smoggy-Link impacts ZigBee, WiFi, and Bluetooth:

ZigBee. In the design of Smoggy-Link, the sender tries to find an interference-free link for concurrent transmission with the interference. However, the error in either interference identification or link estimation may lead to false concurrency, which further leads to signal collision and thus high retransmission. Fortunately, our experimental results in Section 8.4 show that the retransmission of Smoggy-Link is only slightly higher than that of the CSMA-based method, but the throughput of Smoggy-Link increases by near 100%.

WiFi. To enable concurrency, Smoggy-Link requires that the interferer, which locates within the CCA range of ZigBee sender, locates outside the interference range of ZigBee receiver. Thus if concurrency is permitted, namely any one of the potential receivers of a ZigBee sender can successfully

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing



(a) Static (b) Environmental mobility (c) Micro mobility

Fig. 12: Performance of interference identification component under different scenarios.

decode the sender's packet under the interference of WiFi, it is likely that the sender is located at the boundary of WiFi's communication range. Since the transmit power of WiFi devices is much higher than that of ZigBee (the maximum transmit powers of WiFi and ZigBee are 14 and 0 dBm, respectively), the communication range of WiFi is much broader than that of ZigBee. Thus we can infer that the WiFi interferer is probably located outside the interference range of the ZigBee sender and thus the transmission of ZigBee would have little impact on the performance of WiFi.

Bluetooth. Bluetooth uses the adaptive frequency hopping technique across 79 MHz of bandwidth in the 2.4 GHz band. The hopping occurs at a rate of 1600 hops/s, hence it occupies a 1 MHz channel for only 625 μs. This improves its robustness against the interference from other 2.4GHz technologies. In addition, since Bluetooth spans over the whole 2.4 GHz ISM band while ZigBee only occupies a 2 MHz channel during transmissions, ZigBee would have little impact on Bluetooth during the concurrency.

7.6 Handling other interferences

In-network interference. Although Smoggy-Link is designed to mitigate cross-technology interference, it can also handle in-network interferences (i.e., the interference from other ZigBee devices). First, similar to other wireless technologies, ZigBee signal also has its own features that can be used to detect the presence of ZigBee interference, as shown in [26]. Then, once a ZigBee interference is detected, a node can directly identify the ZigBee device by overhearing, as the author of [24] did in their work. After this, Smoggy-Link can perform link selection and transmission schedule as it does with other interference.

Non-standard interference. Indeed, besides the signal from standardized communication technologies, e.g. WiFi and Bluetooth, non-standard signal (e.g., the signal from microwave ovens) is also a source of interference. According to the result in [26], such non-standard signal also has very distinct features that can be used as a fingerprint for interference identification. However, since traffic pattern of the non-standard interference is not stable and hard to characterize, it is difficult to perform accurate ACK transmission schedule when microwave signal occurs. Therefore, once non-standard signal is detected, the ZigBee sender can directly choose to avoid such an interference.

8 Implementation and Evaluation

8.1 Implementation

Smoggy-Link is implemented on TelosB motes under TinyOS 2.1.2. In our implementation, we increase the register reading rate to capture the RSSI features of the interference signal. Specifically, we increase the SPI speed and simplify the interfaces to quickly fetch RSSI readings from the register. The sampling frequency is increased to 31.25KHz. The RSSI.RSSI_VAL register in CC22420 always has valid RSSI value when reception has been enabled at least 8 symbol periods (128μs). Hence, our implementation does not change the hardware RSSI sampling and just increase the register reading rate, which does not incur much extra energy consumption [26].

The implementation of Smoggy-Link takes more space to store the RSSI sequence and algorithm codes. It consumes extra 1233 bytes RAM and 8123 bytes ROM, 54.1% and 57.1% more than the default implementation. However, the extra consumption is usually affordable for most of the sensor application programs on TelosB motes.

8.2 Experiment setup

The experiment parameters are specified in Table 4, and the detailed settings are as follows.

TABLE 4: Summary of parameters in Smoggy-Link.

Power	Throughput	Ch.	W	λ	θ	E_{th}
Level 5	$2Kbps$	12	2 ms	0.9	0.9	$0.1/mJ$

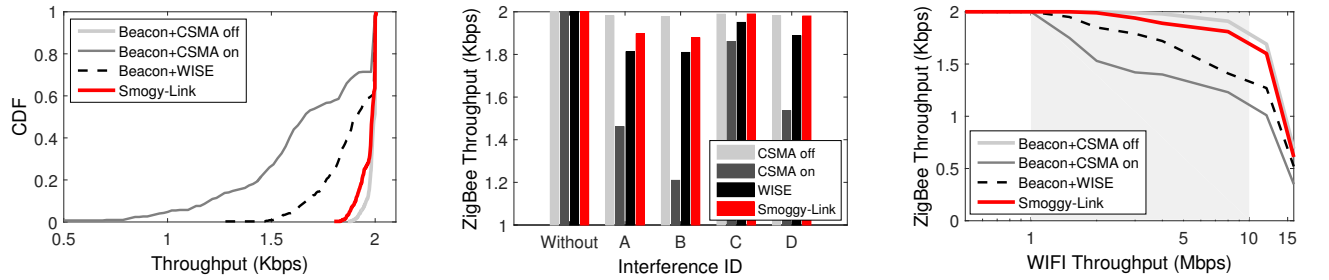
Interfering Technologies. We focus on two interferer technologies that are prevalent in today's environments: WiFi and Bluetooth. Specifically, we generate WiFi signal using a TP-Link WR740N router as AP and an iPhone 6S as clint. The smartphone download a large file at a data rate of $2 \sim 8Mbps$ during the experiment. We generate Bluetooth signal using two iPhone 6S smartphones transferring a large file at a data rate of $1 \sim 2Mbps$.

Compared schemes:

- *Beacon+CSMA off:* the method with CSMA disabled using beacon-based link estimation.
- *Beacon+CSMA on:* the method using beacon-based link estimation and using CSMA for interference avoidance.

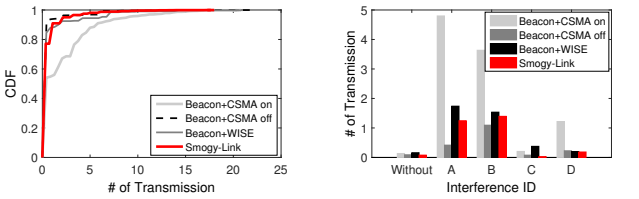
1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing



(a) Distribution (b) Under different interference (c) ZigBee throughput vs WiFi throughput

Fig. 13: Comparison of schemes on throughput.



(a) Distribution (b) Under different interference

Fig. 14: Comparison of schemes on # of retransmissions.

- *Beacon+WISE:* the method using beacon-based link estimation and using WISE [4][1] for interference avoidance.

Smoggy-Link and Beacon+CSMA off adopt interference concurrency. Beacon+CSMA on and Beacon+WISE adopt interference avoidance. Here, the beacon-based link estimation refers to the method which estimate link quality only based on periodical probing. Such method is inapplicable in the interference-rich environment where the ACKs will collide with the interference, leading to link estimation error.

8.3 Evaluation of interference identification module

TABLE 5: The settings of the interferences.

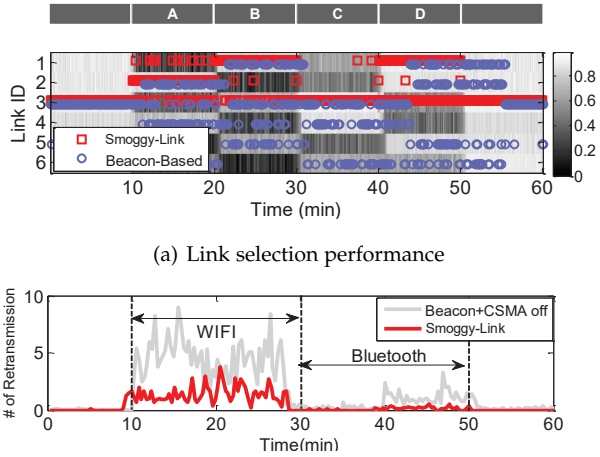
ID	A	B	C	D	E	F	G	H
Distance (m)	15	10	5	0.5	15	5	10	0.5
Technology	WIFI				Bluetooth		ZigBee	

We first evaluate the performance of the interference identification module. In the experiment, we have 8 different interferers. The interferers take turn to generate interference for 10 min individually, during which the node collects the RSSI segments of the ongoing interference and identify it. Detailed setting of the 8 interferences is shown in Table 5.

We conduct the experiment in three scenarios: i) the static scenario where the devices are placed in a empty room; ii) the environmental mobility scenario where peoples are allowed to walk around in the room; and iii) the micro mobility scenario where we picked up the devices and moved it around within one meter of its location.

Figure 12 (a) shows the confusion matrix of interference identification in the static scenario. It shows that the identification of bluetooth has lower accuracy than that of WiFi and ZigBee. This is primarily because that the energy

1. WISE is a interference avoidance based protocol which enable ZigBee to transmit on the white space of the interference



(a) Link selection performance

(b) Number of retransmission

Fig. 15: Comparison of schemes on link estimation accuracy.

level of Bluetooth is not so stable as that of WIFI and ZigBee. Further, by comparing Figures 12 (a)-(c), we find that both the environmental mobility and micro mobility lead to performance degradation. Fortunately, the identification accuracy is consistently higher than 85% in all the cases.

TABLE 6: Identification delay under different scenarios.

Static	Env	Micro
2.9ms	3.5ms	3.8ms

Table 6 further shows the identification delay under different scenarios. We can see that the mobility incurs slight increase in identification delay. This is because that Smoggy-Link will trigger the robust identification module more frequently under mobility scenarios.

8.4 Network Performance

We use the same testbed in Figure 1 to evaluate the network performance of Smoggy-Link. In the experiment, the sender periodically broadcasts packets at a rate of $2Kbps$. The experiment lasts for 60min, during which we periodically turn ON the interferers to generate interference signal.

Throughput. Figure 13 compares the throughput of different schemes. As shown in Figure 13(a), the throughput of interference avoidance based methods (i.e., Beacon+'CSMA on' and Beacon+WISE) is far less than that of concurrency

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

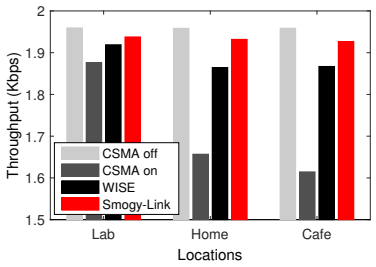


Fig. 16: Throughput comparison under different scenarios.

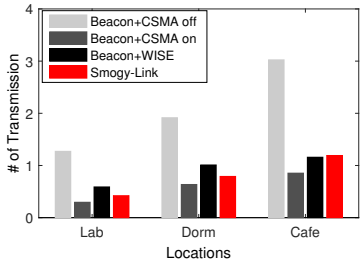


Fig. 17: Number of re-transmission under different scenarios.

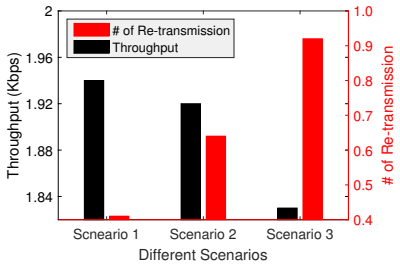


Fig. 18: Performance of Smoggy-Link under dynamic environments.

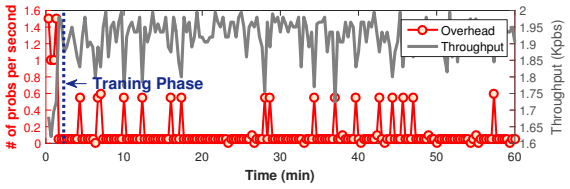


Fig. 19: Performance of Smoggy-Link in cafe.

methods (i.e., Beacon+'CSMA off' and Smoggy-Link). Beacon+'CSMA off' attains the largest throughput, because it can maximally exploit the concurrent opportunities by completely turn off the CSMA. This however leads to poor PRR and thus high energy consumption for packet retransmissions. Compared with the interference avoidance based methods, Smoggy-Link can improve the throughput by 33%, because Smoggy-Link can quickly confirm the feasibility of exploiting concurrent opportunity by considering the PRR of the outbound links under the current interference.

In addition, Figure 13(b) shows how different interference impact the throughput of ZigBee. We can see that, throughput of Smoggy-Link only decreases slightly even under strong interferences (such as Interference A and B). While the interference avoidance based methods suffer serious decline on throughput, especially under strong interferences (with a degradation of 45% compared with the scenario that there is no interference). This phenomenon is illustrated more clearly in Figure 13(c) which shows the impact of WiFi data rate (iperf [29] is used to generate WiFi traffic at different rates) on the throughput of ZigBee. We can see that Smoggy-Link achieves high throughput when WiFi data rate $\leq 10Mbps$. When WiFi data rate $> 10Mbps$, all the four methods suffers serious throughput decline.

Retransmission. Although concurrent transmission is adopted in Smoggy-Link, the retransmission count is comparable with that of interference avoidance based approaches, as shown in Figure 14. Figure 14(a) shows that as Smoggy-Link can adaptively select the strongest link under different interferences, its cost for retransmissions is obviously less than that of Beacon+'CSMA off' and almost equal to that of interference avoidance based approaches. In addition, Figure 14(b) shows that even under the strong interferences, the transmission count of Smoggy-Link increases slightly compared with that of Beacon+ 'CSMA on'.

Figure 15 depicts an explanation of Figure 14. The figure tells that: (i) Smoggy-Link makes better link selection and thus (ii) experiences lower transmission cost. In Figure 15(a), the matrix shows the PRR of different links under different interferences. The square and circle marks the link selection trace of Smoggy-Link and beacon-based estimator, respectively. We can see that Smoggy-Link always select the best link and thus its transmission cost is consistently less than that of beacon-based estimator (Figure 15(b)). In addition, collision between the ACK and the interferences would undermine the performance of the beacon-based estimator.

8.5 Real World Experiment

In this section, we conduct experiments under practical environment. We deploy 5 nodes in an office, a dorm and a cafe. The transmission power is set at level 8 to assure NLOS propagation. Other parameters are kept the same as previous experiments.

Results on throughput (Figure 16) suggests that Smoggy-Link's performance gain is more significant in the cafe and dorm than in the office. This is because that the interference in cafe and dorm is more serious than that in the office. For example, users always watching movie or playing computer games in the cafe or dorm, which create excessive amount of WiFi interference. In the office, however, users just use WiFi to send email or download documents. High load interference would force the avoidance based methods to suppress transmissions and thus leads to poor network throughput. Results on number of retransmissions is shown in Figure 17. Due to the same reason as in the throughput case, smaller degree of reduction is observed in the office, compared with what is achieved in the cafe and dorm.

To further evaluate the robustness of Smoggy-Link in dynamic environments, we repeat the experiment in our office under three different scenarios: i) 10 volunteers move around the office; ii) 5 volunteers move around the office; iii) all the 10 volunteers sit on their seats. The experimental results are shown in Figure 18. We can see that the throughput of Smoggy-Link only slightly decreases when the environment becomes highly dynamic.

We further evaluate how the training phase affects the performance of Smoggy-Link, in term of the communication overhead and the throughput. Figure 19 shows the experimental results in cafe. We can observe that: i) the throughput of Smoggy-Link increases significantly after the training phase; ii) the time cost of the training phase is low (about 3 mins). Obviously, the training phase has ignorable impact on the overall throughput of Smoggy-Link. We further plot the time costs of the training phases under different scenarios[2] in Figure 20. We can see that the time cost keeps

2. Offices 1, 2, and 3 refer to the experiment scenarios in the previous experiment).

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

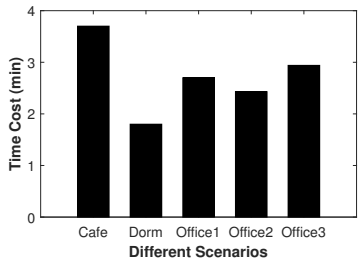


Fig. 20: Time cost of the training phase under different scenarios.

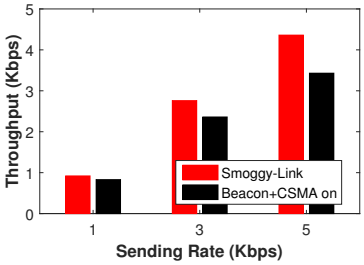


Fig. 21: Throughput comparison in multi-hop network.

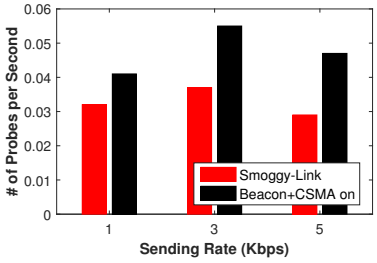


Fig. 22: Overhead comparison in multi-hop network.

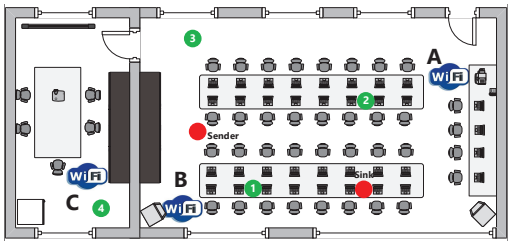


Fig. 23: Multi-hop experiment setup.

less than 4 minutes under all the scenarios.

8.6 Evaluation in a multi-hop network

Considering that sensor networks are often multi-hop, we evaluate the performance of Smoggy-Link in a multi-hop network deployed in our office. In the experiments, 6 nodes form a two-hop network, where two of them are the sender and the sink, as shown in Figure 23. Node 1, Node 2 and the sender are first-layer nodes. Nodes 3 and 4 are second-layer nodes. We use CTP (Collection Tree Protocol) as the relaying protocol. We can see that for the sender, only Nodes 1 and 2 can be considered as a forwarder. As a result, when both Interferences A and B are on, the sender cannot find an interference-free link for transmission.

In the experiments, we tune data rate of the sender from 1Kbps to 5Kbps, and compare the throughput and overhead of Smoggy-Link and Beacon+'CSMA on' under different data rates. The results are shown in Figures 21 and 22. We can see that the throughput of Smoggy-Link is always higher than that of Beacon+'CSMA on', especially when the data rate of the sender is high. The overhead of Smoggy-Link is lower than that of Beacon+'CSMA on'.

9 Conclusion

In this paper, we propose Smoggy-Link to exploit potential concurrency for adaptive transmission under interference. Smoggy-Link can use low-cost interference information to get fine-grained link information which is further utilized for adaptive link selection and intelligent transmission schedule under different interference. Experimental results in indoor and outdoor scenarios show that Smoggy-Link can significantly improve the performance of the state-of-the-art interference avoidance schemes.

Acknowledgments

This work was supported by National Basic Research Program (973 program) under Grant of 2014CB347800, National Natural Science Fund of China for Excellent Young Scientist No.61422207, The NSFC (61672428, 61902213, 61672320, 61602381, 61572402).

References

[1] S. Gollakota, F. Adib, D. Katabi, and S. Seshan. Clearing the RF Smog: Making 802.11n Robust to Cross-technology Interference. In *ACM SIGCOMM*, pages 170–181, 2011.

[2] X. Zhang and K. G. Shin. Enabling Coexistence of Heterogeneous Wireless Systems: Case for Zigbee and Wifi. In *ACM MobiHoc*, pages 3967–3974, 2011.

[3] B. Radunovic, R. Chandra, and D. Gunawardena. Weeble: Enabling Low-power Nodes to Coexist with High-power Nodes in White Space Networks. In *ACM CONEXT*, pages 205–216, 2012.

[4] J. Huang, G. Xing, G. Zhou, and R. Zhou. Beyond Co-existence: Exploiting WiFi White Space for ZigBee Performance Assurance. In *IEEE ICNP*, pages 305–314, 2010.

[5] Y. Yan, P. Yang, X. Li, Y. Tao, L. Zhang, and L. You. ZIMO: Building Cross-Technology MIMO to Harmonize ZigBee Smog with WiFi Flash without Intervention. In *ACM MobiCom*, pages 465–476, 2013.

[6] A. Hithnawi, H. Shafagh, and S. Duquennoy. TIIM: Technology-independent Interference Mitigation for Low-power Wireless Networks. In *ACM/IEEE IPSN*, pages 1–12, 2015.

[7] A. Hithnawi, S. Li, H. Shafagh, J. Gross, and S. Duquennoy. CrossZig: Combating Cross-Technology Interference in Low-power Wireless Networks. In *ACM/IEEE IPSN*, pages 1–12, 2016.

[8] H. Rahul, N. Kushman, D. Katabi, C. Sodini, and F. Edalat. Learning to Share: Narrowband-friendly Wideband Networks. In *ACM SIGCOMM*, pages 147–158, 2008.

[9] L. Yang, W. Hou, L. Cao, B. Y. Zhao, and H. Zheng. Supporting Demanding Wireless Applications with Frequency-agile Radios. In *USENIX NSDI*, pages 65–80, 2010.

[10] R. Musaloiu-E and A. Terzis. Minimising the Effect of WiFi Interference in 802.15.4 Wireless Sensor Networks. *International Journal of Sensor Networks*, 3(1):43–54, 2008.

[11] M. Garetto, T. Salonidis, and E. W. Knightly. Modeling Per-flow Throughput and Capturing Starvation in CSMA Multi-hop Wireless Networks. *IEEE/ACM Transactions on Networking*, 16(4):864–877, 2008.

[12] X. Zhang and K. G. Shin. Gap Sense: Lightweight Coordination of Heterogeneous Wireless Devices. In *IEEE INFOCOM*, pages 3094 – 3101, 2013.

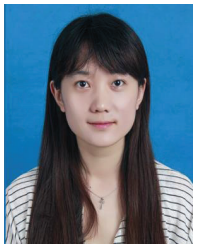
[13] Z. Li, Y. Xie, M. Li, and K. Jamieson. Recitation: Rehearsing Wireless Packet Reception in Software. In *ACM MobiCom*, pages 291–303, 2015.

[14] X. Zheng, J. Wang, W. Dong, Y. He, and Y. Liu. Bulk Data Dissemination in Wireless Sensor Networks: Analysis, Implications and Improvement. *IEEE Transactions on Computers*, 65(5):1428–1439, 2016.

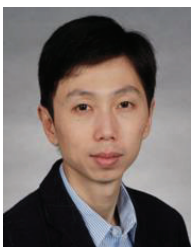
1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.

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.2020.2978205, IEEE Transactions on Mobile Computing

[15] K. Zeng, Z. Yang, and W. Lou. Opportunistic routing in multi-radio multi-channel multi-hop wireless networks. *IEEE Transactions on Wireless Communications*, 9(11):5328–5340, 2010.
[16] Y. Hou, M. Li, X. Yuan, Y. Hou, and W. Lou. Cooperative Interference Mitigation for Heterogeneous Multi-hop MIMO Wireless Networks. *IEEE Transactions on Wireless Communications*, 15(8):5328–5340, 2016.
[17] J. Kwak, C. Lee, and D. Y. Eun. A High-order Markov Chain Based Scheduling Algorithm for Low Delay in CSMA Networks. In *IEEE INFOCOM*, pages 1662–1670, 2014.
[18] H. Rahul, N. Kushman, D. Katabi, C. Sodini, and F. Edalat. Learning to Share: Narrowband-Friendly Wideband Networks. In *ACM SIGCOMM*, pages 147–158, 2008.
[19] L. Yang, W. Hou, L. Cao, B. Y. Zhao, and H. Zheng. Supporting Demanding Wireless Applications with Frequency-Agile Radios. In *USENIX NSDI*, 2010.
[20] J. Lu and K. Whitehouse. Flash Flooding: Exploiting the Capture Effect for Rapid Flooding in Wireless Sensor Networks. In *IEEE INFOCOM*, pages 2491–2499, 2009.
[21] X. Zhang and K. G. Shin. Chorus: Collision resolution for efficient wireless broadcast. In *IEEE INFOCOM*, pages 1747–1755, 2011.
[22] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. Efficient network flooding and time synchronization with glossy. In *ACM/IEEE IPSN*, pages 73–84, 2011.
[23] X. Ji, Y. He, J. Wang, K. Wu, K. Yi, and Y. Liu. Voice over the Dins: Improving Wireless Channel Utilization with Collision Tolerance. In *IEEE ICNP*, pages 1–10, 2013.
[24] D. Liu, M. Hou, Z. Cao, Y. He, X. Ji, and X. Zheng. COF: Exploiting Concurrency for Low Power Opportunistic Forwarding. In *IEEE ICNP*, pages 32–42, 2015.
[25] M. Vutukuru, K. Jamieson, and H. Balakrishnan. Harnessing Exposed Terminals in Wireless Networks. In *USENIX NSDI*, 2008.
[26] X. Zheng, Z. Cao, J. Wang, Y. He, and Y. Liu. ZiSense: Towards Interference Resilient Duty Cycling in Wireless Sensor Networks. In *ACM SenSys*, pages 119–133, 2014.
[27] F. Hermans, O. Rensfelt, T. Voigt, and P. Gunningberg. SoNIC: Classifying Interference in 802.15.4 Sensor Networks. In *ACM/IEEE IPSN*, pages 55–66, 2013.
[28] S. M. Kim, S. Wang, and T. He. cETX: Incorporating Spatiotemporal Correlation for Better Wireless Networking. In *ACM SenSys*, pages 323–336, 2015.
[29] Sourceforge. iperf.sourceforge.net.



Meng Jin is currently a post-doctoral researcher at School of Software and BNRist, Tsinghua University. She received the B.S., M.S. and Ph.d. degrees in Computer Science from Northwest University, Xian, China, in 2012, 2015, and 2018, respectively. Her main research interests include backscatter communication, wireless network co-existence at 2.4GHz, mobile sensing and clock synchronization.



Yuan He is an associate professor in the School of Software and TNLIST of Tsinghua University. He received his BE degree in University of Science and Technology of China, his ME degree in Institute of Software, Chinese Academy of Sciences, and his PhD degree in Hong Kong University of Science and Technology. His research interests include wireless networks, Internet of Things, mobile & pervasive computing. He is a member of the IEEE and ACM.

Xiaolong Zheng received the BE degree in the School of Software Technology from Dalian University of Technology in 2011, and the PhD degree in the Department of Computer Science and Engineering from Hong Kong University of Science and Technology in 2015. He is currently a member of Tsinghua National Lab for Information Science and Technology. His research interests include wireless sensor networks and pervasive computing. He is a member of the IEEE.

Dingyi Fang currently a Professor with the School of Information Science and Technology, Northwest University, Xi'an, China. He received his Ph.D. degree from Northwestern Polytechnic University of China. His current research interests include mobile computing, Internet of Things, and information security. He is a member of the IEEE and ACM.

Dan Xu received the B.S. degree in Software Engineering from Northwest University, Xi'an, China, in 2011. She is currently working toward the Ph.D. degree in the School of Information Science and Technology. Her current research interests include data transmission in wireless networks.

Tianzhang Xing is currently an assistant professor in School of Information and Technology, Northwest University, Xi'an, China. He received his B.E. degree in the School of Telecommunications Engineering, Xidian university, and his Ph.D. degree in School of Information and Technology, Northwest University. His current research interests include wireless communication networks, with emphasis on the localization problem and the location-based services.

Xiaojiang Chen received the Ph.D. degree in computer software and theory from Northwest University, Xi'an, China, in 2010. He is currently a Professor with the School of Information Science and Technology, Northwest University. His current research interests include localization and performance issues in wireless ad hoc, mesh, and sensor networks and named data networks. He is a member of the IEEE and ACM.

1536-1233 (c) 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Auckland University of Technology. Downloaded on May 24,2020 at 13:33:18 UTC from IEEE Xplore. Restrictions apply.