Exploiting Link Diversity for Performance-Aware and Repeatable Simulation in Low-Power Wireless Networks

Zhiwei Zhao, *Member, IEEE*, *ACM*, Geyong Min, *Member, IEEE*, Wei Dong, *Member, IEEE*, *ACM*, Xue Liu, *Fellow, IEEE*, Weifeng Gao, *Student Member, IEEE*, Tao Gu, *Senior Member, IEEE*, *Member, ACM*, and Minghang Yang

***Abstract*— Network simulation is a fundamental service for performance testing and protocol design in wireless networks. Due to the wireless dynamics, it is highly challenging to provide repeatable and reliable simulation results that are comparable to the empirical experimental results. To achieve repeatability for simulation, the existing works focus on reproducing the behaviors on individual links. However, as observed in recent works, individual link behaviors alone are far from enough to characterize the protocol-level performance. As a result, even if the link behaviors can be simulated very closely, these works often fail to simulate the protocol performance with high reliability. In this article, we propose a novel performance-aware simulation approach which can preserve not only the link-level behaviors but also the performance-level behaviors. We first combine the spatial-temporal link diversity to devise an accurate performance model. Based on the model, we then propose a Performance Aware Hidden Markov Model (PA-HMM), where the protocol performance is directly fed into the Markov state transitions. Compared to the existing works, PA-HMM is able to simulate both link-level behaviors and high-level protocol performance. We conduct extensive testbed and simulation experiments with broadcast and anycast protocols. The results show that 1) the proposed model is able to accurately characterize communication performance for both broadcast and anycast and 2) the protocol performance is closely simulated as compared to the empirical results and the PA-HMM based simulation is more repeatable compared to the existing works.**

***Index Terms*— Low-power wireless networks, performance modeling, repeatable simulation, link diversity, Markov model.**

Manuscript received November 19, 2018; revised March 27, 2020; accepted July 20, 2020; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor K. Jamieson. Date of publication September 2, 2020; date of current version December 16, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61972074 and Grant 61972075, in part by the China Post-Doctoral Science Foundation under Grant 2018M640909, and in part by the National Post-Doctoral Program for Innovative Talents of China under Grant BX201700046. *(Corresponding author: Geyong Min.)*

Zhiwei Zhao, Weifeng Gao, and Minghang Yang are with the College of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610051, China (e-mail: zzw@uestc.edu.cn; weifeng@mobinets.org; minghang@mobinets.org).

Geyong Min is with the Department of Computer Science, University of Exeter, Exeter EX4 4QF, U.K. (e-mail: g.min@exeter.ac.uk).

Wei Dong is with the School of Computer Science, Zhejiang University, Hangzhou 310027, China (e-mail: dongw@zju.edu.cn).

Xue Liu is with the School of Computer Science, McGill University, Montreal, QC H3A 2A7, Canada (e-mail: xueliu@cs.mcgill.ca).

Tao Gu is with the School of Computer Science and Information Technology, RMIT University, Melbourne, VIC 3000, Australia (e-mail: tao.gu@rmit.edu.au).

Digital Object Identifier 10.1109/TNET.2020.3016056

I. INTRODUCTION

WIRELESS network simulation is a fundamental service aiming at providing a controlled and repeatable environment for protocol design, performance testing, algorithm analysis, etc., [1]–[4]. A good simulator is designed to generate the packet traces, based on which the *end-to-end protocol performance* reflects the empirical performance as closely as possible. Some research efforts have been devoted to developing repeatable simulation, such as TOSSIM [5] and M&M [6]. These works attempt to reproduce and simulate the link-level behaviors of the empirical traces in order to generate repeatable protocol performance. Specifically, TOSSIM [5] employs the Signal-to-Noise-Ratio (SNR) model to simulate packet traces for individual links. M&M [6] uses multi-level Markov model to simulate both long-term and short-term behaviors of individual links. The rationale is that the protocol performance can be characterized if the individual links are closely simulated.

However, many recent studies have shown that individual link behaviors alone are not enough for capturing the wireless protocol performance, especially for anycast and broadcast/multicast [7]–[9]. The correlation among adjacent links is also highly impactive to the protocol performance. As a result, simulation of individual links can hardly provide repeatable performance simulation for the protocols involving multiple links (analyzed in Section III). In this article, we aim to design a simulation approach which preserves both link-level behavior and the end-to-end protocol performance.

To this end, there are two key challenges: Firstly, we need to model the mathematical relationship between packet traces on different links and end-to-end protocol performance, such that we can determine the specific traces that preserve protocol performance. Secondly, we need to find a way to automatically generate the packet traces with desired properties to preserve both link-level and protocol-level behaviors.

As to the first challenge, most protocol performances are determined by each single-hop performance and the propagation path in the network. Since the propagation path is determined by the simulated routing protocol, modeling the single-hop performance becomes the essential key problem. Similar to [10], [11], we choose the expected number of transmissions (ETX) as the key metric for protocol performance as most of other performance metrics can be derived with ETX [10]. Specifically, we separately define the ETX metrics for the three transmission modes for single-hop communications [12]: uETX for unicast, aETX for anycast, and bETX for broadcast/multicast. The detailed definitions are described in Section II.

While modeling uETX is straightforward, it is much more challenging to model aETX and bETX, which involves multiple links. Although some existing works use packet reception ratio (PRR) on individual links to model aETX/bETX, many recent studies [8], [13] indicate that the temporal and spatial correlations among links also have a significant impact on anycast/broadcast besides individual PRRs [14]. However, combining all three kinds of information (PRR, temporal and spatial correlations) is a non-trivial task. The reason is that the metrics for them essentially contain overlapped information (as analyzed in Section II).

For example, when we calculate the bETX for a pair of links. If we separately take the PRRs of both links and the correlation between them in the modeling, one link's PRR is actually accounted twice. Considering that the problem of bETX modeling is an inclusion-exclusion problem (as analyzed in Section IV.B and Figure 4). When the overlapped information (i.e., the correlated packet loss of the two links) is accounted multiple times, the probability that not all nodes receive the packet will be over-estimated, which further leads to the situation that bETX is under-estimated. The existing works such as [15], [16] account such overlapped information multiple times during the model iteration, which can lead to largely inaccurate performance characterization. To deal with the overlapping problem, the independent link metrics κ factor [7] and β factor [17] are potential alternatives. Unfortunately, the isolation of different dimensional information leads to too complex metric designs, which can hardly be utilized for ETX modeling.

As to the second challenge, Markov model has been proved to be effective for packet trace simulation [6], [18], [19]. When the ETX performance is accurately characterized, the packet traces can be generated using a Markov model fed with the derived performance states. Hence the key problems are 1) to define the appropriate Markov states representing both link-level and performance level behaviors and 2) to obtain appropriate parameters to generate packet sequences preserving the performance states.

To address the above two challenges, we first propose an accurate performance modeling approach for anycast and broadcast/multicast (aETX and bETX), which considers PRR, temporal and spatial link correlations. Based on this new modeling approach, we devise a Performance Aware Hidden Markov Model (PA-HMM), in which the aETX/bETX combinations are used as the underlying unobserved performance states; and an abstraction of link features containing both spatial and temporal link correlation is used as the observed states. With the accurate performance model and PA-HMM, the proposed work can simulate not only the link-level behaviors but also the protocol performance, providing a more repeatable and reliable simulation environment for wireless protocols.

We implement the performance model and PA-HMM. The experimental results show that, (1) The performance model provides more accurate single-hop aETX/bETX modeling than the existing works. (2) Compared to the existing simulators (TOSSIM [5] and M&M [6]), the proposed work can achieve more repeatable wireless network simulation in terms of both link-level behaviors and protocol performance.

The main contributions of this article are summarized as follows:

1) We propose an accurate performance model for anycast and broadcast/multicast, which jointly considers PRR, spatial and temporal link correlation. With the model, aETX and bETX can be accurately obtained from the packet traces.
2) Based on the performance model, we propose a Performance Aware Hidden Markov Model (PA-HMM) for wireless network simulation, which can simulate both link-level and performance-level behaviors of wireless networks.
3) We explore the optimal parameters for PA-HMM to achieve accurate simulation for different scenarios, such that the model could adapt to various application demands.
4) We implement PA-HMM and evaluate the simulation based on PA-HMM. The results show that more repeatable simulation is achieved compared to the existing works in terms of protocol performance.

The rest of the article is organized as follows: Section II presents related works on wireless network simulation and performance characterization. Section III analyzes the necessary link features that should be considered for accurate performance modeling with measurement study. Section IV presents the performance modeling and PA-HMM in detail. Section V evaluates the performance model and the simulation with PA-HMM in comparison with the state-of-the-art works. Section VII concludes this work.

II. Related Works

Simulation has always been one of the most important means to study protocol behaviors and evaluate protocol performance in wireless networks. To this end, the existing works have utilized different techniques to simulate packet reception/loss behaviors on individual links. However, according to recent observations on wireless dynamics, the protocol performance cannot be simulated solely by reproducing individual link behaviors. In this section, we will review the existing works on wireless simulations as well as the modeling of protocol performance. The comparison between our work and the existing works will also be discussed.

A. Wireless Network Simulation

Many existing works utilize Markov model for network simulation [6], [18], [20]–[22]. The Gilbert model [20] is a probabilistic model for simulating bursty noise in wireless channels. A hidden Markov model with two states is employed, where the first state has a zero transmission error rate (perfect links) and the other state has a given nonzero probability of transmission error rate (intermediate links). Nguyen *et al.* [21] employ the exponential and Pareto distributions to model the packet traces. Markov-based trace analysis decompose the packet trace with non-stationary properties into stationary pieces consisting of lossy and error-free states. Khayam and Radha [22] focus on 802.11b networks in terms of both bit errors and packet errors. From these works, we can see that wireless simulation needs to consider both long-term and short-term link variations. The simulated traces should also be close to the input traces.

Next, we focus on two representative works on low-power wireless simulation, i.e., TOSSIM and M&M, and then introduce some other recent efforts.

TOSSIM [5] is a discrete-event simulator for wireless sensor networks operating the TinyOS system. The input of TOSSIM is the physical layer signal powers (RSSI) for each link and background noise, where the RSSI values are given by simulation users and the background noise is generated based on the historical environmental noise traces with the Closest Pattern Matching (CPM) model. The packet reception/loss traces are then generated using the Signal-to-Noise-Ratio (SNR) model.

M&M [6] is a Markov model based approach, which is directly built upon the packet traces instead of the physical layer indicators. A multi-level Markov model is employed in M&M, where the higher-level states capture the long-term link behavior and the lower level states capture the short-term link behavior. The transition probabilities control the duration of long-term and short-term behaviors and are extracted from the collected traces in real environments. M&M has two subsequent works [18], [19], which focus on boosting the training process of the Markov model with limited time. Compared to the M&M model, the main difference of PA-HMM is that we additionally consider the multi-link correlations and directly employ the multi-link performance metrics as the hidden Markov states. As a result, both single-link behavior (unicast) and multi-link behavior (broadcast and anycast) can be preserved by PA-HMM.

Other recent efforts on the wireless simulation There are a number of recent works on wireless simulations. IoTNetSim [23] is a modeling and simulation platform for end-to-end IoT services and networking. It considers the heterogeneity of the IoT devices and the novel network architectures with cloud/edge/fog computing. In terms of the simulation of link-level behaviors, it employs the existing SNR-based approaches designed for different communications such as ZigBee and WiFi. Morpheus [24] is a simulation-based work for network management (deployment and parameter planning). The module of performance evaluation utilizes a modified version of the Cooja simulator [25]. The link properties are based on RSSI and PDR traces. Similarly, DrySim [26] simulation also combines RSSI and PDR traces as the target link properties. Reference [27] is another work on single-link simulations focusing on the burstiness. It employs the Markov model to characterize the number of consecutive packet losses and receptions. The core ideas of the above works are either based on the single-link radio models or based on the trace-driven models.

We can see that the existing works focus on simulating single links. The rationale behind is that once the individual links are simulated, the protocol performance can be simulated as well. However, many recent studies observed that the spatial correlation between adjacent links can significantly affect the communication performance (anycast and broadcast). Srinivasan *et al.* [7] observed correlation on the packet receptions and losses among different links and analyzed the impact of link correlation on broadcast performance. Many following works [8], [28], [29] also confirmed the impact of link correlation on anycast and broadcast/multicast performance. As a result, simulation of individual links is not enough to simulate the protocol performance.

The main difference of PA-HMM with the above works is as follows: 1) PA-HMM considers the multi-link properties, which are important for preserving the multi-link behaviors and performance in the simulated trace. 2) PA-HMM employs a two-layer Markov model, which is driven by the performance states. As a result, the simulated trace can better represent the multi-link performance. The limitation of PA-HMM, compared to the above works, is that its training process takes much more space and time than the simulation based on radio models. Besides, we directly adopt both end-to-end performance and link-level behaviors into the proposed simulation model. Both the link-level behaviors and the end-to-end performance can be preserved.

B. Performance Characterization

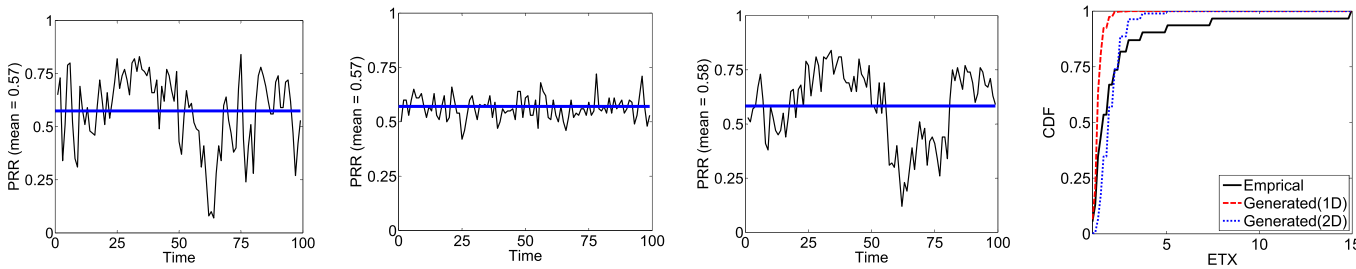
The expected number of transmissions (ETX) has been widely used as the performance metric for various protocols. While ETX for unicast (uETX) is easy to calculate, it is challenging to characterize the ETX of anycast and broadcast/multicast. Similar to the existing works [8], we denote the ETX for anycast and broadcast/multicast as aETX and bETX. Specifically,

- aETX is the ETX for a sender to successfully deliver one packet to *at least one* of its receivers.
- bETX is the ETX for a sender to successfully deliver one packet to *all* of its receivers.

aETX for a sender s is often calculated as $\frac{1}{P^s_{S_R}}$, where $P^s_{S_R}$ is the probability that at least one node in its receiver set (S_R) receives the packet. To calculate $P^s_{S_R}$, the work [15] uses the multiplication of the link quality of all outbound links of s. However, due to the spatial link correlation, the result is often over-estimated as the correlated information is accounted multiple times. In [30], link correlation is additionally considered for accurate aETX calculation. The calculation of bETX is more complex as given by:

$$bETX = \sum_{k=1}^{+\infty} kP(X = k) \tag{1}$$

where $P(X = k)$ denotes the probability that all nodes in S_R receive the packet after k transmissions. The existing approaches [8], [31] differ from each other mainly in the way of calculating $P(X = k)$. In [31], topology and link quality are considered and in [8], link correlation is



(a) Empirical PRR trace (b) Generated trace with 1D(PRR) (c) Generated trace with 2D (d) ETX of empirical/generated trace

Fig. 1. Characterizing single link performance.

additionally considered. However, the overlapped information between link correlation and individual link quality is also accounted multiple times in the model iteration.

The modeling of aETX/bETX in this article differs from existing performance models in the following ways. Firstly, we jointly consider link quality and the temporal-spatial link correlation to achieve accurate performance estimation. Secondly, to improve the model efficiency, we propose a packet trace abstraction scheme which can efficiently extract the three kinds of information without explicitly calculating the separate link metrics. In addition to the modeling of aETX/bETX, we further incorporate the model to reversely generate the link metrics using the aETX/bETX sequences.

III. Measurement Study on Link Features for Modeling Wireless Communication Performance

In this section, we analyze the impacting factors of unicast, anycast and broadcast/multicast performance and find the necessary information required for uETX/aETX/bETX modeling.

A. Characterizing Single Link Performance (uETX)

Link quality (1D). Link quality is one of the most widely identified impacting factors for wireless communication performance. Packet reception ratio (PRR) is a typical characterizing metric for link quality. We denote link quality as the one-dimension (1D) link feature.

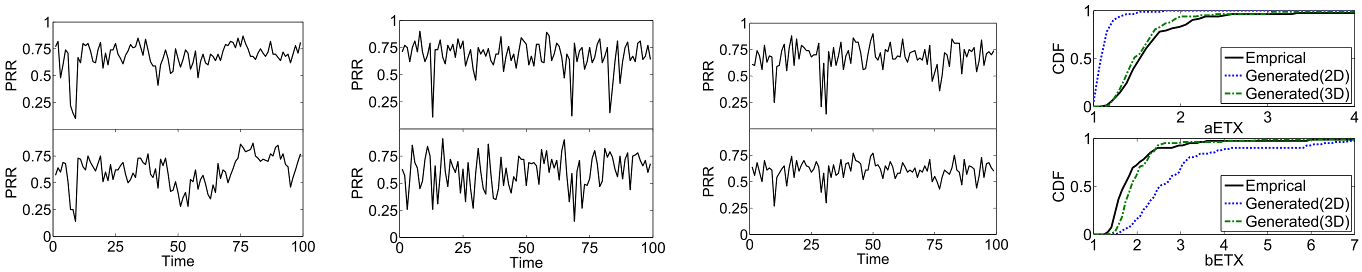
Figure 1(a) shows an empirical PRR trace of a wireless link. If we simulate this link using the average PRR value (0.57) with random variations, the generated PRR is shown in Figure 1(b). Now with the two packet traces generated based on the two links, we check whether the uETX is characterized. The uETX is obtained from the packet traces as $n_{tx} = n_{loss} + 1$, where n_{loss} is the number of losses before a packet reception. We repeat the experiments and obtain the average uETX values for both links.

Figure 1(d) shows the CDF of uETX values for both empirical and the 1D generated traces (1D). We can see that, the uETX values using 1D information are largely different from the empirical uETX. The reason is that the average PRR captures long term link behaviors, which could derive the long term overall uETX of the empirical trace. However, due to the PRR variations and the nature of reciprocal relationship between PRR and ETX, the fine-grained short term uETX cannot be captured, resulting in the inaccurate performance characterization. For example, the average PRR of 0.2 and 0.8 is 0.5. The ETX is normally calculated as $\frac{1}{0.5} = 2$. However, the actual ETX should be $\frac{1}{2}(\frac{1}{0.2} + \frac{1}{0.8}) = 3.125$, which is quite different from the result directly obtained by the average PRR. Therefore, a long-term PRR metric is not enough to accurately capture the uETX performance.

PRR and its temporal distribution (2D). It has been observed that both packet receptions and losses have clear temporal behaviors [32]. These works try to characterize the temporal distribution using various metrics such as μ [33], β [17], etc. To study the impact of the temporal features on the protocol performance, we manually generate packet traces for the simulated link preserving both PRR and the temporal packet distributions (using the metric μ and a simulation approach similar to [6]). Figure 1(c) shows the generated PRR trace, preserving 2D properties (the long term PRR and temporal distributions). Intuitively, it is much more close to the original link in Figure 1(a) than the 1D simulated link in Figure 1(b). Figure 1(d) shows the uETX comparison between the empirical trace and the 2D generated trace. We can see that uETX is also characterized more accurately. Preserving the 2D properties (i.e., PRR and its temporal distribution) seems good enough for characterizing uETX.

B. Characterizing the Performance for Multiple Links (aETX/bETX)

Next, we study whether the above 2D information can characterize transmission performance involving multiple links (aETX and bETX). Figure 2(a) shows the empirical PRR traces of a link pair. We can see that these two links have a high positive correlation. Figure 2(b) shows the simulated link pair preserving both PRR and the temporal distributions of the links in Figure 2(a). Obviously, the correlation between the links is not captured by the generated link pair. Then we further investigate whether aETX and bETX are characterized by the simulated link pair. Using the packet reception traces for the two links, we can directly obtain the number of transmissions for delivering one packet to at least one receiver (aETX) and the number of transmissions for delivering one packet to both receivers (bETX). Figure 2(d) depicts the CDF of the aETX/bETX values for the empirical traces and the generated traces with simulated link pair. We can see that there exist large errors on both aETX and bETX with 2D. The reason is



(a) Empirical trace of a link pair (b) Generated trace with 2D (c) Generated trace with 3D (d) ETX of empirical/generated trace

Fig. 2. Characterizing broadcast/anycast performance.

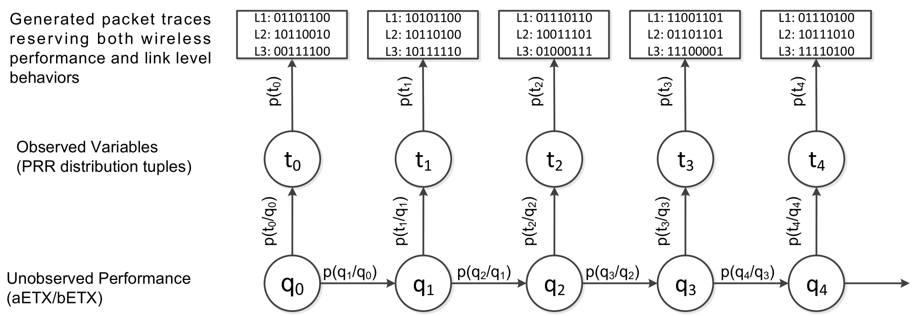


Fig. 3. The PA-HMM. The unobserved q states capture the performance state transitions (with aETX and bETX). The observed t states capture the link level behavior transitions (with PRR tuples, where a tuple indicates the PRR distribution at multiple receivers).

that, anycast and broadcast can be greatly affected by spatial correlation, which is not captured by the 2D link features. For example, if the receptions of two links are strongly correlated, aETX tends to be large [30] and bETX tends to be small [7] for the same generated packet traces on both links.

PRR and the temporal-spatial distributions (3D). The spatial distribution of PRRs essentially reflects the relationship among different links, which has been observed by the existing works [7]. Now we manually set packet traces for the simulated link pair preserving PRR, temporal and spatial distributions. Figure 2(c) shows the generated traces. We can see that the relationship between two links is similar to that in Figure 2(a). As shown in Figure 2(d), both aETX and bETX are much more accurately characterized with 3D information.

We also repeat the experiment under various different environments (e.g., indoor, outdoor, WiFi-interfered [34], pedestrians, etc.) and obtain similar observations to the above results.

C. Short Summary on Characterizing Link Features

From the above study, we can see that 1) PRR characterizes the long term property of a link; 2) The temporal distribution characterizes how a link's PRR variates. The two-dimension information can accurately characterize single link performance. 3) The spatial distribution characterizes the relationship between different links. With the three-dimension information, each link's PRR, PRR variations and its correlation with other links can be determined, which essentially determines the performance of anycast and broadcast/multicast.

Therefore, to *accurately* infer the link behaviors as well as wireless communication performance in terms of uETX, aETX and bETX, we need to consider all the three kinds of information. In the next section, we will present our performance modeling approach and the simulation approach based on the performance model.

IV. The Performance Aware Markov Model for Wireless Simulation

In this section, we present the Performance Aware Hidden Markov Model (PA-HMM) for repeatable wireless network simulation, which is based on the performance modeling considering the aforementioned link features. The proposed work can preserve both wireless communication performance and the link-level behaviors. We will first present the overview of the simulation based on HMM and then present the details of each building block of PA-HMM including the wireless communication performance modeling and the HMM.

A. Overview

The performance aware hidden Markov model (PA-HMM) is shown in Figure 3. We denote the packet receptions and losses on each links with binary sequences, where a "0" denotes a packet loss and a "1" denotes a packet reception. The binary sequences on multiple links of a sender are generated at the same time using the PRR tuples represented by t_n as shown in the figure, where n denotes the n-th time window. Each performance state q (unobserved) is a combination of aETX and bETX and has its own probability distribution $p(t_n/q_n)$ of emitting the PRR tuple distributions (t_n). The performance states (q) capture the performance-level behaviors and the link states (t) capture the link-level behaviors (as will be described in Section IV-D). The transition probability of the performance states controls the performance variations. For each performance state q_n, the emission distribution contains m component for the m links in t_n (m is the number of outbound links of a sender). Each component contains d elements controlling the PRR temporal variation in a duration of $d \times W$ slots, where W denotes the number of packets sampled in one PRR window. It is worth mentioning that wireless communication is inherently based on broadcast and packet receptions/losses happen at the same time. Hence, compared to the existing "link-wise" approaches, a more reasonable simulation manner is to generate the packet traces for multiple adjacent links at the same time.

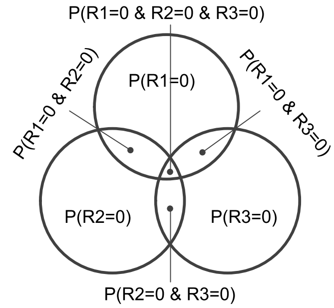


Fig. 4. Calculation of bETX: The case of three receivers.

B. PA-HMM

Our focus is to preserve the multi-link behaviors. Considering that most multi-link wireless protocols are based on broadcast/multicast and anycast, in PA-HMM, single-hop protocol performance is denoted by the performance states using aETX/bETX combinations (q). The link level behavior is captured by the PRR tuple distributions (t), where both link correlation and link quality are preserved. The input parameters include:

1) Performance states (aETX/bETX pairs);
2) PRR distribution tuples (described in Section IV-D);
3) The transition probability matrix between performance states, $p(q_n/q_{n-1})$;
4) The emission probability distribution for each performance state, $p(t_n/q_n)$.

These parameters can be either manually set by the simulation users or extracted from the target empirical traces (the simulated results will have similar performance and link-level behaviors with those of the empirical trace).

The following example shows how the PA-HMM is used to generate network traces for a three-link neighborhood. Starting from the initial state, say q_0 =(aETX $= 1.2$,bETX $= 2.1$), the current state transits from one to another following the matrix of transition probability, which is trained using the raw packet trace. Then for each state, it generates different tuples of packet reception rates (PRRs) according to its emission rates to these tuples. In each PRR tuple, one PRR value corresponds to a single link in the tuple. For example, if the state (aETX $= 1.2$, bETX $= 2.1$) has emission rates to two PRR tuples (each contains three link PRRs): $t_0 = (0.7, 0.6, 0.9)$ and $t_1 = (0.4, 0.6, 0.8)$. Following the emission rates $p(t_0/q_0) = 0.4$ and $p(t_0/q_0) = 0.6$, a number of instances for the two PRR tuples will be generated. For each PRR tuple, we further generate the single-link packet receptions using the corresponding PRR in the tuple. For example, if the current tuple is t_0, then the three links will generate packet receptions using the probability of 0.7, 0.6 and 0.9. As the hidden state transits from one state to another, the above process repeats and thus the traces are generated.

Obtaining the aETX/bETX states. As the aETX/bETX pairs are used as the hidden states, we need to extract those aETX/bETX values from the packet trace. However, it may not be feasible to obtain the raw packet traces in practical scenarios. Therefore, we design an estimation scheme for aETX and bETX, which can convert the link metrics (packet reception rates and link correlation) to the performance metrics (uETX, aETX and bETX). The estimated metrics are then used as hidden network states in the PA-HMM model.

C. Performance Modeling for aETX and bETX

aETX. Recall that aETX is the number of transmissions for a sender to deliver one packet to at least one node of its receivers. Similar to the existing works [15], [35], the aETX is calculated as:

$$aETX = \frac{1}{p^s_{S_R}} \tag{2}$$

where $p^s_{S_R}$ is the probability that at least one node in S_R receives the packet. Since we extract the metric from packet reception/loss traces, $p^s_{S_R}$ can be obtained as follows:

$$\begin{aligned} p^s_{S_R} &= 1 - p_{0*} \\ &= 1 - \sum_{\forall t_i \in T} p(t_i) p_{t_i}(0*) \end{aligned} \tag{3}$$

where p_{0*} denotes the probability that all receivers lose the packet ("0" stands for a packet loss), T denotes the PRR tuple set, $p(t_i)$ denotes the probability of PRR tuple t_i, and $p_{t_i}(0*)$ denotes the probability that all receivers lose the packet given the PRR values in tuple t_i.

bETX. Recall that bETX is the expected number of transmissions for a sender to deliver one packet to all its receivers. *Note that the receivers are not restricted to receive the packet at the same time.*

For simplicity, we start from the case of three receivers R1, R2, and R3. Basically, bETX can be calculated as:

$$bETX = \sum_{k=1}^{+\infty} kP(X = k) \tag{4}$$

where $P(X = k)$ is the probability that k transmissions cover all three receivers. It can be calculated as

$$P(X = k) = P(X > k - 1) - P(X > k) \tag{5}$$

where $P(X > k)$ is the probability that after k transmissions, at least one receiver has not received the packet.

The calculation of $P(X > k)$ turns out to be an inclusion-exclusion problem as shown in Figure 4. Note that P(R1 $= 0$) denotes the probability that *after k transmissions*, R1 has not received the packet, P(R1 = 0&R2 = 0) denotes the probability that after k transmissions, both R1 and R2 have not received the packet, and P(R1 = 0&R2 = 0&R3 = 0) denotes the probability that after k transmissions R1, R2, and R3 have not received the packet. With the above information,

$$\begin{aligned}
&P(X > k)\\
&= P(R1 = 0) + P(R2 = 0) + P(R3 = 0)\\
&\quad - P(R1 = 0 \& R2 = 0) - P(R1 = 0 \& R3 = 0)\\
&\quad - P(R2 = 0 \& R3 = 0) + P(R1 = 0 \& R2 = 0 \& R3 = 0)\\
&= P_{n_0=1} - P_{n_0=2} + P_{n_0=3} \qquad (6)
\end{aligned}$$

where $P_{n_0=1}$ denotes the probability that $n_0(= 1)$ receivers lose the packet k times. With the input, we get:

$$\begin{aligned}
P_{n_0=1} &= (p_{000} + p_{001} + p_{010} + p_{011})^k\\
&\quad + (p_{000} + p_{001} + p_{100} + p_{101})^k\\
&\quad + (p_{000} + p_{010} + p_{100} + p_{110})^k\\
P_{n_0=2} &= (p_{000} + p_{001})^k\\
&\quad + (p_{000} + p_{010})^k\\
&\quad + (p_{000} + p_{100})^k\\
P_{n_0=3} &= (p_{000})^k \qquad (7)
\end{aligned}$$

where p_{ijk} denotes the probability that $R1 = i, R2 = j, R3 = k$ $(i, j, k \in [0, 1])$. Combining Eqs. (4)-(7), we can obtain the bETX to cover the three nodes.

n-receivers case for bETX. Now we move to calculate the bETX for n receivers, which is an extension of Eq. (4). The key is to calculate $P(X > k)$, the probability that not all n receivers received the packet after k transmissions. We use an n-bit bitmap to denote the case of packet reception distribution. For example, a bitmap of "0101" denotes the case in which the first and third receivers lose the packet and the second and the fourth receivers receive the packet. Then $P(X > k)$ is given as:

$$\begin{aligned}
P(X > k) &= \sum_{m=1}^{n} (-1)^{m-1} P_{n_0=m}\\
&= \sum_{m=1}^{n} (-1)^{m-1} \sum_{\forall S_m} e_{S_m}^k \qquad (8)
\end{aligned}$$

where $P_{n_0=m}$ is the probability that m receivers do not receive the packet by k transmissions, S_m is set of bitmaps with m "0"s and e_{S_m} is the probability with m uncovered receivers. e_{S_m} is calculated as:

$$e_{S_m} = \sum_{\forall b \in S_m} p_b = \sum_{\forall b \in S_m} \sum_{\forall t_i \in T} p_{t_i}(b) \qquad (9)$$

where b is a bitmap with m "0"s and $p_{t_i}(b)$ denotes the probability of the bitmap b given the PRR tuple of t_i.

Combining Eqs. (4), (5) and (8), the bETX to cover n receivers is given by:

$$\begin{aligned}
bETX &= \sum_{k=1}^{+\infty} kP(X = k)\\
&= \sum_{m=1}^{n} (-1)^{m-1} \sum_{\forall S_m} \frac{1}{1 - e_{S_m}} \qquad (10)
\end{aligned}$$

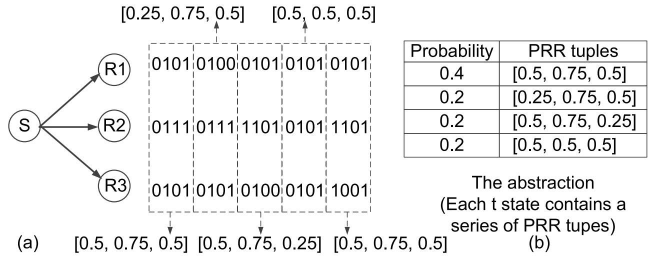


Fig. 5. Illustration of the packet trace abstraction.

The combinations of the extracted aETX and bETX values are then used as the performance-level states.

D. Link-Level States: PRR Tuple Distribution

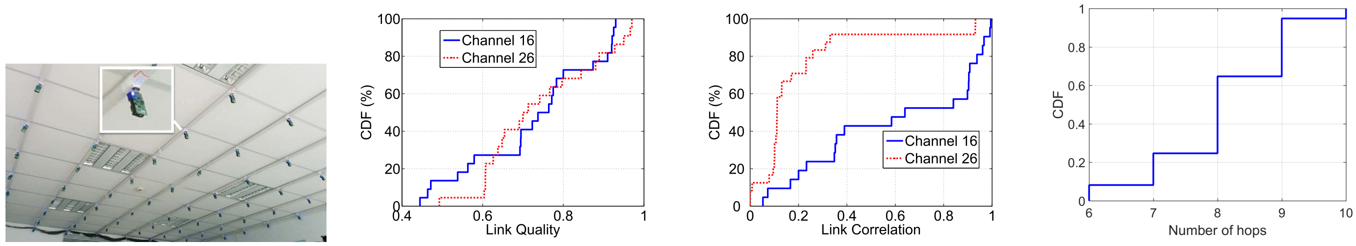
Link level states are responsible to represent the link level metrics including PRR, temporal and spatial correlations. Different from the existing modeling approaches, we do not utilize separate link metrics such as κ for link correlation or β for burstiness. Instead, we abstract a PRR tuple distribution from the packet reception traces on multiple links, which essentially stores the PRR and temporal-spatial distributions. With the PRR tuples, link-level behaviors can be preserved.

Given packet reception traces on different links, we first slice time into many short periods and obtain a series of PRR values for each link. The period length can be set according to user's granularity requirement. After that, we combine PRR values at the same period in a PRR tuple and account the overall probability of each different PRR tuples. After that, we obtain a table storing PRR tuples and its distribution probabilities. One different PRR tuple represents one different spatial distribution for a short period. The probabilities for PRR tuples represent the temporal distributions and variations.

Figure 5(a) shows an illustrating example, where S is a sender; $R1, R2$ and $R3$ are three receivers. The "0" and "1" represent packet losses and receptions. Taking the packet reception traces as input, we first slice the traces into several windows (each window contains four packets). In each window, we can obtain a PRR tuple, e.g., the PRR tuple of the first window is [0.5, 0.75, 0.5], indicating that PRRs on the three links are 0.5, 0.75 and 0.5 within the window. After that, we can obtain the probabilities of all different PRR tuples as shown in Figure 5(b). This table is the packet trace abstraction and used as the states for link features (t). We can see that PRR values, spatial distributions and temporal distributions are all covered the abstraction. For each aETX/bETX state, there are several corresponding link states of PRR tuples and probability distributions. It is worth noting that, *during the abstraction, each packet reception/loss is accounted only once*, inherently avoiding the information overlapping problem.

E. Reducing Input Size

With the performance modeling approach, we can obtain the performance states (aETX/bETX pairs) from the packet traces (It is also worth noting that the states can be set manually



(a) The 8×10 testbed (b) CDF of Link quality (c) CDF of link correlation (d) CDF of hop count

Fig. 6. Link and performance features of the testbed.

for the simulation users to explore all possible performance space). However, there will be infinite values for aETX/bETX states, which will significantly increase the complexity for PA-HMM. For example, if we divide the range of aETX/bETX values into 100 sections, there will be 100×100 different (aETX,bETX) states. The transitional matrix size will be $(100 \times 100)^2$. To reduce the overhead, we can decrease the number of performance states using k-means clustering. The cluster centers can then be used as the performance states.

Apparently, there exists a tradeoff between accuracy and the computational overhead. With a small k, the model accuracy decreases and memory overhead decreases. With a large k, the model accuracy increases yet the memory overhead increases.

F. Discussion

In the existing simulations, users can easily tune certain parameters to study different network conditions. For example, we can tune the pairwise RSSI for each link in TOSSIM [5] to change network conditions. However, in PA-HMM simulation, although we can generate network traces to achieve closely simulated performance and link behaviors, we are unable to *freely* change the network conditions. To achieve tunable and flexible simulation based on PA-HMM, there are two challenges as follows. First, it is hard to determine which parameters to tune. As we have analyzed, link quality and its temporal-spatial variations together determine the end-to-end anycast/broadcast performance. However, it is not easy for users to manually set the spatial-temporal variations of wireless links. Second, the PA-HMM simulation requires initial input of the empirical traces, which are collected from real networks. If we change the network conditions and still require repeatable simulated performance, we need to collect the network traces that are similar to the tuned target network conditions, which makes it impossible to *freely* change the network conditions.

For the first challenge, considering the ease of use for the simulation, we choose the performance metrics aETX/bETX as tunable parameters instead of the spatial-temporal variations of the PRRs. The reason is two-fold: 1) The performance metrics are more straightforward and meaningful than other intermediate metrics. 2) The performance metrics are easier to be tuned compared to the PRR variations. For the second challenge, using the proposed performance model, we can reversely derive the PRR tuples according to the user set performance metrics in terms of aETX/bETX. The problem is that to achieve a specific aETX/bETX value, we can derive a number of different PRR tuples. Based on these PRR tuples, we further allow to tune the spatial-temporal behaviors such that the candidate PRR tuples can be specified.

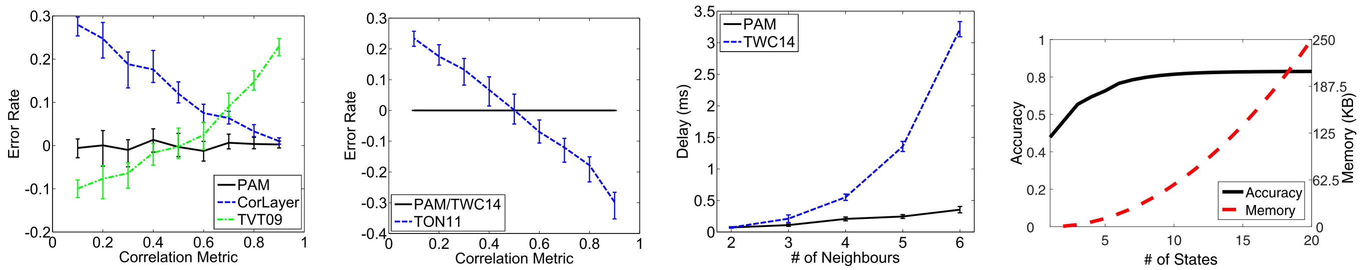
V. EVALUATION

We conduct both indoor and outdoor experiments. For the indoor experiment, we use our 8×10 TelosB nodes testbed (Figure 6(a)) to collect packet traces. The radio power is set to -32.5 dBm to enable a 6-10 hop network. Each node periodically broadcasts packets and records the packet receptions from neighboring nodes. The packet receptions on each link are sent to the serial ports on PC via USB cables. With 80 network nodes, we need to record 80×80 binary traces for all links (some of which are empty). It is worth noting that the packet traces can be from any networks or generated according to the user demands. With the traces, we then study the model accuracy as well as the repeatable simulation performance. In order to explore the full potential of the PA-HMM simulation we utilize the exact aETX/bETX values as performance states. The outdoor experiments are described in Section V.D.

Figure 6 shows the link quality, link correlation and the hop count of the testbed. Link quality is measured by PRR, link correlation is measured by the conditional probability and the average aETX/bETX is measured by the aETX/bETX values from each node to all its subsequent receivers/forwarders. As a result, when specific protocols that have limits on the number of receivers/forwarders, the aETX/bETX values may be different from the measurements. We can see that the wireless features are highly diverse: 1) There are approximately 30% links with PRR smaller than 0.6 and 25% links with PRR larger than 0.8. 2) In channel 26, most link correlation values (around 80%) are smaller than 0.4. Considering we are using the conditional probability as the correlation metric, most links are independent with each other. In channel 16, which overlaps with WiFi, the link correlation becomes stronger where more than 60% links are highly correlated (> 0.8). Considering the existing works have mainly focused on the independent cases, we mainly study the performance simulation in the channel with stronger correlation. The multi-hop anycast/broadcast protocols are studied in a network with 6-10 hops.

A. Evaluation Setup

Implementation. PA-HMM and M&M are separately implemented in python. The input of the simulator is the



(a) Accuracy of bETX modeling (b) Accuracy of aETX modeling (c) Computational overhead (d) Impact of link states

Fig. 7. Evaluation results on the PA-HMM modeling approach.

packet trace collected from the real-world testbed, i.e., the time series of packet receptions and losses per link. The output is a new packet trace that preserves both performance behaviors and the single link behaviors. Both PA-HMM and M&M do not rely on any underlying PHY layer specifications or SNR models.

It is also possible to embed the simulation model into TOSSIM. To this end, we need to replace the SNR-PRR module using PA-HMM. The challenge of implementation is in that our model generates the packet receptions at multiple nodes at the same time. To address this challenge, in the process of sensing a packet, we need to extract the destination nodes and combine them with the PA-HMM packet generation.

Relative errors. We mainly compare the relative errors of the simulated trace to the original trace. Specifically, The relative error is calculated as $\frac{m_g - m_o}{m_0}$, where m_g is the performance metric accounted with the generated bitmap trace, and m_o is the performance metric accounted with the original bitmap trace. The performance metrics are uETX for unicast, aETX for anycast and bETX for broadcast. Specifically, as defined in Section II, uETX is accounted as the number of transmissions (ETX) with which the receiver receives the packet; aETX is accounted as the ETX with which at least one of its receivers receive the packet; and bETX is accounted as the ETX with which all its receivers receive the packet. When accounting the performance metrics, we use the same number of bits (packets) for the generated trace and original trace.

Trace collection. To feed the PA-HMM model, we need to collect the network traces. However, it is usually difficult to know whether all broadcast nodes receive a packet. In our experiments, we manually record the packet receptions (source nodes and packet sequences) at each node, and infer which nodes are in the vicinity. Specifically, we use local logging in our work. Each network node has no prior knowledge of its neighbors. We require each node to broadcast a number of packets, with its own node ID and packet sequence. Each node records a bitmap for the packet receptions from any sender as long as it receives packets from that sender. After that, we collect all bitmaps from all nodes. Then we are able to calculate the broadcast performance for each network node.

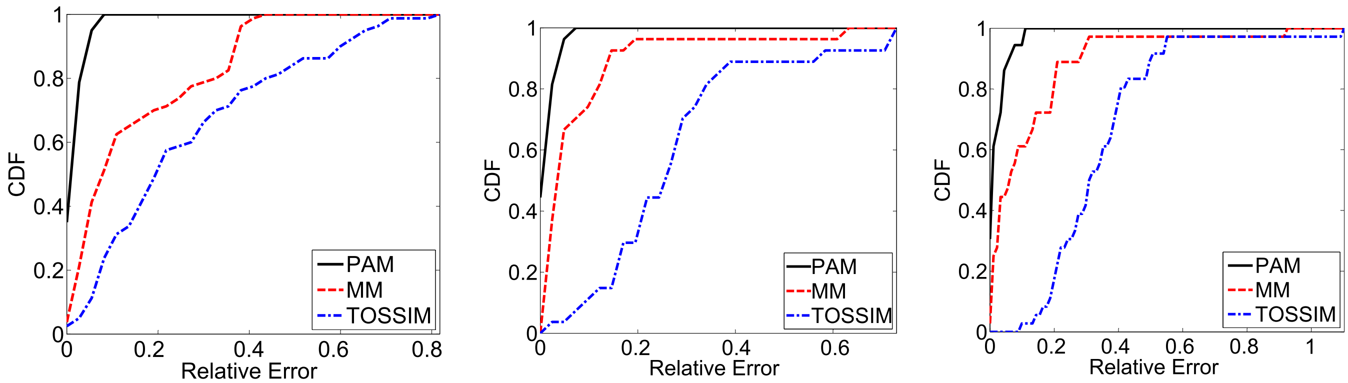
B. Performance Modeling

We study the model accuracy of both aETX and bETX for anycast and broadcast by repeating experiments with varying number of receivers. We manually tune the PRR, temporal and spatial distributions by introducing intentional packet losses to compare the modeling accuracy under various environments. Performance models of aETX and bETX are separately evaluated, in terms of accuracy and computation overhead.

Baseline approaches. For the performance model, We use the approaches introduced in Section II as baseline approaches. Specifically, for aETX we compare PA-HMM with two existing works TON11 and TWC14 [15], [30]. TON11 considers only link quality and TWC14 considers both link quality and link correlation. For bETX, we also compare PA-HMM with two existing works TVT09 and CorLayer [8], [31]. TVT09 [31] considers only link quality and CorLayer considers both link quality and link correlation. For repeatable simulation, we compare our work with TOSSIM [5] and M&M [6]. Corlayer and TVT09 are inspiring works that can be used to characterize multi-link bETX performance. Thus, the comparison of PA-HMM with Corlayer and TVT09 mainly aims at studying the estimation errors of the bETX performance metric.

Figure 7(a) compares the bETX modeling accuracy of the proposed work (denoted as PAM) and other approaches. We can see that, (1) TVT09 [31] is accurate only when link correlation is around 0.5, i.e., the spatial distribution is random. The reason is that it does not consider the spatial distributions and implicitly assumes that the PRRs are independently distributed at different receivers. (2) CorLayer [8] is accurate when link correlation is strong and inaccurate when link correlation is weak. The reason is that it is based on the assumption that receivers of the better-quality links receive the packets earlier than other receivers. When link correlation is 1, it means all receivers receive the packet at the same time, which minimizes the negative impact of the assumption.

Figure 7(b) compares the aETX modeling accuracy. We can see that (1) the proposed model is more accurate than the approach in [15]. The reason is the spatial and temporal link characteristics are additionally considered. (2) The proposed model and TWC14 have the same accuracy. The reason is that TWC14's modeling essentially takes 2^n link correlation values for n links, which implicitly takes the spatial distribution as well as its temporal distributions. Therefore, although they do not explicitly reduce the overlapped information, the modeling results are as accurate as our work PAM. We further compare the computation overhead of PAM and TWC14 on



(a) Comparison on representing link correlation (b) Comparison on opp. routing performance (c) Comparison on dissemination performance

Fig. 8. Evaluation results on the aETX/bETX performance modeling.

the MSP430 platform. Figure 7(c) compares the computation overhead of the proposed model and TWC14 [30]. We can see that when the number of receivers increases, TWC14 incurs much more delay. It implies that the proposed performance model could also be used as an alternative for link estimation in various wireless protocols. The reason is that in the proposed model, the probability of all zeros can be directly extracted by p_{0*} (Eq.(3)) while TWC14 has to translate the PRR and link correlation metrics for calculating the aETX.

There exists a tradeoff between accuracy and overhead on determining the number of aETX/bETX states (denoted as k). Figure 7(d) shows the model accuracy of presenting the aETX/bETX states and memory overhead with varying k values. We can select $k = 7$ to achieve a good tradeoff between efficiency and accuracy because it achieves relatively high accuracy (nearly 0.8) while incurring a smaller memory overhead. Please refer to our technical report for detailed experimental settings.

C. Comparison With TOSSIM and M&M Simulation

Using the collected packet traces at all links, we can simulate the network using PA-HMM based simulation (denoted as PAM in the figures). For fair comparison, we use the same measured traces to drive the M&M and PA-HMM. For TOSSIM, we measure the noise traces on our testbed and feed it into the TOSSIM simulation. We first study whether the packet traces can be characterized by letting nodes periodically transmit packets in both the testbed experiments and all the three simulations Since both PA-HMM and M&M consider PRR and the temporal distribution, the key is to check whether link correlation among different links on the testbed can be preserved and simulated. We run the protocols 1000 times and the results are shown in Figure 8(a), where the relative-errors between the empirical and simulated link correlation by different simulation approaches are demonstrated. We can see that, most of the simulation errors of PA-HMM are smaller than 9%. Conversely, Only about 50% cases in M&M simulates the transmission count with relative errors smaller than 9%. The relative error of TOSSIM is even larger.

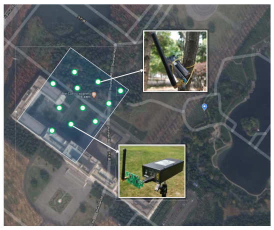
Next we compare our work with M&M and TOSSIM in terms of protocol performance. We test two popular protocols: opportunistic routing [36] and data dissemination [37], which are based on anycast and broadcast/multicast respectively. We initiate the PA-HMM, TOSSIM and M&M using the same packet traces collected from our testbed. Then we study whether the simulated packet traces can support repeatable protocol performance (number of transmissions).

Figure 8(b) shows the comparison of PA-HMM, TOSSIM and M&M for opportunistic routing protocol. Figure 8(c) shows the comparison of PA-HMM, TOSSIM and M&M for bulk data dissemination protocol. We can see that for both opportunistic routing and dissemination, the protocol performance based on PA-HMM is much more closer to the empirical results (the relative error between simulation results and empirical results are greatly reduced compared to the other approaches). The reason is that 1) Our work explicitly use wireless communication performance (aETX and bETX) as the underlying performance states, thus the generated trace can preserve the single hop wireless communication performance. 2) The aETX/bETX errors in the other works can further lead to incorrect routing decisions, which makes the difference between simulated performance and the empirical performance even larger in TOSSIM and M&M. 3) The relative error of bETX is generally larger than that of aETX. The reason is that aETX is determined by the number of common packet losses while bETX is affected by a number of factors such as the explicit values of link quality and link correlation among links. Similarly, the link level states (PRR tuples) can also be optimized by clustering the traces into k states.

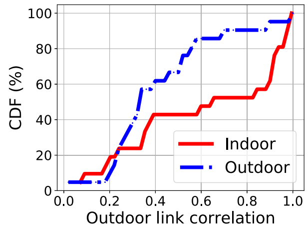
D. Outdoor Experiments

To study the simulation for more realistic outdoor networks, we further conduct outdoor experiments. The collected trace from outdoor network is fed into the PA-HMM and then we study the simulated link behaviors. **Network setup.**

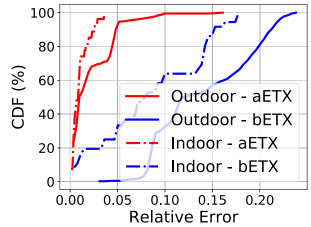
As shown in Figure 9(a), we place the TelosB nodes (CC2420 and MSP430) in an area covering buildings and woods. The average distance between any two nodes is around 40 meters. No artificial interference is introduced to the network. The nodes are powered by either mobile chargers or AA batteries.



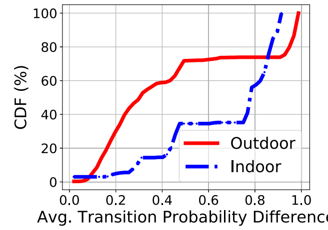
(a) Outdoor network setup



(b) Average link correlation

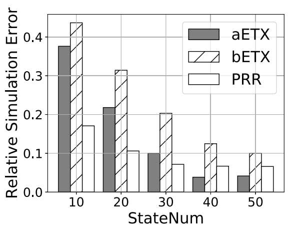


(c) aETX/bETX simulation

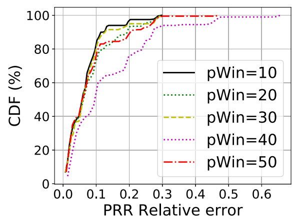


(d) Transition matrix

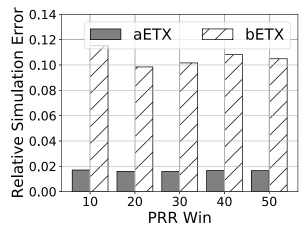
Fig. 9. Outdoor evaluation of the PA-HMM.



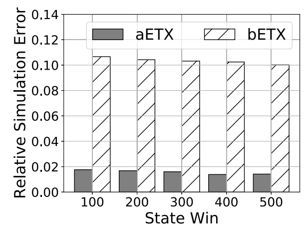
(a) State number with k-means



(b) PRR with varying PRRWin



(c) aETX/bETX with varying PRRWin



(d) aETX/bETX with varying StateWin

Fig. 10. The impact of the PA-HMM parameters.

Simulation results. Figure 9(b) demonstrates the average link correlation measured with different channels in indoor/outdoor environments. Over 80% link correlation values in outdoor environment are smaller than 0.6, while around 50% link correlation values in indoor environment are larger than 0.8. The measurements show that link correlation is generally weaker in outdoor.

Figure 9(c) shows the comparison of the relative error of simulated aETX/bETX in indoor/outdoor environments. We can see that 1) aETX is preserved more accurately than bETX. The reason is that given the same network neighborhood, aETX has fewer possible values and is often dominated by the best link. Differently, bETX has more possible values and can be affected by multiple poor links. As a result, the inter-link relationship has a larger impact on bETX and thus bETX is more difficult to characterize. 2) For both aETX and bETX, the simulation errors are smaller in indoor environment than outdoor environment. The results are somewhat beyond our expectation as we assume outdoor environments should be easier to characterize. We further analyze the model details and the state transition matrix, as the transition matrix plays a significant role in the Markov model for simulating the packet trace. We use the difference between the largest transition probability and the average transition probability as an indicator to evaluate whether matrix is more deterministic.

Figure 9(d) shows the metric comparison of indoor and outdoor models. We can see that, for more than 70% outdoor states, the difference is smaller than 0.5. While for more than 60% indoor states, the difference larger than 0.78. When the difference is large, it means the state transitions are more deterministic and more repeatable. Otherwise, it means the transition probability to each different state is similar, thus the simulated trace tends to have more variations.

E. Impact of the Parameters in the PA-HMM Model

To further reveal the impact of the timing factors, we conduct experiments to study the following important parameters in PA-HMM:

- StateNum: As we have discussed in Section IV-E, we can use state clustering to reduce the number of states and reduce the simulation overhead. To study the impact of k-means on the simulation accuracy, we vary the number of k and study the simulated aETX/bETX.
- PRRWin: The window size for accounting packet reception ratio (PRRs).
- StateWin: The window size for accounting the performance metrics (aETX and bETX).

For both PRRWin and StateWin, a larger window size captures the long-term behaviors while a smaller window size captures the short-term behaviors.

Figure 10 shows the result. As shown in Figure 10(a), as the number of clusted states increases, all the relative simulation errors for PRR, aETX and bETX decrease. The reason is that when merging all states to a small number of clustered states, the state transitions and the emission rates are not as representative as those without state clustering. However, we can also infer that when the number of states is set to 50, the accuracy is similar with that without clustering, which means in our network conditions, there are around 50 essential network states that have the most significant impact on the simulation.

Figure 10(b) shows the PRR simulation errors with varying PRRWin. We can see that PRR Win $=$ 50 achieves the most accurate simulation. This is because long-term PRR trace is more stable. However, PRRWin $=$ 20, instead of the smallest PRRWin $=$ 20, yields the least accurate simulation. Figure 10(c) shows the aETX/bETX simulation with varying PRRWin. Interestingly, unlike the single-link simulation, in the multi-link simulation, PRRWin $= 20$ achieves the most accurate results, which means that the high-level performance is more accurately preserved while the low-level link behaviors are less accurately preserved. The possible reason is that when accounting PRR using PRRWin $= 20$, the emission rates to the PRR tuples are more evenly distributed, and thus the single link PRR will experience larger errors. As long as the probability of each PRR tuples remains similar to the initial trace, the aETX/bETX performance can be preserved.

Figure 10(d) shows the aETX/bETX simulation with varying StateWin. Compared to the simulation with varying PRRWin, we can see that larger StateWin will yield lower simulation errors. The reason is two fold: a) long-term behaviors are more stable; b) Different from the PRRWin, the step used in the StateWin is 100 packets, thus the StateWin demonstrates clearer trend that long-term behaviors can be more accurately characterized.

VI. Discussion on Open Issues

In this section, we discuss the open issues and possible future directions for the PA-HMM based simulation.

A. Impact of Packet Collisions

In practical scenarios, packet collisions are inevitable and common. The packet collisions will affect PRR for single link behavior and link correlation for multi-link behavior. In our model, the impact of packet collisions is inherently characterized in the single-link and multi-link features.

However, if the level of packet collisions changes drastically before and after the simulation modeling process, the accuracy can still be affected. Specifically,

1) The case when there are heavy collisions during the trace collection for modeling but light collisions when protocols run in practice. In this case, the input metrics of the model are essentially affected by the collisions, which can further lead to under-estimated performance (uETX/aETX/bETX) and link features (PRR) in the simulation.
2) The case when there are light collisions during the trace collection but heavy collisions when protocols run in practice. In this case, the simulated trace will yield over-estimated performance and link features. It is worth noting that, if the level of packet collisions remains similar in trace collection and in the practical protocol running, the simulation results will not be affected.

As packet collisions are protocol specific and hard to predict, it is difficult to explicitly incorporate the information of packet collisions to the simulation model. A possible way to deal with the impact is to trade the generality of the model for collision-awareness. In the initial data collection, we collect the trace of packet receptions separately for different protocols running in the network. When simulating a given protocol, we use the model obtained from the trace with the same protocol. Then the impact of packet collisions could be reduced.

B. Extension to Mobile Networks

Our trace-driven simulation does not work for mobile networks, as the spatial-temporal diversities vary drastically with the user mobility.

We discuss how to accommodate our work to the mobile networks as follows. The key challenge is that user mobility will change the link features and performance characterized in the collected trace. Intuitively, if we can establish a link between the geo-information and link features, we can adapt the approach to mobile networks. In mobile networks, it is reasonable to assume that 1) pair-wise user distance and 2) the environmental noise are known or could be estimated. Then we need to find a way to infer the link features using the above information. Fortunately, existing works on modeling of link features can be used and combined to implement the above idea. Specifically, the single-link feature, packet reception rate (PRR), can be characterized using the radio propagation model and the link quality estimation models in [35]. Similarly, the multi-link correlation can be characterized using the models [38]. It is worth mentioning that other models on propagation, link quality/correlation estimation can also be considered as alternatives. With the modules, we can obtain single-link and multi-link features from the geo-info, and further derive the performance states using the proposed performance model.

C. Limitations of the Trace-Driven Simulation Compared to the Event-Driven Simulation

We summarize the limitations of the proposed work, compared to the event-driven simulations.

1) Compared to the trace generation based on event-driven simulation, the trace-driven simulation (trace generated using initial trace) strictly conforms with the observed trace properties. While it is good for repeatable simulations, it can be a limitation when we want to freely explore different network conditions.
2) The trace-driven simulation requires a long initial process of trace-collection and model training, while the event-driven simulations like TOSSIM only require measurement of RSSI and noise traces.
3) The requirement of preserving the high-level properties adds additional complexity to the simulation model. Though it is affordable for simulations, under some scenarios that require fast execution of the simulation, it can be a limitation.

From the above discussion, we can see that the proposed model is more suitable for simulation to achieve repeatable network conditions and performance, while it is not favorable when used to explore various network conditions or a fast execution is required.

VII. Conclusion

In this article, we investigate the problem of repeatable wireless network simulation. We first propose a performance model that considers spatial-temporal link correlation to accurately characterize the single-hop ETX performance. Based on the performance model, we further propose a novel Performance Aware Hidden Markov Model for wireless network simulation. The evaluation results show that the performance model achieves more accurate ETX modeling for both anycast and broadcast/multicast, and the PA-HMM based simulation can simulate both the link-level behaviors as well as the protocol performance. We will focus on the combination of event-driven and trace-driven simulations, and add flexibility to the model while preserving the simulation accuracy.

REFERENCES

[1] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, "An empirical study of low-power wireless," *ACM Trans. Sensor Netw.*, vol. 6, no. 2, p. 16, 2010.

[2] F. Ding, A. Song, D. Zhang, E. Tong, Z. Pan, and X. You, "Interference-aware wireless networks for home monitoring and performance evaluation," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 3, pp. 1286–1297, Jul. 2018.

[3] D. Liu, Z. Cao, Y. He, X. Ji, M. Hou, and H. Jiang, "Exploiting concurrency for opportunistic forwarding in duty-cycled IoT networks," *ACM Trans. Sensor Netw.*, vol. 15, no. 3, pp. 1–33, Aug. 2019.

[4] C. Yan, G. Zhang, X. Ji, T. Zhang, T. Zhang, and W. Xu, "The feasibility of injecting inaudible voice commands to voice assistants," *IEEE Trans. Dependable Secure Comput.*, early access, Mar. 19, 2019, doi: 10.1109/TDSC.2019.2906165.

[5] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: Accurate and scalable simulation of entire TinyOS applications," in *Proc. ACM SenSys*, 2003, pp. 126–137.

[6] A. Kamthe, M. A. Carreira-Perpinán, and A. E. Cerpa, "M&M: Multi-level Markov model for wireless link simulations," in *Proc. ACM SenSys*, 2009, pp. 57–70.

[7] K. Srinivasan *et al.*, "The κ factor: Inferring protocol performance using inter-link reception correlation," in *Proc. ACM MobiCom*, 2010, pp. 317–328.

[8] S. Wang, S. M. Kim, Y. Liu, G. Tan, and T. He, "CorLayer: A transparent link correlation layer for energy efficient broadcast," in *Proc. ACM MobiCom*, 2013, pp. 51–62.

[9] S. Wang, A. Basalamah, S. M. Kim, G. Tan, Y. Liu, and T. He, "A unified metric for correlated diversity in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6215–6227, Sep. 2016.

[10] Y. Liu, Y. He, M. Li, J. Wang, K. Liu, and X. Li, "Does wireless sensor network scale? A measurement study on GreenOrbs," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 10, pp. 1983–1993, Oct. 2013.

[11] J. Wang, Y. Liu, Y. He, W. Dong, and M. Li, "QoF: Towards comprehensive path quality measurement in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 4, pp. 1003–1013, Apr. 2014.

[12] T. S. Rappaport *et al.*, *Wireless Communications: Principles and Practice*, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[13] Z. Zhao, Z. Wang, G. Min, and Y. Cao, "Highly-efficient bulk data transfer for structured dissemination in wireless embedded network systems," *J. Syst. Archit.*, vol. 72, pp. 19–28, Jan. 2017.

[14] X. Zheng, D. Xia, X. Guo, L. Liu, Y. He, and H. Ma, "Portal: Transparent cross-technology opportunistic forwarding for low-power wireless networks," in *Proc. ACM MobiHoc*, 2020, pp. 1–12.

[15] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli, "Valuable detours: Least-cost anypath routing," *IEEE/ACM Trans. Netw.*, vol. 19, no. 2, pp. 333–346, Apr. 2011.

[16] A. Basalamah, S. M. Kim, S. Guo, T. He, and Y. Tobe, "Link correlation aware opportunistic routing," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 3036–3040.

[17] K. Srinivasan, M. A. Kazandjieva, S. Agarwal, and P. Levis, "The β-factor: Measuring wireless link burstiness," in *Proc. SenSys*, 2008, pp. 29–42.

[18] A. Kamthe, M. Á. Carreira-Perpiñán, and A. E. Cerpa, "Improving wireless link simulation using multilevel Markov models," *ACM Trans. Sensor Netw.*, vol. 10, no. 1, pp. 1–28, Nov. 2013.

[19] A. Kamthe, M. A. Carreira-Perpinan, and A. E. Cerpa, "Quick construction of data-driven models of the short-term behavior of wireless links," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 160–164.

[20] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, no. 5, pp. 1253–1265, Sep. 1960.

[21] G. T. Nguyen, R. H. Katz, B. Noble, and M. Satyanarayanan, "A trace-based approach for modeling wireless channel behavior," in *Proc. Conf. Winter Simulation*, 1996, pp. 597–604.

[22] S. A. Khayam and H. Radha, "Markov-based modeling of wireless local area networks," in *Proc. Int. Workshop Modeling Anal. Simulation Wireless Mobile Syst.*, 2003, pp. 100–107.

[23] M. Salama, Y. Elkhatib, and G. Blair, "IoTNetSim: A modelling and simulation platform for end-to-end IoT services and networking," in *Proc. 12th IEEE/ACM Int. Conf. Utility Cloud Comput.*, Dec. 2019, pp. 251–261.

[24] R. Figura, M. Ceriotti, S. Jungen, S. Hevelke, T. Hagemeier, and P. J. Marrón, "M$o\rho\phi\varepsilon\upsilon\varsigma$: Simulate reality for the orchestration of deployed networked embedded systems," in *Proc. ACM EWSN*, 2018, pp. 145–156.

[25] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level sensor network simulation with COOJA," in *Proc. 31st IEEE Conf. Local Comput. Netw.* Nov. 2006, pp. 641–648.

[26] M. Strübe, F. Lukas, B. Li, and R. Kapitza, "DrySim: Simulation-aided deployment-specific tailoring of mote-class wsn software," in *Proc. 17th ACM Int. Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, 2014, pp. 3–11.

[27] Y. Li, D. Graham, G. Zhou, X. Qi, S. Deng, and D. Xiao, "Discrete-time Markov model for wireless link burstiness simulations," *Wireless Pers. Commun.*, vol. 72, no. 2, pp. 987–1004, Sep. 2013.

[28] T. Zhu, Z. Zhong, T. He, and Z. Zhang, "Achieving efficient flooding by utilizing link correlation in wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 21, no. 1, pp. 121–134, Feb. 2013.

[29] Z. Zhao, W. Dong, J. Bu, T. Gu, and G. Min, "Accurate and generic sender selection for bulk data dissemination in low-power wireless networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 2, pp. 948–959, Apr. 2017.

[30] S. Wang, A. Basalamah, S. M. Kim, S. Guo, Y. Tobe, and T. He, "Link-correlation-aware opportunistic routing in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 47–56, Jan. 2015.

[31] D. Nguyen, T. Tran, T. Nguyen, and B. Bose, "Wireless broadcast using network coding," *IEEE Trans. Veh. Technol.*, vol. 58, no. 2, pp. 914–925, Feb. 2009.

[32] A. Cerpa, J. L. Wong, M. Potkonjak, and D. Estrin, "Temporal properties of low power wireless links: Modeling and implications on multi-hop routing," in *Proc. ACM MobiHoc*, 2005, pp. 414–425.

[33] A. Köpke, A. Willig, and H. Karl, "Chaotic maps as parsimonious bit error models of wireless channels," in *Proc. IEEE INFOCOM*, Mar. 2003, pp. 513–523.

[34] X. Zheng, Z. Cao, J. Wang, Y. He, and Y. Liu, "Interference resilient duty cycling for sensor networks under co-existing environments," *IEEE Trans. Commun.*, vol. 65, no. 7, pp. 2971–2984, Jul. 2017.

[35] D. Liu, Z. Cao, Y. Zhang, and M. Hou, "Achieving accurate and real-time link estimation for low power wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2096–2109, Aug. 2017.

[36] O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson, "Low power, low delay: Opportunistic routing meets duty cycling," in *Proc. IEEE IPSN*, Apr. 2012, pp. 185–196.

[37] Z. Zhao, W. Dong, J. Bu, T. Gu, and C. Chen, "Exploiting link correlation for core-based dissemination in wireless sensor networks," in *Proc. IEEE SECON*, Jun. 2014, pp. 372–380.

[38] Z. Zhao, W. Dong, G. Guan, J. Bu, T. Gu, and C. Chen, "Modeling link correlation in low-power wireless networks," in *Proc. IEEE INFOCOM*, Apr. 2015, pp. 990–998.

Zhiwei Zhao (Member, IEEE) received the B.S. degree from Xi'an Jiaotong University in 2010 and the Ph.D. degree from the College of Computer Science, Zhejiang University, in 2015. He is currently an Associate Professor with the School of Computer Science and Engineering, University of Electronic Science and Technology of China (UESTC). His research interests include edge computing, the IoT systems, heterogeneous wireless networks, and protocol design. He is a member of the ACM and CCF.

Geyong Min (Member, IEEE) received the B.Sc. degree in computer science from the Huazhong University of Science and Technology, China, in 1995, and the Ph.D. degree in computing science from the University of Glasgow, U.K., in 2003. He joined the University of Bradford as a Lecturer in 2002, became a Senior Lecturer in 2005 and a Reader in 2007, and was promoted to a Professor in computer science in 2012. He is currently the Chair Professor and the Director of the High Performance Computing and Networking (HPCN) Research Group, University of Exeter, U.K. His main research interests include next-generation Internet, wireless networks, mobile computing, cloud computing, big data, multimedia systems, information security, system modeling, and performance optimization.

Wei Dong (Member, IEEE) received the B.S. and Ph.D. degrees in computer science from Zhejiang University, Hangzhou, China, in 2005 and 2011, respectively. He is currently a Full Professor with the College of Computer Science, Zhejiang University. His research interests include network measurement, wireless and mobile computing, and sensor networks. He is a member of the ACM and a Senior Member of CCF.

Xue Liu (Fellow, IEEE) received the B.Sc. degree in applied mathematics and the master's degree in automation from Tsinghua University, and the M.Sc. and Ph.D. degrees (Hons.) in computer science from the University of Illinois at Urbana–Champaign. He is currently a Full Professor and a William Dawson Scholar with the School of Computer Science, McGill University. He is also a VP R&D, the Chief Scientist, and the Co-Director of the Samsung AI Center (SAIC) Montreal. He has also worked as the Samuel R. Thompson Chair Associate Professor with the University of Nebraska-Lincoln and as a Visiting Research Scientist with HP Labs, Palo Alto, CA, USA. His research interests are in AI/machine learning and its applications, computer systems and networks, cyber-physical systems and IoT, green computing, and smart energy technologies. He has published more than 300 research articles in major peer-reviewed international journals and conference proceedings in these areas and received several best paper awards. His research has been covered by multiple news and media reports.

Weifeng Gao (Student Member, IEEE) received the B.S. degree from the School of Computer Science and Engineering, University of Electronic Science and Technology of China (UESTC) in 2015, where he is currently pursuing the Ph.D. degree. His research interests include wireless computing, low-power networks, and resource optimization.

Tao Gu (Senior Member, IEEE) received the bachelor's degree from the Huazhong University of Science and Technology, the M.Sc. degree from Nanyang Technological University, Singapore, and the Ph.D. degree in computer science from the National University of Singapore. He is currently an Associate Professor in computer science with RMIT University, Australia. His current research interests include mobile computing, ubiquitous/pervasive computing, wireless sensor networks, distributed network systems, sensor data analytics, the Internet of Things, and online social networks. He is a member of the ACM.

Minghang Yang received the B.S. and M.S. degrees from the School of Computer Science and Engineering, University of Electronic Science and Technology of China (UESTC). He is currently a member of the FMI Group, UESTC, and he will be joining Bytedance, as an Engineer. His research interests include wireless communications and embedded computing.