Graph-Theoretic Complexity Reduction for Markovian Wireless Channel Models

Hassaan Khaliq Qureshi • Junaid Jameel Ahmad • Syed Ali Khayam • Veselin Rakocevic • Muttukrishnan Rajarajan

Abstract Accurate simulation and analysis of wireless networks are inherently dependent on accurate models which are able to provide real-time channel characterization. High-order Markov chains are typically used to model errors and losses over wireless channels. However, complexity (i.e., the number of states) of a high-order Markov model increases exponentially with the memory-length of the underlying channel. In this paper, we present a novel graphtheoretic methodology that uses Hamiltonian circuits to reduce the complexity of a high-order Markov model to a desired state budget. We also demonstrate the implication of unused states in complexity reduction of higher order Markov model. Our trace-driven performance evaluations for real wireless local area network (WLAN) and wireless sensor network (WSN) channels demonstrate that the proposed Hamiltonian Model, while providing orders of magnitude reduction in complexity, renders an accuracy that is comparable to the Markov model and better than the existing reduced state models.

Part of this work has appeared in the proceedings of the IEEE Milcom, San Diego CA, November 2008 [17].

H. K. Qureshi · J. J. Ahmad · S. A. Khayam School of Electrical Engineering & Computer Science, National University of Sciences and Technology, Sector H-12, Islamabad, Pakistan

S. A. Khayam e-mail: ali.khayam@seecs.edu.pk

J. J. Ahmad Universität Konstanz, Konstanz, Germany e-mail: junaid.ahmad@uni-konstanz.de

H. K. Qureshi (⊠) · V. Rakocevic · M. Rajarajan School of Engineering and Mathematical Sciences, City University, London, UK e-mail: hassaan.qureshi.1@city.ac.uk

V. Rakocevic e-mail: V.Rakocevic@city.ac.uk

M. Rajarajan e-mail: R.Muttukrishnan@city.ac.uk **Keywords** Channel modeling · Complexity reduction · Hamiltonian model · Wireless networks

1 Introduction

Due to a lack of available infrastructure to perform realistic wireless experiments, systemlevel simulations are used to evaluate the performance of emerging wireless protocols and services. An accurate model of the wireless channel is an important component of such simulation-based performance evaluation. In the past three decades, channel error modeling techniques have been used extensively to improve the design of communication channels and the protocols that operate on these channels [7,18,19,21]. Using an accurate channel model, one can simulate the channel and can gain insights into the channel's underlying behavior. More importantly, an accurate and low-complexity channel model can be used to tune critical parameters of network protocols and applications at design time and in real-time. Lastly, a low-complexity channel model also allows real-time channel characterization and prediction which is required by rate adaptive protocols and applications.

In the channel modeling context, stochastic models have gained significant research attention [2,3,6,13]. In particular, high-order Markov channel models have been shown to be quite accurate in modeling link layer bit-errors and packet losses [2,11,13,14,20]. Unfortunately, the complexity of Markov models increases with their memory length and consequently the viability of using Markov models in resource-constrained wireless environment is very limited. Thus, accurate approximations of high-order Markov channel models are needed for wireless environments.

Many models have been proposed in recent literature to reduce the complexity of highorder Markov chains [11, 14, 15, 20]. While there exists a clear tradeoff between complexity and accuracy (lower the model complexity, lower the accuracy), existing low-complexity channel models [16] (with the exception of the bipartite model [20]) reduce the channel model's complexity to a fixed level and therefore do not cater for the emerging heterogeneous communication devices; for instance, on a given channel, high-end wireless devices (e.g., desktop and laptop computers) can afford higher complexity channel models than lowend devices (e.g., PDAs and smart phones.) To cater for such device heterogeneity, we need channel models that can adapt their complexity to an arbitrary level in accordance with the resources available at a wireless device.

In this paper, we propose a new variable-complexity wireless channel model referred to as the Hamiltonian Model (HM). The HM reduces the complexity of high-order Markov channel models by identifying and aggregating Hamiltonian circuit present in the states of the Markov chain. Given a desired complexity budget in terms of the total number of Markov states, the proposed model identifies a Hamiltonian circuit in the Markov chain, finds cycles of the needed complexity, and then aggregates these cycles into odd and even states based on the number of total states present in those cycles. We further analyze the transition probability matrices for bit patterns that never occur in the collected traces and are referred to as the unused states. We further demonstrate that when the unused state parameter is available, we can reduce the complexity associated with higher order Markov channel models.

The performance of the proposed model is compared with the Bipartite Model (BM) of Willig [20]. The performance characteristics of both models are evaluated using a comprehensive dataset of actual traces collected in two different environments: (i) 802.11 MAC layer bit errors at 5.5 Mbps; (ii) 802.15.4 MAC layer bit errors at 250 Kbps. We compare the models' accuracies by measuring their closeness to the actual wireless channel traces using

an information-theoretic Kullback–Leibler-based divergence measure and by comparing the Cumulative Distribution Functions (CDF) of bit errors. Our results demonstrate that HM has significantly higher modeling accuracy than BM.

The rest of this paper is organized as follows. Section 2 describes related work in this area. Section 3 provides the background that is required to understand the contribution of this paper. Section 4 describes error trace collection on Wireless Sensor and Local Area Networks. Section 5 describes the proposed Hamiltonian model. Section 6 outlines performance evaluation of the HM model and compares its performance with the BM model. Section 7 summarizes the key conclusions of this paper.

2 Related Work

Some previous studies have proposed methods to reduce the complexity of high-order Markov models [2,11,20]. In Khayam and Radha [11], guidelines were proposed to accurately model Markov based wireless channels and a constant complexity probabilistic model was proposed. In Khayam and Radha [10], authors showed empirically that low complexity hierarchical and hidden Markov models cannot characterize the bit error processes and proposes to employ high-order Markov chains for accurate channel characterization. These studies, however, resulted in models with fixed, non-scalable complexities.

Since complexity and accuracy of a model generally exhibit a direct proportionality relationship, we argue that, in view of the heterogeneity of contemporary wireless devices, a model should be able to scale its complexity in accordance with the complexity that can be afforded at a wireless device. More specifically, given a complexity budget (for instance, in terms of the number of model states,) a scalable channel modeling algorithm should be able to produce a channel model to satisfy that budget. We are only aware of two studies ([2] and [20]) which approach the wireless channel modeling problem in this way.

Chen and Rao [2,3] used the lumpability framework to reduce the order of a Markov channel model. However, the lumpability conditions place very stringent constraints on the transition probabilities of a Markov chain. These constraints are generally not satisfied by real-life wireless channel models. Therefore, we do not compare performance of our proposed technique with lumped Markov chains.

Willig [20] proposed a scalable-complexity Bipartite Model for wireless channels. The bipartite model uses the notion of bipartite graphs and divides the graph according to its burst order based on the binary indicator sequence and then according to burst order Probability Mass Function (PMF) is calculated. The PMF $f_x(x)$ on a countable sample space $s \subseteq R$ is given by

$$f_X(X) = \begin{cases} \Pr(X = x), & x \in s \\ 0, & x \in R \setminus s \end{cases}.$$

In order to build a model from the traces one needs to choose the numbers of states K1 and K2 i.e., total number of odd or even states. Every state is assigned a discrete random variable with probability distribution. This random variable takes values on a finite interval of the natural numbers based on a specific state which determines the burst length of good and bad bits. Based on the burst intervals the transition matrix p is generated for the transitions from the good states to the bad states and vice versa.

Our data analysis on the collected traces revealed generalized probability distributions of good and bad-bursts for a Full State Markov (FSM) chain of arbitrary order. The probability distributions are derived in terms of FSM chain transition and steady-state probabilities and

we noticed that generally 99% of the times the transitions are in either good state (0) or in a bad state (1). This observation renders useful insights into important FSM characteristics, which helped us to develop guidelines for defining FSM state space partitions for our proposed Hamiltonian model (HM) which is not used by the BM. The Bipartite model uses the notion of a bipartite graph and divides any order of Markov chain into only two states namely an odd state and even state and do not separate the good state and the bad state. This is a very important property because it is undesirable to merge good and bad states together with an odd states or even states. Moreover, the accuracy of BM depends on a selected value of complexity. This fundamental flaw further degrades the performance of the BM when compared with the proposed HM.

3 Background and Notation

In this section, we present brief description of the binary nature of traces and the commonly used high-order Markov chain models of wireless channels.

3.1 Representation of Binary Wireless Traces

Traces collected over a wireless medium generally represent two states. One state is the good state and the other state is the bad state or the lossy state. Hence we can characterize wireless traces generally as a binary time series $\{x(n)\}_{n=1}^{l}$, where $x(n) \in \{0, 1\}$ and l is the length of the error trace.

Without loss of generality, throughout this paper we use zero to represent an error-free bit and one for a bit in error. The sequence of these bits forms alternating bursts of zeros and ones. If the burst consists of number of zeros then we refer to it as a *good burst* and if the burst consists of ones then we call it a *bad burst*. The trace can hence be represented as pairs of good and bad bursts: $(N_1, G_1), (N_2, G_2), \ldots, (N_n, G_n)$, where N_n and G_n represent the length of the *n*th good and the bad bursts, respectively [17]. Many channel modeling studies have showed that this binary representation is suitable for representing channel traces [2,9,10,15,20].

3.2 *K*th Order Markov Chains

A Markov chain of memory K is a discrete time random process whose probabilities for going to future states at a given present state are independent of the past states. For a memory length of K, the Markov chain comprises of 2^{K} possible combinations of K consecutive bits. If we have a set of states K which consists of $S = \{S_1, S_2, S_3, \ldots, S_n\}$, then the process starts in one of these states and moves successively from one state to another. If we define S_i as a current state then for moving to next state S_j , the probability will be denoted by P_{ij} . The probabilities are called transition probabilities and are computed by sliding bit by bit a K bit memory window over the data [5,8]. The sum of all transitions from a Markov state must sum to one, for any state S we have $P_{(2S+1) \mod 2^{K}} = 1 - P_{(2S) \mod 2^{K}}$. For calculating the transition probabilities, we fixed the order of Markov chain and performed the above operation on real 802.15.4 and 802.11b traces to generate the probability transition matrix. The probability transition matrix for a 3-rd order (memory length=3) Markov chain is shown below in Table 1.



Table 1 Transition probability matrix for 3-RD order Markov chain

Fig. 1 A 3-rd order (memory length = 3) Markov chain

In this paper, Markov chain corresponding to even (odd) decimal numbers is referred as even (odd) states. Using the above notation, an example 3-rd order Markov chain is shown in Fig. 1; only transitions to even states are shown. If the Markov chain is in an even state, the last received bit (i.e., the least significant bit position in the memory window) must be error-free. Similarly, a Markov chain in the odd states implies that the last bit was corrupted. Due to the binary nature of the underlying wireless bit-error process, each Markov chain state can transit to only two other states [8, 12]. This is due to the Markov chain definition in which the memory-window at each time instance is left-shifted by one bit and a one or a zero bit is added to the least-significant bit position. Thus from state *S*, a Markov chain can transit either to even state (2*S*) mod 2^K or to odd state (2*S* + 1) mod 2^K. Since the sum of all transitions from a Markov state must sum to one, for any state *S* we have $P_{(2S+1) \mod 2^K} = 1 - P_{(2S) \mod 2^K}$. It should also be emphasized that once a corrupted bit is received, a *K*-th order Markov chain will return to state 0 (i.e., the no error state) only from state 2^{K-1} after *K* transitions; see in Fig. 1 that at state $2^{3-1} = 4$, the Markov chain wraps around to state 0.

4 Data Collection

To perform realistic performance evaluation over operational channels, we collected a comprehensive dataset of wireless error traces over two different channels: (1) an 802.15.4 WSN channel, and (2) an 802.11b WLAN channel. All traces were collected at the MAC layer after physical layer processing; MAC layer channels are referred to as *residual channels* in prior literature [6,11]. This section describes the data collection and some preliminary trace statistics.



Fig. 2 Setup for 802.15.4 bit error traces

4.1 802.15.4 Data Collection

We used Crossbow's Micaz motes to collect residual bit-error traces over wireless sensor networks. These motes operate on the ISM frequency band of 2.4 GHz and support a peak data rate of 250 Kbps. Sensor motes were running the open-source TinyOS operating system. We modified the source code of TinyOS applications to disable the MAC layer checksum feature at the receiver. Hence, corrupted packets were not dropped in a receiver's kernel, and were passed to a data logging application. The application logged all packets on an attached computer through the serial port. We collected the traces at four different locations or setups. These setups are named according to their geographical location as shown in Fig. 2. The light-shaded mote in Fig. 2 is the base station which received and logged data, while the remaining motes are sending motes which transmitted packets with predefined contents to the base station. In each experiment, one sender transmitted unicast data to the base station and the other senders were inactive; i.e., in each trace collection, there was no channel contention and collisions as there was only a single sender and a single receiver. While performing experiments, motes were kept stationary. In order to collect traces with varying error behaviors, the distance between the motes and the base station was varied from 5 to 12 m. The senders transmitted 20-byte fixed-sized frames at a rate of 10 frames per second. We first performed the experiments by having a direct line of sight (LoS) between the sender and the base station but the error rates observed in those experiments were too low to warrant further analysis. Therefore, throughout this paper, we focus on non-LoS traces.

The average number of frames per trace was approximately 31,000 frames. Thus, the average length of each trace was approximately 5 million bits [6]. For evaluation of HM and BM we used five traces for each setup and a total of 20 traces for all setups. While we collected more actual wireless sensor network traces but the trends observed in 20 traces (5 per setup) are representative of the trends that we observed in other traces. More standardized approaches for data collection can be found in [1].

4.2 802.11 Data Collection

For 802.11b traces, we repeated the same process as the 802.15.4 traces using the topology shown in Fig. 3 and AP (access point) was operating in Distributed Coordination Function (DCF) mode and three wireless stations communicating in the infrastructure network configuration. One of the stations was operating as the server and the remaining two as multicast clients. All wireless stations were Linux boxes using Dlink wireless cards with Prism2



Fig. 3 Setup for 802.11b bit error traces

chipset device drivers. The server was stationary and transmitted a continuous stream of predetermined patterns to the multicast clients. Traces were generated for each bitrates at different stationary client positions with and without LoS. It was observed that, with clear LoS, the error rate at all bitrates was extremely low. Such excellent performance deemed further LoS study inconsequential. Hence, both clients were positioned in a separate room across two walls in order to simulate a more realistic business/classroom/home-network wireless setup and forced to transmit non Line of Sight frames of 512-bytes at a physical layer data rate of 5.5 Mbps. The average length of each trace was approximately 6 million bits. We collected 3 traces at different locations for 5.5 Mbps.

4.3 Kth Order Markov Chain Statistics

In order to accurately and efficiently represent transition probability data and to reduce that complexity associated with higher order Markov chains we investigated the transition probability matrices for the 802.15.4 WSN and 802.11b WLAN collected traces and noticed the bit patterns that never occur in transition probability matrices and are referred to as unused states. In other words, such states result in all-zero columns of the transition probability matrix. An all-zero column implies that the probability of jumping to that state from any state is zero which is observed because that state was not present in the training data. The number of unused states is larger for the traces with lower BER because all the error states are not observed in a finite length trace. Conversely, as the error rate increases, the number of observing unused states decreases proportionally. We noticed many unused states in 802.15.4 WSN traces and the number of unused states grew as the order of Markov chain [9]. The total number of unused states per setup for 802.15.4 WSN traces is shown in Table 2. It shows that the number of unused states increase with the increase in the order of Markov chain and maximum number of unused states are observed at K = 9. The unused states in Room 3 are fewer as compared with other setups since the sender and receiver in Room 3 were separated by a concrete wall. Therefore, this setup has a high BER when compared with other setups. We witnessed more unused states in case of Upper floor in which sender and receiver were at the farthest distance from each other. However in our 802.11b WLAN traces we noticed that there is no unused state up to K = 9 i.e., the order by which we tested our model for complexity reduction.

4.4 Trace Bit Error Rate (BER) Statistics

Table 3 shows the average bit error rates of 802.15.4 and 802.11b original traces, respectively. It can be seen that highest bit error rate was observed for Room 3 as in this case the sender

Setup	Number of states						
	16	32	64	128	256	512	
Room2	0	0	6	41	209	772	
Room3	0	0	5	36	138	425	
Stairs	0	0	0	13	129	595	
Upper floor	0	4	46	205	610	1,512	

 Table 2
 Total number of unused states per setup

Table 3Average bit error rateof actual traces

Setup	BER
802.15.4	
Room2	0.00085689
Room3	0.01519663
Stairs	0.00737452
Upper floor	0.00735198
802.11b	
Location 1,2,3	0.003067968

and receiver were separated by a concrete wall which was in between them. Room 2 has the lowest bit error rate as in this case the sender and receiver were at the closest distance from each other and there was a glass window in between them. The bit error rate of Upper floor and Stairs are similar because of being in the same vicinity but the distance of Upper floor was more as compared to Stairs setup. In case of 802.11b where the receiver's were kept stationary, the average bit error rate observed at different locations was 0.003.

5 The Hamiltonian Wireless Channel Model

Let a K-th order Markov chain be represented as a K vertex connected digraph G = (V, E)with positive edge weights. Markov chain of any order forms a connected graph as long as both state transitions probability in each Markov state are greater than zero. We observed that Markov chains exhibit many interesting graph-theoretic properties which can be used to reduce the complexity associated with higher order Markov chains and those properties can be used to develop a scalable model. A graph in which we can traverse each vertex or node exactly once forms a Hamiltonian circuit. A Hamiltonian circuit can be identified in Markov chains of any arbitrary order. By using this property, we can further arrange the states according to the nodes traversed, which gives an easy method for aggregating states that comprise the Hamiltonian circuit. Since a cycle of arbitrary length can be identified in the Markov digraph, the states of the circuit can be aggregated to a desired state budget. These characteristics are a consequence of the Markov chain construction and are therefore present at all orders of the Markov chains. Moreover, the Hamiltonian circuit formed in the Markov digraph clearly identifies the good and bad nodes which remain separated during state aggregation. This is a very important property because it is generally undesirable to merge good (even) and bad (odd) states together [20]. After Hamiltonian state aggregation, the merged states probabilities are aggregated and normalized into one aggregate state of the



Fig. 4 An example of complexity reduction using the Hamiltonian Model. a Markov chain of order K = 4; b Hamiltonian circuit; c Hamiltonian circuit arranged by vertices traversed; d Hamiltonian circuit reduced to K = 3

low-complexity model. This graph-theoretic realization helps us in reducing the complexity in a finite time and in a simple and easy manner. In the following discussion, we provide an example of applying the proposed Hamiltonian Model based state aggregation on a Markov chain of order K = 4.

A Markov chain of order K = 4 is shown in Fig. 4a. If we start from state zero and start traversing all the other vertices, the order of traversed vertices in the consequent Hamiltonian circuit will be: 0-1-3-7-15-14-13-11-6-12-9-2-5-10-4-8-0. This is depicted with dotted line or edges in Fig. 4b. After finding a Hamiltonian cycle we observed that there are *n* cycles in the graph. Based on the cycles still left in the graph we applied our algorithm to work in such a way to find a cycle exactly aggregating the graph into K = 3 states. This procedure is shown in Fig. 4c. We first found a cycle consisting of 5 vertices (7, 5, 14, 13, 11,) and then again we found another edge coming from vertex 5 towards vertex 11 which was already merged. Hence, we looked for cycles consisting of ten nodes and then we separated the cycle into odd and consecutive even nodes.

Figure 4d shows the final transformation from K = 4 to K = 3. We then normalized the probabilities of the states aggregated in odd and even parts. We first compressed our model from $2^9 = 512$ to $2^8 = 256$ and then to $2^7 = 128$ and so on up to $2^3 = 8$. We applied the algorithm on the graph as depicted in the figures and at each state we generated artificial traces generated by our model. It is also possible to find the Hamiltonian circuit by traversing vertices different than the traversed vertices shown in Fig. 4. For that the model remains the same and is still applicable in reducing the needed complexity.

The HM does not merge FSM states 0, 1 and 2^{K-1} and keeps them in a separate partition, while grouping all the remaining FSM states into two partitions namely odd states and even

states. The notion of separating the 2^{K-1} state comes from the wrapping of Markov chain from state 2^{K-1} after K transitions to zero state as shown in Fig. 1. The possibilities of finding a Hamiltonian cycle increases with an increase in the order of Markov chain. In order to find a Hamiltonian cycle, Depth First Search (DFS) is used to compute Hamiltonian cycle and the big-O complexity of finding a Hamiltonian cycle is O(max{V, E}). The pseudo code of the algorithm is given below.

Algorithm: A systematic way to reduce the complexity of a *K*-th order Markov chain using Hamiltonian Circuit.

Input: Markov chains of any arbitrary order. Part 1: Generate Hamiltonian Circuit

- 1. Let G be the Markov chain graph with a set of Vertices (states) denoted by $V = \{V_1, V_2, V_3, \ldots, V_n\}$. Select V_1 as starting vertex and add V_1 to P which is defined to be the set of visited vertices in the graph.
- 2. On each traversal through the graph:
 - a. If V_i is *already not* visited (does not exist in P) then, add V_i to P.
 - b. If V_i is already visited and leads towards the dead end then visit the other child of the last parent node and subtract the visited vertex from P and continue adding unvisited vertices to P.
- 3. If P forms a Hamiltonian cycle then go to step 4, if not then iterate step 2 until all vertices are traversed to the order of Markov chain and Markov chain wraps around to state zero or P forms a Hamiltonian cycle.

Part II: Reduce the K- th order Hamiltonian Circuit to (K - 1)-th order

- 4. Let P be the graph with a set of Vertices (states) denoted by a vertex set of P.
- 5. Arrange the vertices according to the traversed order in P.
- 6. Reduce the Markov chain to its lower order (K 1) by:
 - a. Separate 0, 1, and 2^{K-1} states and make them independent states.
 - b. Merge and classify all other states into two states (odd and even).

7. Iterate step 6 until the markov chain does not wrap around to form a new graph H.

The problem of finding a Hamiltonian cycle is NP-complete. However, to determine whether a given order of Markov chain has a Hamiltonian cycle undertakes a non-exhaustive search method as each Markov state transits either to an even state $(2S) \mod 2^K$ or an odd state $(2S + 1) \mod 2^K$ (see Fig. 1). These special properties make the process of finding a Hamiltonian cycle easier in a *K*-th order Markov chain when compared with general graphs.

6 Performance Evaluation

In this section, we use the error traces to compare the performance of the Hamiltonian Model with the Bipartite Model using Bit Error Rate (BER), Kullback-Leibler Divergence (KLD) and bit error distributions.

6.1 Bit Error Rate (BER)

We calculated the BER of 802.15.4 and 802.11b synthetic traces generated by Hamiltonian Model and the Bipartite Model. Table 4 shows the Average BER per setup of Hamiltonian

Model	Setup	Number of states			
	e	64	128	256	
Hamiltonain_802.15.4	Room2	0.004119	0.001383	0.000416	
	Room3	0.016309	0.016735	0.021816	
	Stairs	0.021501	0.009547	0.011524	
	Upper floor	0.01853	0.008242	0.014039	
Hamiltonian_802.11b	Location 1,2,3	0.007559	0.002649	0.005284	
Bipartite_802.15.4	Room2	0.0259419	0.0388292	0.0540207	
	Room3	0.0203247	0.0324104	0.0725722	
e	Stairs	0.0066553	0.0099122	0.0453318	
	Upper floor	0.0373605	0.0800648	0.1279515	
Bipartite_802.11b	Location 1,2,3	0.038968	0.03524	0.030718	

 Table 4
 Average bit error rate of synthetic traces

Model traces parameterized from actual traces at K = 6, K = 7 and K = 8. It also shows the BER of Bipartite Model for varying number of K-states. Comparing with Table 3, it can be observed that BER of Hamiltonian Model (HM) traces are closer to the actual traces' BER than the Bipartite Model (BM). Overall, the inaccuracy in BER estimates shows the opposite trends in HM and BM. For the HM, the inaccuracy decreases with an increase in the number of states except for Room 3 which has a larger number of unused states when compared with other setups. The BM, on the other hand, incurs more inaccuracy for higher number of states. Thus for the BM introducing more state does not necessarily increase the accuracy of the model.

Typically, an increase in the complexity of a model (e.g., with an increase in the order of a Markov chain) causes the accuracy of the model to improve. However, for the HM model we did not observe this proportionality trend between complexity and accuracy. On the contrary, after empirically evaluating the HM model's accuracy for varying state merging orders, we observed that a particular HM order provides better accuracy than orders above and below it when unused states are not used. When the unused state parameter is available, we can reduce the complexity associated with the model by tuning the unused state parameter. This is also shown in Table 5 for K = 6, K = 7 and K = 8. For generating HM synthetic traces we tuned our model in such a way that when unused states occur in transition probability matrix the model should always jump to a good state. The notion of jumping to good state when zero columns occur in transition probability matrix is taken from the overall behavior of the WSN traces which shows that 99% of the times the state transitions are in good state. Hence, when compared with Table 3, the model shows very similar bit error rate for varying number of K-states in case of 802.15.4 WSN traces and is shown in Table 5.

While the BER estimates give us an overall picture of the accuracy of a model, for more elaborate performance comparison, we need to compare the burst distributions of the HM and BM. To this end, we compare the good- and bad-bursts' distributions of the two models in the next two sections.

6.2 Kullback-Leibler Divergence (KLD) of Good- and Bad-Bursts

Entropy is a measure of the average number of bits required to represent all outcomes of a probability distribution. The Kullback-Leibler divergence quantifies the difference in the

Model	Setup	Number of states			
		64	128	256	
Hamiltonain_802.15.4	Room2	0.000898	0.000841	0.000835	
	Room3	0.01614	0.015131	0.015937	
	Stairs	0.007777	0.007411	0.006951	
	Upper floor	0.00782	0.007335	0.007236	

 Table 5
 Average bit error rate of synthetic traces with unused states used

entropies of two probability distributions [4]. The KLD divergence quantifies the sourcecoding-like overhead incurred by employing a model instead of the actual source. For two probabilities distributions p and q defined over a common alphabet Ψ , the KL divergence is defined as:

$$D(p||q) = \sum_{x} p(x) \log_2 \frac{p(x)}{q(x)}.$$

The Kullback-Leibler Divergence (KLD) is not a true distance metric and has the following properties:

(a) Non-negativity, $D(p||q) \ge 0$.

(b) Non-symmetry, $D(p||q) \neq D(q||p)$.

(c) Zero if, $D(p||q) = 0 \Leftrightarrow p = q$.

The KLD has shortcoming of non symmetry and it requires two distributions to be continuous with respect to each other. Therefore, instead of Kullback-Leibler we used the KL based Resistive (R) divergence measure defined as

$$\frac{1}{R} \equiv \frac{1}{D(p \parallel q)} + \frac{1}{D(q \parallel (p))}.$$

As with the KL measure, small values of R represent more similarity between p and q. Therefore, we employ the KL-based R divergence measure for model verification on 802.15.4 and 802.11b wireless traces. For performance evaluation reference, p(x) and q(x) respectively represent the burst distributions from an actual trace and a trace artificially generated by a model, where x is a good-bursts random variable. Similarly, R(p(y), q(y)) is also computed for the bad-bursts random variable y. We used the KL based Resistor-Average (R)divergence measure defined in Cover and Thomas [4]. For accuracy evaluation of HM and BM, we compare the R divergence of good- and bad-bursts distributions derived from actual traces and the models. The R divergence observed in Figs. 5, 6, 7, and 8 at different states demonstrate that the HM shows results demonstrating very small R values for good bursts and outperforms the BM. The results for R divergence of 802.15.4 HM traces and BM are demonstrated in Fig. 5. It shows consistent performance with increasing order of K and renders good behavior with increasing state compression. Figure 6 shows the R divergence of bad-bursts for 802.15.4 traces generated from HM and BM. For bad bursts, it demonstrates very small R values elaborating similarity with actual traces when compared with BM and showing lower R divergence at K = 5 because of occurrence of unused states at K = 6. The increase in the order of Markov chain causes occurrence of more unused states and overall in our 802.15.4 WSN traces these states start at K = 6 which is also shown in Table 2. It can be clearly observed that at increasing order of K the HM has very small R divergence



Fig. 5 Resister average divergence of good bursts versus complexity for the 802.15.4 bit error process



Fig. 6 Resister average divergence of bad bursts versus complexity for the 802.15.4 bit error process

in case of bad bursts. On the other hand, the good-bursts and bad-bursts distributions of the BM diverge quite significantly from actual traces. R divergences of good and bad bursts for the 802.11b traces are shown in Figs. 7 and 8. It can be clearly observed that at increasing order of K the HM has very small R divergence in the good bursts case. On the other hand, the good-bursts distributions of the BM diverge quite significantly from actual traces. For bad-bursts, the R divergence of the HM is slightly higher than the BM at K = 9. The slight variations in bad burst R divergence values can be removed by averaging over more traces. Nevertheless, both models are able to capture the bad-bursts behavior quite accurately.

6.3 Cumulative Distribution Function (CDF) of Good and Bad Bursts

We derive CDFs of good and bad bursts from actual network traces and traces artificially synthesized by our model and the BM. For evaluation we took one trace per setup from each of the model at different memory length. The cumulative distribution function for good and bad burst length at K = 5 and K = 8 is shown in Figs. 9, 10, 11 and 12 respectively. The CDF of the HM clearly follows the CDF of actual 802.15.4 traces for both the good and bad



Fig. 7 Resister average divergence of good bursts versus complexity for the 802.11b bit error process



burst distributions. The CDF for K = 8 is also plotted for good and bad burst lengths and is shown in Figs. 13 and 14. The CDF of the HM clearly follows the CDF of actual 802.11b bit error traces in both the good and bad burst length distributions, respectively. While the BM model captures the bad burst CDF reasonably accurately, the HM outperforms BM in capturing bad bursts as shown in Figs. 10 and 12, respectively.

7 Conclusions

In this paper, we presented a novel algorithm to reduce the complexity of high-order Markov channel models to a desired state budget. We performed state aggregation by identifying Hamiltonian Circuits in the Markov digraph. States in the circuit were then aggregated to a given and arbitrary state budget. We demonstrated that the HM provides orders of magnitude reduction in complexity and renders very accurate performance when compared with BM. We showed that by using the unused state parameter we can reduce the model complexity and can have results very similar to actual traces at varying number of K-states. When this parameter is not available, the model complexity renders behavior similar to actual traces at



Fig. 9 Good burst length distribution for 802.15.4 bit error traces at K = 5



Fig. 10 Bad burst length distribution for 802.15.4 bit error traces at K = 5



Fig. 11 Good burst length distribution for 802.15.4 bit error traces at K = 8



Fig. 13 Good burst length distribution for 802.11b bit error traces at K = 8



Fig. 14 Bad burst length distribution for 802.11b bit error traces at K = 8

846

particular order of K-states. However, the HM model still performs better than the Bipartite model in capturing good and bad bursts.

References

- 1. Aguayo, D., Bicket, J., Biswas, S., Judd, G., & Morris, R. (2004). Link-level measurements from an 802.11b mesh network. ACM Sigcomm, 121–132.
- 2. Chen, A. M., & Rao, R. R. (1999). Wireless channel models—coping with complexity. Wireless multimedia network technologies (pp. 271–288). Dordrecht: Kluwer Academic Publishers.
- 3. Chen, A. M., & Rao, R. R. (1998). On tractable wireless channel models. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)* (vol. 2, pp. 825–830).
- Cover, T., & Thomas, J. A. Elements of Information Theory, Second edition. ISBN: 0-471-24195-4.
 Grinstead, C. M., & Snell, J. L. (1997). Introduction to probability: Second revised edition.
- Providence: American Mathematical Society (AMS). ISBN: 0821807498.
 6. Iqbal, A. & Khayam, S. A. (2008). Improving WSN simulation and analysis accuracy using two-tier channel models. *IEEE International Conference on Communications*, 349–353.
- 7. Ji, P., Liu, B., Towsley, D., & Kurose, J. (2002). Modeling frame-level errors in GSM wireless channels. *IEEE Globecom*, *3*, 2483–2487.
- 8. Kemeny, J. G., & Snell, J. L. (1976). *Finite markov chains*. New York: Springer-Verlag. ISBN: 0387901922.
- 9. Khayam, S. A., & Radha, H. (2003). Markov-based modeling of wireless local area networks. ACM MSWIM, 100-107.
- 10. Khayam, S. A., & Radha, H. (2005). Linear-complexity models for wireless MAC-to-MAC channels. ACM WINET, 11(5), 543-555.
- 11. Khayam, S. A., & Radha, H. (2006). Constant-complexity models for wireless channels. *IEEE Infocom*, 1–11.
- 12. Khayam, S. A, & Radha, H. (2007). On the impact of ignoring Markovian channel memory on the analysis of wireless systems. *IEEE International Conference on Communications*, 199–204.
- 13. Khayam, S. A., Radha, H., Aviyente, S., & Deller, J. R. Jr. (2007). Markov and multifractal wavelet models for wireless MAC-to-MAC channels. *Elsevier Performance Evaluation*, 64(4), 298–314.
- 14. Konrad, A., Zhao, B. Y., Joseph, A. D., & Ludwig, R. (2003). A Markov based channel model algorithm for wireless networks. ACM WINET, 9(3), 189–199.
- 15. Köpke, A., Willig, A., & Carl, H. (2003). Chaotic maps as parsimonious bit error models of wireless channels. *IEEE Infocom*, 1, 513–523.
- 16. Mcdougall, J. M. (2003). Low complexity channel models for approximating flat Rayleigh fading in network simulations. PhD thesis from Texas A&M University.
- 17. Qureshi, H. K., Shahzad, K., Khayam, S. A., Rajarajan, M., & Rakocevic, V. (2008). Complexity reduction of Markov channel models for wireless networks using graph theory. *IEEE Military Communications Conference*, 1–7.
- 18. Turin, W., & Nobelen, R.van (1998). Hidden Markov modeling of flat fading channels. *IEEE Journal of Selected Areas in Communications, 16*(8), 1809–1817.
- 19. Wang, H. S. (1995). Finite-state Markov channel—a useful model for radio communication channels. *IEEE Transactions on Vehicular Technology*, 44(1), 163–171.
- 20. Willig, A. (2002). A new class of packet- and bit-level models for wireless channels. TKN-02-009, telecommunication networks group, Technical University, Berlin.
- 21. Zorzi, M., Rao, R., & Milstein, L. (1995). On the accuracy of a first-order Markov model for data transmission on fading channels. *Proceedings of IEEE ICUPC'95*, 211–215.



Hassaan Khaliq Qureshi completed the bachelor degree in Electrical Engineering with a first class honors. He worked in A.E.S.L (Caterpillar) Private Ltd. as a trainee service engineer and then moved to Sweden to complete Master's of Electrical Engineering with specialization in telecommunications from Blekinge Institute of Technology. He remained Research Assistant in Technical University of Dresden, Germany and worked on European Union OPERA project. He also served as a reviewer for IEEE conferences, including Globecom and DCOSS and for Springer's Wireless Personal communications Journal. Presently he is reading for his Split Ph.D. at City University, London, England and National University of Sciences and Technology, Pakistan. His main research interests include channel modeling, network optimization techniques, Topology control algorithms for wireless networks and graph theory.



Junaid Jameel Ahmad received his BSc. (Hons) degree in Computer Science from the University of the Punjab, Pakistan, in 2005 and his M.S. degree in Computer Software Engineering from National University of Sciences and Technology (NUST), Pakistan, in 2009. He holds the honors of securing the university wide 1st and 3rd positions in his B.Sc. and M.S. degrees respectively. During his M.S. thesis, he worked on a research project funded by Nokia Research Center, China. Currently, he is working as a Research Assistant at the Department of Computer and Information Sciences, Universität Konstanz, Germany. Before that, he worked as Research Assistant at the Wireless and Secure Networks (WiSNet) Research Lab of the School of Electrical Engineering & Computer Science (SEECS), National University of Sciences & Technology (NUST), Pakistan, and as Software Engineer at Q-Solutions Inc. His research interests include video compression. multimedia security, multimedia communications, energy efficiency in wireless sensor networks and wireless channel modeling.



Syed Ali Khayam received his B.E. degree in Computer Systems Engineering from National University of Sciences and Technology (NUST), Pakistan, in 1999 and his M.S. and Ph.D. degree in Electrical Engineering from Michigan State University in 2003 and 2006, respectively. In February 2007, he joined the School of Electrical Engineering & Computer Science (SEECS), National University of Sciences & Technology (NUST), Pakistan, as an assistant professor. AT NUST-SEECS, he directs the Wireless and Secure Networks (WiSNet) Research Lab. Khayam has received research awards from Nokia Research, Korean Research Foundation and Pakistan National ICT R&D Fund. He offers consultancy for some Silicon Valley based technology companies. He also worked at Communications Enabling Technologies as a Design Engineer from October 2000 to August 2001. His research interests include analysis and modeling of statistical phenomena in computer networks, network security, cross-layer design for wireless networks, and real-time multimedia communications. He has more than 50 publications and 4 pending patents in this area.



Veselin Rakocevic is a Senior Lecturer at City University London, UK. His research interests include cross-layer control in wireless networks, especially ad hoc and mesh networks. He holds a Ph.D. degree in Electronic Engineering from Queen Mary, University of London (2002), and a Dipl-Ing degree from the University of Belgrade, Serbia (1998). He published over 40 journal and conference papers, and served on a number of Technical Program Committees for IEEE conferences, including Globecom, ICC and MASS.



Muttukrishnan Rajarajan received his B.E. and Ph.D. degree from City University in 1994 and 1999 respectively. From 1999 he worked at City University as a Research Fellow. In August 2000 he left City University to join LogicaCMG as a Telecommunication consultant. He joined City as a lecturer in January 2002 where he is now a Senior Lecturer. In 2005 the online travel web services company GOQuo appointed Rajarajan as their Chief Technical Advisor. Dr. Rajarajan is a senior member of IEEE, a member of IET and a member of Technical Committees for ICOCN, PIERS 2008, eHealth 2008, Pervasive Health 2009 and has published more than 60 Journal and conference papers.

849