Coding Opportunity Aware Backbone Metrics for Broadcast in Wireless Networks

Shuai Wang, *Student Member, IEEE*, Guang Tan, *Member, IEEE*, Yunhuai Liu, *Member, IEEE*, Hongbo Jiang, *Member, IEEE*, and Tian He, *Member, IEEE*

Abstract—Reducing transmission redundancy is key to efficient broadcast in wireless networks. A standard approach to achieving this goal is to create a *network backbone* consisting of a subset of nodes that are responsible for data forwarding, while other nodes act as passive receivers. On top of this, network coding (NC) is often used to further reduce unnecessary transmissions. The main problem with existing backbone and NC combinations is that the backbone construction process is blind of what is needed by NC, thus may produce a structure that limits the power of NC algorithms. To address this problem, we propose Coding Opportunity Aware Backbone (COAB) metrics, which seek to maximize coding opportunities when selecting backbone forwarders. We show that the backbone construction process guided by our metrics leads to significantly increased coding frequency, at the cost of minimal localized information exchange. The highlight of our work is COAB's broad applicability and effectiveness. We integrate the COAB metrics with ten state-of-the-art broadcast algorithms specified in eight publications [1]-[8], and evaluate COAB with a running testbed of 30 MICAz nodes and extensively simulations. The experimental results show that our design outperforms the existing schemes substantially.

Index Terms—Broadcast, network coding, connected dominating set, wireless networks

1 Introduction

Reducing transmission redundancy is key to optimal energy efficiency of broadcast in wireless networks. Existing optimization schemes (e.g., [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13]) can be divided into two categories: probabilistic and deterministic. In probabilistic approaches (e.g., [9], [10]), each node rebroadcasts packets to its neighbors with a given forwarding probability. In contrast, deterministic approaches predetermine particular nodes that forward the broadcast packet. In this method, a virtual network backbone is created. Nodes on the backbone are called the *forwarders*, which take the responsibility of delivering packets to their neighbors, while other nodes act as passive receivers. The backbone can be constructed with tree based methods [3], cluster based methods [1], [7], [8], [13], and pruning based methods [2], [4], [5], [6].

Running on top of network backbones, network coding (NC) techniques can be used to further reduce unnecessary transmissions. Originally proposed by R. Ahlswede *et al.* [14], network coding has been adapted to support broadcast applications in wireless networks [15], [16], [17], [18], [19], [20]. In these works, two coding strategies, that is, COPE type network coding (XOR) [21] and random linear network coding (RLNC) [22], are used. XOR coding strategy is applied to the deterministic approach [17], [20], while RLNC is usually used with the probabilistic approach [15], [16].

The main problem with traditional designs is that the backbone construction process is independent of NC, meaning that it is unaware of what is needed by NC. This may lead to a network structure that fails to exploit the full power of NC. It is known that the power of NC heavily depends on the availability of coding opportunities [23], which is a function of packet reception status at the nodes. If such status information can be used by the backbone construction algorithm in such a way that the coding opportunities are maximized, then we can hopefully obtain more benefit from NC.

In this paper, we consider the combination of network coding (NC) with the deterministic approach. At the heart of our design is a *forwarder selection metric*, which considers not only link quality, but also the reception status of neighbors, based on which we estimate the coding opportunity and measure the broadcast efficiency of each link. In addition, our design also introduces a *node association metric*, which assists a downstream node to choose the most efficient upstream forwarder, thus improving broadcast efficiency further.

The main contribution of our work is two Coding Opportunity Aware Backbone (COAB) metrics that have broad applicability and effectiveness. Both the forwarder selection and node association metrics can be easily combined with existing backbone construction algorithms to make the broadcast more efficient. We augment ten backbone construction algorithms, i.e., 1) tree based

- S. Wang and T. He are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455. E-mail: {shuaiw; tianhe}@cs.umn.edu.
- G. Tan is with Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences, China. E-mail: guang.tan@siat.ac.cn.
- Y. Liu is with Third Research Institute of the Ministry of Public Security, Shanghai, China. E-mail: yunhuai.liu@gmail.com.
- H. Jiang is with the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China. E-mail: hongbojiang2004@gmail.com.

Manuscript received 10 June 2013; revised 22 Aug. 2013; accepted 23 Aug. 2013. Date of publication 8 Sept. 2013; date of current version 16 July 2014. Recommended for acceptance by J. Chen.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2013.229

methods [3], 2) cluster based methods [1], [7], [8], and 3) pruning based methods [2], [4], [5], [6], with the COAB metrics. We evaluate the energy efficiency of COAB with both testbed implementations with 30 MICAz nodes and simulations. Experimental results show that compared to the traditional backbone schemes, the COAB-augmented protocols save up to 50 percent of the broadcast transmissions. Our algorithm increases the coding opportunities by up to 50 percent compared to the backbone+NC schemes, resulting in an additional energy gain of 20–30 percent for typical network settings.

The remainder of the paper is structured as follows. Section 2 reviews related work. Section 3 presents the motivation. Section 4 introduces the model, followed by the main design in Section 5. Section 6 explains how to integrate COAB with previous broadcast algorithms. Evaluation results from testbed experiments and simulations are shown in Sections 7 and 8, respectively. Finally, Section 9 concludes the paper.

2 Related Work

Research on efficient broadcast in wireless networks can be divided into two categories: probabilistic and deterministic. In probabilistic methods [9], [10], each node rebroadcasts the packet to its neighbors with a given forwarding probability. In contrast, deterministic approaches predetermine and select forwarders to relay the broadcast packet. To ensure that the broadcast packets reach all the nodes with minimal redundancy, a network backbone is constructed. It has been shown that finding a backbone with a minimum size is NP-hard. Several good approximation algorithms [5], [6], [8], [11], [12], [13] have been proposed.

Network coding [14], which allows intermediate nodes to combine packets before forwarding, has been shown to significantly improve the energy efficiency in wireless networks. The problem of minimizing energy per bit during multicast can be formulated as a linear program and thus can be solved with a polynomial-time algorithm [24]. The authors in [15], [25] analyzed the benefit of network coding for broadcast and proved that the energy gain is bounded by a constant factor. Liu *et al.* [25] considered the case where network coding is based on all the information possessed by a node and showed an upper bound of 3 for the energy gain. While applying network coding to broadcast, two coding techniques, i.e., XOR [21] and RLNC [22], are widely used. Broadcast algorithms that use linear network coding were studied in [15], [16]. In [17], [20], [26], [27], XOR typed network coding is applied upon deterministic algorithms. Li *et al.* [17] applied network coding directly upon a deterministic broadcast algorithm named PDP [5], which potentially misses some coding opportunities of improving the broadcast efficiency. The authors in [26] consider deterministic broadcasting in MANETs using XOR typed network coding and directional antennas.

In this paper, we consider XOR typed network coding upon the deterministic broadcast algorithms. Compared with previous deterministic approaches (with or without NC) [1], [2], [3], [4], [5], [6], [7], [8], [17], [20], [26] which predetermine the virtual network backbone for broadcast, COAB decides the forwarder with knowledge of the best forwarding structure under the current packet reception status. Consequently, our forwarder selection metric can maximize the coding opportunity while the backbone+NC broadcast approaches [17], [20] are blind of coding opportunity.

This paper extends our previous work [27] that focused on design challenges in coding opportunity aware backbone construction. Different from previous work, we provide a more efficient algorithm to approximate the number of transmissions needed by a source node to reliably broadcast a packet to all its covered nodes. The heuristic method significantly reduces the computational complexity. Besides, we supply a node association strategy which can help the covered nodes find a better forwarder. We discuss how to integrate the node association strategy with previous reliable backbone based broadcast algorithms.

3 Motivation

3.1 Network Coding Based Broadcast Rule

Network coding has great potential to improve broadcast efficiency by saving redundant transmissions in wireless networks. When a source node 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 new one. We specify the broadcast coding rule as follows:

Definition 1. ***(Broadcast Coding Rule)*** *Consider a node u transmitting an encoded packet $p' = \oplus(p_1, p_2, \ldots, p_K)$. To decode p', each receiver should have already received $K-1$ packets among p_i, $i = 1, 2, \ldots, K$.*

For NC based broadcast, we seek to encode as many packets as possible. To transmit, a node picks the first packet p_1 in its output queue, checks whether the remaining packets can be encoded with p_1 (i.e., checking the broadcast coding rule) and encodes as many packets as possible. Normally the number of packets that can be encoded into a single packet is small (bounded by the node's degree). Therefore, the computational overhead is insignificant.

3.2 Coding Opportunity in Broadcast

We use an example to show how coding opportunity affects the efficiency of broadcast. Fig. 1 shows two broadcast routes in a network, where the source node u wants to broadcast packets to the other nodes. In Fig. 1a, after u sends the packet, v_1 is selected as the forwarder, and the nodes v_2, v_3 and v_4 are *covered* by the forwarder v_1. The node v_1 broadcasts the received packet (from u) to all the nodes it covers to accomplish the broadcast task. In Fig. 1b, similarly, v_2 is selected as the forwarder. The broadcast task completes when v_2 successfully delivers the packet to its covered nodes.

A node's packet reception information can be found from the packet reception bitmaps at each node in Figs. 1a and 1b, where a block with a thick borderline means a packet being received, and a block with a thin borderline means a packet being missed. Now let's examine the number of packet transmissions needed for the two cases separately.

1. CASE 1 (Fig. 1a): Node v_1 is selected as the forwarder and it needs to retransmit packets $\{p_2, p_3, p_4, p_5, p_6\}$. With the help of NC, v_1 needs to retransmit packet $\{p_2 \oplus p_3, p_4, p_5, p_6\}$ to make sure all the nodes it covers receive all the packets. It is clear that CASE 1 only has one coding opportunity with 2 original packets XORed together.
2. CASE 2 (Fig. 1b): Node v_2 is selected as the forwarder and it needs to retransmit all the six packets. With the help of NC, v_2 only needs to retransmit three packets $\{p_1 \oplus p_2 \oplus p_3, p_4 \oplus p_5, p_6\}$ to make up the losses on v_1, v_3 and v_4. CASE 2 has two coding opportunities where the first XORed packet involves 3 original packets and the second involves 2.

Let's compare two cases: the total number of retransmissions for CASE 1 is 4 while that for CASE 2 is 3. This suggests that in broadcast, if we can manage to increase the coding opportunities when we select the forwarder, then the number of transmissions can be reduced.

4 Model Analysis

Our objective is to reduce transmissions by increasing NC opportunities. The natural question, then, is: how much benefit can be obtained from NC in broadcast? To answer the question, we first estimate the expected number of transmissions needed for reliable delivery of a packet from a source to all its receivers without considering NC. Then, we quantify the benefit of coding opportunities in reducing transmissions when NC is considered. Some notations used in this paper are listed in Table 1.

Here, we assume a widely used ARQ model for reliable delivery. In ARQ, if a forwarder does not receive an ACK before timeout, it retransmits the packet until it receives an ACK. With ARQ, for each link e with round-trip link quality of $p(e)$, the expected number of transmissions needed to successfully send a packet over a single link e is $\frac{1}{p(e)}$. We also assume that although link quality of wireless links changes over time, it can be measured and refreshed through periodic beacons, sequenced data packets or LQI [28].

TABLE 1
Notation Used in This Paper

Notation	Description
$e_j(u) = \{u, v_j\}$	A link from node u to v_j, we use e_j for short when u is clear from the context
$p(e)$	The link quality, measured by the transmission success rate
$\varepsilon(u), \hat{\varepsilon}(u)$	The number of transmissions for u to reliable broadcast one packet, $\hat{\varepsilon}(u)$ is an approximation of $\varepsilon(u)$
$\beta_{nc}(u)$	The total number of reduced broadcast packets on node u with NC
$\xi_{nc}(V(u))$	The per-link covering cost of u to broadcast a packet to the node set $V(u)$ with NC
$\eta_{nc}(e)$	Link e's forwarding cost with NC

4.1 Expected Transmission Count

We denote $\varepsilon(u)$ as the number of transmissions needed by forwarder u to deliver one packet to all its covered nodes without considering NC. Clearly, the total number of transmissions for the broadcast is thus the summation of ε of all the forwarders. Let the set of nodes covered by forwarder u be $V(u) = \{v_1, v_2, \ldots v_M\}$, where $M = |V(u)|$. Let the link quality between u and its covered node v_j be $p(e_j)$, $j = 1, 2, \ldots, M$. The corresponding packet loss probability is denoted $p(\overline{e_j}) = 1 - p(e_j)$. Without loss of generality, we assume $p(e_1) \geq p(e_2) \geq p(e_3) \geq \cdots \geq p(e_M)$.

Three covered nodes case: We start by considering the three covered nodes case as in Fig. 2a, where node u is the forwarder and nodes v_1, v_2 and v_3 are covered by u. Fig. 2b shows a diagram representing the events where covered nodes receive a transmission from u. $p(e_1 \cap e_2 \cap e_3)$ is the probability that all three receivers successfully receive a packet. Without correlated shadowing and severe interference [29], wireless links are considered to be independent [30]. This means $p(e_1 \cap e_2 \cap e_3) = p(e_1)p(e_2)p(e_3)$.

Let $\Pr(\varepsilon(u) > k)$ be the probability that u needs more than k transmissions to deliver a packet to all the three receivers, then we have

$$Pr(\varepsilon(u) > k) = p(\bar{e_1})^k + p(\bar{e_2})^k + p(\bar{e_3})^k - (p(\bar{e_1} \cap \bar{e_2}))^k - (p(\bar{e_1} \cap \bar{e_3}))^k - (p(\bar{e_2} \cap \bar{e_3}))^k + (p(\bar{e_1} \cap \bar{e_2} \cap \bar{e_3}))^k.$$

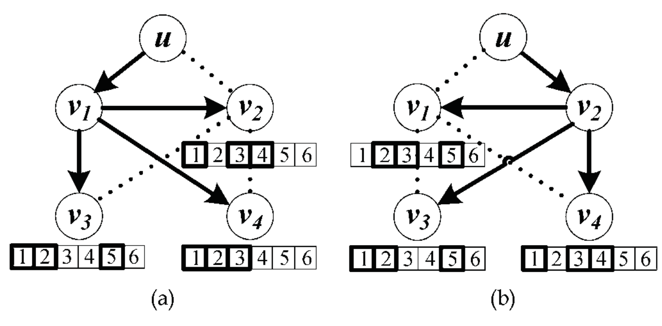

Fig. 1. Impact of coding opportunity on broadcast. In the packet reception bitmap, a block with a thick borderline means received packets, and a block with a thin borderline means a lost one. (a) Case 1: v_1 is the forwarder. (b) Case 2: v_2 is the forwarder.

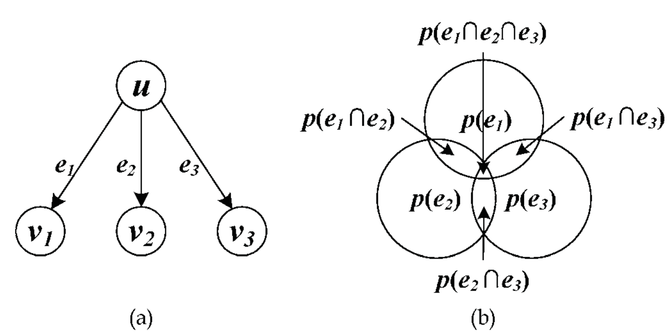

Fig. 2. Three covered nodes case. (a) A forwarder with 3 covered nodes. (b) Joint reception.

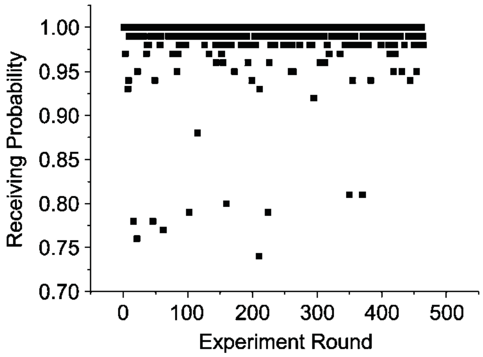

Fig. 3. Statistics of the proportion of packets received earlier or at the same time from a better link.

Moreover,

$$Pr(\varepsilon(u) = k) = Pr(\varepsilon(u) > k - 1) - Pr(\varepsilon(u) > k).$$

Thus,

$$\begin{aligned}
E[\varepsilon(u)] &= \sum_{k=1}^{+\infty} k \cdot Pr(\varepsilon(u) = k) \\
&= \frac{1}{1 - p(\bar{e_1})} + \frac{1}{1 - p(\bar{e_2})} + \frac{1}{1 - p(\bar{e_3})} \\
&\quad - \frac{1}{1 - p(\bar{e_1} \cap \bar{e_2})} - \frac{1}{1 - p(\bar{e_1} \cap \bar{e_3})} \\
&\quad - \frac{1}{1 - p(\bar{e_2} \cap \bar{e_3})} + \frac{1}{1 - p(\bar{e_1} \cap \bar{e_2} \cap \bar{e_3})} \\
&= \sum_{i=1}^{3} \frac{1}{1 - p(\bar{e_i})} - \sum_{i \neq j} \frac{1}{1 - p(\bar{e_i} \cap \bar{e_j})} \\
&\quad + \frac{1}{1 - p(\bar{e_1} \cap \bar{e_2} \cap \bar{e_3})}.
\end{aligned}$$

To get $\varepsilon(u)$ with three covered nodes, we need to calculate $C_3^1 + C_3^2 + C_3^3 = 7$ polynomial terms. More generally, for M covered nodes, the computational complexity of calculating $\varepsilon(u)$ is $C_M^1 + C_M^2 + C_M^3 + \cdots + C_M^M = 2^M - 1$. Although in wireless networks, the number of covered nodes M is relatively small, the exponential growth of complexity with M shall be avoided when possible. In the following section, we present an approximation to simplify the calculation.

4.2 Approximation With Reduced Complexity

Due to the high cost of computing $\varepsilon(u)$, we seek a more efficient algorithm to approximate $\varepsilon(u)$ with lower computational complexity. Through extensive empirical studies, we observe that the nodes with a higher link quality usually receive the broadcast packet before those with a lower link quality. We deploy 31 MICAz nodes near a sender u, which broadcasts a packet every 0.2 s. The total number of packet broadcasts is 1000. The receivers keep the packet sequence number and time stamp. After collecting the packet reception trace, for each packet, we compare the reception between each link pair (there are $\binom{2}{31} = 465$ such pairs). Fig. 3 shows that the node with a better link from u receives about 98 percent of the packets earlier (or at the same time) than the node with a worse link from u. Based on this observation, we propose an approximate method to estimate $\varepsilon(u)$. We first estimate the number of transmissions for the source node u to reliably send a packet to the node v_i with a better link. Then we consider the transmissions of delivering a packet to the node v_j with a worse link under the situation that v_j fails to receive the packet when u sends it to v_i.

Lemma 1.

$$\hat{\varepsilon}(u) = \sum_{j=1}^{M} \frac{1}{p_j} - M + 1. \tag{1}$$

Proof. Let $K_i(u)$ be the set of i nodes with the highest link qualities among u's covered nodes.

$$\begin{aligned}
\hat{\varepsilon}(u) &= \frac{1}{p(e_1)} + \frac{\Pr(\bar{e_2}|p(e_1))}{p(e_2)} + \cdots + \frac{\Pr\left(\bar{e_M}| \bigcap_{i=1}^{M-1} p(e_i)\right)}{p(e_M)} \\
&= \frac{1}{p(e_1)} + \frac{p(K_1(u)) - p(K_2(u))}{p(e_1) \cdot p(e_2)} + \dots \\
&\quad + \frac{p(K_{M-1}(u)) - p(K_M(u))}{p(K_{M-1}(u))p(e_M)} \\
&= \sum_{i=1}^{M} \frac{1}{p(e_i)} - \sum_{i=2}^{M} \frac{1}{p(e_i)} \cdot \frac{p(K_i(u))}{p(K_{i-1}(u))}.
\end{aligned}$$

Because of link independence, we have

$$p(K_i(u)) = p(e_i) \cdot p(K_{i-1}(u)).$$

Thus,

$$\hat{\varepsilon}(u) = \sum_{i=1}^{M} \frac{1}{p(e_i)} - M + 1.$$

□

Validation of Eq. (1): To verify the correctness of Eq. (1), we did an experiment on an 802.15.4 testbed. In this testbed, 10 MICAz nodes are deployed to form a single-hop network. A randomly selected node serves as the transmitter and broadcasts packets to five arbitrary nodes under channel 26, which is free of external interference (e.g., WiFi). The five receivers report their reception results to a sink node after all of them receive 100 packets. The X-axis of Fig. 4 is the real transmission count used by the transmitter to cover five arbitrary nodes with one packet, while the Y-axis is the corresponding estimated transmission count using Eq. (1) with WMEWMA [31] parameter $\alpha = 0.1$ which means the expected transmission count calculation gives 90 percent of the weight to the current $\hat{\varepsilon}$ and 10 percent of the weight to the historical value. The window size in the experiment is 5. From Fig. 4, we can see that the estimated transmission count is quite close to the real one.

4.3 Coding Opportunities Estimation

From the example in Section 3, we can find that the coding opportunity is crucially dependent on the forwarder selection: we can get more benefit from NC if node v_2 (Fig. 1b) is selected as the forwarder. Therefore, it is imperative to

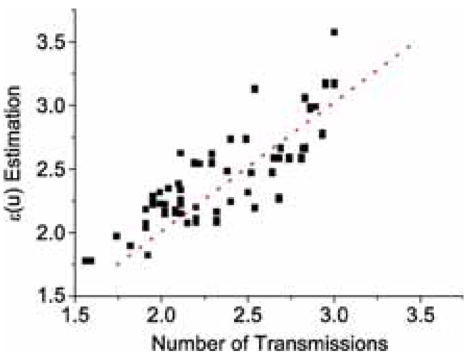

Fig. 4. Validation of Eq. (1): the estimated transmission count is quite close to the real one.

estimate the benefit of NC for each forwarder candidate. First, let's give the formal definition of coding opportunity:

Definition 2. *(**Coding Opportunity**) For packets buffered in an output queue, if there exist a group of packets that satisfy the broadcast coding rule and thus can be encoded together, we call this condition a coding opportunity.*

Let the number of coding opportunities with k_i original packets involved in an encoded packet be t_i, $2 \leq k_i \leq M$. Node u's total reduced number of broadcast packets by using network coding $\beta_{nc}(u)$ is given by

$$\beta_{nc}(u) = \sum_{i=2}^{M} (k_i - 1)t_i. \tag{2}$$

Note that each broadcast packet may need multiple retransmissions to ensure it be received by all the receivers. This makes significant room for NC to reduce transmissions.

5 COAB Metric

Although deterministic broadcast protocols are highly diverse, they all need to address two issues: (ii) how to choose backbone forwarders, and (ii) if a node can hear packets from multiple upstream forwarder nodes, which upstream forwarder should it get associated with? This section presents two metrics that address these questions.

5.1 Forwarder Selection Metric

We use the forwarder selection metric to measure the forwarding capability of a node, which is defined as following:

Definition 3. *(**Forwarder Selection Metric**) The forwarder selection metric is defined as the number of transmissions needed by u to deliver a packet to all of its covered nodes, divided by the number of u's covered nodes.*

The Case without NC: If NC is not used, the forwarder selection metric, denoted as $\xi(V(u))$, is:

$$\xi(V(u)) = \frac{\hat{\varepsilon}(u)}{M}, \tag{3}$$

where M is the number of u's covered nodes. $\xi(V(u))$ offers a good estimate for the expected transmission count for a successful packet delivery without NC. It captures a basic characteristic of lossy links. In a nutshell, $\xi(V(u))$ suggests that selecting a proper forwarder should consider covered nodes with good link qualities.

To calculate $\xi(V(u))$, we need to know $\sum_{v_j \in V(u)} 1/p(e_j)$, which in turns requires the knowledge of link quality $p(e_j)$. In wireless networks, link quality is known to be dynamic and thus online measurement is needed. Many existing measurement methods [28] can be used in COAB. For example, every node can periodically send out a HELLO message at an adaptive time interval T which is increased or decreased based on the link's stability. Every HELLO message is identified by the node ID and a packet sequence number. The message is used not only for one-hop neighbor discovery, but also for updating $p(e_j)$. The calculation of link quality is straightforward. Every node maintains a reception record of all HELLO messages from its neighboring nodes within a time window WT. To reduce the required memory space and mitigate the overhead of control messages, the record is represented in a bitmap format (e.g., [110010]) for each neighbor. Such records are exchanged within a HELLO message every WT seconds among neighboring nodes. Take the network topology in Fig. 1a for an example. The link qualities of link (v_1, v_2), (v_1, v_3), and (v_1, v_4) are 0.5, 0.5 and 0.5 respectively. Therefore, $\xi(V(v_1)) = \frac{\frac{1}{0.5}+\frac{1}{0.5}+\frac{1}{0.5}-3+1}{3} = \frac{4}{3}$. Similarly, $\xi(V(v_2))$ in Fig. 1b is equal to $\frac{4}{3}$.

The Case with NC: If NC is used, the forwarder selection metric, denoted as $\xi_{nc}(V(u))$, is:

$$\xi_{nc}(V(u)) = \frac{(|\Phi(u)| - \beta_{nc}(u))}{|\Phi(u)|}\xi(V(u)). \tag{4}$$

The calculation of $\xi_{nc}(V(u))$ in Eq. (4) involves two terms: 1) $\xi(V(u))$ is the forwarder selection metric in the case without NC. and 2) $\frac{(|\Phi(u)|-\beta_{nc}(u))}{|\Phi(u)|}$ is the percentage of packets left in the queue after NC, where $\Phi(u)$ is the set of packets in node u's output queue and $|\Phi(u)|$ is the size of the queue. For example, in Fig. 1b, node v_1 needs to transmit packets $\Phi(v_1) = \{p_2, p_3, p_4, p_5, p_6\}$ and thus $|\Phi(v_1)| = 5$. While $\Phi(u)$ is straightforward, the calculation of $\beta_{nc}(u)$, total number of reduced broadcast packets, deserves a little more explanation. In Fig. 1b for example, $\beta_{nc}(u)$ is calculated based on the packet reception information of node v_2's covered nodes, which is obtained through the periodical HELLO messages. It is easy to see that there exist two coding opportunities: one is $p_4 \oplus p_5$ which involves two packets, and the other is $p_1 \oplus p_2 \oplus p_3$ which involves three packets. Essentially we encode five packets $\{p_2, p_3, p_4, p_5, p_6\}$ into two packets $p_4 \oplus p_5$, $p_1 \oplus p_2 \oplus p_3$, with a total reduction $\beta_{nc}(v_2) = 3$. Similarly in Fig. 1a, $\beta_{nc}(v_1) = 1$.

Utilization of Forwarder Selection Metric: In Fig. 1, for example, the Forwarder Selection Metric for forwarder candidate v_1 is $\xi_{nc}(V(v_1)) = \frac{(5-1)}{5} \times \frac{4}{3} = \frac{16}{15}$, while that's for forwarder candidate v_2 is $\xi_{nc}(V(v_2)) = \frac{(6-3)}{6} \times \frac{4}{3} = \frac{2}{3}$. Since, $\xi_{nc}(V(v_1)) > \xi_{nc}(V(v_2))$, our metric indicates that node v_2 is a better forwarder.

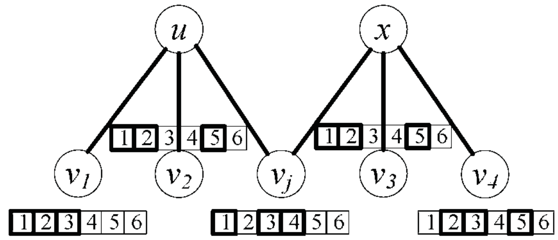

Fig. 5. Example of node association.

Algorithm 1 Node Association (v_j).

1: v_j is COVERED by u
2: **if** Received a FORWARDER message from x**then**
3: **if** $\eta_{nc}(e_j(u)) > \eta_{nc}(e_j(x))$ **then**
4: Node v_j is associated with forwarder x;
5: **end if**
6: **end if**

5.2 Node Association Metric

In wireless broadcast, a node v_j may hear packets from multiple upstream forwarder nodes. In deterministic broadcast, node v_j needs to associate itself with the most efficient upstream forwarder u_i. The key idea for the node association metric is to choose an upstream forwarder with the minimal forwarding cost.

Definition 4. *(**Node Association Metric**) Given a forwarder u_i and its covered node set $V(u_i) = \{v_1, v_2, \dots v_M\}$. The node association metric is the forwarding cost of the link $e(u_i, v_j)$, $j = 1, 2, \dots M$*

$$\eta_{nc}\big(e(u_i, v_j)\big) = M\xi_{nc}(V(u_i)) - (M-1)\xi_{nc}\big(V(u_i) - \{v_j\}\big). \quad (5)$$

The node association metric is used to measure the cost for a forwarder u_i broadcasting a packet to a specified downstream node v_j, which hears packets from other forwarders.

Utilization of Node Association Metric: Here we use a simple example to show how the node association metric is used. The pseudo-code is shown in Algorithm 1. For node v_j (initially dominated by forwarder u), if it receives another FORWARDER message from forwarder x (lines 1 & 2), it calculates the forwarding costs of link $e(u, v_j)$ and $e(x, v_j)$. If the forwarding cost of $e(u, v_j)$ is greater than that of $e(x, v_j)$ (line 3), node v_j re-selects node x as its forwarder. It is easy to prove that the selection based on our node association metric always reduces the expected total transmissions. Due to space constraints, we omit such proof.

Let's illustrate further with a more concrete example. In Fig. 5, node v_j can hear packets from either forwarder u or forwarder x. Node v_j judges which node it should associate with by comparing broadcast link costs. The costs of links $e(u, v_j)$ and $e(x, v_j)$ are $\eta_{nc}(e(u, v_j)) = 3\frac{(5-1)}{6}(\frac{3}{0.5} - 3 + 1) - 2\frac{(4-1)}{4}(\frac{2}{0.5} - 3 + 2) = \frac{7}{2}$, and $\eta_{nc}(e(x, v_j)) = 3\frac{(6-3)}{6}(\frac{3}{0.5} - 3 + 1) - 2\frac{(4-1)}{4}(\frac{2}{0.5} - 3 + 2) = \frac{3}{2}$ respectively. Therefore, v_j chooses node x as its upstream forwarder.

6 Integrating COAB Metrics with Backbones

We classify the existing reliable broadcast algorithms into tree-based [3], cluster-based [1], [7], [8], and pruning-based [2], [4], [5], [6] methods. Thus far, we have successfully implemented ten classical algorithms and embedded COAB metrics with them. The basic information of these algorithms is shown in Table 2. We briefly introduce how to embed our design into these tree backbone construction algorithms, and thus bringing them an improvement on energy efficiency. In Tree+COAB, instead to find the nodes with maximum leaves, we choose the nodes with $\min(\xi_{nc})$ as the tree nodes. To combine cluster based broadcast with COAB, the algorithm Cluster+COAB first selects nodes with $\min(\xi_{nc})$ to form a maximum independent set (MIS). Then, Cluster+COAB finds connectors to link the nodes in MIS. In Pruning+COAB, each forwarder adds its one-hop neighbors with $\min(\xi_{nc})$ to forwarder set to cover its two-hop neighbors.

In Tree+COAB, Cluster+COAB and Pruning+COAB, if a covered node receives a message from the nodes in tree, MIS or forwarder set, the node association metric is used to help the covered node find a better forwarder.

Running the COAB metric introduces little additional communication cost. The main overhead is from two sources. One is packet reception bitmap exchange between neighboring nodes which is used to calculate the expected transmission count, coding opportunity and the broadcast link cost. The exchange of bitmap is *already* required by previous network coding schemes [17], [20], [21]. Besides, the bitmap is designed to be very short (e.g., 2 bytes) so this overhead is negligible. The other part of overhead is the

TABLE 2
Ten State-of-Art Protocols Supported by COAB

Protocol Name	Reference	Network Info.	Hello Msg	Broadcast Msg	Category
Spanning Tree	[3]	One-hop	ID	Msg only	Tree-based
Cluster Tree	[1]	Quazi-Global	Global	Msg only	Tree and Cluster-based
Forwarding Node Cluster	[8]	Local	ID	Covered set	Tree and Cluster-based
Clustering	[7]	Quazi-Local	Degree	Msg only	Cluster-based
Multi-Point Relay	[6]	Two-hop	One-hop	Msg + Covered set	Pruning-based
Self Pruning	[4]	One-hop	One-hop	Msg + Covered set	Pruning-based
Partial Dominating Pruning	[5]	Two-hop	One-hop	Msg + Covered set	Pruning-based
Dominating Pruning	[4]	Two-hop	One-hop	Msg + Covered set	Pruning-based
Total Dominating Pruning	[5]	Two-hop	One-hop	Msg + Covered set	Pruning-based
RNG Relay Subset	[2]	Two-hop	One-hop	Msg only	Pruning-based

exchange of one-hop neighbor information, which is required by backbone construction algorithms [5], [6], [8]. Thus, applying COAB will not noticeably affect the system's overall overhead.

7 Testbed Implementation

In this section, we report the experiment results of ten state-of-art protocols integrating the COAB metrics on a TinyOS/Mote platform consisting of 30 MICAz nodes.

7.1 Experiment Setup

We deploy 30 MICAz nodes randomly on an in-door testbed shown in Fig. 6. In the beginning of the experiment, a control node is used to remotely configure radio parameters, i.e., transmission power and channel. According to the testbed size, i.e., 8 m $\times$ 3 m, the power is set to be -25 dBm. We use 802.15.4's channel 26, which is free of external interference (e.g.,WiFi). Based on these radio settings, each node broadcasts 100 HELLO packets in turn. Each packet was identified by a sequence number. The transmission rate is 5 packets/sec. All the received packets are recorded in the MICAz 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 PC. We thus obtain the information required by COAB, i.e., link qualities and packet receiving patterns, from packet reception history, and calculate the backbone for broadcast using the forwarder selection method and node association strategy. Then, the corresponding nodes in the testbed are selected as forwarders (the backbone). The forwarders keep on broadcasting packets until all their covered nodes receive 100 packets. Based on the packet reception records, the average link quality of the testbed scenario is about 0.85.

7.2 Performance Metrics

We use two metrics for performance evaluation:

1. **Number of Transmissions**, which is defined as the number of transmissions needed by a broadcast scheme to reliably broadcast 100 packets to the whole network. We define energy gain as the percentage of saved transmissions.
2. **Number of Coding Operations**, defined as the number of times that network coding occurs during the simulation. It is used to measure coding opportunities.

Fig. 6. Testbed.

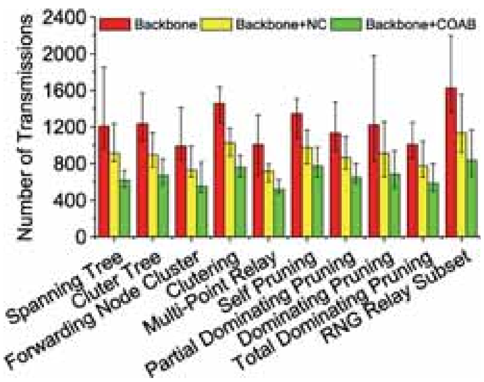

Fig. 7. Number of transmissions.

7.3 Main Performance Results

The experimental results of the ten classical reliable broadcast protocols are shown in Fig. 7. The first bar (in red) in each set of data represents the broadcast transmissions needed by the backbone schemes, while the second bar (in yellow) and the third bar (in green) represent the transmissions needed by backbone+NC and backbone+COAB schemes separately. For example, for the Spanning Tree (backbone) algorithm, the nodes need 1208 transmissions on average to guarantee that every node in the network receives 100 packets, while the number is 616 when COAB is combined with Spanning Tree, achieving a reduction of 49 percent. The average transmission of backbone+NC and backbone+COAB is 892 and 662, respectively. On average, our design COAB reduces transmissions of backbone+NC by 26 percent. For the number of coding operations in Fig. 8, we see that on average, backbone+COAB produces 43 percent more coding opportunities than backbone+NC. These improvements turn out to be very helpful for broadcast efficiency.

Although we have collected results for all ten protocols, space constraints do not allow presenting all of them here. Therefore, we have chosen three representative broadcast algorithms, namely Spanning Tree [3] (Tree for short),

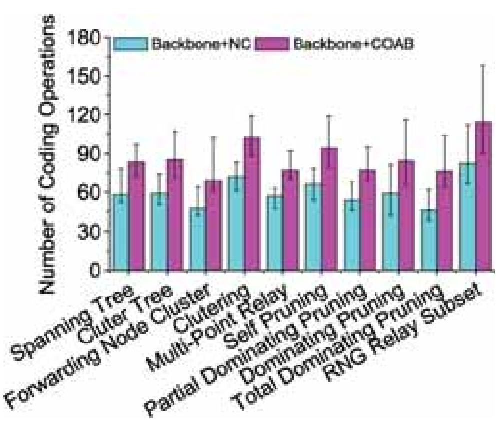

Fig. 8. Number of coding operations.

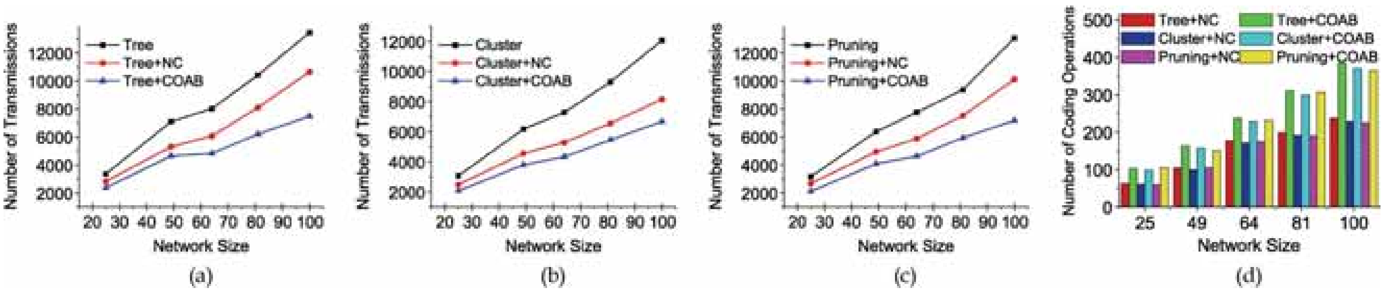

Fig. 9. Performance in uniform networks with different sizes. (a) Tree based backbone. (b) Cluster based backbone. (c) Pruning based backbone. (d) Number of coding operations.

Forwarder Node Cluster [8] (Cluster for short), and Multi-Point Relay [6] (Pruning for short) for the rest of the experiments in the simulation.

8 Simulation

In this section, we present simulation results for large-scale networks under different settings.

8.1 Simulation Setup

We generate both uniform and non-uniform network topologies with different network sizes and densities. Given a scenario, we generate independent reception bitmaps for all the sender-receiver pairs by modifying the sampling algorithm for Bernoulli random variables in [32]. For a particular packet, the reception status at a receiving node can be either 0 or 1. We assume that the bitmaps at different nodes are of the same length. By default the network size is 64, the average link quality is 0.6, and the field size is 800 m $\times$ 800 m with a communication range of 160 m. In the experiment, the source (e.g., node 1) broadcasts 100 packets, and we record the number of network coding operations and the number of transmissions required to finish broadcasting the 100 packets. The experimental results of each scenario are the average values of 100 rounds over different bitmaps.

8.2 Simulation Results

8.2.1 Impact of Network Size

Fig. 9 shows the performance comparison of our COAB schemes (i.e., Tree+COAB, Cluster+COAB and Pruning+-COAB) and Backbone+NC schemes (i.e., Tree+NC, Cluster+NC and Pruning+NC) with networks size ranging from 25 to 100. Figs. 9a, 9b, and 9c show the results of tree, cluster and pruning based broadcast schemes respectively. It can be seen from Fig. 9b that the average transmission count of our design is 4466, while those of Cluster and Cluster+NC are 7586 and 5373 respectively. Our design saves 41 percent of transmissions compared to the cluster based scheme without using NC. Compared with Cluster+NC, Cluster+-COAB saves about 20 percent of transmissions because Cluster+COAB better exploits the power of NC. From Figs. 9a, 9b, and 9c, we can also see that the trend of energy gain with increasing network size is quite stable, suggesting that our design scales well with large networks.

Fig. 9d shows the number of coding operations of COAB and Backbone+NC under different network sizes. In all tree, cluster and pruning based schemes, we find that COAB produces much more coding operations than Back-bone+NC which applies NC directly to the backbone. On average, our scheme increases the number of coding operations by about 50 percent, which greatly reduces broadcast transmission.

Fig. 10 shows the performance of the COAB scheme and Backbone+NC scheme in non-uniform network topologies. The results are quite similar to those in uniform network topologies. On average, compared with backbone based broadcast algorithm (i.e., Tree, Cluster and Pruning), our COAB scheme achieves an energy gain of 40 percent. Also it finds 53 percent more coding opportunities than Backbone+NC, leading to about 20 percent fewer transmissions.

8.2.2 Impact of Link Quality

Let's consider the energy gain of COAB and Backbone+NC for networks with different link qualities. The results are

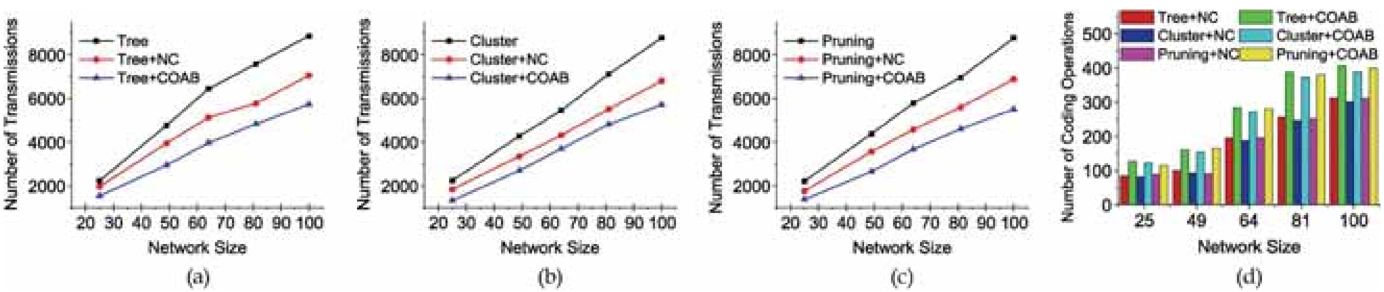

Fig. 10. Performance in non-uniform networks with different sizes. (a) Tree based backbone. (b) Cluster based backbone. (c) Pruning based backbone. (d) Number of coding operations.

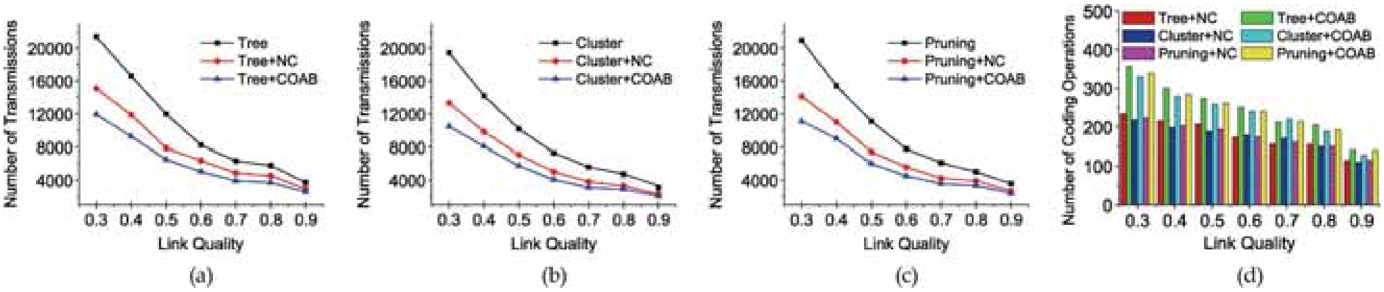

Fig. 11. Performance in networks with different link qualities. (a) Tree based backbone. (b) Cluster based backbone. (c) Pruning based backbone. (d) Number of coding operations.

shown in Figs. 11a, 11b and 11c. From Fig. 11c, we can see that the transmission count of our design varies from 11927 to 2346 when the link quality varies from 0.3 to 0.9. Compared with Pruning algorithm, the energy gain of Pruning+COAB decreases from 47 percent to 32 percent when the link quality increases. The reason is that with higher link quality, the transmission count of a forwarder to send a packet to its covered nodes is already small, leaving only a marginal room for the algorithm to improve the energy gain. For the same reason, the number of coding operations in Fig. 11d also decreases when the link quality improves.

8.2.3 *Impact of Network Density*

In this experiment, we consider both uniform (Fig. 12) and non-uniform (Fig. 13) node distributions. Figs. 12a, 12b, and 12c show the number of transmission of Backbone schemes, Backbone+NC schemes and COAB for uniform networks, under different network densities. The average node degrees for side length (of the simulated square sensing field—800 m $\times$ 800 m) 0.6, 0.8, 1, 1.2, 1.4 are respectively 20.2, 13.0, 8.4, 5.9, and 3.9. From Figs. 12 and 13, we can see that with variation in density, the number of transmissions does not change monotonically. This is because with the increase of network density, a forwarder has more receivers and needs more transmissions to cover them, but the number of forwarders decreases in a fixed size network. The energy gain of our design decreases as the side length increases (and thus the density decreases). For example, in Fig. 12b, the energy gain of Cluster+COAB is 47 percent at an average degree of 20.2, and it drops to 30 percent when the average degree is only 3.9. This is because as the network becomes denser, a forwarder tends to covers more nodes. This increases the possibility that links with poor qualities are put into the same cluster, thus giving our node association algorithm more opportunities to find a suitable forwarder for a node. This explains the increasing energy gain when the node density grows.

Figs. 12d and 13d show the number of coding operations of COAB and Backbone+NC under different network densities. We find that the number of coding operations of our COAB scheme increases as the density decreases. There are two reasons for this. In this experiment, to make sure the network density is the unique affecting factor, we fix the network size (i.e., 64 nodes). There are fewer forwarders in a denser network. In COAB, only the forwarders perform packet encoding. Therefore, a denser network has fewer coding operations. Also, a forwarder covers more nodes in a denser network. It is more difficult to satisfy the broadcast rule because a node needs to ensure that all the covered nodes be able to decode the packet.

9 Conclusion

In this paper we have studied the effect of network coding opportunity on the performance of broadcast. The key novelty of this work lies in two generic metrics that can be used in a wide range of deterministic flooding algorithms. Specifically, we developed a new forwarder selection metric to capture potential coding opportunities and a node association metric to help each node get associated with a suitable upstream forwarder. Our effort to demonstrate COAB's broad applicability and effectiveness is comprehensive. We integrate COAB with ten state-of-the-art broadcast algorithms, and evaluate our design with testbed experiments and extensive simulations. The results

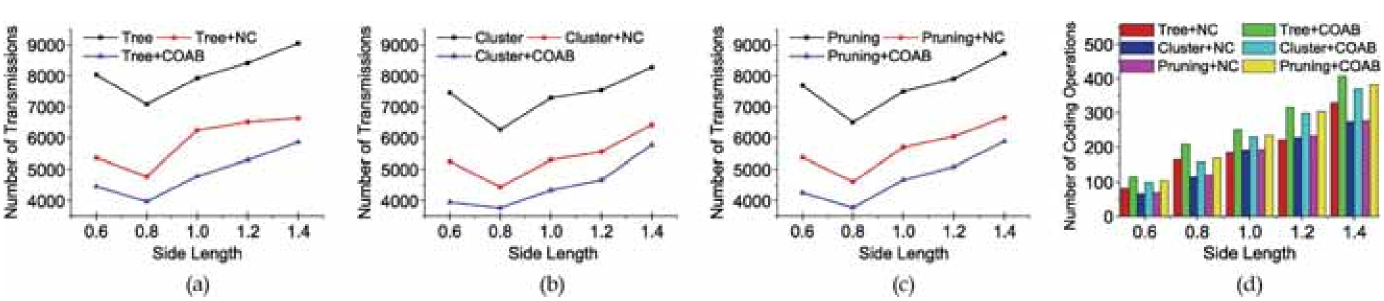

Fig. 12. Performance in uniform networks with different network densities. (a) Tree based backbone. (b) Cluster based backbone. (c) Pruning based backbone. (d) Number of coding operations.

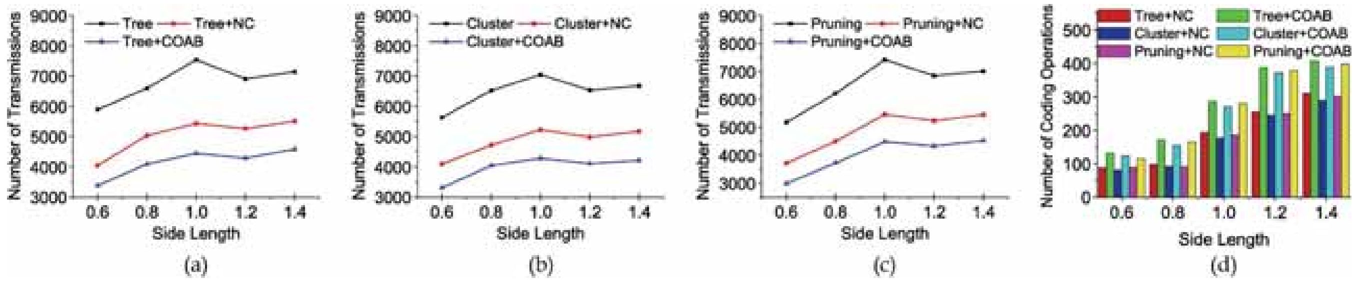

Fig. 13. Performance in non-uniform networks with different network densities. (a) Tree based backbone; (b) Cluster based backbone; (c) Pruning based backbone. (d) Number of coding operations.

confirm the effectiveness of our design compared with the independent combinations of NC and backbone algorithms schemes under wider range of system settings.

ACKNOWLEDGMENT

G. Tan's work was supported by the National Science Foundation of China (NSFC) under Grants 61103243 and 61379135, and Youth Innovation Promotion Association, Chinese Academy of Sciences.

REFERENCES

[1] K.M. Alzoubi, P.J. Wan, and O. Frieder, ''New Distributed Algorithm for Connected Dominating Set in Wireless Ad Hoc Networks,'' in *Proc. Hawaii Int'l Conf. Syst. Sci.*, 2002, pp. 3849-3855.

[2] J. Cartigny, F. Ingelrest, and D. Simplot, ''RNG Relay Subset Flooding Protocols in Mobile Ad Hoc Networks,'' *Int'l J. Found. Comput. Sci.*, vol. 14, no. 2, pp. 253-265, Apr. 2003.

[3] A. Juttner and A. Magi, ''Tree Based Broadcast in Ad Hoc Networks,'' *Mobile Netw. Appl.*, vol. 10, no. 5, pp. 753-762, Oct. 2005.

[4] H. Lim and C. Kim, ''Flooding in Wireless Ad Hoc Networks,'' *Comput. Commun. J.*, vol. 24, no. 3/4, pp. 353-363, Feb. 2001.

[5] W. Lou and J. Wu, ''On Reducing Broadcast Redundancy in Ad Hoc Wireless Networks,'' *IEEE Trans. Mobile Comput.*, vol. 1, no. 2, pp. 111-122, Apr.–June 2002.

[6] A. Qayyum, L. Viennot, and A. Laouiti, ''Multipoint Relaying for Flooding Broadcast Messages in Mobile Wireless Networks,'' in *Proc. HICSS*, 2002, pp. 3866-3875.

[7] I. Stojmenovic, M. Seddigh, and J. Zunic, ''Dominating Sets and Neighbor Elimination Based Broadcasting Algorithms in Wireless Networks,'' *IEEE Trans. Parallel Distrib. Syst.*, vol. 13, no. 1, pp. 14-25, Jan. 2002.

[8] J. Wu and W. Lou, ''Forward-Node-Set-Based Broadcast in Clustered Mobile Ad Hoc Networks,'' *Wireless Commun. Mobile Comput.*, vol. 3, no. 2, pp. 155-173, 2003.

[9] Z. Haas, J. Halpern, and L. Li, ''Gossip-Based Ad Hoc Routing,'' in *Proc. IEEE INFOCOM*, 2002, pp. 1707-1716.

[10] J. Subramanian, R. Morris, and H. Balakrishnan, ''Uflood: High-Throughput Flooding Over Wireless Mesh Networks,'' in *Proc. IEEE INFOCOM*, 2012, pp. 82-90.

[11] M. Li, W. Cheng, K. Liu, Y. He, X. Li, and X. Liao, ''Sweep Coverage With Mobile Sensors,'' *IEEE Trans. Mobile Comput.*, vol. 10, no. 11, pp. 1534-1545, Nov. 2011.

[12] M. Li and Y. Liu, ''Underground Coal Mine Monitoring With Wireless Sensor Networks,'' *ACM Trans. Sens. Netw.*, vol. 5, no. 2, p. 10, Mar. 2009.

[13] A.E. Abdallah, T. Fevens, and J. Opatrny, ''3D Local Algorithm for Dominating Sets of Unit Disk Graphs,'' *Adhoc Sens. Wireless Netw.*, vol. 19, no. 1/2, pp. 21-41, 2013.

[14] R. Ahlswede, N. Cai, S.Y. Li, and R. Yeung, ''Network Information Flow,'' *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204-1216, July 2000.

[15] C. Fragouli, J. Widmer, and J. Boudec, ''Efficient Broadcasting Using Network Coding,'' *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, pp. 450-463, Apr. 2008.

[16] A. Asterjadhi, E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, ''Toward Network Coding-Based Protocols for Data Broadcasting in Wireless Ad Hoc Networks,'' *IEEE Trans. Wireless Commun.*, vol. 9, no. 2, pp. 662-673, Feb. 2010.

[17] E.L. Li, R. Ramjee, M.M. Buddhikot, and S.C. Miller, ''Network Coding-Based Broadcast in Mobile Ad-Hoc Networks,'' in *Proc. IEEE INFOCOM*, 2007, pp. 1739-1747.

[18] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, ''Trading Structure for Randomness in Wireless Opportunistic Routing,'' in *Proc. ACM SIGCOMM*, 2007, pp. 169-180.

[19] D. Koutsonikolas, C. Wang, and Y. Hu, ''Ccack: Efficient Network Coding Based Opportunistic Routing Through Cumulative Coded Acknowledgments,'' in *Proc. IEEE INFOCOM*, 2010, pp. 1-9.

[20] 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 *Proc. IEEE ICC*, 2011, pp. 1-5.

[21] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, ''Xors in the Air: Practical Wireless Network Coding,'' in *Proc. ACM SIGCOMM*, 2006, pp. 243-254.

[22] T. Ho, M. Medard, R. Koetter, D.R. Karger, M. Effros, J. Shi, and B. Leong, ''A Random Linear Network Coding Approach to Multicast,'' *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413-4430, Oct. 2006.

[23] S. Sengupta, S. Rayanchu, and S. Banerjee, ''Network Coding-Aware Routing in Wireless Networks,'' *IEEE Trans. Netw.*, vol. 18, no. 4, pp. 1158-1170, Aug. 2010.

[24] Y. Wu, P.A. Chou, and S.Y. Kung, ''Minimum-Energy Multicast in Mobile Ad Hoc Networks Using Network Coding,'' *IEEE Trans. Commun.*, vol. 53, no. 11, pp. 1906-1918, Nov. 2005.

[25] J. Liu, D. Goeckel, and D. Towsley, ''Bounds on the Gain of Network Coding and Broadcasting in Wireless Networks,'' in *Proc. IEEE INFOCOM*, 2007, pp. 724-732.

[26] S. Yang and J. Wu, ''Efficient Broadcasting Using Network Coding and Directional Antennas in MANETs,'' *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 2, pp. 148-161, Feb. 2010.

[27] S. Wang, G. Tan, Y. Liu, H. Jiang, and T. He, ''Coding Opportunity Aware Backbone Metrics for Broadcast in Wireless Networks,'' in *Proc. IEEE INFOCOM*, 2013, pp. 275-279.

[28] S. Lin, J. Zhang, G. Zhou, L. Gu, T. He, and J.A. Stankovic, ''ATPC: Adaptive Transmission Power Control for Wireless Sensor Networks,'' in *Proc. Sensys*, 2006, pp. 223-236.

[29] K. Srinivasan, M. Jain, J. Choi, T. Azim, E. Kim, P. Levis, and B. Krishnamachari, ''Theκ Factor: Inferring Protocol Performance Using Inter-Link Reception Correlation,'' in *Proc. ACM MOBICOM*, 2010, pp. 317-328.

[30] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, ''Measurement-Based Models of Delivery and Interference,'' in *Proc. ACM SIGCOMM*, 2006, pp. 51-62.

[31] A. Woo and D. Culler*Evaluation of Efficient Link Reliability Estimators for Low-Power Wireless Networks*,'' EECS Dept., Unive. California, Berkeley, CA, USA, Tech. Rep., 2003.

[32] J. Macke, P. Berens, A. Ecker, A. Tolias, and M. Bethge, ''Generating Spike Trains With Specified Correlation Coefficients,'' *Neural Comput.*, vol. 21, no. 2, pp. 397-423, 2009.

Shuai Wang received the BS and MS degrees from Huazhong University of Science and Technology, China. He is currently working toward the PhD degree with the Department of Computer Science and Engineering at the University of Minnesota, Twin Cities. His research interests include wireless networks and sensors, distributed systems, network coding and related areas. He is a Student Member of the IEEE.

Guang Tan received the PhD degree in computer science from the University of Warwick, U.K., in 2007. He is currently an Associate Professor at Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences, China, where he works on the design of distributed systems and networks. From 2007 to 2010 he was a postdoctoral researcher at INRIA-Rennes, France. He has published more than 30 research articles in the areas of peer-to-peer computing, wireless sensor networks, and mobile computing. His research is sponsored by National Science Foundation of China and Chinese Academy of Sciences. He is a member of ACM and CCF. He is a member of the IEEE.

Yunhuai Liu received the PhD degree in computer science and engineering from Hong Kong University of Science and Technology in 2008. From 2008 to 2010 he worked in Hong Kong University of Science and Technology as a Research Assistant Professor. In the year 2010, he joined Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences as an Associate Professor. And in the year 2011, he joined the Research Center of Internet of Things, Third Research Institute of Ministry of Public Security located in Shanghai, China. His research interests include wireless sensor networks, cognitive radio networks, and extreme-scale datacenter and data networks. His research papers have been published in many prestigious conferences and journals such as ACM Mobicom, IEEE INFOCOM, IEEE ICDCS, IEEE ICPADS, and *IEEE TPDS, IEEE TMC*. The paper titled "Opportunity-Based Topology Control in Wireless Sensor Networks" obtained the only Best Paper Award in IEEE ICDCS 2008 (1 out of 638). He is an ACM member. He is a member of the IEEE.

Hongbo Jiang received the BS and MS degrees from Huazhong University of Science and Technology, China. He received his PhD from Case Western Reserve University in 2008. After that he joined the faculty of Huazhong University of Science and Technology as an associate professor. His research concerns computer networking, especially algorithms and architectures for high-performance networks and wireless networks. He is a member of the IEEE.

Tian He is currently an Associate Professor in the Department of Computer Science and Engineering at the University of Minnesota-Twin City. He is the author and co-author of over 150 papers in premier sensor network journals and conferences with over 11,000 citations (H-Index 43). He has received a number of research awards in the area of networking, including five best paper awards. He is also the recipient of the NSF CAREER Award 2009 and McKnight Land-Grant Professorship. Dr. He served a few program chair positions in international conferences and on many program committees, and also currently serves as an editorial board member for seven international journals including *ACM Transactions on Sensor Networks*. His research includes wireless sensor networks, cyber-physical systems, intelligent transportation systems, real-time embedded systems and distributed systems. He is a member of the IEEE.