Rate Diverse Network Coding:
Breaking the Broadcast Bottleneck

Sangki Yun
Department of Computer Science and Engineering
Korea University, Seoul, Korea
sangkiyun@gmail.com

Hyogon Kim
Department of Computer Science and Engineering
Korea University, Seoul, Korea
hyogon@korea.ac.kr

ABSTRACT

An inherent limitation of the existing digital wireless network coding is that the relay node has to settle for a single broadcast rate for the coded packet transmission. Since the selected rate should be supported on the worst quality links to the intended receivers, the throughput gain by network coding is essentially bound to the capacity of the worst link. Worse yet, the bottleneck capacity diminishes as the diversity of links increases, which generally happens when the nodes participating in network coding operation grow in number. In this paper, we solve this "broadcast bottleneck" by using a novel symbol-level network coding scheme called Rate Diverse Network Coding (RDNC). With RDNC, the relay node can deal with receivers under disparate channel conditions with a single coded data stream, eliminating the single-rate broadcast bottleneck. Through extensive simulation, we find that RDNC significantly boosts the coding gain and the throughput, more when the given topology provides richer opportunities for coding. Specifically, RDNC is as good as COPE in the worst case, but can achieve up to 2.5 times the coding gain if the network topology permits.

Categories and Subject Descriptors

C.2.2 [**Computer Systems Organization**]: Computer-Communications Networks

General Terms

Algorithms, Design, Performance, Theory

Keywords

Network coding, broadcast bottleneck, rate-diverse broadcast, wireless link quality

1. INTRODUCTION

Wireless network coding is a technique to increase the utilization of wireless links by exploiting the broadcast nature of wireless transmission. It is frequently illustrated by the Alice-and-Bob example of Fig. 1, in which Alice and Bob are exchanging data via the relay node. Traditional wireless communication would require four transmissions to exchange two crossing packets: Alice to the relay, and the relay to Bob, and *vice versa*. In the network coding approach, on the other hand, Alice and Bob temporarily store their transmitted packet, where the relay codes (*e.g.* XORs) two crossing packets before broadcasting it. Upon receiving the coded packet from the relay, Alice and Bob each recover their packets by decoding (*e.g.* XOR-ing) the received packet against the stored packet. The number of transmissions reduces to three, one less than in the traditional approach.

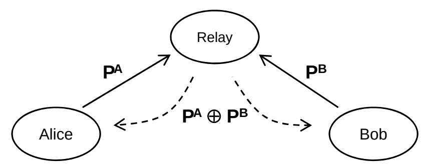


Figure 1: Network coding in Alice-and-Bob topology.

Unfortunately, however, the existing network coding approach still does not fully exploit the potential of the wireless channel. This is because the relay node has to settle for a single, low broadcast rate for the coded packet transmission, usable on the lowest quality links to the possible receivers. For instance, suppose the channel to Bob is good enough to support Quadrature-Phase Shift Keying (QPSK) modulation while that to Alice can barely sustain Binary-Phase Shift Keying (BPSK). Inevitably, the relay node has to choose BPSK, as the broadcast coded packet should be demodulated by both Alice and Bob. In essence, the current network coding approach effectively forces the throughput gain bound to the capacity of the worst link, which tends to fall with the diversity of links.

The practical performance implication of the "broadcast bottleneck" can be significant. If wireless network coding is used on the IEEE 802.11 mesh networks for instance, IEEE 802.11 a/g [1] devices can choose from eight different rates, ranging from 6Mbps to 54Mbps. Depending on the choice of the rate, the nominal throughput can diverge by a factor of 9. If the relay nodes should select a lower rate, it may cancel out substantial fraction of the throughput gain obtained by network coding.

To tackle the broadcast bottleneck problem of the wireless network coding, we propose a novel symbol-level net-

MobiHoc'10, September 20–24, 2010, Chicago, Illinois, USA.

work coding scheme coupled with modulation, called the Rate Diverse Network Coding (RDNC). In RDNC, the symbols transmitted by the encoder is differently demodulated by the receivers according to their individual channel conditions. Take Fig. 2, where again the links to Alice and Bob can sustain BPSK and QPSK, respectively. The relay node

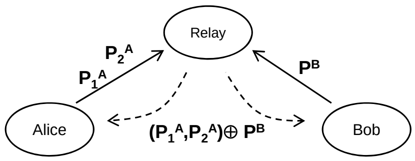


Figure 2: Rate diverse network coding.

under RDNC, unlike in the conventional network coding, codes *two* packets going to Bob with one packet going to Alice, and broadcasts it in QPSK modulation. Since a QPSK symbol can carry twice the bits of a BPSK symbol, a single QPSK-modulated packet can pair with two packets sent in BPSK. On Bob's part, it has no problem recovering the QPSK modulated symbols, so it decodes Alice's two packets by combining it with its own stored copy P_B. Namely, $[(P_1^A, P_2^A) \oplus P^B] \oplus P^B \rightarrow (P_1^A, P_2^A)$ where (P_1^A, P_2^A) represents the concatenation of two packets transmitted by Alice. On Alice's part, however, it cannot demodulate the QPSK modulated symbols of P_B as such. RDNC helps overcome this difficulty by letting Alice perform a BPSK-like demodulation with the help of the known information P_1^A and P_2^A. Compared with the traditional network coding where both Alice and Bob use BPSK, the coded packet in RDNC in this example has 50% more information.

Besides PSK, Quadrature Amplitude Modulation (QAM) can also be used with RDNC. RDNC can readily work with more than two link speeds, and with the situation where some links use PSK and others use QAM. In addition, RDNC can handle more than two receivers. In fact, RDNC in Fig. 2 yields a throughput gain only marginally higher than COPE [3] if there are no other nodes. This is because the end-to-end throughput will be throttled by the rate at which the slow Alice-Relay link provides packets. But we demonstrate below that as the network topology departs from the simplest chain and becomes richer, the benefit from RDNC is increasingly pronounced. Extensive evaluation result shows that RDNC significantly boosts the coding gain and the throughput. Depending on topologies that give opportunities for coding, RDNC can achieve up to 250% performance gain over COPE. In our arbitrary topology evaluation where 16 nodes are randomly scattered, RDNC has 36% and 77% more average throughput than COPE and the traditional unicast based data transmission, respectively.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 gives a detailed description of the design of the RDNC scheme. Section 4 describes implementation detail of RDNC. Section 5 evaluates the performance of the RDNC. Finally, Section 6 concludes the paper.

2. RELATED WORK

Network coding was originally proposed to mix multiple packets at routers to maximize the capacity of a wired network [2]. It has been successfully applied to wireless network context to improve the throughput of bidirectional traffic using overheard packets. COPE [3] is the first system architecture that shows network coding has practical gain in a real-life wireless network. It is especially salient in that it provides many functionalities to make network coding work in the IEEE 802.11-based wireless network, which are also used by RDNC. Another contribution of COPE is that it showed the throughput gain can be higher than the theoretical network coding gain. This is because the network coding mitigates the MAC layer bottleneck at the relay nodes. According to the results reported in [3], the gain reaches 3 or even 4, beyond the theoretical gain of 2. A disadvantage of COPE compared with RDNC is only one transmission rate is picked for the encoded packet broadcasting despite channel condition difference among receivers. If the rate is selected for the high channel quality receiver, some of the receivers cannot correctly receive the encoded packet, which loses the gain of network coding. Consequently, though not explicitly specified, the broadcast rate is selected as the lowest among all receivers' rates, which is inefficient.

Some recent works propose symbol level network coding performed at the physical layer [4, 5, 6, 7, 9]. Analog Network Coding (ANC) [4] is an inter-flow network coding that combines the signals of packets rather than bits. In ANC, senders transmit simultaneously, which the relay amplifies and forwards, so that the receiver extracts the packet using the mixed signal and the transmitted packet. As there is no need to separately receive packets to be mixed at the relay, it provides throughput improvement over digital network coding. It bears similarity to RDNC in that coding is done at the PHY layer, but has different objective: ANC tries to reduce the number of transmission for packet exchange while RDNC exploits the channel condition difference among receivers.

Our novel network coding method that utilizes known bit information in demodulation comes from our previous work [5]. In Zero-Cost Retransmission (ZCR), the retransmitted packet piggybacks a new packet where piggybacking is achieved through symbol-level network coding which uses higher modulation. The receiver first tries to decode the retransmitted packet through maximal ratio combining with the previously received (corrupted) packet. If the packet recovery is successful, the retransmitted packet is used to decode the piggybacked packet. With the support of the retransmitted packet, the piggybacked packet is demodulated as if it is a base-level modulated packet. RDNC applies this idea to inter-flow network coding context where each node has a different channel condition. Using the same symbol-level packet coding method, RDNC allows receivers to exploit diverse demodulation.

The problem of rate difference among receivers in network coding has been addressed in [6, 7, 8, 9]. Zhang [8] *et al.* approach the problem of having to use the lowest rate for the broadcast network coded packets from the routing angle. Instead of directly tackling the problem as RDNC does, they find the routing paths that minimize the capacity loss from asymmetric link capacity matches at relay nodes. On the other hand, Wu [6, 7] and Alimi *et al.* [9] show similar approaches to RDNC. In [6], Wu shows the gain of physical layer network coding that exploits channel condition difference, which analyzes the expansion of capacity when receivers know some messages a priori. However, it is confined to showing one coding method as an example [7]. It does not address detailed design issues to apply it to wireless networks such as packet encoding rule when multiple receivers exist, extension with opportunistic listening and protocol design that allows easy integration with existing WLAN standard. Also, its work is confined to showing its information theoretic gain, while we provide extensive evaluation results in wireless ad-hoc network and show its gain when it works over IEEE 802.11. Alimi *et al.* [9] tackles the problem by transporting different amount of information depending on the channel condition using superposition coding. This work is quite close to our approach, but superposition coding has a practicality issue unless precise power control is tightly coupled with it. As the comparison of the superposition coding and RDNC requires an involved discussion, we defer it until after we have explained sufficient details of RDNC, and revisit it in Section 3.6.

3. RATE DIVERSE NETWORK CODING

In this section, we discuss how the RDNC scheme works in detail. We start with the packet encoding and decoding methods used in RDNC, with the emphasis on how they work when receivers are under disparate channel conditions. Throughout the discussion, we use the notations summarized in Table 1. What is unique in this paper is the packet encoding function F. It is essentially bitwise XOR, but when the packets to be XORed do not have matching number of bits, the operation allows the packets to be still XORed (see Fig. 3). The bit pairing patterns in this function depends on the rate difference of the coded packets, and they will be defined below.

Table 1: Notations

Notation	Meaning
$(\cdot)^l$	low channel quality
$(\cdot)^m$	medium channel quality
$(\cdot)^h$	high channel quality
N	a node
P	a packet (destined to a node)
L	packet length (in bits)
M	modulation
$\oplus$	bitwise XOR
$\bullet$	packet concatenation
$(\cdot)^c$	coded
F	packet encoding function

The transmission rate of a link is determined by both channel coding and modulation. But in this paper, we fix the former at a single rate. Therefore, among the convolution coding rates of 1/2, 2/3, 3/4 available in the IEEE 802.11a that we assume in this paper, the 3/4 rate is used. Permitting the full degrees of freedom in both modulation and channel coding would certainly be desirable, but we stop at showing the potential of the RDNC approach in the current work. The extension to accommodate different channel coding rates will be considered in the future. The 3/4 rate convolution coding that we use has higher information rate than other rates, but has been shown to exhibit small decoding performance degradation [10]. The use of 3/4 rate in the 802.11a network means that the links can choose the transmission rate from 9, 18, and 36Mbps with BPSK, QPSK, and 16QAM, respectively.

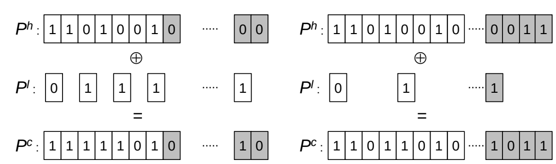
Now, we start the description of RDNC from the simple topology depicted in Fig. 2 where Alice is N^l and Bob is N^h, and extend it to the cases where the relay node communicates with more than two nodes.

3.1 Packet encoding rule

In RDNC, the relay has two tasks: one is mixing packets at the bit level and the other is modulating it for receivers with disparate decodable rates. Since symbols in different modulation methods contain different number of bits, a problem arises when we attempt to mix the bits into a single coded packet that is broadcasted to both N^h and N^l. Suppose M^l = BPSK and M^h = QPSK in Fig. 2. Choosing BPSK for the common modulation is what conventional network coding does, and we argue that it is wasteful since the high-quality link would not make the fullest use of its capacity. So let us assume that QPSK is used. Setting aside the question of how N^l can demodulate the packet as if in BPSK, consider the bit mixing operation that occurs before the modulation. In the number of QPSK symbols that N^l receives, N^h can receive twice the bits. If one-to-one packet mixing is done as in the conventional network coding, however, we would be again wasting half the capacity of the link to N^h. Therefore, we concatenate two packets destined for N^h before mixing with a single packet destined for N^l. Given the same packet length,the number of concatenated packets is proportional to the information ratio of M^h to M^l. Let k be the ratio, then the concatenated packet P^h for N^h is given as

$$P^h = P_1^h \bullet P_2^h \bullet \cdots \bullet P_k^h, \tag{1}$$

where P_i^h is a native packet headed for N^h. As a consequence, the length of P^h is $k \times L$. Note that we are not inflating the coded packet any more than the conventional network coding since the number of symbols in both schemes are the same.



(a) $k = 2$ (b) $k = 4$

Figure 3: Packet encoding examples.

The RDNC packet mixing essentially uses the bitwise XOR as in COPE [3]. But since P^l and P^h can have different number of bits, we need to redefine the encoding function $F(P^h, P^l, k)$ to substitute for $\oplus$. $F(P^h, P^l, k)$ mixes k bits from P^h with 1 bits from P^l. Fig. 3(a) shows an example for $k = 2$, which corresponds to the BPSK-QPSK and the QPSK-16QAM configurations. In this case, every odd bit of P^h is XOR-ed with a bit of P^l, but the even bits of P^h are left intact. Likewise, for $k = 4$, every fourth coded bit is an XOR-ed one but the other 3 bits are native from P^h, as shown in Fig. 3(b). This is to allow N^l to recover P^l after receiving the coded transmission of L symbols in M^h.

Using its stored packets, N^l eliminates the redundancy from the symbols, which leads to the desired "modulation reduction." So it recovers L bits (*i.e.*, P^l) from the M^h-modulated symbols, where the remaining L bits from the symbols are discarded. The pseudocode for the encoding function appears in Algorithm 1. It is general for any combination of modulations.

Algorithm 1 Encoding function $P^c = F(P^h, P^l, k)$

```
for i = 1 to L do
    for j = k-1 to 0 do
        if j=k-1 then
            P^c[i*k - j] = P^l[i] ⊕ P^h[i*k - j]
            //Here, P[x] = x^th bit of P.
        else
            P^c[i*k - j] = P^h[i*k - j]
        end if
    end for
end for
P^c
```

3.2 BPSK-QPSK case

We start with the case of Fig. 2 where Alice (N^l) uses M^l=BPSK while Bob (N^h) can support M^h= QPSK.

3.2.1 At the sender

Once the relay digitally mixes the packets into P^c as above, it next modulates the coded packet. Among the used modulation schemes, we always choose M^h to carry P^c. In this case M^h is QPSK. So two bits b^c_{2i-1} and b^c_{2i} are modulated together in a single symbol.

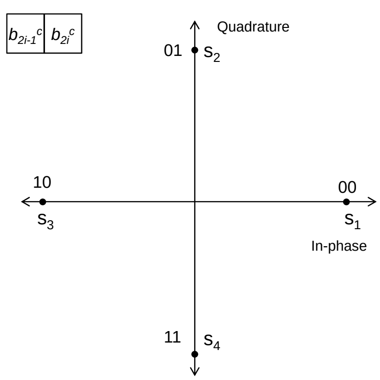


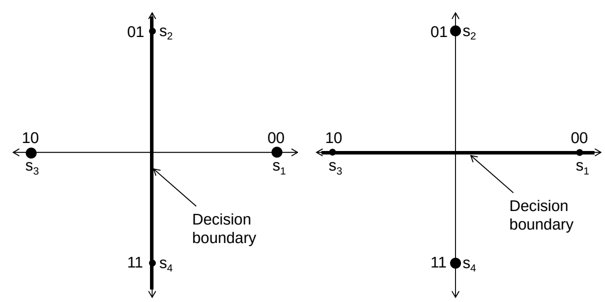
Figure 4: Signal space representation of the RDNC modulated symbol for BPSK-QPSK encoded packet.

Fig. 4 represents the QPSK modulated symbols in the signal space. To each constellation point, the two bits (e.g. "00") are the bits mapped, where the left and right bits are respectively given by b^c_{2i-1} and b^c_{2i}. Note that bits-to-constellation mapping of the coded bits should be done exactly as shown in Fig. 4. This mapping is done for each bit of P^c. Finally, the modulated symbols are up-converted to the carrier frequency and broadcasted to N^l and N^h.

3.2.2 At the receiver

The decoding process at N^h that can use QPSK is straightforward. N^h demodulates the received symbols of P^c, and XORs it with the stored packet P^h on the odd bit positions. For even bit positions, it simply takes the value as such. Thus the two packets from N^l (P^h_1 and P^h_2) are decoded.

On the other hand, N^l can demodulate only BPSK. In order to demodulate the QPSK symbols in P^c, it has to go through a different process. Essentially, N^l exploits the even bits of P^c (namely $b^c_{2i}, 0 < i \leq L$) that it already knows from P^h. Suppose b^c_{2i} is known to be 0. Then N^l recognizes that the transmitted symbol is either s_1 ("1**0**") or s_3 ("0**0**"), ruling out s_2 and s_4. So the decision process can focus on the determination of the first bit, as if this was BPSK. The situation is depicted in Fig. 5(a). Similarly, it chooses between s_2 and s_4 if the given bit b^c_{2i} is 1. As a result, the demodulation performance and the amount of information per symbol becomes identical to those of BPSK.



(a) $b^c_{2i} = 0$ (b) $b^c_{2i} = 1$

Figure 5: Signal space and decision boundary with prior knowledge.

The RDNC demodulation rule for N^l in the BPSK-QPSK case is written as follows. If the received symbol is r, we estimate it to be s_i where

$$\hat{s}_i = \begin{cases} \arg \min\limits_{s \in \{s_1, s_3\}} |r - s|^2 & \text{if } b^c_{2i} = 0, \\ \arg \min\limits_{s \in \{s_2, s_4\}} |r - s|^2 & \text{if } b^c_{2i} = 1. \end{cases}$$

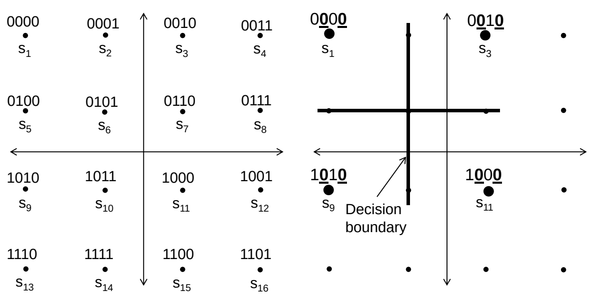
After the demodulation, N^l XORs b^c_{2i-1} with b^l_{2i-1} for each i (the even bits b^c_{2i} is dropped as it is useless now), and obtains P^l.

3.3 B(Q)PSK-16QAM case

Now, we go up one notch in the modulation hierarchy and consider the case where N^l uses BPSK or QPSK and N^h uses 16QAM. In the BPSK-to-QPSK case above, it is evident that N^l gets the equivalent demodulation performance compared with the pure BPSK case. Unfortunately in B(Q)PSK-to-16QAM case, however, N^l experiences slight loss in demodulation performance due to the different ways the constellation points are set in PSK and QAM. Let us first see how QPSK-to-16QAM case works. In this case, the information ratio of 16QAM to QPSK is two, so the packet encoding method at the relay is the same as in the BPSK-QPSK case ($F(P^h, P^l, k)$). The only difference is that N^l uses the prior knowledge of two bits b^c_{4i-2} and b^c_{4i} among 4 bits b^c_{4i-3}, b^c_{4i-2}, b^c_{4i-1} and b^c_{4i} modulated in the same symbol.

Fig. 6(a) is the constellation for 16QAM, and 6(b) is the RDNC-QPSK after the reduction by knowledge of two even bits from the packets kept at the receiver. The known bits information reduces the candidates from 16 points to 4 points marked with thick dots. The surviving candidates after the elimination are (0**000**, 0**010**, 1**010**, 1**000**) in the signal space, and correspondingly the decision boundaries form as if this is QPSK demodulation.

The BPSK-16QAM case is not much different from the QPSK-16QAM except that $k = 4$ in F and N^l removes 14 constellation points using the three known bits b^c_{4i-2}, b^c_{4i-1}, and b^c_{4i} in the demodulation process. For instance, if the known three bits are "001," only s_2 and s_{12} are the points to be considered, with the decision boundary demarcated down the equidistant positions from the two points.



(a) (b)

Figure 6: 16QAM constellation and an example of the effect of modulation reduction.

Although through the modulation reduction the *number* of constellation points to consider is reduced to that of B(Q)PSK, the *distance* among the points in the signal space gets smaller. The consequence is that the decoding probability for N^l gets smaller than in the original B(Q)PSK. Fortunately, however, the degradation is not substantial. Specifically, the distances between constellation points after the modulation reduction are approximately 88% of that in B(Q)PSK. We show this in Fig. 7 with QPSK-16QAM example. In the figure, the hollow dots represent the 16QAM constellation points, and the filled dots are those of QPSK. The dashed circle represents the feasible locations for PSK symbols. Let us call the symbol that represents the bits mixed from the previously QPSK and 16QAM modulated symbols and is demodulated on QPSK level, the "RDNC-QPSK" symbol. We can compare the decoding performance of QPSK and RDNC-QPSK as follows. The performance of a given modulation method is determined by the minimum Euclidean distance between its constellation points. Suppose the mean energy associated with a single symbol is E_s. Since PSK is an equal energy modulation, all constellation points exist on a circle with the radius of $\sqrt{E_s}$. On the other hand, since QAM utilizes the amplitude in modulation, symbols can have different energy levels. If the average energy per symbol is E_s and the constellation points have an equal probability, the point farthest from the origin has approximately $1.32\sqrt{E_s}$ energy. Now, the minimum distance of QPSK constellation points is $\sqrt{2E_s} \approx 1.414\sqrt{E_s}$, and that of RDNC-QPSK is

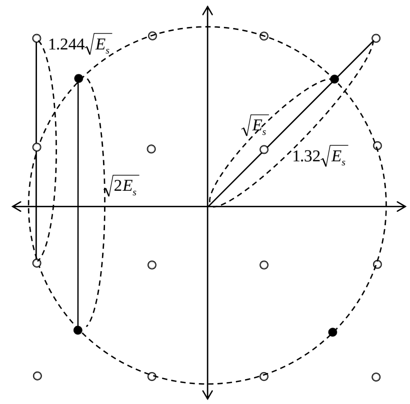


Figure 7: Comparison of the minimum distances between constellation points in QPSK and RDNC QPSK.

$2/3 \times 1.32\sqrt{2E_s} \approx 1.244\sqrt{E_s}$. So the minimum distance of RDNC-QPSK is 88% of that of the QPSK. So is RDNC-BPSK.

In order to see the impact of this reduction on the demodulation performance, we conducted simulation experiments. Fig. 8 shows that there is approximately 1dB loss in the SNR under the AWGN channel.

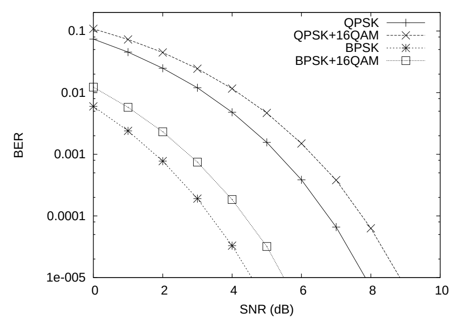


Figure 8: BER performance loss due to constellation contraction.

3.4 RDNC with opportunistic listening

So far, we have discussed RDNC with two receivers (Alice and Bob) for ease of explanation. But in practice, using RDNC would yield only marginal throughput improvement over COPE in the Alice-and-Bob type of topologies. This is because the slower Alice-Relay link is less likely to provide enough packets to enable RDNC to mix multiple such packets with a packet from Bob. But if RDNC works with more than two receivers where the receivers allow the use of the opportunistic listening, RDNC can yield much higher coding gain. In this section, we explain how RDNC works with multiple receivers that can overhear other nodes' transmission.

To begin with, we take a simple example of network coding under opportunistic listening. Fig. 9(a) depicts the situation where N_1 is sending to N_5, and N_4 to N_2 (solid arrows). We assume N_2 and N_5 can overhear N_1 and N_4, respectively (dashed arrows). So when the relay node mixes P_1 and P_4, N_2 and N_5 can decode the coded packet with the assistance

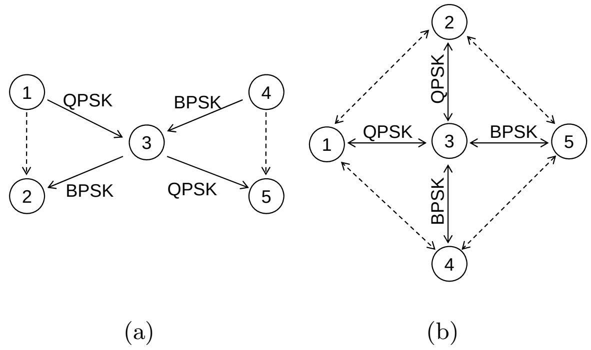


Figure 9: Example topologies with opportunistic listening.

of overheard packets. Now, RDNC enables network coding to be used even when the links use different modulations, as marked on the links in Fig. 9(a). Suppose N_5 can accept QPSK-modulated packets, but N_2 can do only BPSK. Then the two packets going to the N_5 are concatenated and coded with a packet destined to N_2, where we have $P^c = F(P_1^{N_1} \bullet P_2^{N_1}, P^{N_4}, 2)$. One caveat here is that N_1 should select the rate that the overhearing nodes can accept. For instance, if N_2 cannot accept QPSK-modulated packets from N_1, N_1 should use BPSK modulation. But this restriction is not specific to RDNC; it also applies to COPE.

Fig. 9(b) depicts a situation where N_1 and N_2 respectively exchange packets with N_5 and N_4. Each node can overhear others except the one in the opposite end. In COPE, N_3 can code the packets from the neighbors ($P_1 \oplus P_2 \oplus P_3 \oplus P_4$) in the lowest rate of the four links. The receivers can decode the packet using the copy of its previous transmission and the overheard packets. In case there is difference in channel quality as in Fig. 9(b), however, RDNC can help avoid the broadcast bottleneck. The encoder first concatenates the packets to send to N_1 and N_2, respectively. Then it XORs them: $((P_1^{N_4} \bullet P_2^{N_4}) \oplus (P_1^{N_5} \bullet P_2^{N_5}))$. The encoder also XORs the packets to send to bad channel receivers: $P^{N_1} \oplus P^{N_2}$. Finally, RDNC encoding is applied, so that the coded packet becomes

$$P^c = F((P_1^{N_4} \bullet P_2^{N_4}) \oplus (P_1^{N_5} \bullet P_2^{N_5}), P^{N_1} \oplus P^{N_2}, 2).$$

Note in this example that the number of native packets coded in P^c is six, 1.5 times that of COPE.

3.5 BPSK-QPSK-16QAM case

Now, we come back to the RDNC operation in face of more than two modulations used by links. For simplicity, we assume there is a single node for each modulation level, denoted by N^h, N^m, and N^l (so step 2 is omitted from the aforementioned workflow). As discussed above, we require that all nodes can exploit the opportunistic routing, *i.e.*, each overhears all the packets used for encoding except the packets destined to it. Then, P^c is iteratively encoded as

$$P^c = F(F(P^h, P^l, 4), P^m, 2), \tag{2}$$

where $P^h = P_1^h \bullet P_2^h \bullet P_3^h \bullet P_4^h$ and $P^m = P_1^m \bullet P_2^m$. Bits in P_c are consequently computed as follows.

$$\begin{aligned} b_{4i-3}^c &= b_{4i-3}^h \oplus b_{2i-1}^m \oplus b_i^l, \quad 0 < i \le L \\ b_{4i-2}^c &= b_{4i-2}^h, \quad 0 < i \le L \\ b_{4i-1}^c &= b_{4i-1}^h \oplus b_{2i}^m, \quad 0 < i \le L \\ b_{4i}^c &= b_{4i}^h, \quad 0 < i \le L, \end{aligned}$$

where b_{4i-3}^c, b_{4i-2}^c, b_{4i-1}^c and b_{4i}^c are four bits modulated in a 16QAM symbol. The bits-to-constellation mapping for RDNC is as shown in Fig. 6. Of the four bits in each symbol, N^l knows three bits, and N^m knows two bits. Using the known information either from its own stored packet(s) or from overhearing, each node can demodulate the 16QAM symbol.

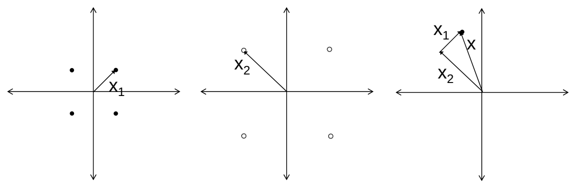
3.6 RDNC vs. superposition coding

Superposition coding (SC) was proposed to increase the capacity of the multi-user downlink, but in the broadcast context it can also be used to transfer different amount of information depending on the receiver's channel condition [12]. In the multi-rate network coding context, it can be also used to differentiate the rate of the receivers [9] as RDNC does. For example, suppose a sender should send a symbol x_1 to node N_1, and both x_1 and x_2 to N_2, where N_1 is in a better channel condition. Instead of separately sending x_1 and x_2 to the receivers, the sender codes them together and sends them in one broadcast transmission. Namely, it broadcasts

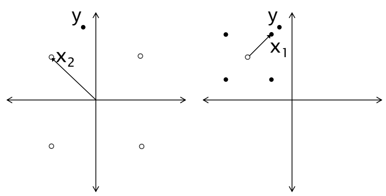
$$x = x_1 +_s x_2,$$

where $+_s$ denotes the superposition operation, which is simply the vector addition in the signal space (Fig. 10(a)). When N_2 receives x, it decodes x_2, regarding x_1 as interference. On the other hand, N_1 performs the successive interference cancelation (SIC) to recover both symbols. It first decodes x_2 regarding x_1 as noise, and then it subtracts x_2 from x to obtain x_1, eventually gets both x_1 and x_2. Fig. 10 shows this process of QPSK as an example. N_1 which receives y in Fig. 10(b) first decodes x_2, subtracts it from y and estimates the second symbol. As the subtracted value is closer to x_1 than the other three constellation points, it is estimated as x_1.

Information-theoretically, SC increases the capacity, compared with the separate transmissions of symbols [12]. But in real situations, it is not easy to apply SC because the effect of the noise due to superposition is not negligible to the nodes with bad channel condition. Fig. 10 shows this fact intuitively. In the figure, we notice that the symbol x would be closer to the decision boundaries for x_2 when decoded (y is the received symbol corresponding to x). If N_2 is in a channel condition to barely support QPSK, the superpositioned symbol can make the decoding difficult. In order to confirm this, we conducted a simulation study for the SC under AWGN channel. Here, we assume asymmetric channel condition such as in Fig. 2, and the relay encodes different number of bits per symbol using SC rather than RDNC, as in iPack [9]. Fig. 11 shows the demodulation performance results when the base modulation is QPSK. We easily notice the performance difference between RDNC-QPSK and SC-QPSK. For instance, given the SNR that would map to a BER of 10^{-3} in QPSK, the RDNC-QPSK degrades by only 1 dB, but the SC-based QPSK degenerates by as much as 6 dBs. The significant gap stems from the fact that RDNC



(a) Encoding



(b) Decoding

Figure 10: Superposition coding - QPSK example.

rules out constellation points using the known bits, but SC is subject to the noise from the superposition.

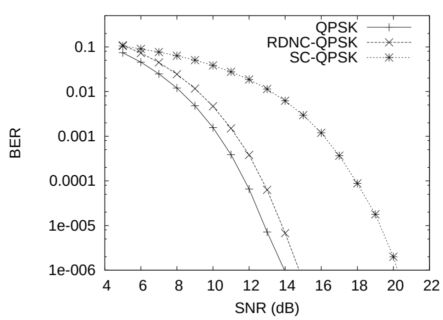


Figure 11: Comparison of QPSK, RDNC-QPSK and SC-QPSK under AWGN channel.

4. IMPLEMENTATION DETAILS

Since RDNC requires the cross-layer operation with the physical layer, we need to modify the PHY and MAC layers. In this section, we explain the implementation detail of RDNC assuming that the IEEE 802.11a PHY/MAC is used. We assume that the most basic functionalities of traditional network coding are available, as in COPE. Specifically, we use pseudo-broadcasting, opportunistic listening, reception report, maintaining packet pool and asynchronous ACK as defined in COPE.

4.1 Frame format

In order to implement RDNC, we need a RDNC header to identify the intended receivers of the coded packet and the modulation information. We insert it between the 802.11 PLCP header and the MAC header so that it is transmitted in a basic rate regardless of the receiver's channel condition, and is understood by every receiver. Fig. 12 shows the 802.11 PHY layer frame structure with the RDNC shim header. Since RDNC modulates the coded packet according to the channel condition of N^h, the rate field in the PLCP header has the rate for N^h. On the other hand, the rate information of N^l and N^m go into the `Low rate` and `Med. rate` fields of the RDNC shim header. The `Nexthop` and the `Packet ID` fields are repeated as many times as there are receivers. Note that these two fields have the same role with the same fields in COPE header, so the values in them are also generated in the same way. The identities of N^h's come first, followed by those of N^m's and N^l's. Exploiting COPE, the sender needs to insert the COPE header into the transmitted packet when it use asynchronous ACK or reception report. We assume the COPE header is included in the RDNC header with the same format in COPE.

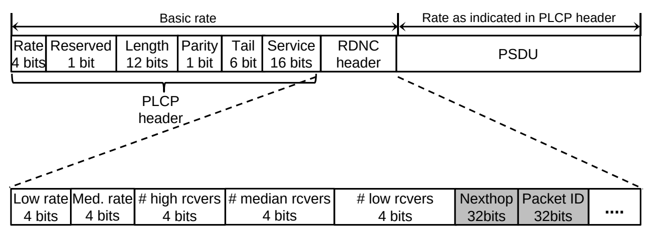


Figure 12: 802.11 physical layer frame format with RDNC shim header.

4.2 Interaction with the Physical layer operation

Assuming that IEEE 802.11a is used as a PHY/MAC protocol, a challenge is how to interact with the PHY layer operation. Especially, channel coding makes it difficult to prepare known bits for demodulation at the receiver side. Even though the receiver knows some bits of the encoded packet a priori, it is difficult to expect how the encoded packet is transformed after channel coding, which makes the known information useless. Therefore, RDNC encoding should be performed after the native packets are separately channel coded. If RDNC encoding is performed after channel coding, receiver can expect the RDNC encoded bits by performing channel coding to the packets used for RDNC decoding.

Considering this, RDNC encoding and decoding procedures that interact with the 802.11a PHY layer operation are described as follows. Similar to COPE, the sender first decides which packets to encode by examining the nexthop of the packets and reception reports from the receivers, in order to guarantee the decodability of the receivers. If a receiver is determined to be included in the N^h or N^m group, more packets destined to it are extracted from the packet queue and concatenated, as discussed in Section 3. After that, the sender separately performs scrambling, convolutional coding and interleaving to those packets, and encodes them. Finally, RDNC header is generated and the encoded packet is transmitted. At the receiver side, the receiver first demodulates PLCP and RDNC headers and checks the `Nexthop` field to see if it is one of the intended receivers and if so, which group it belongs to (*i.e.*, N^l or N^h). Once the check is done, it sees if it has enough packets in the packet pool with which to decode P^c by examining the `Nexthop` and the `Packet ID` fields. If the receiver belongs to either N^l or N^m, it prepares the known bits by performing scrambling, convolutional coding and interleaving to the packets for decoding, and demodulates the encoded packet with the support of them. Finally, after the demodulation, the receiver decodes the packet it wants by XORing the P^c with the packets it has kept.

4.3 Rate Selection

One of the tasks a RDNC encoder should perform is to choose the rates for adjacent links. Specifically, it may need to determine M^h and M^l (and M^m in case three rates are to be used) that fit the given links best. Although the rate adaptation in RDNC is an interesting subject, it is beyond the scope of the current work. In this paper, we simply assume that the sender has an SNR threshold table that tells the best rate for each link. Note that if 16-QAM is involved in the encoding, the decoding probability of the low quality channel receiver decreases, as discussed in Section 3.3. Therefore, in such a case the selection of the rate should be more conservative, which makes it difficult to design of the rate selection algorithm. In this paper, we leave it as a future work and assume that the relay has two different SNR thresholds for each rate depending on the involvement of 16-QAM modulation.

5. PERFORMANCE EVALUATION

In this section, we conduct a comparative study of the RDNC performance using extensive simulation. We compare the throughput and the coding gain with those of no network coding case ("no NC") and of COPE. A slight difference between COPE in [3] and what we compare is that for fair comparison the latter also opportunistically uses higher rates rather than keeping to the basic rate. For more controlled experiments, we start with a few special topologies: Alice-and-Bob, Y, backhaul, and dumbbell. They are shown in Fig. 13. The solid arrows mean good quality links, whereas dashed arrows represent bad quality links.

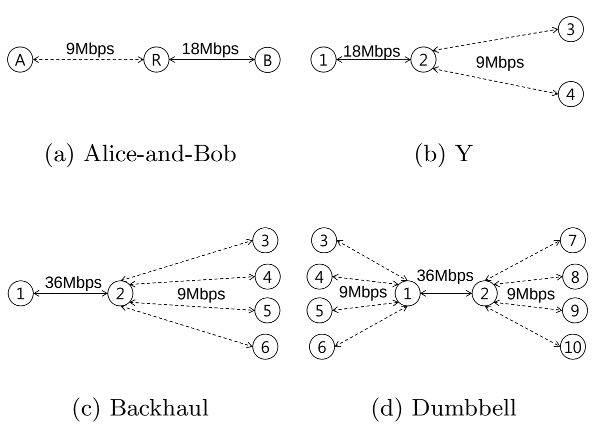


Figure 13: Tested topologies.

In the first set of experiments, we create flows in both directions on the given topologies, so the coding opportunity occurs at the intersections. The flows are constant bit rate (CBR) traffic encapsulated in UDP/IP, where the interpacket gap and the packet size is 1ms and 1500 bytes, respectively. We used the Qualnet 4.5 simulator, and modified its IEEE 802.11a PHY/MAC for RDNC and COPE simulations. RDNC is implemented as described in Section 4 and all additional header and message overheads are reflected in the simulation. Additionally, we used MATLAB as the symbol level coding is difficult to simulate in Qualnet. Using MATLAB, we first generated the SNR vs. BER table for RDNC modulations, which Qualnet uses. For this purpose, we modified the MATALB 802.11a PHY simulator provided in [10] to reflect the OFDM structure, channel coding, and multipath fading on the simulation. In this simulator, the effect of multipath fading is controlled by the root mean square (rms) value of the delay spread parameter, which we set it to 5 nanoseconds. Based on the result from the simulator, the performance degradation of RDNC modulation when channel coding and multipath fading effect is considered is similar to AWGN channel evaluation result in Fig. 8. Compared with normal modulation, RDNC modulation has approximately 1dB SNR loss across all measured SNR ranges. In the QualNet simulator, SNR is calculated according to the distance between the sender and the receiver, where the transmission power, thermal noise and path loss exponent is set to 20dBm, -93dBm and 4, respectively. Throughout simulation, RTS/CTS and frame bursting are turned off, and each simulation run simulates 100 seconds of the given system operation.

5.1 Special topologies

Fig. 14 compares the aggregate throughput in the special topologies in Fig. 13, where all links are assumed to be symmetric and all traffic flows are bidirectional. Fig. 15 shows the distribution of the number of packets mixed into the coded packets at the relay nodes.

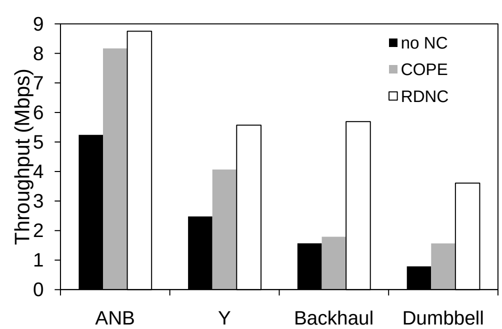
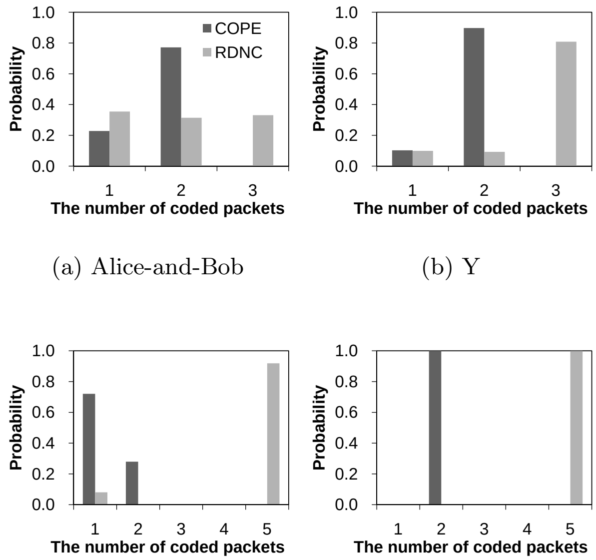


Figure 14: Throughput comparison.

In Alice-and-Bob topology, the throughput gains of COPE and RDNC are 1.56 and 1.67, respectively. Essentially, RDNC does not show great improvement over COPE in the topology. This is because RDNC is not given enough chances for coding. In order for RDNC encoding to happen, the relay node should receive twice as many packets. However, the IEEE 802.11 MAC gives the nodes equal chances for transmission given the same traffic load at the two senders, and only occasional randomness in packet arrivals allows the RDNC encoding to happen.

Both COPE and RDNC fall short of the coding gain of 2 as suggested by [3]. In COPE the reason is claimed to be the COPE header overhead, but we observe that it is



(a) Alice-and-Bob (b) Y

(c) Backhaul (d) Dumbbell

Figure 15: Comparison of coding patterns.

due to the shortage of coding opportunities. For network coding to happen, the relay node should have packets from both Alice and Bob. But depending on the packet arrival pattern there are cases where the condition does not hold. From Fig. 15(a), it is observed that approximately 20% of the packets are transmitted uncoded in COPE. RDNC has even more uncoded packets than COPE. This is because RDNC consumes more packets in the transmission queue since it uses maximally three packets in each coding opportunity. This ironically reduces the number of coding opportunities for RDNC, which is corroborated by Fig. 15(a) where more than 30% of packets are transmitted uncoded. Consequently, the performance gap between RDNC and COPE is small.

Given the topology, we added another evaluation that intentionally gives coding opportunity for RDNC. We control the transmission interval of the CBR traffic so that the ratio of packets from A to those from B in the queue of R becomes approximately 2. In this evaluation, the gain of RDNC from COPE is 1.34, which is less than our expectation, 1.5. Though an RDNC encoded packet has 50% more information than COPE's encoded packet, the uncoded packet transmitted from R to B is transmitted by 18Mbps rate, which reduces the gain by RDNC encoding.

In the Y topology, RDNC begins to exhibit a visible throughput difference. After all senders have a transmission opportunity, the relay node gets to have three packets in the queue. With COPE, it can code only two packets. But RDNC can code three packets, so the theoretical throughput is 1.5 times that of CODE. Fig. 15(b) confirms that most coded packets in RDNC uses three packets in packet mixing. Note that in this evaluation, COPE and the uncoded transmission drop packets in the node 4's queue due to the limited draining rate at the relay, while we intentionally avoid that situation in the previous evaluation.

In the backhaul topology, we can explore the potential of RDNC in faster modulations such as 16QAM. The relay node that uses RDNC can put at most 5 packets in one coded packet, thus yields up to 2.5 times the throughput than COPE. But the COPE result is unexpectedly small, being only marginally better than the no network coding case. This is because only two packets are coded, leaving three others uncoded, as Fig. 15(c) shows. In RDNC, in most cases five packets are coded together, leading to much higher throughput. Although the given rate assignments for the links are purposely set to show off the prowess of RDNC, it clearly illustrates the limitation of COPE that RDNC overcomes.

In Dumbbell topology, both COPE and RDNC approach the maximum coding gain of two and five, respectively. This is evident from the coding pattern in Fig. 15(d). The improved coding gain of COPE stems from the symmetric nature of the traffic. For the intersection nodes 1 and 2, the same number of packets from the two directions are matched. But the match is achieved at the congestion of the intersection nodes since it cannot clear the packets fast enough, since it has to use the lowest rate of all links, 9Mbps, for the coded broadcast. As a consequence of ill using the fast center link, the throughput is not high even though the coding gain is pulled up to the maximum. For RDNC, the much higher coding gain helps cope with the congestion better, and a significant throughput gain over COPE is again obtained.

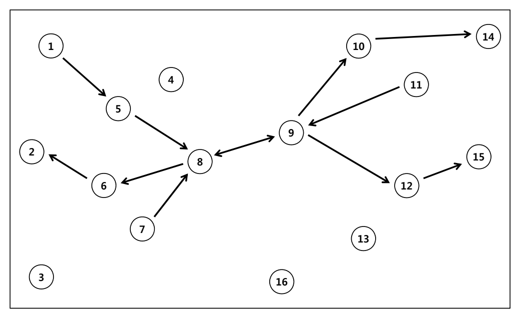


Figure 16: An example of the arbitrary topology.

5.2 Arbitrary topology

Finally, we experiment with a large number of arbitrary topologies as exemplified in Fig. 16, where the arrows represent the direction of the flows. The figure shows there are three flows $1 \rightarrow 15$, $11 \rightarrow 2$ and $7 \rightarrow 14$ for example. We randomly scattered 16 nodes in a 900×1300 m^2 rectangle. All links are lossy, and experience up to 10% packet error rate (PER). We observe the distribution of transmission rates of the links to be: 42% are 9Mbps, 35% are 18Mbps, and 21% are 36Mbps. In the topology, we varied the number of flows from 2 to 6, and measure the performance with no NC, COPE, and RDNC.

Fig. 18 shows the average throughput from eight simulation runs for each tested number of flows created between random source-destination pairs, where error bars represent 95% confidence interval. The throughput for no NC, COPE, and RDNC averaged over the number of flows are 2.35Mbps, 3.05Mbps, and 4.17Mbps, respectively. The reason of high variance in the result is because each run selects different source-destination pairs. We notice that as the number of flows increases, the throughput generally decreases. This is because the intermediate nodes that cross the flows are subject to increasingly severe bottleneck. But in case of RDNC it can code more packets than COPE, so its throughput performance does not significantly drop even if the number of flows increases. The reason that RDNC shows comparable performance with COPE in the two flows case is that the opportunities for coding more packets than COPE is rare. But as the number of flows increases and the bottleneck at the intermediate nodes becomes significant, RDNC and COPE diverge.

Fig. 18 compares the coding gain of COPE and RDNC for the given number of flows. Again, when the number of flows is two, RDNC is similar to COPE. But as it increases, the coding gain difference becomes wider. When we allow link quality diversity, the coding gain of COPE begins to descend early, whereas RDNC shows no sign of coming down up to six flows. It clearly demonstrates the robustness of RDNC in face of varying channel qualities, which are more common than not in real-life networks.

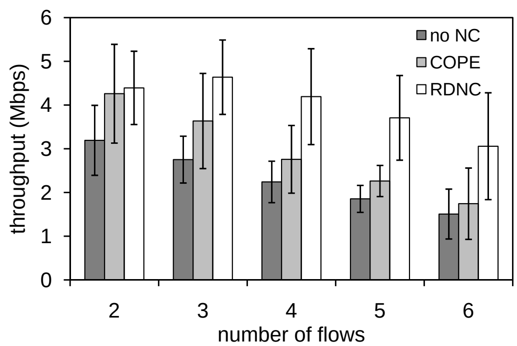


Figure 17: Throughput comparison in the arbitrary topology.

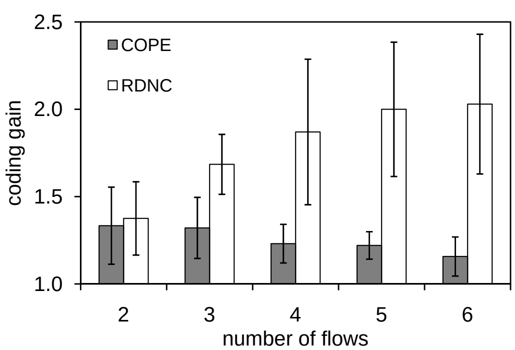


Figure 18: Coding gain comparison in the arbitrary topology.

6. CONCLUSION

In this paper, we show that the gain from the existing network coding approach quickly degenerates to a marginal number in face of diverse channel qualities, not much better than in the no coding case. It is due to what we call the broadcast bottleneck, where the coded broadcast must use the lowest rate of all links involved in the network coding operation. To solve this problem, we propose a novel symbol level network coding scheme that allows the higher quality links to receive the coded broadcast at their individual rates instead of the least common rate. The core idea is to let the low channel quality receivers use their previous transmission in demodulating the high rate coded broadcast. Through extensive simulations over various topologies, we demonstrate that the proposed scheme is highly robust against channel quality variations, and well outperforms the existing network coding approach. With 16QAM-QPSK-BPSK hierarchy, for instance, the proposed scheme can achieve up to 250% coding gain over COPE. In future work, we will extend the scheme so that it works under different channel coding rates as well. It will enable the entire spectrum of transmission rates to be used in coding, and completely eliminate the broadcast bottleneck.

Acknowledgement

This work was supported in part by Mid-career Researcher Program through NRF grant funded by the MEST (No.2009-0080413), and the Ubiquitous Computing and Network (UCN) Project, the Ministry of Knowledge and Economy(MKE) Knowledge and Economy Frontier R&D Program in Korea as a result of UCN's subproject 10C2-C1-10M.

7. REFERENCES

[1] IEEE 802.11 WG, *IEEE 802.11g-2003: Further Higher Data Rate Extension in the 2.4 GHz Band*, IEEE 802.11g-2003, 2003.

[2] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network Information Flow," *IEEE Transactions on Information Theory*, pp. 1204–1216, July 2000.

[3] S. Katti, H. Rahul, W. Hu, D. Katabi, and M. Medard, and J. Crowcroft, "XORs in the Air: Practical Wireless Network Coding," in proceedings of ACM SIGCOMM, 2006.

[4] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: Analog network coding,", in proceedings of ACM SIGCOMM, 2007

[5] S. Yun, H. Kim "Towards Zero-Cost Retransmission through Physical-Layer Network Coding in Wireless Networks," ACM SIGCOMM (poster abstract), 2008

[6] Yunnan Wu, "Broadcasting when receivers know some priori," in proceedings of International Symposium of Information Theory, IEEE, 2007.

[7] Yunnan Wu, "Network Coding for Wireless Networks," Technical Report, MSR-TR-2007-90.

[8] J. Zhang, Q. Zhang "Cooperative Network Coding-Aware Routing for Multi-Rate Wireless Networks," in proceedings of IEEE INFOCOM, 2009.

[9] R. Alimi, Li Li, R. Ramjee, H. Viswanathan, Y. Yang, "iPack: in-Network Packet Mixing for High Throughput Wireless Mesh Networks," IEEE INFOCOM 2008.

[10] O. Awoniyi and F. A. Tobagi, "Packet Error Rate in OFDM-based Wireless LANs Operating in Frequency Selective Channels," in proceedings of IEEE INFOCOM, 2006.

[11] J. Heiskala and J. Terry, *OFDM Wireless LANs: A Theoretical and Practical Guide*, SAMS.

[12] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge University Press, 2005.

[13] R. Draves, J. Padhey, B. Zill "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," in proceedings of ACM MOBICOM, 2004.