

Correlated Coding: Efficient Network Coding under Correlated Unreliable Wireless Links

Shuai Wang, Song Min Kim, Zhimeng Yin, and Tian He

Department of Computer Science and Engineering, University of Minnesota, U.S.
{shuaiw, ksong, tianhe}@cs.umn.edu, yinzhimeng@gmail.com

Abstract—Diversity-based protocols such as network coding and opportunistic routing have been proposed in recent years to exploit spatial diversity in wireless communication. By utilizing concurrent links, these protocols achieve significantly better performance than traditional unicast-based approaches. However, they explicitly or implicitly assume that wireless links are independent, which overestimates the true spatial diversity in reality. For the first time, this paper analyzes the impact of link correlation on network coding and introduces Correlated Coding, a link correlation-aware design that seeks to optimize the transmission efficiency by maximizing *necessary* coding opportunities. Correlated coding uses only one-hop information, which makes it work in a fully distributed manner and introduces minimal communication overhead. The highlight of our design is its broad applicability and effectiveness. We implement our design with four broadcast protocols and three unicast protocols, and evaluate them extensively with one 802.11 testbed and three 802.15.4 testbeds running TelosB, MICAZ, and GreenOrbs nodes. The experiment results show that (i) more coding opportunities do not lead to more transmission benefits, and (ii) compared to coding aware protocols, the number of coding operations is reduced while the transmission efficiency is improved.

I. INTRODUCTION

Wireless communication is essentially based on the broadcast medium with concurrent receptions. Extensive diversity based routing protocols, e.g., network coding [3] and opportunistic routing [6], have been proposed to exploit the broadcast diversity benefit. For example, under the lossy broadcast channel, several nodes may lose different packets. With network coding, multiple missed packets are encoded together and then broadcast in a single transmission, thus improving the transmission efficiency.

In prior work, researchers explicitly or implicitly assume that the wireless links are independent [6] when they exploit the broadcast diversity benefit. This assumption, however, contradicts the empirical evidence that wireless links are correlated. Recent work [7], [23] shows that transmissions over a wireless link within a short time interval are highly dependent, indicating the existence of temporal correlation. On the other hand, Srinivasan et al. [22] and Zhu et al. [29] highlight the existence of spatial correlation among multiple receivers because of the cross-network interference and correlated shadowing.

Link correlation can greatly impact the performance of (i) traditional network protocols, such as broadcast and multi-path

routing, or (ii) diversity-based protocols such as opportunistic routing and network coding. Wang et al. improve the performance of broadcast protocols by eliminating certain poorly correlated wireless links and forming better-correlated clusters [24]. Basalamah et al. reveal the impact of link correlation on opportunistic routing and build a link-diverse forwarder set which reduces the risk that all the forwarder fails to receive a packet [5]. For network coding, although Srinivasan et al. point out its performance varies dramatically under different link correlation environments [22], little work has been done on link correlation aware network coding protocol design.

Network coding consists of two operations: coding and broadcast. On the one hand, the coding operation prefers high spatial diversity (i.e., *low* link correlation). This is because when all receivers lose the same packets, network coding will not work better than the traditional routing protocols, as there is no coding opportunity [14] to exploit. On the other hand, the broadcast operation, in contrast, prefers low spatial diversity (i.e., *high* link correlation). This is because it takes fewer number of transmissions to deliver a coded packet to all receivers when the receptions of these nodes are correlated [11]. Clearly there exists a trade-off between the coding opportunity and broadcast effectiveness on the preference of link correlation. Ignoring this correlation in network coding protocol design may result in under-utilized diversity benefits [14], [22]. Even worse, because of the inaccurate link independent assumption, unnecessary coding operations exploited by coding aware routings [16], [21], [25], [28] may lead to extra energy consumption.

In this paper, we propose correlated coding, a link correlation aware network coding design, which estimates the coding opportunity and measures the transmission efficiency, for both broadcast and unicast applications. Guided by correlated coding, a coding operation is executed only when necessary and the diversity gain is maximized. In summary, our contributions are as follows:

- We experimentally show the existence of link correlation, i.e., the reception results of broadcasting packets at multiple receivers are not independent. We introduce link correlation to analyze the efficiency of network coding. We determine whether a network coding operation is necessary or not based on link correlation, avoiding unnecessary operations.
- We propose correlated coding, which captures the expected number of transmissions with network coding under the effect

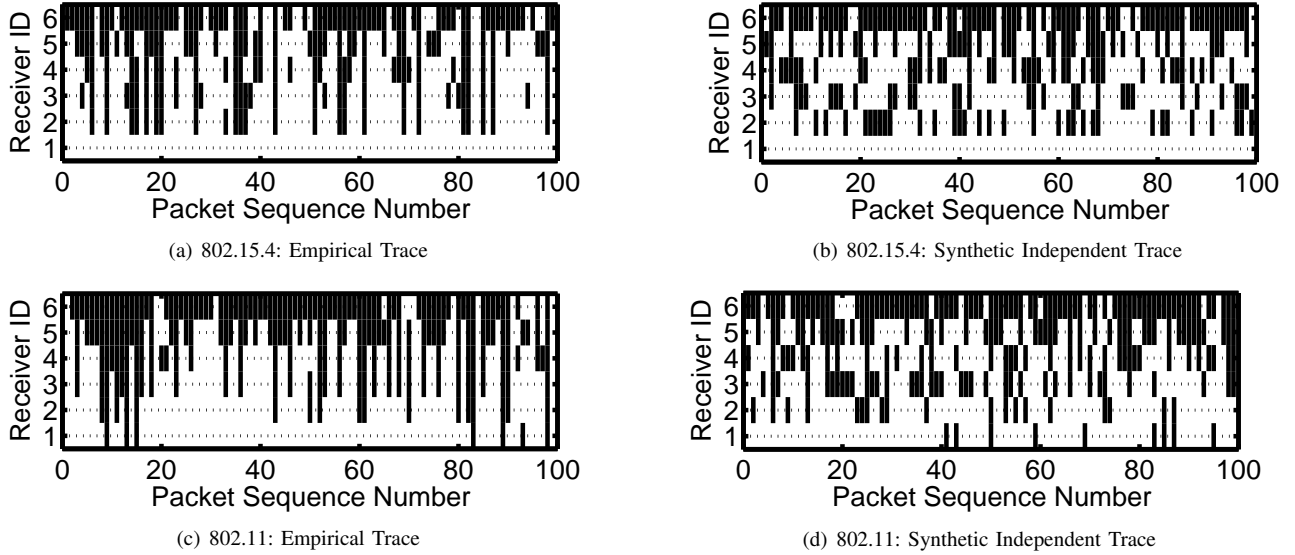


Fig. 1. Packet receptions at six receivers when a single transmitter broadcasts 100 packets; (a)&(c) empirical trace from the 802.15.4 and 802.11 testbeds. (b)&(d) the corresponding synthetic trace with the independent links under the same packet reception ratio. Packet losses are marked by black bends. Compared with synthetic independent trace, empirical trace has more correlated losses.

of link correlation. Our design can be intergraded with both unicast and broadcast protocols and help them minimize the transmission cost with the optimized coding opportunity. As far as we know, this is the first work that explores link correlation both mathematically and experimentally in network coding.

- We experimentally verify the impact of correlated coding on three unicast and three broadcast protocols with three 802.15.4 testbeds running TelosB, MICAz, and GreenOrbs nodes, and one 802.11 testbed. Both testbed and simulation results show that our design consistently enhances the performance of these protocols. Experimental results show that the transmission efficiency is improved by 30% ~ 50%.

The rest of the paper is structured as follows. Section II presents the diversity of network coding. Section III introduces the main design followed by its applications in Section IV. Evaluation results from testbed experiments and simulations are shown in Sections V and VI. Section VII reviews related work. Finally, Section VIII concludes the paper.

II. DIVERSITY OF NETWORK CODING

In this section, we first demonstrate the existence of link correlation. Then, we present the benefit nature behind network coding. Finally, we present the impact of link correlation on network coding.

A. Existence of Link Correlation

To verify the existence of link correlation, we conduct two small experiments on 802.15.4 and 802.11 testbeds separately. In both experiments, seven nodes are deployed to form a star topology. The central node is selected as the sender and the other six nodes are receivers. The sender broadcasts 100 packets to the receivers. on the 802.15.4 testbed, we use channel 16 which overlaps with the Wi-Fi signals. On the

802.11 testbed, channel 6 is utilized. After the broadcast task, a sink node collects the reception results of the receivers.

Figure 1(a) and 1(c) show the packet receptions at the six receivers from empirical measurements. The lost packet is marked by black bands and long vertical black bands indicate that packets are lost at multiple receivers. As a comparison, Figure 1(b) and 1(d) plot the independent synthetically generated traces with the same PRR where few multiple simultaneous receptions and losses are observed. The comparison indicates that the packet receptions at multiple receivers in Figure 1(a) are correlated. The experiment result reaffirms the empirical study reported in recent work [22], [24], [29].

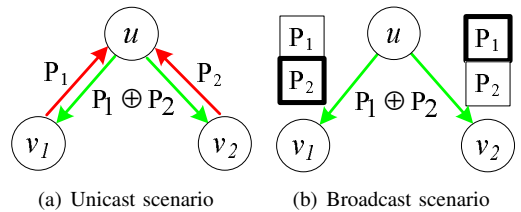


Fig. 2. Examples of network coding occurs at (a) unicast scenario and (b) broadcast scenario. From both scenarios, network coding gets benefit because it broadcasts a coded packet that involved with multiple original packets.

B. The Nature Behind Network Coding

Network coding has great potential to improve the performance of both unicast and broadcast applications by allowing intermediate nodes to encode multiple packets together before forwarding. Figure 2 shows how network coding benefits both unicast and broadcast. In Figure 2(a), after node v_1 and v_2 send their packets to the relay node u , instead of sending packets p_1 and p_2 separately, node u broadcasts a coded packet $p_1 \oplus p_2$ with one transmission. In the broadcast scenario, a packet reception report is shown in Figure 2(b), in which a block

with a thick borderline means a received packet, and a block with a thin borderline means a lost one. The receiver v_1 and v_2 lose packet p_1 and p_2 respectively. In traditional designs, to make sure that both receivers get the two broadcast packets, the source node u needs to send packets p_1 and p_2 using two transmissions. With the help of network coding, node u broadcasts an XORed packet $p_1 \oplus p_2$ using one transmission, thus saving one transmissions. From the examples in Figure 2, we see that the nature behind network coding is that instead of sending each target packet one by one, the forwarder encodes all of them together and broadcasts them with a coded packet using one transmission.

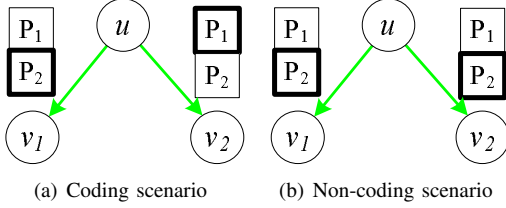


Fig. 3. Impact of link correlation on network coding. In scenario (a), the links are negative correlated and coding opportunity occurs. In scenario (b), the links are positive correlated and there are no coding opportunities. The expected number of transmissions to deliver the coded packet to the two receivers is (a) 3, and (b) 2.

C. Impact of Link Correlation on Network Coding

From the above discussion, we learn that network coding obtains benefit following two steps. First, a node finds the coding opportunity based on the reception of its neighbors. Second, the node encodes several packets together and broadcasts with one transmission. Correspondingly, the impact of link correlation on network coding comes from two aspects. On the one hand, the coding opportunity highly depends on the diversity of each link's receptions, i.e., link correlation. For example, in Figure 3(b), both receivers lose the same packet p_1 , i.e., the link uv_1 and uv_2 are positive correlated. There are no coding opportunities in this scenario. Compared with the coding scenario in Figure 3(a), we find that we have more coding opportunities when the links are *lower* correlated.

On the other hand, network coding could be effective if and only if the encoded packet is broadcast and received by all receivers that are involved in the coded packet. We theoretically analyze the expected number of transmissions to cover all potential receivers and demonstrate that the source node needs *fewer* transmissions when the links are *higher* correlated. We use $p(e_i) \in (0, 1]$ to denote the probability that a source node can directly deliver a packet via link e_i . Let $p(e_1)$ and $p(e_2)$ denote the link qualities for the two receivers respectively. Corresponding packet loss probabilities are denoted as $p(\bar{e}_1) = 1 - p(e_1)$ and $p(\bar{e}_2) = 1 - p(e_2)$. Let $p(\bar{e}_1 \cap \bar{e}_2)$ denote the probability that a coded packet from the source node is not received by either receiver. Then the expected number of transmissions $E[\varepsilon]$ to deliver the coded packet to the two receivers can be calculated as

$$E[\varepsilon] = \sum_{i=1}^2 \frac{1}{p(e_i)} - \frac{1}{1 - p(\bar{e}_1 \cap \bar{e}_2)} \quad (1)$$

The proof of Eq.(1) is in the Appendix. We note that $p(\bar{e}_1 \cap \bar{e}_2)$ obtains its maximum value when links e_1 and e_2 are perfectly positive correlated while it gets a minimum value when links e_1 and e_2 are perfectly negative correlated. Let us revisit the two scenarios in Figure 3(a) and 3(b). For the scenario in Figure 3(a), the expected number of transmissions $E[\varepsilon]$ is 3 based on Eq.(1), given that $p(e_1) = p(e_2) = 0.5$ and $p(\bar{e}_1 \cap \bar{e}_2) = 0$ since uv_1 and uv_2 are perfectly negative correlated. In Figure 3(b), however, since uv_1 and uv_2 are perfectly positive correlated, it is not difficult to get that $p(\bar{e}_1 \cap \bar{e}_2)$ is 0.5. As a result, $E[\varepsilon]$ is 2 which is less than the cluster in Figure 3(a). This suggests that in the broadcast procedure, positive correlation is preferred since all the receivers lose the same packets and few retransmissions are needed. Therefore, there exists a trade-off between the number of coding opportunity and the broadcast efficiency. Only considering coding opportunity without taking link correlation into account may not effectively utilize the broadcast diversity, or even worse – it may lead to a higher transmission cost when an undesired coding operation is executed.

III. MAIN DESIGN

From the previous section, we know that there exists a trade-off between the broadcast efficiency and coding opportunity. In this section, we propose Correlated Coding to optimize the transmission efficiency of network coding with link correlation. We aim to build a general model that can be used for both unicast and broadcast protocols. The unnecessary coding operation should be avoided by the design.

In the following section, we theoretically analyze the expected transmission count for a sender to reliably broadcast a coded packet to all the receivers in Section III-A. Section III-B proposes a heuristic to reduce the computational cost, followed by overhead and accuracy discussions in Section III-C. In Section III-D, we provide a model to estimate the transmission benefit from coding opportunity. Finally, Section III-E introduces two correlated coding metrics which quantify the benefit of coding opportunities in reducing transmissions under the consideration of link correlation.

A. Expected Transmission Count

We first examine the number of transmissions for node u to reliably broadcast the coded packet to all the potential receivers such that they can extract the original packet from the coded one. We denote ε as the number of transmissions needed by sender u to reliably broadcast a coded packet involved with K original packets to the K potential receivers – $V(u) = \{v_1, v_2, \dots, v_K\}$.

We assume a widely used ARQ model for the reliable delivery. In ARQ, if a sender does not receive an ACK before the timeout, it retransmits the packet until it receives an ACK. With ARQ, for each link e with link quality $p(e)$, the expected number of transmissions needed to successfully send a packet over a single link e is $\frac{1}{p(e)}$. Although link quality of wireless links changes over time, it can be measured and refreshed through normal data traffic or periodic beacons. Let the link

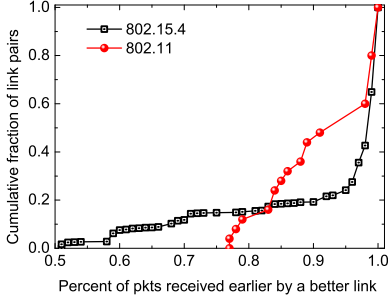


Fig. 4. Statistics of receiving probability. The node with a better link receives about 92.7% (802.15.4 testbed) and 92.1% (802.11 testbed) of the packets earlier or at the same time than the node with a worse link.

quality between u and the potential receiver v_j be $p(e_j)$, $j = 1, 2, \dots, K$. The corresponding packet loss probability is denoted $p(\bar{e}_j) = 1 - p(e_j)$. Without loss of generality, we assume $p(e_1) \geq p(e_2) \geq p(e_3) \geq \dots \geq p(e_K)$. The expectation of ε can be calculated as

$$\begin{aligned} E[\varepsilon] &= \sum_{i=1}^K \frac{1}{p(e_i)} - \sum_{1 \leq i < j \leq K} \frac{1}{1 - p(\bar{e}_i \cap \bar{e}_j)} \\ &+ \sum_{1 \leq i < j < l \leq K} \frac{1}{1 - p(\bar{e}_i \cap \bar{e}_j \cap \bar{e}_l)} + \dots \\ &+ (-1)^{K-1} \frac{1}{1 - p(\bar{e}_1 \cap \bar{e}_2 \cap \dots \cap \bar{e}_K)}. \end{aligned} \quad (2)$$

The proof of Eq.(2) is in the Appendix. To get ε with K potential receivers, we need to compute $\binom{K}{1} + \binom{K}{2} + \dots + \binom{K}{K} = 2^K - 1$ polynomial terms where $\binom{a}{b}$ is the number of selecting b items from a ones. In network coding, although the number of packets that can be encoded together is relatively small (and thus the number of the potential receivers, i.e., K is small), the exponential growth of complexity with K shall be avoided when possible. In the following section, we present a novel approach to simplify the calculation.

B. Transmission Count Approximation

Due to the high cost of computing ε , we seek a more efficient algorithm to approximate ε with less computational complexity. In wireless networks, the nodes with a higher link quality usually receive the broadcast packet before (or at the same time) those with a lower link quality. To confirm this observation, we deploy 30 MICAz near a sender u to form a star topology. The source node keeps broadcasting packets in every 0.2s until all the receivers receive 100 packets. In each packet we include sequence number and time stamp. After collecting the packet reception trace, we compare the reception between each link pair (there are $\binom{30}{2} = 435$ such pairs). Figure 4 shows that the node with a better link from u receives more than 90% of the packets earlier (or at the same time) than the node with a worse link from u on both 802.15.4 and 802.11 testbed. In other words, statistically, the node with a better link needs fewer transmissions for a specified packet. Based on this observation, we propose an approximate algorithm to estimate ε , denoted as $\hat{\varepsilon}$.

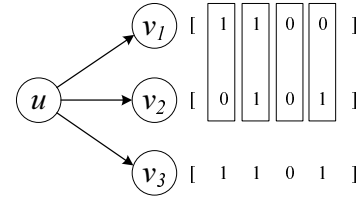


Fig. 5. Example of Calculating u 's $p(S_i(u))$ for $\{v_1, v_2\}$

Lemma 1: (Approximation of ε) Assuming nodes with higher link quality receive the broadcast packet earlier than those with lower link quality, u 's expected transmission count with K potential receivers is approximated by

$$\hat{\varepsilon} = \sum_{i=1}^K \frac{1}{p(e_i)} - \sum_{i=2}^K \frac{1}{p(e_i)} \cdot \frac{p(S_i(u))}{p(S_{i-1}(u))} \quad (3)$$

where $S_i(u)$ is a subset of i nodes with the highest link quality among u 's neighbors, and $p(S_i(u))$ is the probability that all the i nodes in $S_i(u)$ successfully receive a packet. The proof of Lemma 1 is omitted due to the space constrain.

Implementation: To calculate $\hat{\varepsilon}$, we need to get each receiver's link quality $p(e)$ and $p(S_i(u))$ for $\forall i = 2, \dots, K$. Suppose each receiver maintains a packet reception report (e.g., [1100]) recording the reception status of a fixed number (e.g., 4) of most recent packets. In our design, we use normal data traffic to update the reception report. With the reception report, the link quality is given simply by the number of 1s in the reception report divided by the reception report length. To calculate $p(S_i(u))$, every node exchanges its reception report with its neighbors. Let the reception report length be W , we have

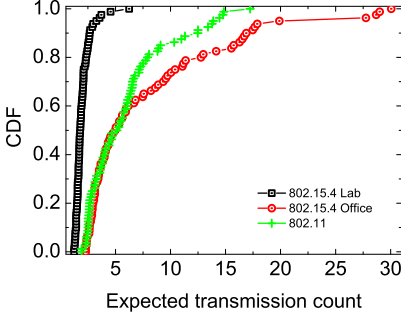
$$p(S_i(u)) = \frac{1}{W} \sum_{j=1}^W B_{v_1}(j) \& B_{v_2}(j) \& \dots \& B_{v_i}(j), \quad (4)$$

where $B_{v_i}(j)$ is a bit representing the receiver v_i 's reception status of the j th packet. $B_{v_i}(j) = 1$ represents node v_i receives the packet, otherwise $B_{v_i}(j) = 0$. For example, in Figure 5, node u has 3 neighbors. We calculate u 's $p(S_i(u))$ for $\{v_1, v_2\}$. Suppose the report of node v_1 is [1100], which indicates that v_1 receives the 1st and 2nd packets and loses the 3rd and 4th packets. When node u receives the reception reports from the receivers, it uses Eq.(4) to calculate $p(S_i(u))$, i.e., $p(S_2(u)) = (1\&0 + 1\&1 + 0\&0 + 0\&1)/4 = 25\%$.

C. Estimation Overhead and Accuracy

In real-world environments, both link quality and link correlation change over time. A nature question is that how much overhead is required to maintain these values fresh under practical scenarios, especially a link dynamic scenario? To answer it, we conduct a set of experiments on both 802.11 and 802.15.4 testbeds in a period of 50 minutes.

The main overhead comes from two sources. The first comes from link status measurement, which is accomplished using reception report from normal traffic data. This part overhead is thus negligible. Second, we need to exchange packet reception reports (i.e., [1100]) among one-hop neighbors in order to

Fig. 6. CDF of $\hat{\varepsilon}$

calculate $\hat{\varepsilon}$. The exchange of reception report has already been required by network coding schemes, e.g., COPE [14]. The difference is that we use the reception report not only for the capture of coding opportunity but also for the calculation of link correlation. Besides, the binary report is small and much less frequently exchanged.

In the experiments, the source node keeps broadcasting packets to six receivers in every 0.3s. The reception report is sent in every 30s to fresh link quality and correlation values. We run the 802.15.4 experiments on a lab and an open office environment. The 802.11 experiments run on a university building.¹ Figure 6 plots the CDF of $\hat{\varepsilon}$ on the three scenarios. The corresponding estimated number of received packets and the ground truth are shown in Figure 7. Here, the number of received packets (during a time period of 30s) is the number of sent packets (i.e., 100) over $\hat{\varepsilon}$. From the figure, we can find that the estimated values closely follows the ground truth. It indicates that $\hat{\varepsilon}$ is maintained accurately over time. In the above experiment settings (i.e., the reception report is sent in very 30s), the cost of exchanging reception reports occupies a tiny fraction (1%) of the total energy cost. Thus, the estimation of ε will not bring much overhead to the existing protocols.

D. Coding Opportunities Estimation

The coding opportunity in a sender u is crucially dependent on the packet reception patterns in its receivers. When node u broadcasts a coded packet to all its receivers, we need to make sure that all the receivers have already gathered enough packets to decode the original one. We specify the network coding rule as follows:

Definition 1: (Network Coding Rule) Consider a sender u transmitting an encoded packet $p' = \oplus(p_1, p_2, \dots, p_K)$. In order to decode p' , each receiver should have already received $K - 1$ packets among $p_i, i = 1, 2, \dots, K$.

Based on the definition of network coding rule, we estimate the benefit of network coding through the coding opportunity. The formal definition of coding opportunity is given by:

Definition 2: (Coding Opportunity) For packets buffered in an output queue, if there exist a number of packets that

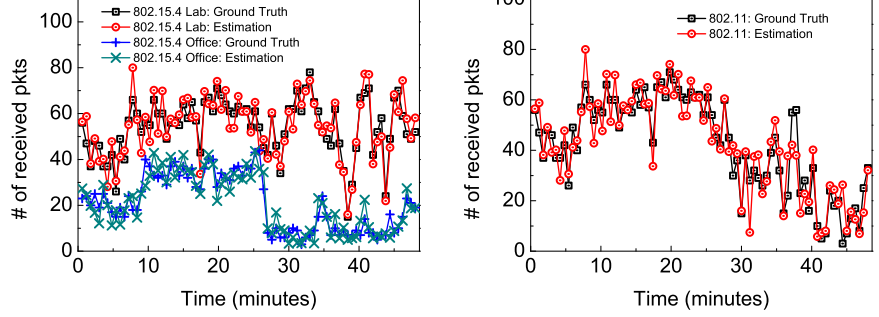


Fig. 7. Received pkts vs. Estimation

satisfy the network coding rule and thus can be encoded together, we call this condition a coding opportunity.

Let the set of nodes involved in node u 's coding operation be $V(u) = \{v_1, v_2, \dots, v_K\}$, where $K = |V(u)|$. Assume the number of coding opportunities with k_i original packets involved in an encoded packet is $\phi(k_i), 2 \leq k_i \leq K$. Assume the packet length in node u 's output is l , with the help of network coding, the total number of packets that node u needs to transmit changes to α , which is given by

$$\alpha = l - \sum_{i=2}^K (k_i - 1)\phi(k_i), \quad (5)$$

where $\sum_{i=2}^K (k_i - 1)\phi(k_i)$ is the number of packets reduced by network coding. From Eq.(5), we find that the total number of packets can be greatly reduced when there are many coding opportunities.

E. Correlated Coding Metrics

In this section, we introduce two metrics, named broadcast correlated coding metric and unicast correlated coding metric, which can be used in broadcast and unicast protocols. First, we introduce the correlated coding metrics for broadcast which is defined as follows:

Definition 3: (Broadcast Correlated Coding Metric) the number of transmissions needed by sender u to reliably broadcast a packet (including both original and coded packet) to all the packet's receivers - $V(u_i) = \{v_1, v_2, \dots, v_K\}$, divided by the number of the potential receivers.

The broadcast correlated coding metric, denoted as αETX , is calculated as follows:

$$\alpha ETX = \frac{\alpha}{l} \cdot \frac{\hat{\varepsilon}}{K}, \quad (6)$$

The calculation of αETX in Eq.(6) involves two terms: (i) $\frac{\alpha}{l}$ is the percentage of packets left in the queue after network coding, and (ii) $\frac{\hat{\varepsilon}}{K}$ measures the broadcast efficiency. αETX offers a good estimate for the expected transmission count for a successful packet delivery with network coding. It captures a basic characteristic of correlated links as well as the coding opportunity. In nutshell, αETX suggests that selecting a proper forwarder (coding node) should consider receivers with high link correlations and good coding opportunities.

¹The detail information of the testbed scenarios is described in Section V.

Protocol Name	Reference	Network Info.	Hello Msg	Broadcast Msg	Routing strategy	Category
Spanning Tree (ST)	[13]	One-hop	ID	Msg only	Tree-based	Broadcast
Forwarding Node Cluster (FNC)	[27]	Local	ID	Covered set	Cluster-based	Broadcast
Partial Dominating Pruning (PDP)	[18]	Two-hop	One-hop	Msg + Covered set	Pruning-based	Broadcast
ZigBee	[10]	One-hop	ID	Msg only	Cluster tree	Unicast
OLSR	[8]	Two-hop	One-hop	Msg + Covered set	Multi-point relay	Unicast
ETX	[9]	One-hop	One-hop	Msg only	Minimal cost path	Unicast
Flexible Multicast Service (FMS)	[1]	One-hop	One-hop	Msg only	Minimal cost	Multicast

TABLE I
SEVEN STATE-OF-THE-ART PROTOCOLS SUPPORTED BY CORRELATED CODING

The optimization tradeoff between the broadcast efficiency and coding benefit is decided by the final value of αETX .

Definition 4: (Unicast Correlated Coding Metric) Given a sender u_i and its potential receiver set $V(u_i) = \{v_1, v_2, \dots, v_K\}$. The unicast correlated coding metric μETX is the forwarding cost of the link $e(u_i, v_j)$, $j = 1, 2, \dots, K$.

$$\mu ETX = K \times \alpha ETX_{V(u_i)} - (K - 1) \times \alpha ETX_{V(u_i) - \{v_j\}}. \quad (7)$$

The calculation of μETX in Eq.(7) involves only one item, i.e., αETX , which can be calculated using Eq.(6). We may use unicast correlated coding metric to build the minimal cost path. Besides, in wireless broadcast, a node v_j could hear packets from multiple upstream forwarders. Node v_j needs to associate itself with the most efficient upstream forwarder u_i based on the unicast correlated coding metric of $u_i \rightarrow v_j$.

IV. APPLICATIONS

The correlated coding metric is a general design which can help a wide range of routing protocols to efficiently exploit the coding benefits. Thus far, we have successfully implemented seven classic algorithms and embedded correlated coding with them. The basic information of these algorithms is shown in Table I.

A. 802.15.4 Networks

1) **Broadcast:** We classify the existing deterministic broadcast algorithms into three categories: i.e., (i) tree-based [13], (ii) cluster-based [4], and (iii) pruning-based [18]. We briefly introduce how to embed correlated coding into these three types of algorithms, thus bring them transmission gain from link correlation and network coding. In a tree based algorithm [13], instead of finding the nodes with maximum leaves, we integrate correlated coding by choosing the nodes with $\min(\alpha ETX)$ as the tree nodes. To combine the cluster based broadcast [4] with correlated coding, the algorithm first selects nodes with $\min(\alpha ETX)$ to form a maximum independent set (MIS). Then, it finds connectors to link the nodes in MIS. In the pruning based scheme [18], each forwarder adds its one-hop neighbors with $\min(\alpha ETX)$ to the forwarder set to cover its two-hop neighbors. In all three algorithms, if a covered node receives a message from different nodes in the tree, MIS or forwarder set, the node selects the node with $\min(\mu ETX)$ as its forwarder.

2) **Unicast:** Unicast routing protocols can be divided into two categories – (i) backbone based, and (ii) flat protocols. In the first category, a backbone will be built using the tree, cluster or pruning based method. For example, in ZigBee [10], a cluster tree is built. In OLSR [8], multi-point relays are selected as backbone nodes. In backbone based unicast routing protocols, if any node has a packet to transmit, it will first send the packet to its nearby backbone node. Then, the packet goes through the backbone to the destination. The idea of integrating correlated coding to this kind of routing protocols is similar to the deterministic broadcast protocols, i.e., we select the nodes with $\min(\alpha ETX)$ to form the cluster tree or multi-point relay set. For the flat protocols, a classic example is ETX [9], which selects a routing path with $\min(\sum \frac{1}{P(e)})$. With correlated coding, we select the path with $\min(\sum \mu ETX)$. Note that when there is only one data flow or multiple data flows do not mixed with each other, correlated coding degenerates into ETX since $K = 1$ and $\alpha = l$.

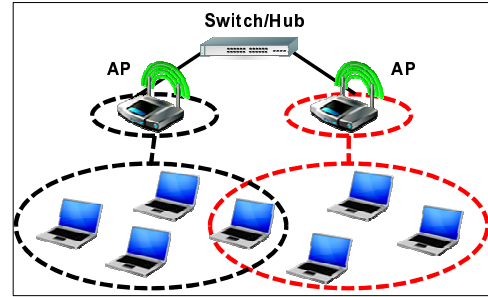


Fig. 8. Collaborative FMS

B. 802.11 Networks

1) **Flexible Multicast Service:** In wireless LAN, FMS is an efficient way to deliver the same contents to a large number of receivers. Notice that in FMS multiple APs may share the same upstream service provider. We thus propose a novel communication paradigm called *collaborative FMS*. As shown in Figure 8, multiple APs are connected via wires and form an infrastructure, while user devices, i.e., the laptops, are one (wireless) hop away from the infrastructure. The collaborative FMS design utilizes the infrastructure for sharing the packet through wires and *collaborates* to cover every user. With correlated coding, we can further improve the performance of collaborative FMS. In detail, we calculate μETX – the

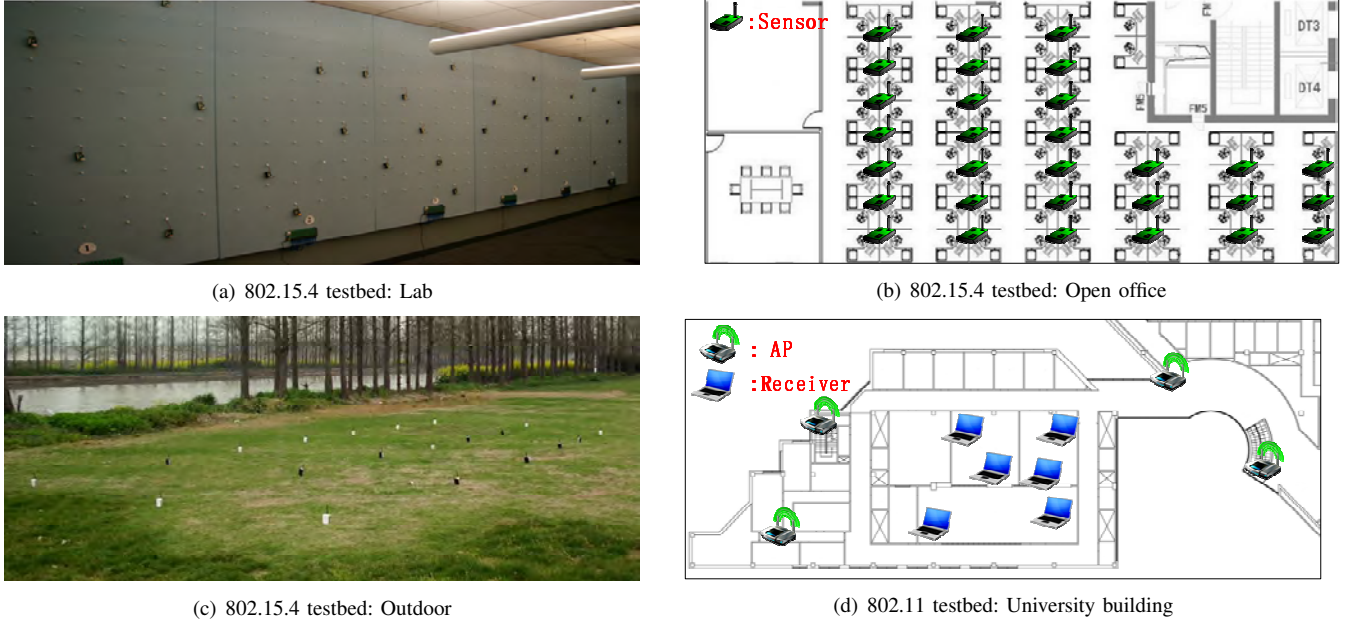


Fig. 9. Testbed environments

Platform	Location	Environment	Physical Size	No. of Nodes	No. of APs	Degree	Channel	Power
MICAz	UMN	Lab	$8m \times 2.5m$	30	54	7 ~ 23	Ch16	-25dBm
TelosB	SIAT	Open office	$18m \times 13m$	30	8	6 ~ 21	Ch16, Ch26	-25dBm
GreenOrbs	TRIMPS	Outdoor	$15m \times 5m$	20	0	4 ~ 13	Ch16	-25, -19.2dBm
802.11g	UMN	University building	$73m \times 30m$	6	54+4	6	Ch3, Ch6	15dBm, 20dBm

TABLE II
TESTBED SETTINGS AND TOPOLOGY PROPERTIES

transmission cost for an AP to cover one receiver under the receiving status of the rest receivers. When one receiver can be covered by multiple APs, as shown in Figure 8, the receiver is assigned to the AP with minimal μ ETX.

V. TESTBED IMPLEMENTATION

The link status as well as the packet reception pattern in different environments varies significantly, which causes the performance of a protocol changes dramatically. In this section, we report the experiment results of seven state-of-the-art protocols supported by correlated coding metrics on one 802.11 platform located in a university department building, and three 802.15.4 platforms in a lab, an open office, and an outdoor environment. The experiment scenarios are shown in Figure 9, and the testbed settings and topology properties are shown in Table II.

A. Experiment Setup

1) *802.15.4 testbed*: We deploy three 802.15.4 testbeds. The first one is located in a lab environment where 30 MICAz nodes randomly on an $8m \times 2.5m$ wall, see Figure 9(a). The second testbed has 30 TelosB nodes which are deployed in an $18m \times 13m$ open office environment, as shown in Figure 9(b). On the third testbed, 20 GreenOrbs nodes are deployed on an open space along a river, as shown in Figure 9(c).

On all the three testbeds, the default power is -25dBm and the default channel is 26. The transmission rate is 5 packets/sec. In the beginning of the experiment, a control node is used to remotely configure radio parameters, i.e., transmission power and channel. Based on these radio settings, each node broadcasts 100 packets in turn. Each packet is identified by a sequence number. All the received packets are recorded in nodes' flash memory. When all the nodes finish broadcasting 100 packets, they send their packet reception information to a sink node which is connected to a PC. We thus obtain the information required by correlated coding, i.e., the packet receiving patterns, based on which we can calculate the broadcast or unicast correlated coding metric. Then, the corresponding nodes in the testbed are selected as forwarders for unicast or broadcast according to the application description in Section IV. In the broadcast application, the forwarders keep on broadcasting packets until all their covered nodes receive 100 packets. In the unicast application, two pairs of data flows are picked up and each source node reliably sends 100 packets to the destination.

2) *802.11 testbed*: This testbed is located on the 4th floor of the Computer Science Department building in University of Minnesota, as shown in Figure 9(d). From the figure, we can see that four APs are deployed at the four corners of the floor, while six receivers are placed in three different

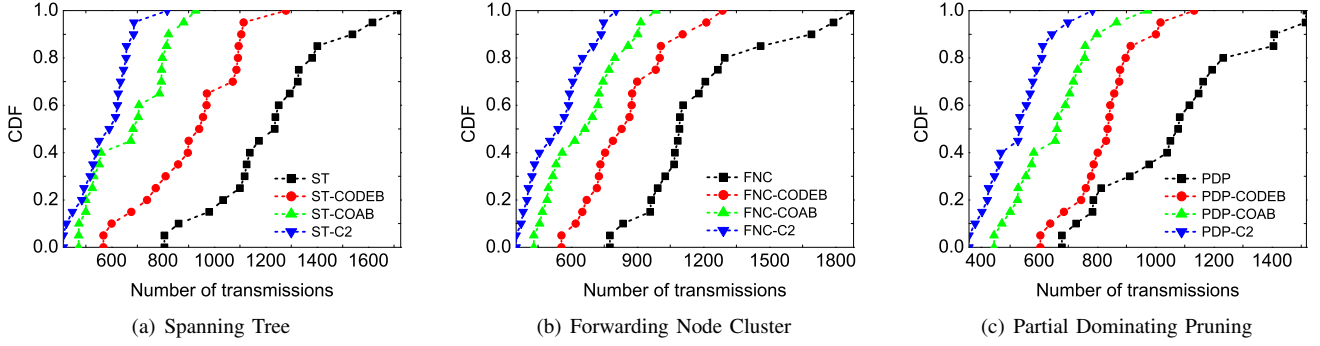


Fig. 10. Main performance results: broadcast protocols in 802.15.4

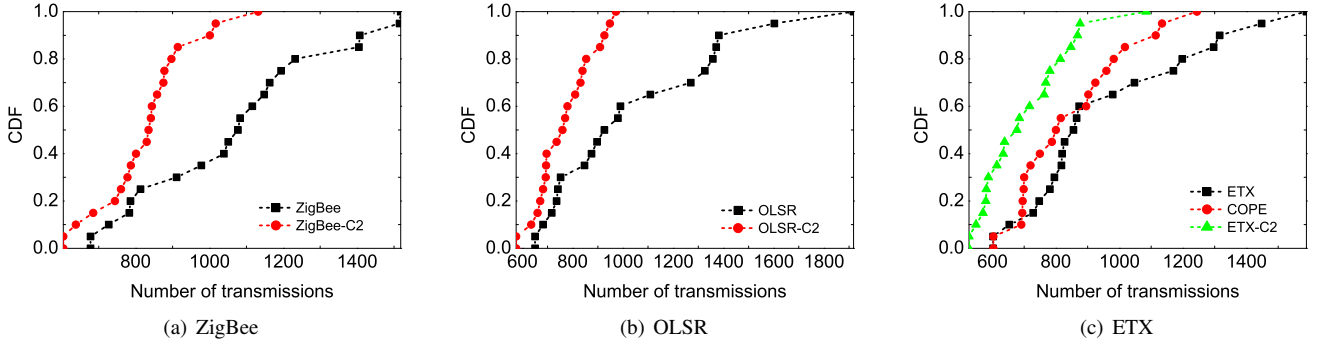


Fig. 11. Main performance results: unicast protocols in 802.15.4

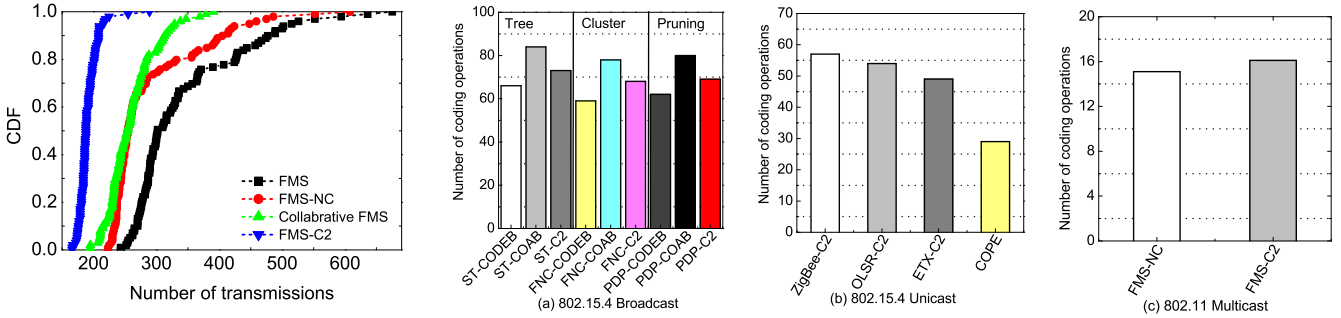


Fig. 12. Main performance results: FMS in 802.11g

Fig. 13. Main performance results in number of coding operations

rooms, separated by concrete walls. We use laptops with the `Lorcon2` packet injection library [2] to generate the traffic. The standard we used is 802.11g. During the experiment, four APs broadcast 10^5 packets in turn and the receivers record the packet reception information. Similar to the experiments on the 802.15.4 testbeds, we obtain the link correlation information for correlated coding. On the 802.11 testbed, the default power is 20dBm and the default channel is 6.

B. Compared Schemes and Performance Metrics

We compare our correlated coding design with COPE [14], CODEB [17], and COAB [25]. Among them, CODEB applies network coding over a wireless backbone built with a pruning method and COAB is a coding aware routing protocol. We use two metrics for the following performance evaluation.

- Number of transmissions** – the number of transmissions

needed by a broadcast scheme to reliably broadcast 100 packets to the whole network.

- Number of coding operations** – the number of times that network coding occurs during the simulation. It is used to measure coding opportunities.

C. Main Performance Results

The experimental results of the seven protocols are shown in Figures 10, 11, 12 and 13. Figure 10 plots the CDF of the transmissions with correlated coding. The figure shows that correlated coding significantly improves the transmission efficiency under all the three different broadcast strategies. For example, for the tree based broadcast algorithm – Spanning Tree (ST), the nodes need 1238 transmissions, on average, to guarantee that all nodes in the network receive 100 packets, while the number is 528 when correlated coding is combined

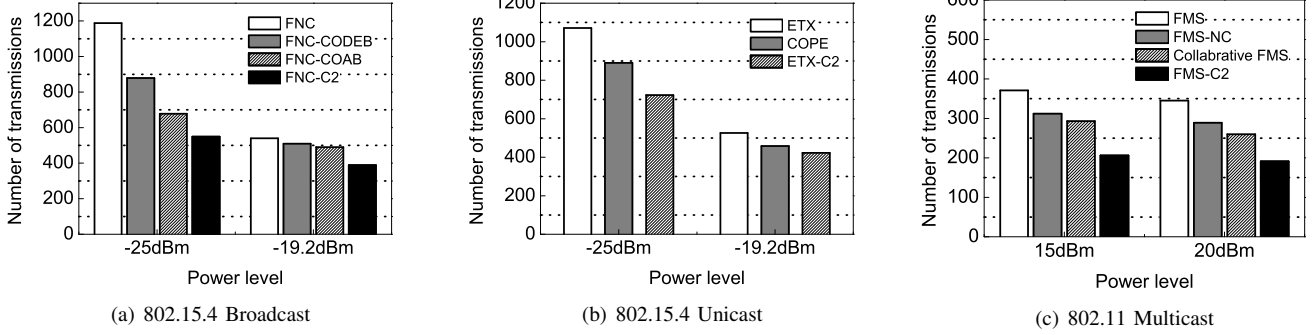


Fig. 14. Impact of power levels

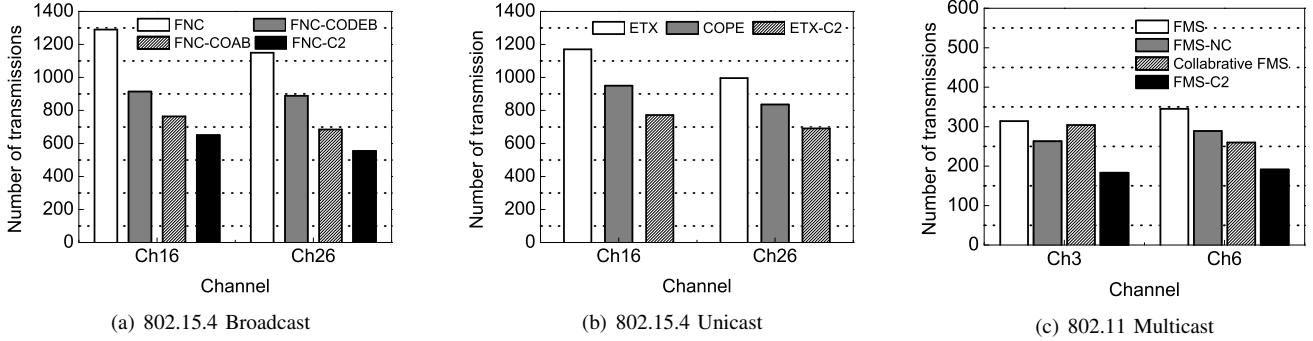


Fig. 15. Impact of channels

with ST (i.e., ST-C2), achieving a reduction of 57%. The average transmission of ST-CODEB and ST-COAB is 940 and 675, respectively. On average, our design reduces transmissions of ST-CODEB and ST-COAB by 44% and 22%.

In Figure 11, we find that correlated coding improves the performance of Zigbee, OLSR, and ETX significantly. The transmission gain is about 35% on average. We note that the benefit of correlated coding in unicast applications is less than broadcast applications since the coding opportunity in unicast is less than it in broadcast. Similar results are also found on the 802.11 testbed. In Figure 12, compared with FMS, FMS-NC, and collaborative FMS, correlated coding, i.e., FMS-C2, saves 45%, 34% and 26% transmissions respectively.

Figure 13 plots the number of coding operations in broadcast, unicast and FMS applications. From figures, we find that the coding aware routing COAB exploits the most coding opportunities while CODEB and COPE can not fully exploit the coding opportunities. Although the coding operation of correlated coding is less than COAB, we find that the performance of correlated coding is better than COAB. This is because correlated coding only encodes a packet when it can optimize the transmission gain, and thus avoid those unnecessary coding operations.

Although we collect results for all seven protocols, space constraints do not allow presenting all of them here. Therefore, we choose one representative algorithm for each application, i.e., Forwarder Node Cluster [27] (FNC for short) for broadcast application, ETX [20] for unicast application, and FMS for the Wi-Fi multicast application. For the rest of the 802.15.4

experiments, we assign correlated coding upon FNC and ETX, and compare them with COPE, CODEB, and COAB. For the 802.11 experiments, we compare correlated coding with FMS, FMS-NC, and collaborative FMS.

D. Impact of Power Level

- **802.15.4 testbed:** The power level for transmission is set from -25dBm to -19.2dBm to form a multi-hop network. Figure 14(a) shows the transmissions of FNC with CODEB, COAB and C2, and Figure 14(b) shows the transmissions of ETX with COPE and C2 under different power levels. We find that correlated coding greatly reduces transmissions for both broadcast and unicast applications. Under power level -25dBm, the transmission count for FNC is 1188, while it is 549 with correlated coding, providing a reduction of 54%. Under power level -19.2dBm, fewer transmissions are needed because a higher power level leads to better link quality. In this case, correlated coding still reduces transmissions by 28%.

- **802.11 testbed:** We examine the performance of correlated coding under power level 15dBm and 20dBm. From Figure 14(c), the number of transmission of C2 under 15dBm is 205 while it's slight lower, i.e., 192 under 20dbm. In both cases, C2 saves collaborative FMS about 30% transmissions.

E. Impact of Different Channels

- **802.15.4 testbed:** In this experiment, we explore the impact of channels on correlated coding. We use two different channels – channels 16 and 26. Note that channel 16 overlaps with a co-habiting access point's 802.11 channel and that channel 26 is free of Wi-Fi interference. The power level for transmission

is set to -25dBm . Figure 15 shows the energy consumption in broadcast and unicast protocols under different channels. The gains of correlated coding under the broadcast and unicast application are 52% and 31% under channel 26, and they are 50% and 34% using channel 16. In addition, we find that on both unicast and broadcast applications, all the algorithms need more transmissions to finish the same task in channel 16. This is because the interference introduced by the overlapped channel causes more packet losses.

- **802.11 testbed:** In this experiment, we examine the performance of C2 under channel 3 and 6. The transmission difference between these two channels is not as obvious as the observation on the 802.15.4 testbeds. That's because both channel 3 and 6 will be impacted by the nearby APs which usually use channel 1, 6 and 11.

VI. SIMULATION

In this section, we provide extensive simulation results about the performance of correlated coding for large-scale networks under different system settings.

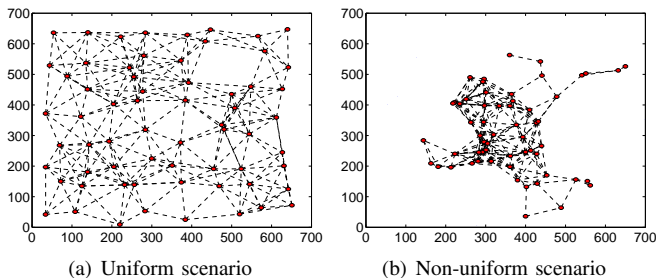


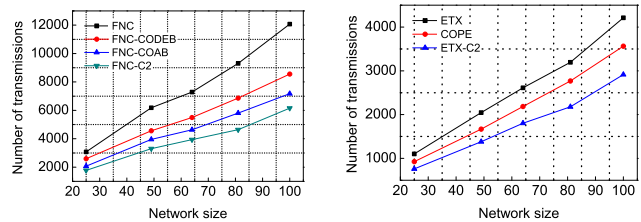
Fig. 16. Simulation topology for 802.15.4 networks

A. Simulation Setup

Given a scenario, we generate correlated reception reports for all the sender-receiver pairs by modifying the sampling algorithm for Bernoulli random variables in [19]. For a particular packet, the reception status at receivers could be either 0 or 1. We assume that the reception reports at different nodes are of the same length.

- **802.15.4 testbed:** We generate network topologies with different network sizes and densities. By default the network size is 64, and the field size is $700\text{m} \times 700\text{m}$ with a communication range of 160m, as shown in Figure 16. In the broadcast application, a random selected source node broadcasts 100 packets, and we record the number of transmissions required to finish broadcasting the 100 packets. In the unicast application, similar to the testbed experiment, we randomly pick up two pairs of data flows. The source nodes keep sending packets until the receivers successfully get 100 packets. The experimental results of each scenario are the average values of 100 rounds over different reception reports.

- **802.11 testbed:** We generate network topologies with varied number of APs and receivers. All the receiver can be connected to arbitrary AP with link quality varies from 0.2 to 1. And the average link quality is 0.6. The default number of AP and receiver is 15 and 300 separately.



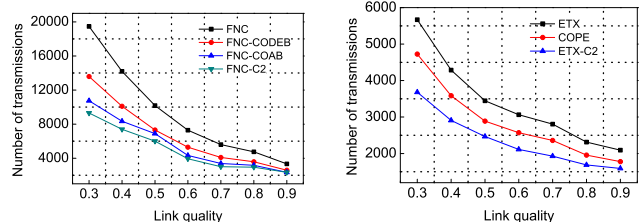
(a) Broadcast scenario

(b) Unicast scenario

Fig. 17. Impact of network sizes

B. Simulation Results on 802.15.4 networks

1) *Impact of Network Size:* Figure 17 shows the performance comparison of our correlated coding schemes (i.e., FNC-C2, and ETX-C2) and other coding schemes (i.e., FNC-CODEB, FNC-COAB, and COPE) with network size ranging from 25 to 100. The left sub-figure in Figures 17 shows the results of FNC, FNC-CODEB, FNC-COAB and FNC-C2 respectively. From this figure, we can find that the average transmission count of our design is 3940, while those of FNC, FNC-CODEB, and FNC-COAB are 7585, 5612 and 4746 respectively. Our design saves 47% of transmissions compared to FNC without using network coding. Compared with FNC-CODEB and FNC-COAB, correlated coding saves about 30% and 20% of transmissions because correlated coding better exploits the necessary coding opportunities. In the unicast application in the right sub-figure in Figures 17, compared with ETX and COPE, the transmission gain of correlated coding is 31% and 18%. From Figures 17, we can also see that the trends of transmission gain with increasing network size in both unicast and broadcast application are quite stable, suggesting that our design scales well with large networks.

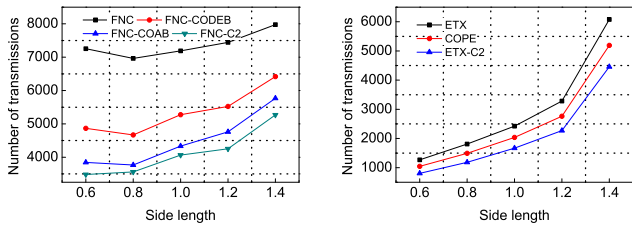


(a) Broadcast scenario

(b) Unicast scenario

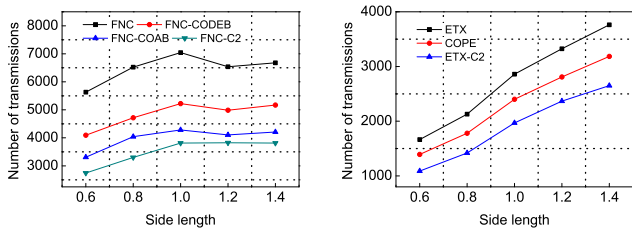
Fig. 18. Impact of link quality

2) *Impact of Link Quality:* Let us consider the transmission gain of correlated coding for networks with different link qualities. The results are shown in Figure 18. From the left sub-figure in Figure 18, we can see that the broadcast transmission count of our design varies from 9302 to 2340 when the link quality varies from 0.3 to 0.9. Compared with FNC, the energy gain of FNC-C2 decreases from 52% to 29% when the link quality increases. A similar result is observed in the unicast application in the right sub-figure in Figure 18, where the transmission gain of ETX-C2 upon ETX decreases from 35% to 23%. The reason is that with higher link quality, the transmission count of a forwarder to send a packet to its destinations is already small, leaving only marginal room for the algorithm to improve the energy gain.



(a) Broadcast scenario (b) Unicast scenario

Fig. 19. Impact of network density (uniform)



(a) Broadcast scenario (b) Unicast scenario

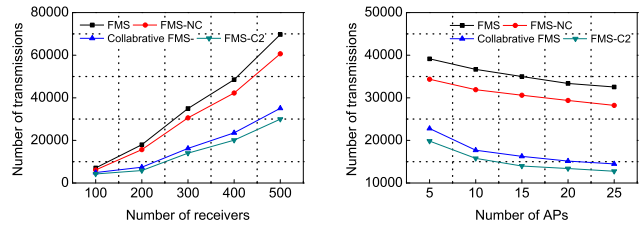
Fig. 20. Impact of network density (non-uniform)

3) *Impact of Network Density*: We consider both uniform (Figure 19) and non-uniform (Figure 20) node distributions. Figures 19 shows the number of transmissions of the four broadcast protocols and three unicast protocols for uniform networks, under different network densities. The average node degrees for side length (of the simulated square sensing field) 0.6, 0.8, 1, 1.2, 1.4 are 20.2, 13.0, 8.4, 5.9, and 3.9, respectively. From the left sub-figures in Figures 19 and 20, we can see that with variation in density, the number of broadcast transmissions does not change monotonically. With the increase of network density, on the one hand, a forwarder has more receivers and needs more transmissions to cover them. On the other hand, the number of forwarders decreases in a fixed size network.

In Figures 19 and 20, the transmission gain of C2 decreases as the side length increases (and thus the density decreases). For example, in the uniform network scenario in Figure 19, the broadcast transmission gain of FNC-C2 over FNC is 52% at node degree 20.2, and it drops to 34% when the average degree is only 3.9. Similarly, the unicast transmission gain of ETX-C2 upon ETX decreases from 37% to 26%. We also find a gain drop in the non-uniform network topology in Figures 20. This is because as the network becomes denser, a node tends to have more one-hop candidates and thus it overhears more packets. This increases C2 the possibility to find more helpful coding opportunities. This explains the increasing energy gain when node density grows.

C. Simulation Results on 802.11 networks

Figure 21(a) shows the performance of C2 with the number of receivers increasing from 100 to 500. Correspondingly, the transmission of FMS (to reliably broadcast 100 packets) increases from 7079 to 69691 while that of C2 increases from 4229 to 29988. In Figure 21(b), with the increase of number



(a) Varied No. of Receivers (b) Varied No. of APs

Fig. 21. Experiments on large scale Wi-Fi networks

of APs from 5 to 25, the transmission of FMS decreases from 39152 to 32536 while that of C2 decreases from 19835 to 12760. With either increased number of receivers or increased number of APs, C2 has more chances to assign the “black sheep” who cause the APs massive retransmissions to the most suitable APs, and thus saving the number of transmissions.

VII. RELATED WORK

Network coding [3], which allows intermediate nodes to combine packets before forwarding, has the great potential to improve the transmission efficiency of both broadcast and unicast applications. The authors in [15] analyze the benefits of network coding for energy savings (in transmissions) and prove that the energy gain is bounded by a constant factor. In practical designs, two coding strategies, i.e., random linear network coding (RLNC) [12] and COPE type network coding [14] are widely used. In RLNC, intermediate nodes may combine incoming packets, by making a random linear combination of them. Once a node receives enough linearly independent combinations, it is able to decode and retrieve the information. The main problem with RLNC is that the size of linear combination of packets is difficult to control.

In COPE [14], the coding strategy exploits the broadcast property of wireless channels and finds coding opportunities. Using this approach, multiple packets are encoded together and then broadcast in a single transmission, thus improving the transmission efficiency. COPE type network coding [14] is applied to both unicast and broadcast protocols [17], [26]. The benefits of network coding in [14], [17], [26] are not fully exploited since the coding opportunity is dependent on the routing path and their coding un-aware routing strategies miss many coding opportunities. The authors in [16], [21], [25], [28] thus propose coding-aware routings to exploit more coding opportunities. These studies implicitly or explicitly assume that wireless links are independent. This unsafe assumption may cause either under-utilized network coding gains or unnecessary coding operations.

While most existing studies in wireless networks focus on individual links or path qualities, Srinivasan and Zhu et al. reveal that the receptions of broadcasting packets at multiple receivers are correlated [22], [29]. In [22], the authors explore a metric called κ that captures the degree of packet reception correlation on different links. The κ value of a network can be used to help network designers decide which protocol should be used for the network. Zhu et al. [29] propose a

probabilistic flooding algorithm to reduce energy consumption in transmissions by using implicit ACKs inferred from link correlation. The authors in [11] propose a particular broadcast design for low-duty-cycle sensor networks, in which nodes with high correlation are assigned to a common sender to save transmissions. In [5], the authors analyze the opportunistic routing gain under the presence of link correlation.

VIII. CONCLUSION

In this paper we study the impact of link correlation on network coding. We find that link correlation can help us decide whether a network coding operation is needed or not. We introduce correlated coding which optimizes the transmission efficiency of network coding. Our design can be applied in both broadcast and unicast protocols. We integrate correlated coding with seven state-of-the-art routing protocols, and evaluate our design with testbed experiments and extensive simulations. The results confirm the effectiveness of our design compared with the existing network coding protocols under a wide range of system settings.

ACKNOWLEDGEMENT

This work was supported in part by National Science Foundation grants CNS-0845994 and CNS-1444021. We thank the reviewers for their valuable comments.

APPENDIX A PROOF OF THE EQUATIONS

Proof of Eq.(1) and Eq.(2): Eq.(1) is a special case of Eq.(2) (when $K = 2$). We now demonstrate why Eq.(2) holds. Let $Pr(\varepsilon > t)$ be the probability that u needs more than t transmissions to deliver a coded packet to K potential receivers, we have

$$\begin{aligned} Pr(\varepsilon > t) &= \sum_{i=1}^K p(\bar{e}_i)^t - \sum_{1 \leq i < j \leq K} p(\bar{e}_i \cap \bar{e}_j)^t \\ &+ \sum_{1 \leq i < j < l \leq K} p(\bar{e}_i \cap \bar{e}_j \cap \bar{e}_l)^t + \dots \\ &+ (-1)^{K-1} p(\bar{e}_1 \cap \bar{e}_2 \cap \dots \cap \bar{e}_K)^t. \end{aligned} \quad (8)$$

Then the expected transmission count for u to reliable broadcast one coded packet can be calculated as

$$\begin{aligned} E[\varepsilon] &= \sum_{t=1}^{+\infty} t \cdot Pr(\varepsilon = t) \\ &= \sum_{t=1}^{+\infty} t \cdot (Pr(\varepsilon > t-1) - Pr(\varepsilon > t)) \\ &= \sum_{t=0}^{+\infty} Pr(\varepsilon > t) \end{aligned} \quad (9)$$

Substitute the right part of the above equation with Eq.(8), we get Eq.(2).

REFERENCES

- [1] *IEEE Standard for Information technology*. IEEE STANDARDS ASSOCIATION, 2012.
- [2] Lorcon wireless packet injection library. <https://code.google.com/p/lorcon/>.
- [3] R. Ahlswede, N. Cai, S. Y. Li, and R. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46:1204–1216, 2000.
- [4] K. Alzoubi, P. Wan, and O. Frieder. New distributed algorithm for connected dominating set in wireless ad hoc networks. In *HICSS*, 2002.
- [5] A. Basalamah, S. M. Kim, S. Guo, T. He, and Y. Tobe. Link correlation aware opportunistic routing. In *IEEE INFOCOM*, 2012.
- [6] S. Biswas and R. Morris. Exor: Opportunistic multi-hop routing for wireless networks. In *ACM SIGCOMM*, 2005.
- [7] 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 *ACM MobiHoc*, 2005.
- [8] T. Clausen, C. Dearlove, and P. Jacquet. The optimized link state routing protocol version 2. In *MANET Working Group*, 2008.
- [9] D. Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *ACM MOBICOM*, 2003.
- [10] G. Ding, Z. Sahinoglu, P. Orlik, J. Zhang, and B. Bhargava. Tree-based data broadcast in ieee 802.15.4 and zigbee networks. *IEEE Transactions on Mobile Computing*, 5:1561–1574, 2006.
- [11] S. Guo, S. M. Kim, T. Zhu, Y. Gu, and T. He. Correlated flooding in low-duty-cycle wireless sensor networks. In *IEEE ICNP*, 2011.
- [12] T. Ho, M. Medard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong. A random linear network coding approach to multicast. *IEEE Transactions on Information Theory*, 52:4413–4430, 2006.
- [13] A. Juttner and A. Magi. Tree based broadcast in ad hoc networks. *Mobile Networks and Applications*, 2005.
- [14] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft. Xors in the air: Practical wireless network coding. In *ACM SIGCOMM*, 2006.
- [15] A. Keshavarz-Haddad and R. Riedi. Bounds on the benefit of network coding: Throughput and energy saving in wireless networks. In *IEEE INFOCOM*, 2008.
- [16] J. Le, C. S. Lui, and D. M. Chiu. Dcar: Distributed coding-aware routing in wireless networks. *IEEE Transactions on Mobile Computing*, 9:596–608, 2010.
- [17] L. E. Li, R. Ramjee, M. Buddhikot, and S. Miller. Network coding-based broadcast in mobile ad-hoc networks. In *IEEE INFOCOM*, 2007.
- [18] W. Lou and J. Wu. On reducing broadcast redundancy in ad hoc wireless networks. *IEEE Transactions on Mobile Computing*, 1:111–122, 2002.
- [19] J. Macke, P. Berens, A. Ecker, A. Toliás, and M. Bethge. Generating spike trains with specified correlation coefficients. *Neural Computation*, 2009.
- [20] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *HICSS*, 2002.
- [21] S. Sengupta, S. Rayanchu, and S. Banerjee. Network coding-aware routing in wireless networks. *IEEE Transactions on Networking*, 18:1158–1170, 2010.
- [22] K. Srinivasan, M. Jain, J. I. Choi, T. Azim, E. S. Kim, P. Levis, and B. Krishnamachari. The κ -factor: Inferring protocol performance using inter-link reception correlation. In *ACM MOBICOM*, 2010.
- [23] K. Srinivasan, M. A. Kazandjieva, S. Agarwal, and P. Levis. The beta-factor: Measuring wireless link burstiness. In *ACM SenSys*, 2008.
- [24] S. Wang, S. Kim, Y. Liu, G. Tan, and T. He. Corlayer: A transparent link correlation layer for energy efficient broadcast. In *ACM MOBICOM*, 2013.
- [25] S. Wang, G. Tan, Y. Liu, H. Jiang, and T. He. Coding opportunity aware backbone metrics for broadcast in wireless networks. In *IEEE INFOCOM*, 2013.
- [26] S. Wang, A. Vasilakos, H. Jiang, X. Ma, W. Liu, K. Peng, B. Liu, and Y. Dong. Energy efficient broadcasting using network coding aware protocol in wireless ad hoc network. In *IEEE ICC*, 2011.
- [27] J. Wu and W. Lou. Forward-node-set-based broadcast in clustered mobile ad hoc networks. In *Wireless Communication and Mobile Computing*, 2003.
- [28] J. Zhang and Q. Zhang. Cooperative network coding-aware routing for multi-rate wireless networks. In *IEEE INFOCOM*, 2009.
- [29] T. Zhu, Z. Zhong, T. He, and Z.-L. Zhang. Exploring link correlation for efficient flooding in wireless sensor networks. In *USENIX NSDI*, 2010.