This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

Network-Coded Multiple Access

Lu Lu, *Member, IEEE,* Lizhao You, *Student Member, IEEE,* and Soung Chang Liew, *Fellow, IEEE*

Abstract—This paper proposes and experimentally demonstrates a first wireless local area network (WLAN) system that jointly exploits physical-layer network coding (PNC) and multiuser decoding (MUD) to boost system throughput. We refer to this multiple access mode as Network-Coded Multiple Access (NCMA). Prior studies on PNC mostly focused on relay networks. NCMA is the first realized multiple access scheme that establishes the usefulness of PNC in a *non-relay setting*. NCMA allows multiple nodes to transmit simultaneously to the access point (AP) to boost throughput. In the non-relay setting, when two nodes A and B transmit to the AP simultaneously, the AP aims to obtain both packet A and packet B rather than their network-coded packet. An interesting question is whether network coding, specifically PNC which extracts packet $A \oplus B$, can still be useful in such a setting. We provide an affirmative answer to this question with a novel two-layer decoding approach amenable to real-time implementation. Our USRP prototype indicates that NCMA can boost throughput by 100% in the medium-high SNR regime ($\geq$10dB). We believe further throughput enhancement is possible by allowing more than two users to transmit together.

Index Terms—network coding, physical-layer network coding, multi-user detection, multiple access, implementation

1 Introduction

This paper proposes and experimentally demonstrates a first wireless local area network (WLAN) system that jointly exploits physical-layer network coding (PNC) [1] and multiuser decoding (MUD) [2] to boost system throughput. We refer to this multiple access mode as Network-Coded Multiple Access (NCMA).

Since its introduction in [1], [3], PNC has developed into a subfield of network coding with a wide following [4], [5]. Nearly all prior studies of PNC, however, focused on *relay* networks. NCMA, as expounded in this paper, is the first realized multiple access scheme that establishes the usefulness of PNC in a *non-relay setting*.

To put things in context, let us first review the application of PNC in a two-way relay network (TWRN), as shown in Fig. 1(a). Here, two nodes A and B wish to send messages to each other via a relay R. With PNC, nodes A and B send their packets, A and B, to relay R simultaneously. Relay R then derives a network-coded packet (e.g., a bit-wise XOR packet $A \oplus B$) from the received overlapping signals. It then broadcasts $A \oplus B$ to nodes A and B. With $A \oplus B$, node A recovers packet B using self-information: $B = A \oplus (A \oplus B)$; likewise for node B. In this way, only two time slots are needed for the two-packet exchange, and 100% throughput improvement can be achieved with respect to the 4-time-slot traditional relaying scheme [1].

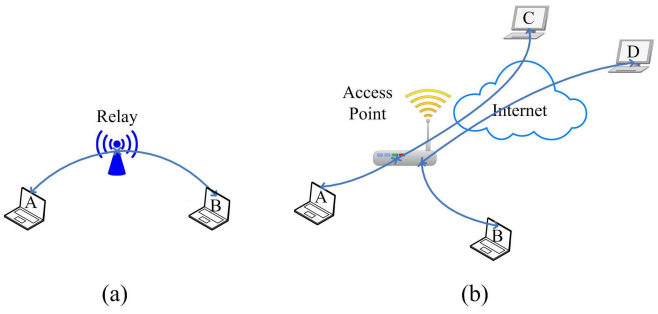

Fig. 1. Traffic patterns of (a) a two-way relay network; (b) a multiple-access WLAN.

In a WLAN setting, as shown in Fig. 1(b), instead of a relay, we have an access point (AP). Oftentimes, nodes A and B are not interested in communicating with each other. Rather, they want to communicate with some other nodes, say in the Internet, as shown in Fig. 1(b). In this case, the communicating counterpart of A is C, not B. For the AP to relay packet $A \oplus B$ to node C is not useful because C lacks packet B. For compatibility with the legacy Internet, the AP has to forward the individual messages of A and B to their respective destinations. As far as the communication within the WLAN is concerned, we have a *non-relay setting* in which the AP serves as the termination point for the messages from A and B, in that the AP's ultimate goal is to derive these individual messages of A and B explicitly rather than their network-coded message. In particular, unlike the previous PNC setting [1], the goal is not to relay messages between nodes A and B.

An interesting question is whether packet $A \oplus B$ is still useful in this non-relay setting. We provide an affirmative answer to this question with a novel two-layer decoding approach amenable to real-time

- *L. Lu is with the Institute of Network Coding, The Chinese University of Hong Kong, Hong Kong Special Administrative Region, China. E-mail: lulu@ie.cuhk.edu.hk*
- *L. You and S. C. Liew are with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong Special Administrative Region, China. E-mail: {yl013, soung}@ie.cuhk.edu.hk*

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

implementation. Two key enabling components are our specially designed 1) PHY-layer channel decoders, and 2) MAC-layer erasure channel decoders.

For component 1), we first use a *MUD channel decoder* to try to decode both packets A and B. If the MUD decoder successfully decodes both packets, our job is done. However, our experimental data indicate that sometimes only packet A or packet B can be obtained, and sometimes none of them. In the event that the MUD decoder can only decode either packet A or packet B, or none of them, we use a *PNC channel decoder* to try to decode $A \oplus B$. The likelihood of the PNC decoder successfully decoding $A \oplus B$ when the MUD decoder does not have complete success in decoding packets A and B can be substantial. For example, at SNR of 8.5dB, with probability 22% the MUD decoder can decode only one of packet A or B. When only one of the packets can be decoded, with probability 85% we can decode $A \oplus B$. In this scenario, we can use $A \oplus B$ and the available native packet to recover the missing native packet, A or B. We refer to this as the *PHY-layer bridging* mechanism of network-coded packets. A native packet can be used with a *complementary network-coded packet* to recover a missing native packet. This is the first way the network-coded packet can be useful.

Component 2) provides a second way to exploit network-coded packets. Our experimental data indicates that at SNR of 8.5dB, with probability 55% the MUD decoder fails to decode both packets A and B. When neither A nor B can be decoded, with probability 40%, the PNC decoder can still decode $A \oplus B$. At first glance, this lone $A \oplus B$ is not useful because we cannot combine it with an available native packet to recover a missing native packet. An interesting question is whether we can still extract utility out of such *lone network-coded packets*. Seemingly, the answer is no because there is no mutual information between the network-coded packet with either of the two native packets when the other native packet is not available.

Component 2), however, provides a way to use the lone network-coded packets by exploiting mutual information at the *message* level. At the MAC layer of NCMA, block messages M^A and M^B from A and B are coded using an erasure channel code (e.g., the Reed Solomon code [6]) and partitioned into smaller constituent packets. For example, the block messages M^A and M^B could be jumbograms from IPv6 or other large messages from the network layer (e.g., those from big data communication [7]). With erasure channel coding, provided enough of the constituent packets of A (or B) can be decoded at the PHY layer, then M^A (or M^B) can be obtained at the MAC layer. Here, we give a simple example on how the lone packets from the PHY layer can be useful. Suppose that at some point in time, the PHY layer has decoded enough packets of A for it to obtain M^A. Having the source message M^A then allows the AP to derive all the missing packets A in the previous PHY transmissions. This includes the time slots in which the PHY decoders could not decode both packets A and B but could decode packet $A \oplus B$. With the newly derived packets A, their corresponding missing packets B can now be recovered through the previously lone $A \oplus B$. We see that once one of M^A or M^B is decoded, the lone $A \oplus B$ packets become complementary and useful. We refer to this as the *MAC-layer bridging* mechanism of network-coded packets.

The contributions of this work are as follows:

1) We present the first conceptualization and experimental demonstration of the usefulness of PNC in a non-relay setting.
2) In terms of concepts and principles, we
   a) design a simple MUD decoder and a simple PNC decoder based on the principle of *reduced constellation* that are amenable to real-time implementation;
   b) devise PHY-layer and MAC-layer bridging algorithms that fully exploit the information contained in the native and network-coded packets decoded by the PHY decoders;
   c) propose an NCMA MAC protocol that can realize the potential throughput gain of our decoding algorithms.
3) In terms of experimentation, we
   a) prototype our NCMA system on the USRP platform to prove its viability;
   b) show that our NCMA system can achieve 100% throughput gain at the medium-high SNR regime ($\geq$10dB) compared with the user-by-user transmission system;
   c) show that our NCMA system can achieve 40-80% throughput gain in the medium SNR regime (between 8dB and 9.5dB) compared with conventional MUD systems;
   d) demonstrate the robustness of our NCMA system under both balanced and unbalanced receive powers from different users.

2 Overview

This section gives a quick overview of the NCMA system architecture and the experimental results that motivate its design.

2.1 NCMA Architecture

Fig. 2 shows the architecture of a node equipped with NCMA capability. For legacy compatibility, the node can revert to 802.11 when necessary. NCMA packets use different preambles than 802.11 so that the receiver can distinguish between NCMA packets and 802.11 packets.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

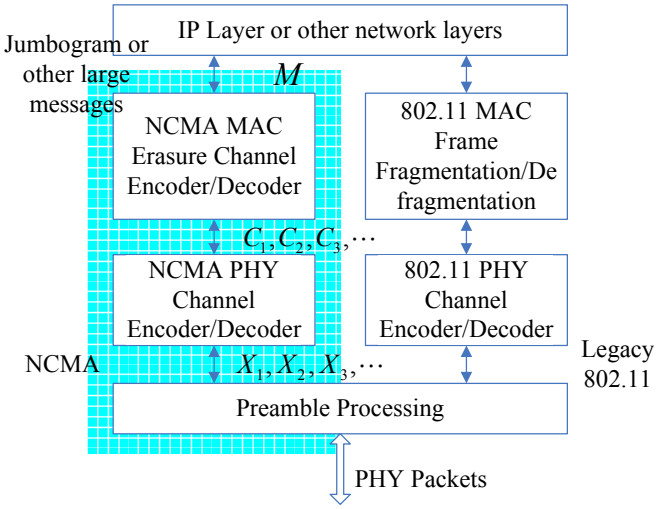

Fig. 2. The architecture of an NCMA node.

Packet Index	Eq^A	$Eq^{A\oplus B}$	Eq^B
1	C_1^A	$C_1^{A\oplus B}$	C_1^B
2	C_2^A	ø	C_2^B
3	C_3^A	$C_3^{A\oplus B}$	ø
4	ø	$C_4^{A\oplus B}$	C_4^B
5	C_5^A	ø	ø
6	ø	ø	C_6^B
7	ø	$C_7^{A\oplus B}$	ø
8	ø	ø	ø

Fig. 3. An example of PHY-layer packet reception patterns for concurrently transmitted packets. Empty entries (ø) mean the corresponding packets cannot be decoded. Each column is labeled Eq^J corresponding to packet type J being decoded. Each column forms a linear equation system for MAC-layer decoding (see Section 3).

NCMA can support IPv6 jumbograms or other large messages from the network layer. Instead of chopping a large message M from the network layer into independent packets, it makes use of an erasure channel code to encode the message into multiple packets, $C_1, C_2, \ldots$. The erasure channel code adopted could be the Reed Solomon (RS) code [6] or a rateless channel code [8]. Provided a sufficient number of these packets are received correctly, then the original source message can be decoded at the receiver. At the PHY layer, each packet C_i is further channel-coded into a packet X_i. For this work, we assume the use of the RS code at the MAC layer and the convolutional code adopted by 802.11 at the PHY layer.

In the NCMA mode, the node transmits packets X_1, $X_2, \ldots$ in different time slots and a packet will not be retransmitted even if it cannot be received successfully. Also, the receiver will not return acknowledgements for individual packets; it will return an acknowledgement only when the overall message M is decoded.

2.2 PHY-Layer Decoding and Bridging

NCMA makes use of two multiuser PHY-layer channel decoders: a MUD decoder and a PNC decoder. We adapt the standard single-user soft-input Viterbi decoding algorithm (VA) [9] for PNC and MUD decoding. The details on how we adapt the standard single-user VA for PNC and MUD decoding can be found in Section 4. We emphasize that we aim for reduced complexity rather than optimality with this adaptation approach. More optimal PNC and MUD decoders are certainly possible; however, the standard VA decoder is a very simple decoder that allows efficient real-time decoding.

When two nodes A and B transmit to the AP simultaneously, the MUD decoder in NCMA attempts to decode both C_i^A and C_i^B based on the overlapped signals of X_i^A and X_i^B; the PNC decoder, on the other hand, attempts to decode $C_i^A \oplus C_i^B$ (the bit-wise XOR of C_i^A and C_i^B) based on the same overlapped signals.

Eight Possible Decoding Events – For the MUD decoder, there are four possible outcomes: (i) both C_i^A and C_i^B are successfully decoded; (ii) only C_i^A is successfully decoded; (iii) only C_i^B is successfully decoded; (iv) both C_i^A and C_i^B cannot be decoded. For the PNC decoder, there are two possible outcomes: (a) $C_i^A \oplus C_i^B$ is successfully decoded; (b) $C_i^A \oplus C_i^B$ cannot be decoded. As a result we have $4 \times 2 = 8$ possible combined outcomes. We will refer to these combined outcomes as "events". Fig. 3 shows a contrived example in which the eight possible events occur in successive time slots.

2.2.1 Leveraging Complementary XOR Packets

In Fig. 3, event (ii)(a) and event (iii)(a) occur in time slots 3 and 4, in which C_3^A and $C_3^A \oplus C_3^B$ (abbreviated as $C_3^{A\oplus B}$), and C_4^B and $C_4^A \oplus C_4^B$ (abbreviated as $C_4^{A\oplus B}$), are decoded, respectively. In these two cases, the complementary XOR packets, $C_3^{A\oplus B}$ and $C_4^{A\oplus B}$, can be used to recover the missing native packets, C_3^B and C_4^A, respectively. That is, these two events are equivalent to event (i) in which both native packets can be decoded.

Let us take a glimpse of the experimental results of Fig. 10, the details of which will be discussed in Section 5. At SNR=8.5dB, we see that the probability of event (ii)(a) or (iii)(a) occurring, annotated as AX | BX in the figure, is not negligible. Specifically, they occur around 20% of the time. This means the complementary XOR packets can help recover missing native packets around 20% of the time.

2.3 MAC-Layer Decoding and Bridging

Returning to Fig. 3, event (iv)(a) occurs in time slot 7. None of the native packets can be decoded by the MUD decoder; on the other hand, the PNC decoder

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

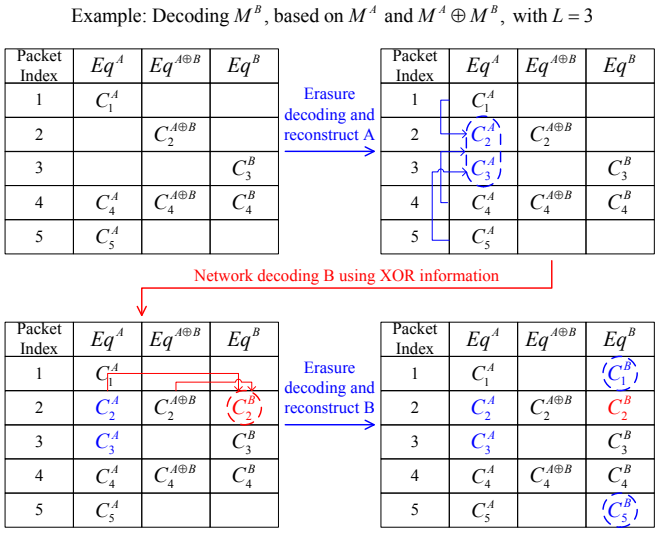

Fig. 4. NCMA decoding process, with $L = 3$ in the RS code as an example.

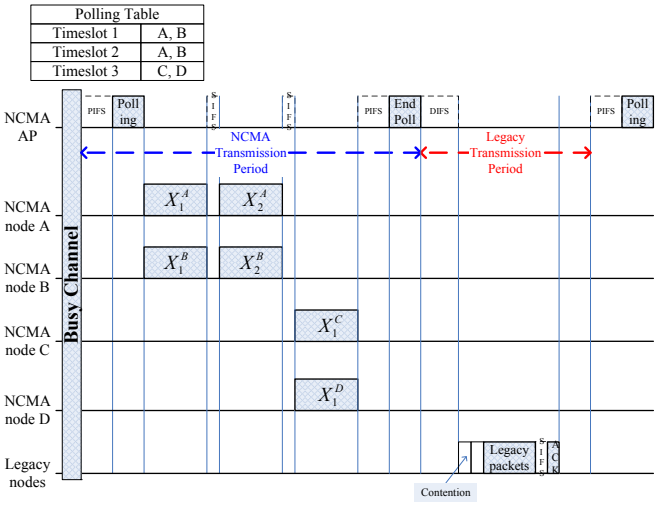

Fig. 5. NCMA uplink media access with NCMA and legacy nodes. The polling table is contained in the *Polling* frame, and the *End Poll* frame terminates the NCMA channel access session.

can decode $C_7^A \oplus C_7^B$. Such lone $C_i^A \oplus C_i^B$ are not useful as far as the recovery of the native packets is concerned. Yet, experimental data indicate that event (iv)(a) is not negligible. As shown in Fig. 10, the event of lone XOR packet, annotated as X in the figure, occurs around 20% of the time at SNR=8.5dB. This suggests that the system performance could be improved by much if we can find ways to exploit the lone XOR packets.

Lone $C_i^A \oplus C_i^B$ turns out to be useful for MAC-layer decoding, where the correlations and mutual information among successive packets can be exploited. Fig. 4 illustrates the idea.

2.3.1 *Leveraging Lone XOR Packets*

In the upper left table of Fig. 4, we suppose that the AP has recovered enough PHY packets C_i^A of different i for it to decode M^A – in this simplified example, $L = 3$ PHY packets are needed to decode the message. Once M^A is decoded, all the C_i^A that could not be obtained by the PHY-layer decoders can now be recovered (through re-encoding based on M^A). In Fig. 4, these are the C_2^A and C_3^A circled in blue as shown in the upper right table. Note that once C_2^A is recovered, the previously lone $C_2^A \oplus C_2^B$ is converted into a complementary XOR packet and can be used to recover C_2^B, shown in red circle in the lower left table. In this example, we now also have enough C_i^B of different i to decode M^B. The details and effectiveness of such MAC-layer bridging by XOR packets will be addressed in Section 3 and Section 5, respectively.

2.4 NCMA MAC Protocol

To exploit the aforementioned decoding and bridging mechanisms, we need a MAC protocol for coordinating concurrent transmissions by multiple nodes to the AP. There are many possibilities. Here we give a simple example that covers only the essence of NCMA.

Fig. 5 illustrates the uplink operation of this protocol. Here, there are four NCMA nodes: A, B, C, and D. Their source messages have been encoded into PHY packets: $X_1^A, X_2^A, \ldots$; $X_1^B, X_2^B, \ldots$; $X_1^C, X_2^C, \ldots$; $X_1^D, X_2^D, \ldots$. As far as the scheduling of the transmissions is concerned, the NCMA operation is similar to, but not exactly the same as, the 802.11 point coordination function (PCF) [10]. As in 802.11 PCF, a transmission round is divided into two periods: contention-free period (CFP) and contention period. NCMA nodes transmit during CFP, and legacy 802.11 nodes transmit during the contention period using 802.11 DCF.

The AP coordinates NCMA transmissions during CFP by polling. Each poll coordinates the transmissions in a number of successive transmission time slots. A major difference compared with 802.11 PCF is that a number of nodes may transmit together in a time slot in NCMA. In other words, instead of avoiding collisions as in the original PCF, here the AP actually encourages "collisions" by polling multiple nodes to transmit in the same time slot. As illustrated in Fig. 5, nodes A and B transmit together in time slots 1 and 2, and nodes C and D transmit together in time slot 3.

Another difference compared with 802.11 PCF is that the NCMA AP does not issue an ACK immediately upon the reception of each packet. Rather, an ACK will be issued only when the whole source message M of a source has been decoded.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

3 MAC-Layer Message Decoding Algorithm

This section delves into the details of NCMA MAC-layer decoding.

3.1 Preliminary: Single-User Erasure Channel Coding and Decoding

We first review the single user case in which only node A transmits. For a focus, we assume the use of the RS erasure channel code [6]. Extensions to other erasure codes, including advanced rateless codes [8], are possible.

We adopt the set-up as expounded in [11]. We organize the symbols of source message M^A of node A in matrix form:

$$M^A = \begin{bmatrix} M_1^A & \dots & M_j^A & \dots & M_K^A \end{bmatrix} = \begin{bmatrix} a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,K} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,K} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{L,1} & \cdots & a_{L,j} & \cdots & a_{L,K} \end{bmatrix}, \quad (1)$$

where $a_{i,j} \in GF(2^s)$. For example, if $s = 8$, each symbol $a_{i,j}$ is a byte and there are altogether LK bytes in the source message. The generator matrix for the RS code is

$$G = \begin{bmatrix} G_1 \\ \vdots \\ G_i \\ \vdots \\ G_N \end{bmatrix} = \begin{bmatrix} g_{1,1} & g_{1,2} & \cdots & g_{1,L} \\ \vdots & \vdots & \ddots & \vdots \\ g_{i,1} & g_{i,2} & \cdots & g_{i,L} \\ \vdots & \vdots & \ddots & \vdots \\ g_{N,1} & g_{N,2} & \cdots & g_{N,L} \end{bmatrix}, \quad (2)$$

where $g_{i,j} \in GF(2^s)$. There are altogether $N = 2^s - 1$ nonzero elements in $GF(2^s)$. Let us denote the nonzero elements by $\alpha_1, \alpha_2, \dots, \alpha_N$. We set

$$G_i = \begin{bmatrix} 1 & \alpha_i & \alpha_i^2 & \cdots & \alpha_i^{L-1} \end{bmatrix}, \quad 1 \le i \le N.$$

Proposition 1: Any L of the vectors $G_1, G_2, \dots, G_{N-1}$ are linearly independent.

Proof: see [11].

For each message M^A, we can generate N coded packets, each of K symbols, by

$$C^A = \begin{bmatrix} C_1^A \\ \vdots \\ C_i^A \\ \vdots \\ C_N^A \end{bmatrix} = GM^A. \quad (3)$$

In the above, a packet C_i^A is represented as a $1 \times K$ row vector. A packet C_i^A will not be retransmitted if it cannot be received; the transmission process just progresses to the next packet C_{i+1}^A. The receiver only needs to receive any L of the N packets in the set $\{C_1^A, \dots, C_i^A, \dots, C_N^A\}$ to decode M^A. Specifically, upon receiving any L packets, denoted by $C_{(1)}^A, C_{(2)}^A, \dots, C_{(L)}^A$, we can arrange them into a matrix $\tilde{C}^A$ as follows:

$$\tilde{C}^A = \begin{bmatrix} C_{(1)}^A \\ \vdots \\ C_{(i)}^A \\ \vdots \\ C_{(L)}^A \end{bmatrix} = \tilde{G}M^A = \begin{bmatrix} G_{(1)} \\ \vdots \\ G_{(i)} \\ \vdots \\ G_{(L)} \end{bmatrix} \begin{bmatrix} M_1^A \cdots M_i^A \cdots M_K^A \end{bmatrix}. \quad (4)$$

By Proposition 1, $\tilde{G}$ is invertible. Thus, we can extract message A by

$$M^A = \tilde{G}^{-1}\tilde{C}^A. \quad (5)$$

With $s = 8$, we have $N = 255$ coded packets. There is a chance that the transmitter runs out of packets C_i^A to transmit after all 255 of them have been transmitted, while the AP still has not acquired enough PHY packets to decode M^A (because many of the 255 packets are corrupted and cannot be decoded). However, in the experiments to be detailed in Section 5, the typical L we need is no more than 32, and the transmitter never runs out of packets to transmit before the message is decoded.

3.2 NCMA Erasure Channel Coding and Decoding

We now consider the NCMA situation in which node A and node B transmit their i-th packets, C_i^A and C_i^B, simultaneously for $i = 1, 2, \dots$ until at least one of the two messages, M^A or M^B, is decoded. For decoding, we set up three interacting equation systems: the first, denoted by Eq^A, is for decoding M^A; the second, denoted by Eq^B, is for decoding M^B; and the third, denoted by $Eq^{A\oplus B}$, is for decoding $M^A \oplus M^B$.

3.2.1 Equation System Eq^A

Recall that there are eight possible events (see Section 2.2 and Fig. 3) for PHY-layer decoding, corresponding to the decoding outcomes of the MUD decoder and PNC decoder for transmission i.

We note that for events (i), (ii), and (iii)(a), C_i^A can be obtained. For events (i) and (ii) (whether the PNC decoding is successful or not), C_i^A is directly given by the MUD decoder. For event (iii)(a), C_i^A is obtained by $C_i^A = C_i^B \oplus C_i^{A\oplus B}$.

Any of the events (i), (ii), or (iii)(a), yields a packet of type C_i^A. As nodes A and B transmit their successive packets, as soon as we have L packet of type C_i^A, we can decode message M^A.

Let $\tilde{C}^A$ denote the set of packets of type C_i^A given by the PHY layer to the MAC layer. The general form of Eq^A, according to (4), is

$$\tilde{G}^A M^A = \tilde{C}^A, \quad (6)$$

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

where the rows in $\tilde{G}^A$ are the subset of the rows of G that correspond to the C_i^A in $\tilde{C}^A$. For example, in Fig. 4, $\tilde{G}^A = \begin{bmatrix} G_1 \\ G_4 \\ G_5 \end{bmatrix}$, and $\tilde{C}^A = \begin{bmatrix} C_1^A \\ C_4^A \\ C_5^A \end{bmatrix}$. Suppose that there are L rows in $\tilde{C}^A$. Then M^A can be decoded by

$$M^A = (\tilde{G}^A)^{-1}\tilde{C}^A. \tag{7}$$

MAC-layer bridging works as follows. Let $\tilde{C}^{A\oplus B} = \tilde{C}^A \oplus \tilde{C}^B$ be the XOR packets decoded by the PHY layer, and let $\bar{C}^{A\oplus B} = \bar{C}^A \oplus \bar{C}^B$ be the resulting $Eq^{A\oplus B}$ that consists of the lone XOR packet only. That is, $\bar{C}^{A\oplus B}$ consists of the XOR packets produced under event (iv)(a) only. Then,

$$\begin{aligned} &\bar{G}^{A\oplus B}(M^A \oplus M^B) \\ &= \bar{G}^{A\oplus B}M^A \oplus \bar{G}^{A\oplus B}M^B = \bar{C}^{A\oplus B}, \end{aligned} \tag{8}$$

where $\bar{G}^{A\oplus B}$ consists of the rows of G that correspond to the $C_i^{A\oplus B}$ in $\bar{C}^{A\oplus B}$. For example, in Fig. 4, $\bar{G}^{A\oplus B} = \left[G_2^{A\oplus B}\right]$ and $\bar{C}^{A\oplus B} = \left[C_2^{A\oplus B}\right]$.

Now, since $M^A = (\tilde{G}^A)^{-1}\tilde{C}^A$ has been decoded by (7), we can write (8) as

$$\bar{G}^{A\oplus B}M^B = \bar{C}^{A\oplus B} \oplus \bar{G}^{A\oplus B}(\tilde{G}^A)^{-1}\tilde{C}^A. \tag{9}$$

We can add the equations in (9) to Eq^B to form a new Eq^B:

$$\begin{bmatrix} \tilde{G}^B \\ \bar{G}^{A\oplus B} \end{bmatrix} M^B = \begin{bmatrix} \tilde{C}^B \\ \bar{C}^{A\oplus B} \oplus \bar{G}^{A\oplus B}(\tilde{G}^A)^{-1}\tilde{C}^A \end{bmatrix}. \tag{10}$$

Thus, if Eq^A is solved before Eq^B and $Eq^{A\oplus B}$, through the above MAC-layer bridging process, we can obtain more equations for Eq^B.

3.2.2 Equation System Eq^B

Similar to Eq^A above. As soon as we have L packets of type C_i^B, we can decode M^B.

3.2.3 Equation System $Eq^{A\oplus B}$

Here, we are interested in event (i) and event (a), since in both these events $C_i^{A\oplus B}$ can be obtained. Similar to Eq^A, as soon as we have L packets of type $C_i^{A\oplus B}$, we have enough equations to decode $M^A \oplus M^B$. In particular, thanks to the linearity of the RS code, the decoding is similar to (5).

In the case when $Eq^{A\oplus B}$ has L equations before Eq^A or Eq^B does, the availability of $M^A \oplus M^B$ gives all the packets $C_i^{A\oplus B}$, $i = 1, \ldots, N$, which can then serve as a bridge to obtain C_i^A and C_i^B that were previously unavailable. In particular, once $Eq^{A\oplus B}$ is solved, after MAC-layer bridging, the number of available equations in Eq^A and Eq^B will be the same from then on, because knowing C_i^A means knowing C_i^B, and vice versa. Thus, once $Eq^{A\oplus B}$ is solved, either both Eq^A and Eq^B are solved at the same time as $Eq^{A\oplus B}$ (when the bridge causes them both to have L or more equations), or both Eq^A and Eq^B need the same number of additional equations before M^A and M^B can be decoded (when they still have fewer than L equations). For the latter, the subsequent transmissions will be more efficient because each time when only C_i^A or C_i^B is decoded at the PHY layer, the missing packet can be derived through the bridge.

More formally, suppose that $Eq^{A\oplus B}$ has L equations before Eq^A and Eq^B. Then, we have

$$M^A \oplus M^B = (\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B}. \tag{11}$$

Additional equations for Eq^A and Eq^B can be obtained as described below. Before bridging, we have

$$\begin{aligned} \tilde{G}^A M^A &= \tilde{C}^A, \\ \tilde{G}^B M^B &= \tilde{C}^B. \end{aligned} \tag{12}$$

Recall that $\bar{C}^{A\oplus B}$ denotes the lone XOR packets corresponding to event (iv)(a). Similarly, we can define $\bar{C}^A$ and $\bar{C}^B$ as the lone user packets corresponding to events (ii)(b) and (iii)(b), respectively. From (11) and (12), we get the following additional packets via MAC-layer bridging

$$\begin{aligned} \bar{G}^A[M^B \oplus (\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B}] &= \bar{C}^A, \\ \bar{G}^B[M^A \oplus (\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B}] &= \bar{C}^B. \end{aligned} \tag{13}$$

Combining (12) and (13), the post-bridging Eq^A and Eq^B are as follows:

$$\begin{aligned} \begin{bmatrix} \tilde{G}^A \\ \bar{G}^B \end{bmatrix} M^A &= \begin{bmatrix} \tilde{C}^A \\ \bar{C}^B \oplus \bar{G}^B(\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B} \end{bmatrix}, \\ \begin{bmatrix} \tilde{G}^B \\ \bar{G}^A \end{bmatrix} M^B &= \begin{bmatrix} \tilde{C}^B \\ \bar{C}^A \oplus \bar{G}^A(\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B} \end{bmatrix}. \end{aligned} \tag{14}$$

3.2.4 Intermixing Message Pairs

In the example of Fig. 4, enough equations are created through the bridge that message M^B can also be solved at the same time as message M^A. In general, it is possible that Eq^B still may not have enough equations. Then M^B of node B can be paired with the next message of node A, or be paired with a message from a different node, say node C. This seamless pairing will ensure that the available equations in Eq^B are not wasted. The new message, of course, will start off with needing L equations while M^B needs fewer equations for decoding. Note that the packets associated with the new message, say M^C, will be sent in the sequence of $C_{j+1}^C, C_{j+2}^C, \ldots, C_1^C, C_2^C, \ldots$, if C_j^B was the last packet sent by node B. This is so as to align the equations of node B and node C in the three equation systems.

4 PHY-Layer Channel Decoding

The preceding section focused on MAC-layer decoding. At the PHY layer, channel coding is applied to C_i^A and C_i^B by node A and node B to form X_i^A and X_i^B for transmission. At the AP, the PHY decoders attempt to

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

obtain C_i^A, C_i^B, and $C_i^{A\oplus B}$ from the overlapped X_i^A and X_i^B.

For PHY channel coding, our NCMA prototype adopts the same convolutional code as in 802.11. A goal of ours is to simplify the PHY decoder design so that real-time decoding is possible in our USRP prototype. The standard VA decoder is widely used in real communication systems thanks to its low complexity. As such, we adapt the standard soft-input Viterbi decoding algorithm (VA) [9] for PNC and MUD decoding. More optimal decoders with better decoding performance are certainly possible for PNC and MUD; however, they are more complex and may not be amenable to real-time decoding. An issue, however, is that the standard VA is meant for single-user systems. NCMA, by contrast, is intrinsically a multiuser system in that the PNC and MUD decoders have to deal with overlapped signals from multiple users. VA cannot be used as is. This section explains how we adapt VA for PNC and MUD decoding.

4.1 Preliminary: Single-User VA

We first review the key idea of the standard VA when applied to the single-user system. Without loss of generality, let us assume an OFDM system (our prototype is an OFDM system). Let X denote the packet transmitted and $x[k]$ denote the value of modulated bit k within the packet X. We assume BPSK modulation for simplicity here. Let $y[k]$ be the received signal corresponding to bit k given by

$$y[k] = h_{s_k} x[k] + n[k], \tag{15}$$

where $x[k] = 1$ if bit k is 0 and $x[k] = -1$ if bit k is 1; and $n[k]$ is the zero mean Gaussian noise with variance σ^2. In the above, h_{s_k} is the channel gain associated with the subcarrier s_k on which bit k is carried.

The idea of the standard VA is to provide a confidence metric for each bit, $\tilde{x}[k]$, for the computation of the edge length in the Viterbi shortest-path algorithm. This confidence metric is also referred to as the soft bit, and is computed by the log likelihood ratio $\log \frac{P_0[k]}{P_1[k]}$, where $P_0[k] = P(x[k] = 1|y[k])$ and $P_1[k] = P(x[k] = -1|y[k])$. Thus, $\tilde{x}[k]$ takes on a real value. The more positive it is, the more confident we are that bit k is 0; and the more negative it is, the more confident we are that bit k is 1. VA then finds the maximum likelihood (ML) codeword $X = \{x[k]\}_{k=1,2,\ldots}$ with the minimum $-\sum_k \tilde{x}[k]x[k]$ (or maximum $\sum_k \tilde{x}[k]x[k]$) [12]. Essentially, a shortest-path algorithm is applied on the trellis associated with the convolutional code.

4.2 Soft Decoders for NCMA

The block diagram of the overall NCMA channel decoder is shown in Fig. 6. Given the input $\{y[k]\}_{k=1,2,\ldots}$,

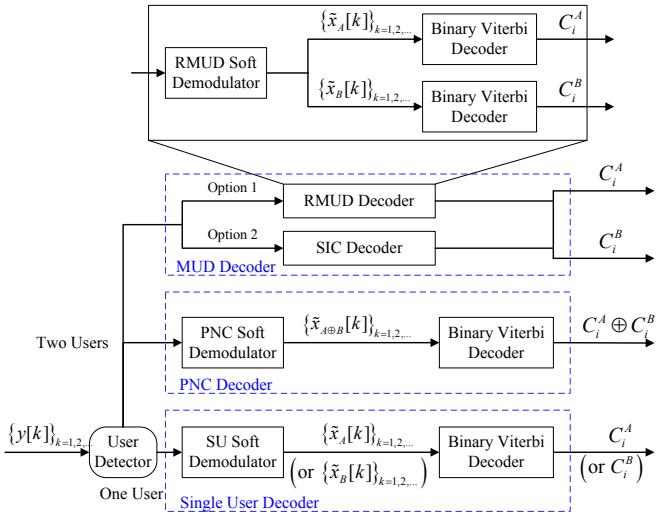

Fig. 6. Single-user decoder, MUD decoder and PNC decoder for overall PHY-layer channel decoding in NCMA. In this figure, the index k is over multiple OFDM symbols in a frame because channel coding is performed over multiple OFDM symbols.

the AP first detects the presence of signals from the users (using correlation on the preambles). If only one user transmits and only the signal from that user is detected, then the standard single-user VA is used (the lowest branch in the block diagram). If there are overlapping signals from two users, then both the MUD decoder and the PNC decoder are used (the upper two branches in the block diagram).

All the decoders use the standard binary Viterbi algorithm for decoding. Our single-user decoder, PNC decoder, and MUD decoder differ only in the ways their demodulators compute the soft information to be fed to the binary Viterbi decoder. In the next two subsections, we will describe how the PNC demodulator and the MUD demodulator compute their soft information.

4.2.1 PNC Decoder

The main essence of our PNC decoder is as follows. To allow the usage of the standard binary VA, we perform simplification by reducing the number of constellation points to two. These two constellation points correspond to the most probable points for the two different XOR values. The log likelihood ratio based on these two constellation points is used as the soft-information to be fed to the standard binary VA. In Fig. 6, the PNC demodulator is responsible for computing this soft information. We refer to this strategy as *reduced-constellation decoding*. In the following paragraphs, we describe the details.

Let $X^A = (x_A[1], \ldots, x_A[k], \ldots)$ and $X^B = (x_B[1], \ldots, x_B[k], \ldots)$ denote the PHY-layer codewords (transmitted packets) of users A and B, and let $\Pi(\cdot)$ denote functional mapping corresponding to

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

convolutional coding. Since $\Pi(\cdot)$ is linear, we have $X^A \oplus X^B = \Pi(C^A) \oplus \Pi(C^B) = \Pi(C^A \oplus C^B)$. This means that we can first detect the XOR of individual bits $\{x_A[k] \oplus x_B[k]\}_{k=1,2,...}$ and then feed the information on these XOR bits to the standard convolutional decoder for decoding [5][1]. For our PNC decoder, the PNC demodulator will feed soft information on $\{x_A[k] \oplus x_B[k]\}_{k=1,2,...}$ to the standard VA. For the subsequent discussion, for brevity, we will drop the index k in our notations. In addition, we will write $x_A[k] \oplus x_B[k]$ simply as $x_{A\oplus B}$.

Our proposed $\tilde{x}_{A\oplus B}$, the soft bit for $x_{A\oplus B}$, follows the same reasoning as for $\tilde{x}$ of the standard single-user VA. Recall from the discussion in Section 4.1 that in the standard VA, $\tilde{x} = \log \frac{P_0}{P_1}$ is the log likelihood ratio, where P_0 is the probability that the bit is 0 and P_1 is the probability that the bit is 1.

When there are signals from both nodes A and B, instead of (15), the received signal is

$$y = h_A x_A + h_B x_B + n, \tag{16}$$

where h_A and h_B are the channel gains associated with A and B respectively, and n is the complex Gaussian noise with variance $2\sigma^2$ (i.e., the variance in each of I and Q dimension is σ^2). For $\tilde{x}_{A\oplus B}$, the log likelihood ratio is given by [13]

$$\begin{aligned}\log \frac{P_0}{P_1} &= \log \frac{\Pr\{x_A=1,x_B=1|y\}+\Pr\{x_A=-1,x_B=-1|y\}}{\Pr\{x_A=-1,x_B=1|y\}+\Pr\{x_A=1,x_B=-1|y\}} \\ &= \log\left(\exp\left\{-\frac{|y-h_A-h_B|^2}{2\sigma^2}\right\}+\exp\left\{-\frac{|y+h_A+h_B|^2}{2\sigma^2}\right\}\right) \\ &\quad - \log\left(\exp\left\{-\frac{|y+h_A-h_B|^2}{2\sigma^2}\right\}+\exp\left\{-\frac{|y-h_A+h_B|^2}{2\sigma^2}\right\}\right).\end{aligned} \tag{17}$$

Unlike the single-user case, it is difficult to further simplify the log likelihood ratio expression above without appromixation. Adopting the log-max approximation, $\log(\sum_i \exp(z_i)) \approx \max_i z_i$), [14] gives

$$\begin{aligned}\frac{\sigma^2}{2}\log\frac{P_0}{P_1} &\approx \frac{1}{4}\max\left\{-|y-h_A-h_B|^2, -|y+h_A+h_B|^2\right\} \\ &\quad - \frac{1}{4}\max\left\{-|y+h_A-h_B|^2, -|y-h_A+h_B|^2\right\}.\end{aligned} \tag{18}$$

We can let $\tilde{x}_{A\oplus B}$ = RHS of (18) since the constant factor $\frac{\sigma^2}{2}$ will not affect the shortest path found by VA.

The RHS of (18) can be further simplified. We illustrate this with the example in Fig. 7. Fig. 7 shows the constellation map of a specific pair of channel gains, h_A and h_B. With BPSK modulation, (x_A, x_B) takes on four possible values, $(\pm 1, \pm 1)$. In (18), either $(+1,+1)$ or $(-1,-1)$ will be chosen in the first $\max(\cdot)$ function, and either $(+1,-1)$ or $(-1,+1)$ will be chosen in the second $\max(\cdot)$ function. Which pair is chosen depends on y. In Fig. 7, we show a particular realization of y such that $(+1,+1)$ and $(-1,+1)$ are chosen. The

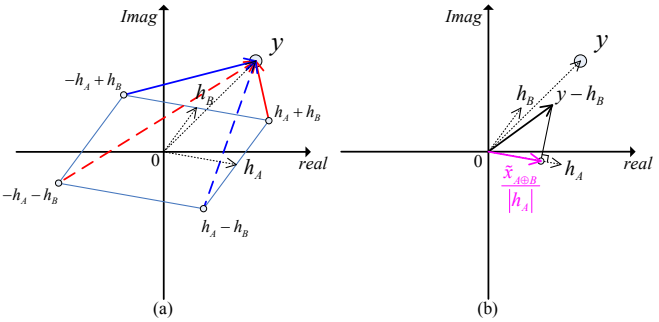

Fig. 7. Illustration of PNC soft demodulation which gives soft information on XOR bit: the solid and dashed lines of the same color represent the two Euclidean distances between the received sample and the two constellation points mapped to the same XOR. (a) the solid blue line and the solid red line are the Euclidean distances to the selected constellation points corresponding to the two different values of the XOR bit $x_{A\oplus B}$; (b) projection of $y - h_B$ onto h_A to get the soft information $\tilde{x}_{A\oplus B}$.

corresponding soft bit for $x_{A\oplus B}$ is

$$\begin{aligned}\tilde{x}_{A\oplus B} &\approx -\frac{1}{4}|y-h_A-h_B|^2 + \frac{1}{4}|y+h_A-h_B|^2 \\ &= h_A \cdot (y-h_B),\end{aligned} \tag{19}$$

where the dot "$\cdot$" above stands for dot product and we represent the complex numbers h_A, h_B, and y as vectors with two elements, one of which is the real part of and one of which is the imaginary part of the corresponding complex number. The intuition for (19) is as follows. The fact that $(+1,+1)$ and $(-1,+1)$ are chosen implicitly means that we have already made a first decision that node B transmits $+1$. Thus, we should then decide the XOR value based on what node A transmits. The term $y - h_B$ in (19) corresponds to subtracting from the received sample y the decision that node B transmits $+1$. The component that contains the signal from node A can therefore be defined as $y_A = y - h_B$. It can be shown that the dot product $h_A \cdot y_A$ is simply the soft information for an equivalent single user case in which only node A transmits[2].

For other realizations of y, $\tilde{x}_{A\oplus B}$ can be found similarly. Specifically, there are four cases. If y is such that:

- $(+1,+1)$ and $(-1,+1)$ are chosen, then $\tilde{x}_{A\oplus B} \approx h_A \cdot (y - h_B)$;
- $(-1,-1)$ and $(-1,+1)$ are chosen, then $\tilde{x}_{A\oplus B} \approx -h_B \cdot (y + h_A)$;
- $(+1,+1)$ and $(+1,-1)$ are chosen, then $\tilde{x}_{A\oplus B} \approx h_B \cdot (y - h_A)$;

1. Note: $x_A[k]$ and $x_B[k]$ adopts a value of $+1$ or -1 rather than 0 or 1; therefore, we define $x_A[k] \oplus x_B[k] = x_A[k]x_B[k]$.

2. Note that for the single user case, when BPSK is used, the component of y_A that is orthogonal to h_A contains noise only. The dot product in the complex plane extracts the component of y_A that contains signal from node A.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

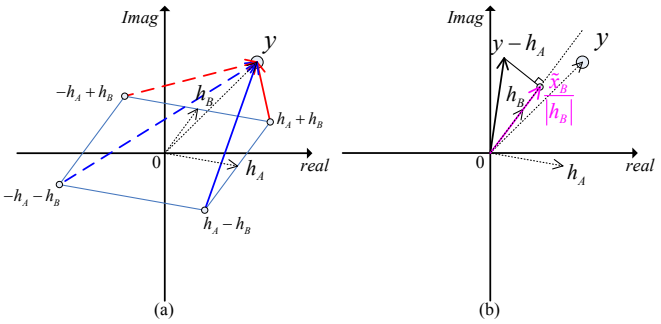

Fig. 8. Illustration of reduced-constellation MUD soft demodulation for x_B: (a) the solid blue line and the solid red lines are the Euclidean distances to the selected constellation points corresponding to the two different values of the bit x_B; (b) projection of $y - h_A$ onto h_B to get the soft information $\tilde{x}_B$.

- $(-1, -1)$ and $(+1, -1)$ are chosen, then $\tilde{x}_{A\oplus B} \approx -h_A \cdot (y + h_B)$.

Feeding $\{\tilde{x}_{A\oplus B}[k]\}_{k=1,2,\ldots}$ to the standard VA allows us to decode for $C^A \oplus C^B$.

4.2.2 MUD Decoders

In this paper, we consider two possible MUD decoders, as shown in the upper portion of Fig. 6. The first MUD decoder is similar to our PNC decoder in that it is based on the principle of reduced constellation. We refer to this MUD decoder as the reduced-constellation MUD (RMUD) decoder. The second MUD decoder is based on the well-known successive interference cancellation (SIC) principle [15]. We detail these two MUD decoders in the following.

RMUD: The RMUD soft demodulator (see Fig. 6) provides the soft information on x_A and x_B, $\tilde{x}_A$ and $\tilde{x}_B$, to two separate standard VA decoders, one for decoding C_i^A and one for decoding C_i^B. Without loss of generality, let us focus on $\tilde{x}_B$. The associated log likelihood function is

$$\log \frac{P_0}{P_1} = \log\left(\exp\left\{-\frac{|y-h_A-h_B|^2}{2\sigma^2}\right\} + \exp\left\{-\frac{|y+h_A-h_B|^2}{2\sigma^2}\right\}\right) - \log\left(\exp\left\{-\frac{|y-h_A+h_B|^2}{2\sigma^2}\right\} + \exp\left\{-\frac{|y+h_A+h_B|^2}{2\sigma^2}\right\}\right). \quad (20)$$

For the same example as in Fig. 7, (20) can be simplified to the following using the same strategy as in the preceding section:

$$\tilde{x}_B \approx h_B \cdot (y - h_A)\,. \quad (21)$$

The procedure for getting $\tilde{x}_B$ is illustrated in Fig. 8.

Similarly, we can get $\tilde{x}_A \approx h_A \cdot (y - h_B)$, whose expression happens to be the same as that of $\tilde{x}_{A\oplus B}$ in (19) for this particular example, but with different physical meanings. As for the PNC modulator in the preceding section, in general there are also four cases to be considered for each of $\tilde{x}_A$ and $\tilde{x}_B$ (omitted here to conserve space).

SIC: With respect to (16), SIC tries to first decode a stronger signal, say x_A, and then substitutes the estimate for x_A, $\hat{x}_A$, into (16) to get $\tilde{y} = (y - h_A\hat{x}_A)/h_B = x_B + h_A(x_A - \hat{x}_A)/h_B + (n/h_B)$. If the overall estimated codeword $\hat{X}^A$ is a valid codeword so that $\hat{X}^A = X^A$, then $y = x_B + (n/h_B)$. Thus the decoding of X^B would be as if X^A did not exist. If, on the other hand, X^A is not decoded correctly, then X^B may likely suffer from decoding errors as well.

Although in principle SIC should decode the stronger signal first, we find that in practice, based on our experimental data, better performance can be obtained by running two parallel SICs, with one decoding signal A first and the other decoding signal B first. It is perhaps due to other distortions and imperfection in channel estimation (in addition to noise) that decoding the stronger signal first is not always the best strategy (although most of time it is). The SIC experimental results that will be presented in Section 5 are based on this parallel version.

4.3 Quantization of Soft Information

So far we have assumed the soft bit, $\tilde{x}$, is a real number. In actual implementation, $\tilde{x}$ needs to be quantized before using VA. In particular, the VA decoder adopted in our system is based on the *Spiral* Viterbi software generator [16] that only accepts 8-bit inputs (from 0 to 255).

In mapping a real $\tilde{x}$ to an 8-bit quantized $\tilde{x}_{quantized}$, the main issue is where to put the constellation points of the signal within the quantized interval [0, 255], and generally there is a trade-off between clipping and quantization errors. We employ a design that optimizes the trade-off point for our multiuser system. Appendix A provides the details of such quantization.

5 Implementation and Experimental Results

This section presents the details of our NCMA implementation over the USRP software radio platform [17] and the experimental results carried over it.

5.1 Implementation

We have implemented the MUD channel decoder and the PNC channel decoder (see Fig. 6). To achieve real-time online decoding, we simplified the decoder designs with approximations as explained in Section 4.

Our system makes use of the USRP hardware [17] and the GNU Radio software with the UHD hardware driver. We extended the RawOFDM single-user point-to-point OFDM transceiver software [18] for the NCMA system. The extensions include:

a) Modifications of the single-user VA software for PHY-layer channel decoding in NCMA.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

b) Modifications of the MAC-layer frame format. For the NCMA system, a message from the higher layer (e.g., IPv6 jumbogram) is divided into smaller packets of fixed size (e.g., 1,500Bytes) at the MAC layer. These smaller packets are then RS encoded and embedded into MAC frames and forwarded to the PHY layer for transmission. Message ID and fragment offset are inserted into the MAC frame header for packet identification to facilitate MAC-layer bridging performed by the AP. Note that, to avoid frame size mismatch between the last frame of a message of one user and the non-last frame of the other message of the other user in paired transmissions, we pad the last frame with zeros to the predefined fixed frame size if necessary.

c) Modifications of the preamble and pilot designs. Different user nodes use orthogonal preambles and frequency-domain pilots so as to enable multiuser signal presence detection, multiuser channel estimation, and multiuser CFO tracking and compensation at the AP.

d) CFO-compensation precoding. Precoding for CFO compensation at the transmitters has been implemented to reduce the relative CFO between the two users at the AP [3]. The two users use the preambles in the poll frames from the AP during the downlink phase to estimate the CFOs between the users and the AP. The estimated CFO is compensated at the users' transmitters during the uplink phase. Thanks to the CFO compensation, the residual relative CFO at the AP is very small ($\sim$100 – 200Hz in our experiments).

e) CRC checking for PNC systems. The mathematics of CRC check for the 802.11 design [10], targeted for single-user system, cannot be directly applied to error detection of PNC decoding. We have modified the CRC check algorithm for PNC error detection.

f) Polling mechanism. The AP uses beacons to poll users to transmit either singly or simultaneously. The signals from multiple users can reach the AP with arrival-time offset that is within the cyclic prefix (CP) of OFDM. Doing so can eliminate OFDM symbol offset between users, thus simplify design [19]–[21]. Time synchronization to within 10^{-6}s has been achieved.

Due to page limit, we omit the details of these modifications in this paper except for a), which has already been explained in Section 4.

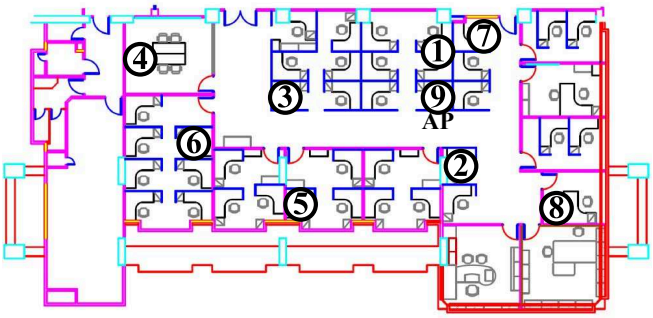

Fig. 9. Layout of an indoor environment for the deployment of 9 USRP N210 nodes for experimentation.

5.2 Experimental Setup and Results

For experimentation, we deployed 9 sets of USRP N210 with XCVR2450 boards [17] indoor to emulate a WLAN system. The topology is shown in Fig. 9. Each node is a USRP-connected PC. Each of the 9 nodes can be chosen to serve as the AP to test different network configurations. The AP can poll any two of the remaining 8 nodes to transmit together. BPSK modulation is used. Our experiments were carried out at 802.11 channel 1 (i.e., 2.412GHz) with 4 MHz bandwidth at midnight to minimize the co-channel interference from nearby ISM band equipment.

To benchmark our NCMA system, we consider the following systems [4]:

1) *Single-User (SU) system*
This is the traditional user-by-user non-overlapped transmission system. We use the same beacon mechanism as in NCMA to poll a pair of users; however, one user delays its transmission until the other user finishes transmission.
2) *MUD system (Multi-User)*
Here, we only use the MUD decoder (and not the PNC decoder) for PHY-layer decoding. The MUD decoder can either be RMUD or SIC. Only two equation systems, Eq^A and Eq^B, are used to do the MAC-layer decoding for A and B separately. There is no PHY-layer bridging or MAC-layer bridging.
3) *NCMA system (Multi-User)*
In full NCMA, both MUD decoder (either RMUD or SIC) and PNC decoder are used at the PHY layer. All three equation systems Eq^A, Eq^B, and $Eq^{A\oplus B}$ are used for MAC-layer decoding. In particular, both PHY-layer bridging and MAC-layer bridging are performed in the decoding process.

3. To estimate the instantaneous CFO between the end node and the AP, the end node makes use of the short training symbols in the Beacon frame (similar to the standard 802.11 point-to-point communication). Since the CFO value is relatively stable (constant over hours) [19], in our realization we average the CFO values obtained from several Beacon frames to filter out the measurement noise.

4. Only the NCMA system adopts the MAC-layer RS code. Neither the SU system nor the MUD system uses MAC-layer erasure codes. The reason is that, for these systems, without the XOR PHY packets from PNC decoding, we cannot do MAC-layer bridging, and MAC-layer channel coding will not make a difference in the throughput.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

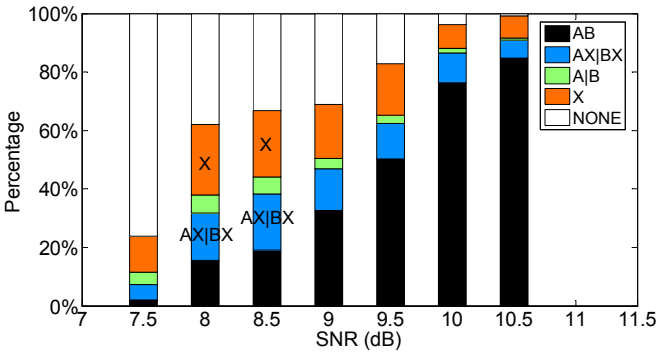

Fig. 10. NCMA PHY-layer packet decoding statistics versus SNR, with two nodes having the same SNR.

5.2.1 PHY-Layer Packet Decoding Statistics

Our first experiment aims to gather PHY-layer decoding statistics. The results are summarized in Fig. 10. Recall that with the combined use of MUD and PNC decoders in NCMA, there are 8 possible events (see Section 2.2). To ease presentation of the key results, we group some events in Section 2.2 together by the following notations in Fig. 10:

- NONE = (iv)(b) (no packet decoded).
- X = (iv)(a) (only XOR packet decoded).
- A|B = (ii)(b) + (iii)(b) (either only packet A or only packet B decoded).
- AX|BX = (ii)(a) + (iii)(a) (XOR packet plus either packet A or packet B decoded).
- AB = (i)(a) + (i)(b) (both packets A and B decoded; XOR packet may or may not be decoded).

We performed controlled experiments for different SNRs. We adopted a coordinated MAC in which the AP transmits beacons to coordinate medium access by users [5]. For our experiments, the AP sent 10,000 beacons to trigger simultaneous transmissions. In our first experiment here, the receive powers of nodes A and B at the AP were adjusted to be balanced, with less than 0.5dB difference between the two powers. We varied the "common" SNR from 7.5 dB to 10dB. The SNR is calculated using the scheme in [22]. We performed the experiment 5 times for each SNR and averaged the results.

Recall from the discussion in Section 3 that the complementary XOR packets are useful for PHY-layer bridging and the lone XOR packets are useful for MAC-layer bridging. As shown in Fig. 10, there are considerable complementary XOR packets (AX|BX) and lone XOR packets (X) across all SNRs. At 8.5dB, in particular, the events associated with complementary packets and lone packets both happen more than 20% of the time. The contributions of these packets to the overall throughput of NCMA will become evident from the experimental results presented in Section 5.2.2.

5. Currently, the end nodes are always A and B. In a complete MAC as depicted in Fig. 5, the addresses of the two nodes being polled should be contained in the two frames. We have not implemented this part.

TABLE 1
Processing time of different PHY-layer decoders

Packet interval \ Decoder Type	RMUD	PNC	RMUD+PNC
6ms	0.949ms	0.809ms	1.217ms
7ms	0.774ms	0.690ms	1.018ms
8ms	0.756ms	0.677ms	0.926ms

In the experimental setting associated with our prototype, many effects (including CFO, channel estimation error, and other effects) may introduce degradations on the PHY-layer packet decoding statistics. To shed further light beyond the experimental results, we have conducted Matlab simulations based on pure AWGN channels without these extraneous effects. Appendix B contains the details. Specifically, the results in Appendix B gives an explanation as to why using a combination of PNC and MUD decoders may be a good idea.

Latency Measurements: As mentioned in Section 5.1, our NCMA PHY decoders are online decoders. To evaluate their real-time decoding performances, we now compare the decoding latencies of different online PHY decoders (namely the SU, RMUD and PNC decoders). For our latency measurements, the PHY-layer parameters were set as follows: (1) packet length = 1,500Bytes, (2) 5MHz channel bandwidth, and (3) BPSK modulation with rate 1/2 convolutional channel code. The accuracy of the USRP FPGA hardware time used to measure latency is on the order of 10^{-9}s. The packet processing delay for successfully decoded packet is defined as the time interval between the packet arrival time (using peak detection) and the time by which the decoded packet passes the CRC check. The average processing times based on 10,000 real runs are shown in Table 1. Thanks to the reduced constellation scheme, the processing delays of RMUD and PNC decoders are on the order of only 1ms. Meanwhile, as shown in Table 1, the packet inter-arrival time is more than 5ms. Thus, real-time decoding is possible.

5.2.2 Overall NCMA Performance

Next, we evaluate the overall NCMA performance. For benchmarking, let us first derive a theoretical upper bound for the overall NCMA normalized throughput imposed by the PHY-layer decoding performance. In each time slot, depending on the event, we could obtain either one or two equations for use in our three equation systems at the MAC layer. For example, when both packets A and B are obtained by PHY-layer decoding, then the normalized throughput for that particular time slot is 2. The upper bound for the normalized NCMA throughput averaged over all time

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

slots can be shown to be (note: a lone XOR packet also counts as one equation):

$$\begin{aligned}\text{Upper Bound} =& 2 \times (\Pr\{\mathsf{AB}\} + \Pr\{\mathsf{AX} \,|\, \mathsf{BX}\}) \\ &+ 1 \times (\Pr\{\mathsf{A} \,|\, \mathsf{B}\} + \Pr\{\mathsf{X}\}) \end{aligned} \quad (22)$$

This upper bound cannot exceed 2 since we allow at most two users to transmit together in our current prototype. It can be shown that even with MAC-layer bridging, the upper bound (22) cannot be exceeded.

We now present experimental results showing that NCMA can achieve normalized throughput close to upper bound (22). We employed **trace-driven simulations** using PHY-layer statistics obtained from our real experiments. Specifically, we first gathered the PHY-layer decoding statistics related to the eight events described in Section 2.2 (i.e., the probability of only packet A being decoded; the probability of only packet B being decoded; the probability of only packet $A \oplus B$ being decoded; the probability of both packets A and B being decoded but not the XOR packet; and so on). We then used the traces to create the events in our simulations. For example, Fig. 11 and Fig. 12 are derived using the PHY statistics shown in Fig. 10.

For NCMA, after the message of a source node is decoded, the source node begins sending another message, paired with the yet-to-be decoded message of the other node (see explanation in Section 3.2.4).

In Fig. 11, we show the normalized throughputs of NCMA with RMUD at SNR = 9.5 dB. In the figure, L_A (L_B) is the number of packets of M^A (M^B) the AP must have before it can decode M^A (M^B). The normalized throughput is defined as $Th = \frac{L_A \times N_A + L_B \times N_B}{N_{Beacon}}$, where N_A (N_B) is the number of messages of node A (B) that have been recovered, and N_{Beacon} is the total number of beacons (recall that each beacon prompts node A and node B to transmit one PHY packet). We can see from Fig. 11(a) that NCMA with RMUD outperforms RMUD with $L_A = 1.5L_B$ without the PNC decoder by 20%.

Note that making $L_A \neq L_B$ improves performance. In particular, making $L_A = 1.5L_B$ allows NCMA throughput to approach the upper bound. Although we have presented the mathematical formulation for the MAC-layer bridging assuming $L_A = L_B$ in Section 3.2, it also works when $L_A \neq L_B$. The mathematical formulation for MAC-layer bridging when $L_A \neq L_B$ can be found in Appendix C. The intuitive reason why unequal L can potentially be better is as follows. With respect to MAC-layer bridging, suppose that M^B is decoded first. As explained in Section 3.2, with the decoded M^B, MAC-layer bridging may allow us to obtain additional equations associated with M^A through lone XOR packets. Each lone XOR packet gives us one more equation for M^A. When $L_A = L_B$, the number of lone XOR packets may be "more than enough" (i.e., we end up having more than L_A equations after bridging). The extra lone XOR packets are

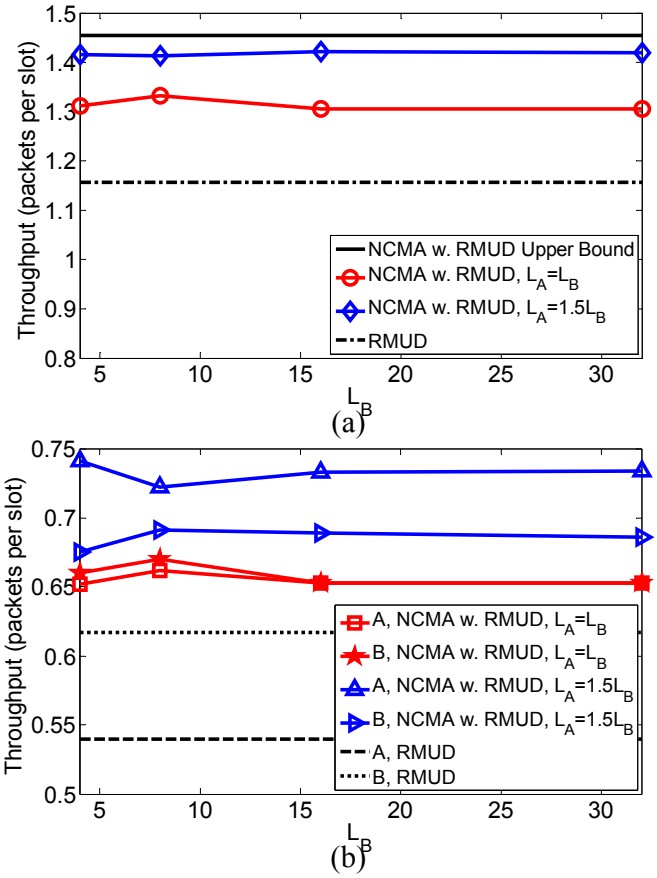

Fig. 11. Throughput comparison of NCMA with RMUD adopting $L_B = 4, 8, 16, 32$, and fixed $SNR_A = SNR_B = 9.5$dB: (a) aggregated throughput; (b) individual throughputs of A and B. Note that the upper bound and RMUD curves are benchmarks with constant values that do not vary with L_B.

then wasted, because they do not contribute to the throughput. By contrast, in upper bound (22), each lone XOR packet contributes 1 unit to the throughput. Thus, each wasted lone packet pulls the throughput further away from the upper bound. Intuitively, we can see that it is better for M^A not to be decoded at exactly the same time as M^B, in which case all the lone XOR packets contribute to the throughput. Making $L_A = 1.5L_B$ disaligns the decoding times of M^A and M^B. Other disalignment strategies for the messages are also possible; to limit scope, we will not delve further into that direction here.

To investigate the potential unfairness introduced by asymmetric L_A and L_B, we separately plot the performances of nodes A and B for NCMA in Fig. 11(b). From Fig. 11(b), we can see that with the asymmetric $L_A = 1.5L_B$, both the performances of node A and node B improve. However, node A improves more with its larger L_A. Thus, there is "unfairness". However, the use of larger L_A also improves B's performance. Specifically, the asymmetric case $L_A = 1.5L_B$ is more *Pareto-optimal* than the symmetric case

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

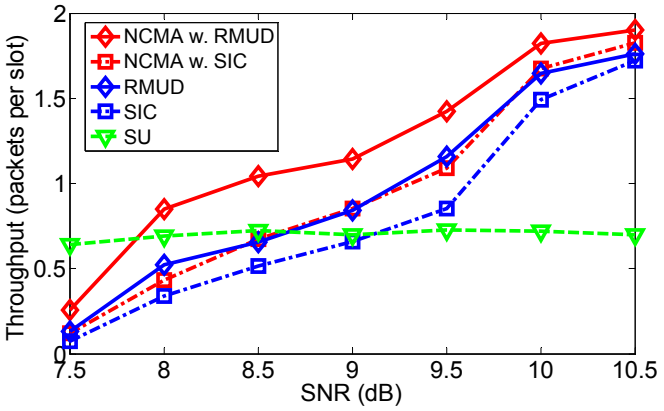

Fig. 12. Throughput comparison of different schemes for different SNRs ($SNR_A = SNR_B$) with $L_A = 1.5 \times L_B = 24$.

$L_A = L_B$, since both the throughputs of A and B are larger in the asymmetric case.

We also see from Fig. 11 that as long as $L_A = 1.5L_B$, the absolute value of L_B is not that important. For the next set of experiments, we fixed $L_B = 16$ and varied the SNR. Fig. 12 shows that NCMA outperforms SU by 100% when the average SNR≥9.5dB. At SNR≥9.5dB, NCMA has significantly better performance than RMUD and SIC without the PNC decoder. When SNR>10.5dB, all multiuser schemes (including MUD and NCMA) have good performance and they start to converge to throughput of 2.

Fig. 12 also includes the results for NCMA with SIC. We see that its performance is not as good as that of NCMA with RMUD. This is understandable because Fig. 12 is related to the balanced-power case, and SIC is known to have poor performance when powers from different users are balanced [15].

5.2.3 Effects of Unbalanced-Power User Pairing

We next investigate what happens when the receive powers of users A and B are different. Fig. 13 presents the results. As with the study in Section 5.2.2, we performed trace-driven simulations based on PHY-layer decoding statistics gathered from real experiments (with $L_B = 16$ and $L_A = 1.5L_B = 24$). The SNR of A was fixed to 7.5dB and 9.5dB in Fig. 13(a) and Fig. 13(b), respectively. For each fixed power, we varied the SNR of B.

Our experiments indicate that NCMA is not only robust against power imbalance, but its performance can actually be better under unbalanced-power user pairing. Our finding that unbalanced power among the colliding packets improves the SIC performance is consistent with the theoretical results on coded random access channel [23], [24].

In Fig. 13(a) and Fig. 13(b), note that for a fixed SNR_A, as SNR_B increases, not only does the throughput of B improve, the throughput of A also improves. For example, when SNR_A is fixed at 7.5dB,

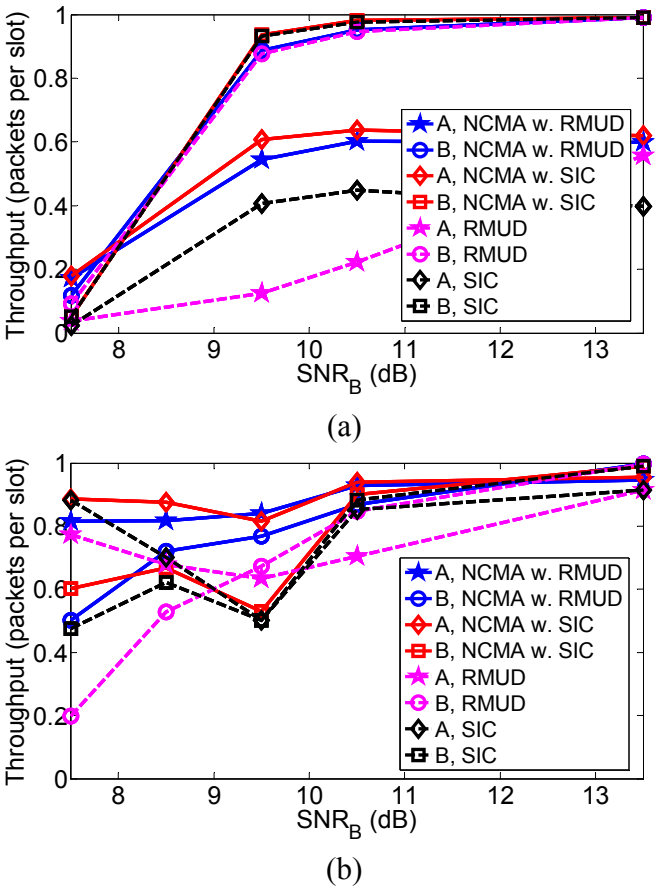

Fig. 13. Throughputs of node A and node B, with SNR of node A fixed at (a) $SNR_A = 7.5$dB and (b) $SNR_A = 9.5$dB, and SNR of node B varied.

the throughput of A could be improved by 400% as SNR_B increases from 7.5dB to 8.5dB.

To better analyze the results, we adopt the following notations:

- $(Th'_A, Th'_B) >> (Th_A, Th_B)$ means $Th'_A > Th_A$ and $Th'_B > Th_B$.
- $(Th'_A, Th'_B) <> (Th_A, Th_B)$ means $Th'_A < Th_A$ and $Th'_B > Th_B$.
- $(Th'_A, Th'_B) >< (Th_A, Th_B)$ means $Th'_A > Th_A$ and $Th'_B < Th_B$.
- $Th^{NR} = (Th^{NR}_A, Th^{NR}_B)$ stands for Throughputs of NCMA-RMUD.
- $Th^{NS} = (Th^{NS}_A, Th^{NS}_B)$ stands for Throughputs of NCMA-SIC.
- $Th^{S} = (Th^{S}_A, Th^{S}_B)$ stands for Throughputs of SIC.
- $Th^{R} = (Th^{R}_A, Th^{R}_B)$ stands for Throughputs of RMUD.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

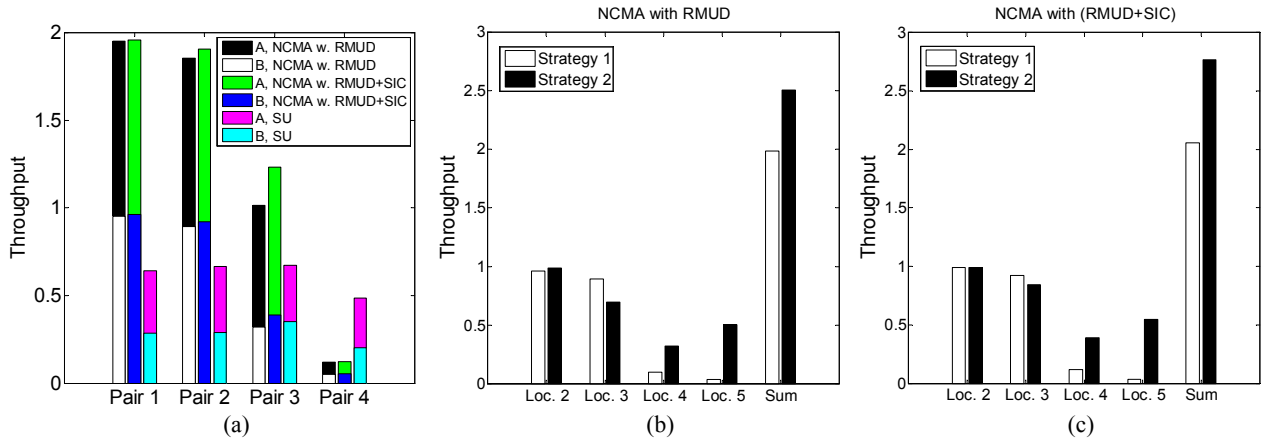

Fig. 14. Throughputs with AP placed at location 9 in Fig. 9: (a) Throughputs of four user pairs, P1, P2, P3, and P4 in Table 2 under NCMA and SU; (b) Throughputs of two different user-pairing strategies under NCMA with RMUD; (c) Throughputs of two different user-pairing strategies under NCMA with (RMUD+SIC).

TABLE 2
User pairing in random topology

User Pair	User A	User B
P1	Location 1 (20dB)	Location 2 (12.3dB)
P2	Location 2 (12.3dB)	Location 3 (9dB)
P3	Location 3 (9dB)	Location 4 (7dB)
P4	Location 4 (7dB)	Location 5 (7.4dB)
P5	Location 2 (12.3dB)	Location 5 (7.4dB)

Let us focus on Fig. 13(b), where $SNR_A = 9.5$dB[6]. At $SNR_B = 7.5$dB, NCMA-SIC has the best performance, i.e., $Th^{NS} >> Th^{NR}$, Th^S, or Th^R. In particular, there is an appreciable gap between Th_B^{NS} and Th_B^S, implying that the use of the PNC decoder can improve the performance of the weak user significantly. At $SNR_B = 8.5$dB, both $Th^{NS} >> Th^S$ or Th^R and $Th^{NR} >> Th^S$ or Th^R. Using SIC alone without the PNC decoder will result in inferior performance. Meanwhile, $Th^{NR} <> Th^{NS}$. At $SNR_B = 9.5$dB, we have the balanced power case. Now, $Th^{NR} >> Th^{NS}$, Th^S, or Th^R, indicating that RMUD is a good complement to SIC so that the Achilles's heel of SIC, the balanced power case, can be overcome. At $SNR_B \geq 10.5$dB, both NCMA-RMUD and NCMA-SIC have good performance. Overall, NCMA-RMUD has the "smoothest performance" (no large fluctuations in performance) when SNR_B is varied.

Which NCMA Variants to Use

Our experimental results have validated the worthiness of introducing PNC decoding in the multi-access system. Overall, NCMA-RMUD and NCMA-SIC, in which PNC decoding is used, have better performance than the corresponding RMUD and SIC, respectively.

A question is which variant of NCMA should be adopted. Fig. 13(b) indicates that compared with NCMA-SIC, NCMA-RMUD has smoother performance transition from the unbalanced power setting to the balanced power setting. That is, it works well for both settings and is more robust if the system does not allow us to fine-tune the relative powers of user pairs. Furthermore, SIC requires a second round of PHY decoding after interference cancellation (which introduces processing delay), while RMUD performs all its decoding in one round. Thus, for a simple implementation that has low latency and good performance across all relative power settings, NCMA-RMUD is perhaps preferred. However, if complexity and latency are not an issue, then NCMA that uses both RMUD and SIC (rather than just one of them), namely NCMA with (RMUD+SIC), should have the best throughput performance.

5.2.4 *Random Topology Experiments*

We now present the experimental results of NCMA with RMUD for a random topology. The NCMA MAC-layer RS code parameters are $L_B = 16$ and $L_A = 1.5L_B = 24$. The random topology was constructed from the set-up in Fig. 9, without deliberate power control, i.e., we placed the USRPs in different locations in an indoor environment, and each of them used a fixed transmit power. We report the results of a particular set of experiments with the AP placed at location 9 in Fig. 9. Users were placed at locations 1, 2, 3, 4, and 5. The associated SNRs for these locations are 20dB, 12.3dB, 9dB, 7dB, and 7.4dB, respectively. Five user pairs P1, P2, P3, P4, and P5 as shown in Table 2 were formed for experimentation purposes.

Fig. 14(a) shows the throughputs of different user pairs P1, P2, P3, and P4, under NCMA and SU. We see

6. We remark that the throughput of NCMA is 1.4 at 9.5dB in Fig. 12. However, in Fig. 13(b) the sum throughput is 0.8 + 0.7 = 1.5dB. This slight difference between Fig. 12 and Fig. 13(b) at 9.5dB is caused by different experimental runs. For example, the reception power of 9.5dB for each run is the mean value of 10,000 packets, and the wireless fading may be slightly different between different runs.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

that NCMA outperforms SU significantly, except for P4. The overall system throughput improvement of NCMA over SU is 100%. In P1, P2, and P3, although the powers are unbalanced, at least one of the users has decent power. We also note that for the power-unbalanced P1, P2, and P3, NCMA with RMUD and SIC has slightly better performance than NCMA with RMUD. In P4, the powers of both users are low, a regime where multi-user decoders do not work well.

The above observation on P1, P2, P3, and P4 implies that it is better to pair a user with strong power with a user with weak power. To better articulate this, consider a situation in which there are four users in the network at locations 2, 3, 4, and 5. We compare two strategies of user pairing. In Strategy 1, we form the user pairs P2 and P4 out of the four users; in Strategy 2, we form user pairs P3 and P5. As shown in Fig. 14(b) and (c), Strategy 2 has better overall performance. Note that in Strategy 2, we avoid pairing two weak users together as in P4 of Strategy 1. As a result, the throughputs of the two weak users at locations 4 and 5 are pulled up significantly by their stronger partners. Not only is Strategy 2 more fair, its overall system throughput is also 20% higher than that of Strategy 1.

6 Related Work

Network coding (NC) has been implemented and evaluated in wireless networks at the PHY layer [1], [14], [25], and the network layer [26]–[28]. However, the previous studies of NC have generally been restricted to relay networks, where NC was originally shown to be helpful for packet exchange via relays. NCMA, on the other hand, targets the non-relay setting. NC has also found use in packet retransmission [27], [28]. NCMA, by contrast, aims directly at packet transmission rather than retransmission.

Refs. [29] and [30] explored how to resolve collisions among packets of different users. Instead of discarding the collided packets, a set of linear equations are formed to exploit information contained in them. The decoding is based on PHY-layer equations only. In addition, the decoding methods are either pure MUD or pure PNC methods. By contrast, NCMA introduces correlations among PHY packets so that another layer of MAC decoding can be used to improve performance. Also, NCMA makes use of both MUD and PNC at the PHY layer in a complementary way.

In OFDMA, different users transmit their signals on different subcarriers [19], [31]. By contrast, in NCMA, multiple users use the same set of subcarriers when they transmit concurrently. This improves spectrum efficiency. In [32], there could be overlapped subcarriers between two users; however, at least some of the subcarriers must be non-overlapping for successful packet recovery. Similarly, as a time-domain WLAN system, the set-up in [33] requires some symbols to be non-overlapping in time to bootstrap the packet recovery algorithm.

Refs. [23] and [24] discussed the power control and performance evaluation on random access channel with success interference cancellation. Recently, interference cancellation techniques have been advanced and applied to the decoding of PHY-layer rateless codes [34]–[36]. For these techniques, the processing is entirely on signal samples rather than on bits. While having good performance, the decoding procedure could incur considerable storage and computation costs. NCMA opts for reduced complexity for simple PHY-layer decoding with real-time performance. The correlations among different PHY packets are exploited in MAC-layer decoding, which deals with bits rather than samples. Our design is largely compatible with the processing flows of the current wireless standards (e.g., 802.11).

7 Discussions

Higher Throughput in Medium to High SNR Regimes – We have presented the performance of our NCMA system based on the BPSK modulation, with SNRs ranging from 7dB to 10.5dB. The highest normalized throughput is bounded by 2 because at most two users are allowed to transmit concurrently. We believe that the throughput can be further increased by allowing more than two users to transmit together.

We also note that when the SNRs are higher than 10dB, higher order modulations (e.g., QPSK and 16-QAM) could be applied [22] to better utilize the available power to improve the absolute throughput (in bps). Appendix D explains the operation principle of RMUD under QPSK. Similar procedures can be extrapolated for even higher-order modulations.

Better Performance in Low SNR Regime – For SNR below 7.5dB, our current decoders do not work well because the approximations adopted (see Section 4) assume the SNR is sufficiently large. The refinement of our approximations for low SNR decoding awaits future work. In the meantime, in this paper, we have shown that pairing a weak user with a strong user can substantially alleviate the problem. In particular, not only the throughput of the strong user improves, the throughput of the weak user also improves, under such pairing.

More Efficient MAC-layer Bridging – In this paper, we use three equation systems to tackle the decoding problem. We can also combine everything into one equation system, in which the information from all individual packets and XOR packets is embedded. In particular, we can put all the received packets in a matrix form. If the rank of the matrix is full, we can decode the packets. This formulation is part of our future work.

Combined Use with Other Advanced Techniques – Recently, distributed MIMO and full-duplex techniques

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

are very active areas in wireless network research. Our current NCMA prototype does not make use of these techniques. NCMA is, however, complementary to these techniques. The combination of MIMO and full-duplex technologies with NCMA is a promising direction for further work.

8 Conclusions

We have conceptualized and experimentally demonstrated a first WLAN system that employs the concept of Network-Coded Multiple Access (NCMA). The key idea of NCMA is to jointly exploits PNC and MUD to decode concurrent transmissions by multiple users, through two processes called PHY-layer bridging and MAC-layer bridging. Our USRP prototype reveals that NCMA can boost throughput by 100% in the medium-high SNR regime ($\geq$10dB) compared with the traditional single-user transmission system.

Appendix A
Quantization of Soft Information

The quantized value of the soft information depends on the quantized values of the reduced constellation points. To set the quantized values of the constellation points, consider the noiseless case. There are four possible distinct values for $y[k]$, corresponding to the four constellation points. The four constellation points need to be reduced to two constellation points before the standard VA can be used. Consider the PNC decoder. Suppose that the constellation point $(+1,+1)$ is transmitted so that $y[k] = h_A[k] + h_B[k]$. Then using the approximation method in Section 4, for this noiseless case, we can get

$$\begin{aligned}\tilde{x}_{A\oplus B}[k] &\approx \min\left(h_A[k], h_B[k]\right) \cdot \left(y[k] - \max(h_A[k], h_B[k])\right) \\ &= \min\left(|h_A[k]|^2, |h_B[k]|^2\right). \end{aligned} \quad (23)$$

Considering all the four constellation points under noiseless reception, it can be shown that there are only two reduced values for $\tilde{x}_{A\oplus B}[k]$, $\pm\min(|h_A[k]|^2, |h_B[k]|^2)$. These two values are the two effective constellation points for bit k. Over all k, we can focus on the pair of reduced constellation points with the largest magnitude given by

$$\pm|h_{\max}|^2 \triangleq \pm \max_k \min(|h_A[k]|^2, |h_B[k]|^2). \quad (24)$$

For quantization, we first map $\pm|h_{\max}|^2$ to two points on the real interval $[-0.5, 0.5]$. Specifically, for a defined $\alpha \in (0, 0.5)$ (e.g., $\alpha = 0.25$), we map $|h_{\max}|^2 \to \alpha$ and $-|h_{\max}|^2 \to -\alpha$. Then, we normalize and quantize $\tilde{x}_{A\oplus B}[k]$ as

$$\tilde{x}_{A\oplus B,quantized}[k] = \left(\frac{\tilde{x}_{A\oplus B}[k]}{|h_{\max}|^2}\alpha + 0.5\right) \times 255. \quad (25)$$

For the MUD decoder that decodes A, using the same reasoning, we have

$$\pm|h_{\max}|^2 \triangleq \pm \max_k |h_A[k]|^2. \quad (26)$$

$$\tilde{x}_{A,quantized}[k] = \left(\frac{\tilde{x}_A[k]}{|h_{\max}|^2}\alpha + 0.5\right) \times 255. \quad (27)$$

For packet B, we just replace A by B in the above formula.

We find that, empirically, optimal performance can be obtained at α between 0.2 and 0.25, and performance does not vary much within this range. Our experiments were conducted with $\alpha = 0.228$.

Quantizations of the RMUD and SIC decoders can be performed based on the same principle as above.

Appendix B
PHY-layer Packet Decoding Statistics in an AWGN Channel

This appendix presents the Matlab simulation results for our PNC decoder and MUD decoder with balanced powers between users A and B, and perfect channel estimation. We assume an ideal AWGN channel and the use of BPSK modulation. We do not assume OFDM in these simulations, and there is no concept of different subcarriers. All symbols of user A experience the same noise and phase. Likewise for user B. The phases of user A and user B, however, may be different.

Fig. 15(a) shows the decoding statistics when the signals of A and B are phase-aligned. In this case, the PNC decoder can often decode the XOR packets, but the MUD decoder fail to decode the individual packets of A and B. This is because it is difficult to distinguish the individual signals of A and B when their phases are aligned. For example, if 0 is received, it is not clear that whether A transmits $+1$ and B transmits -1, or A transmits -1 and B transmits $+1$. Either case will produce a 0 at the receiver. Thus, it is difficult to decode packet A or packet B. However, as far as the XOR bit is concerned, it is 0, and only the XOR bit is required for XOR decoding of the packet.

On the other hand, when the relative phase starts to depart from 0 and becomes $\pi/4$ in Fig. 15(b) and $\pi/2$ in Fig. 15(c) (for $\pi/2$, the signals of A and B are totally orthogonal), it becomes increasingly easier to decode the individual packets of A and B.

The above results give an intuitive explanation as to why it is desirable to use a combination of the PNC decoder and MUD decoder in practice: different relative phases are possible in a practical system. For our OFDM system, things are even more complicated because the different subcarrier pairs of A and B may experience different relative phases, with some phases closer to 0 while other phases closer to $\pi/2$. The combined use of the PNC and MUD decoders

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

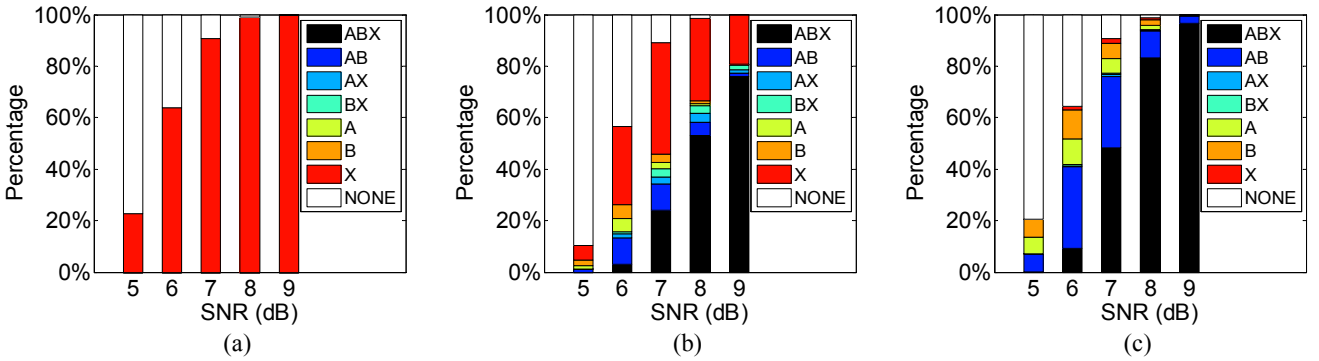

Fig. 15. NCMA PHY-layer packet decoding statistics versus SNR in an AWGN channel with phase difference between two end nodes being (a) 0; (b) $\frac{\pi}{4}$; (c) $\frac{\pi}{2}$ (X means only the XOR packet is decoded; AB means both packet A and packet B are decoded; and so on).

allows the system to adapt to the channel phases dynamically.

APPENDIX C
Eq^A, Eq^B, AND $Eq^{A\oplus B}$ FOR ASYMMETRIC L_A AND L_B

Without loss of generality, suppose that $L_B > L_A$. Let G in (2) be the matrix with $L = L_B$ columns, and let $G^{(-)}$ denote a truncated matrix of G with the last $L_B - L_A$ columns truncated. In other words, $G^{(-)}$ is the generator matrix of A and G is the generator matrix of B:

$$\begin{aligned} C^A &= G^{(-)}M^A, \\ C^B &= GM^B. \end{aligned} \tag{28}$$

Note that with respect to the original form in (1), there are L_B rows in M^B and L_A rows in M^A. Define padded message A as $M^{A(+)} = \begin{bmatrix} M^A \\ 0 \end{bmatrix}$, where $L_B - L_A$ rows of zeros are added to M^B to form the matrix $M^{A(+)}$ with L_A rows. Thus, we can also write

$$C^A = GM^{A(+)}. \tag{29}$$

Thus, we also have

$$C^{A\oplus B} = G(M^{A(+)} \oplus M^B). \tag{30}$$

Henceforth, by the notation $Q^{(-)}$, we mean the matrix Q with the last $L_B - L_A$ columns (or rows) truncated; and by $Q^{(+)}$, we mean the matrix Q with $L_B - L_A$ zero columns (or rows) padded. Whether it is the columns or rows that are truncated or padded depend on whether Q is a generator matrix of type G or a message matrix of type M. If Q is of type G (M), then column (row) truncation or padding applies.

C.1 MAC-layer Bridging if Eq^A is solved first

With respect to the discussions in Section 3.2.1, suppose we can solve Eq^A before Eq^B and $Eq^{A\oplus B}$. Then (7) is replaced by

$$M^A = (\tilde{G}^{A(-)})^{-1}\tilde{C}^A. \tag{31}$$

In addition, (8) is replaced by

$$\begin{aligned} &\bar{G}^{A\oplus B}(M^{A(+)} \oplus M^B) \\ &= \bar{G}^{A\oplus B(-)}M^A \oplus \bar{G}^{A\oplus B}M^B = \bar{C}^{A\oplus B}, \end{aligned} \tag{32}$$

and (10) is replaced by

$$\begin{bmatrix} \tilde{G}^B \\ \tilde{G}^{A\oplus B} \end{bmatrix} M^B = \begin{bmatrix} \tilde{C}^B \\ \bar{C}^{A\oplus B} \oplus \bar{G}^{A\oplus B(-)}(\tilde{G}^{A(-)})^{-1}\tilde{C}^A \end{bmatrix}. \tag{33}$$

C.2 MAC-layer Bridging if Eq^B is solved first

If Eq^B is solved first, then through a similar MAC-layer bridging process, the new Eq^A is

$$\begin{bmatrix} \tilde{G}^{A(-)} \\ \bar{G}^{A\oplus B(-)} \end{bmatrix} M^A = \begin{bmatrix} \tilde{C}^A \\ \bar{C}^{A\oplus B} \oplus \bar{G}^{A\oplus B}(\tilde{G}^A)^{-1}\tilde{C}^A \end{bmatrix}. \tag{34}$$

C.3 MAC-layer Bridging if $Eq^{A\oplus B}$ is solved first

Suppose that $Eq^{A\oplus B}$ is solved first. We need $L_X = \max(L_A, L_B)$ XOR equations in order to declare that $Eq^{A\oplus B}$ can be solved. In our example here, $L_B > L_A$ so that $L_X = L_B$. Now, since $Eq^{A\oplus B}$ is solvable, $\tilde{G}^{A\oplus B}$ in (32) is invertible. We have

$$M^{A(+)} \oplus M^B = (\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B}. \tag{35}$$

(13) can be written as

$$\begin{aligned} \bar{G}^A[M^B \oplus (\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B}] &= \bar{C}^A, \\ \bar{G}^B[M^{A(+)} \oplus (\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B}] &= \bar{C}^B. \end{aligned} \tag{36}$$

And (14) can be written as

$$\begin{aligned} \begin{bmatrix} \tilde{G}^{A(-)} \\ \bar{G}^{B(-)} \end{bmatrix} M^A &= \begin{bmatrix} \tilde{G}^A \\ \bar{G}^B \end{bmatrix} M^{A(+)} \\ &= \begin{bmatrix} \tilde{C}^A \\ \bar{C}^B \oplus \bar{G}^B(\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B} \end{bmatrix}, \\ \begin{bmatrix} \tilde{G}^B \\ \bar{G}^A \end{bmatrix} M^B &= \begin{bmatrix} \tilde{C}^B \\ \bar{C}^A \oplus \bar{G}^A(\tilde{G}^{A\oplus B})^{-1}\tilde{C}^{A\oplus B} \end{bmatrix}. \end{aligned} \tag{37}$$

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

APPENDIX D
REDUCED CONSTELLATION DECODER FOR QPSK MODULATION

Assuming QPSK modulation, each symbol has four possible constellation points. If two users transmit simultaneously, then there are 16 possible constellation points. In our current prototype, we adopt a convolutional code and use the standard Viterbi algorithm (VA) for decoding at the PHY layer. The standard VA, however, is designed for SU operation. In particular, the VA decoder can only accept the binary log-likelihood ratios of the received bits as its inputs.

In order to use the standard VA for RMUD decoding, we first reduce the number of constellation points in the multi-user system from 16 to two (note that each QPSK symbol has two bits and let us focus on the first bit of user A), as follows. Let y_k be the received signal for the k-th symbol. We first find the Euclidean distances of y_k to the eight constellation points corresponding to user A's first bit being a $+1$ (i.e., $(1+j, 1+j)$, $(1+j, -1+j)$, $(1+j, -1-j)$, $(1+j, 1-j)$, $(1-j, 1+j)$, $(1-j, -1+j)$, $(1-j, -1-j)$, and $(1-j, 1-j)$). Among these eight points, we choose the constellation point with the smallest Euclidean distance as the representative and discard the rest. The same is done for user A's first bit being a -1, and we obtain the corresponding Euclidean distance. These two are the more likely constellation points for the first bit's two values. We use the log-likelihood ratio of these two constellation points as the soft input to the standard VA. Similar, we can obtain the soft input for the second bit of user A and the bits for user B.

ACKNOWLEDGMENTS

This work is supported by AoE grant E-02/08 and the General Research Funds Project Number 414812, established under the University Grant Committee of the Hong Kong Special Administrative Region, China. This work is also supported by NSF of China (Project No. 61271277).

REFERENCES

[1] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: physical-layer network coding," in *ACM MOBICOM*, 2006.
[2] S. Verdú, *Multiuser Detection*. Cambridge University Press, 1998.
[3] P. Popovski and H. Yomo, "The anti-packets can increase the achievable throughput of a wireless multi-hop network," in *Proc. IEEE Int. Conf. on Comm. (ICC)*, 2006.
[4] B. Nazer and M. Gastpar, "Reliable physical layer network coding," *Proceedings of the IEEE*, vol. 99, no. 3, pp. 438–460, 2011.
[5] S. Liew, S. Zhang, and L. Lu, "Physical-layer network coding: Tutorial, survey, and beyond," *Physical Communication*, vol. 6, no. 1, pp. 4–42, 2013.
[6] S. Wicker and V. Bhargava, *Reed-Solomon codes and their applications*. Wiley-IEEE Press, 1999.
[7] A. Jacobs, "The pathologies of big data," *ACM Queue*, 2009.
[8] D. MacKay, "Fountain codes," *Communications, IEE Proceedings*, vol. 152, no. 6, pp. 1062–1068, 2005.
[9] G. D. Forney Jr, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.
[10] IEEE 802.11-2009, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 5: Enhancements for higher throughput."
[11] T. Lee, S. Liew, and Q. Ding, "Parallel communications for atm network control and management," *Performance evaluation*, vol. 30, pp. 243–264, 1997.
[12] B. Sklar, *Digital Communications: Fundamentals and Applications (2nd Edition)*. Prentice Hall PTR, 2003.
[13] L. Lu and S. C. Liew, "Asynchronous physical-layer network coding," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 819–831, Feb. 2012.
[14] L. Lu, T. Wang, S. Liew, and S. Zhang, "Implementation of physical-layer network coding," *Physical Communication*, vol. 6, no. 1, pp. 74–87, 2013.
[15] D. Halperin, T. Anderson, and D. Wetherall, "Taking the sting out of carrier sense: interference cancellation for wireless LANs," in *ACM MOBICOM*, 2008.
[16] Spiral Project, "Viterbi decoder software generator," www.spiral.net/software/viterbi.html.
[17] Ettus Inc., "Universal software radio peripheral."
[18] S. Jakubczak, "Rawofdm: A raw I/Q-input OFDM and a fast QAM+FEC implementation for GNURadio," http://people.csail.mit.edu/szym/rawofdm/README.html.
[19] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang, "Fine-grained channel access in wireless LAN," in *ACM SIGCOMM*, 2010.
[20] L. Lu, L. You, Q. Yang, T. Wang, M. Zhang, S. Zhang, and S. C. Liew, "Real-time implementation of physical-layer network coding," in *ACM SRIF*, 2013, pp. 71–76.
[21] H. Rahul, H. Hassanieh, and D. Katabi, "Sourcesync: a distributed wireless architecture for exploiting sender diversity," in *ACM SIGCOMM*, 2010.
[22] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Predictable 802.11 packet delivery from wireless channel measurements," in *ACM SIGCOMM*, 2010.
[23] C. Xu, L. Ping, P. Wang, S. Chan, and X. Lin, "Decentralized power control for random access with successive interference cancellation," *IEEE Jour. Select. Areas in Comm.*, vol. 31, no. 8, pp. 1–10, 2013.
[24] E. Paolini, G. Liva, and M. Chiani, "High throughput random access via codes on graphs: Coded slotted aloha," in *Proc. IEEE Int. Conf. on Comm. (ICC)*, 2011, pp. 1–6.
[25] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: analog network coding," in *ACM SIGCOMM*, 2007.
[26] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: practical wireless network coding," in *ACM SIGCOMM*, 2006.
[27] S. Katti, D. Katabi, H. Balakrishnan, and M. Médard, "Symbol-level network coding for wireless mesh networks," in *ACM SIGCOMM*, 2008.
[28] E. Rozner, A. Iyer, Y. Mehta, L. Qiu, and M. Jafry, "ER: efficient retransmission scheme for wireless lans," in *ACM CoNEXT*, 2007.
[29] T. Li, M. Han, A. Bhartia, L. Qiu, E. Rozner, Y. Zhang, and B. Zarikoff, "CRMA: collision-resistant multiple access," in *ACM MOBICOM*, 2011.
[30] M. Khabbazian, F. Kuhn, N. Lynch, M. Médard, and A. ParandehGheibi, "MAC design for analog network coding," in *ACM FOMC*, 2011.
[31] L. Yang, W. Hou, L. Cao, B. Zhao, and H. Zheng, "Supporting demanding wireless applications with frequency-agile radios," in *USENIX NSDI*, 2010.
[32] L. Li, K. Tan, H. Viswanathan, Y. Xu, and Y. Yang, "Retransmission $\neq$ repeat: simple retransmission permutation can resolve overlapping channel collisions," in *ACM MOBICOM*, 2010.
[33] S. Gollakota and D. Katabi, "Zigzag decoding: combating hidden terminals in wireless networks," in *ACM SIGCOMM*, 2008.
[34] J. Perry, P. Iannucci, K. Fleming, H. Balakrishnan, and D. Shah, "Spinal codes," in *ACM SIGCOMM*, 2012.
[35] A. Gudipati and S. Katti, "Strider: Automatic rate adaptation and collision handling," in *ACM SIGCOMM*, 2011.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2014.2316804, IEEE Transactions on Mobile Computing

[36] A. Gudipati, S. Pereira, and S. Katti, "AutoMAC: rateless wireless concurrent medium access," in *ACM MOBICOM*, 2012.

Lu Lu received his B.E. and Ph.D. degrees from the University of Science and Technology of China (USTC) and The Chinese University of Hong Kong (CUHK), in 2007 and 2012, respectively. He is now a postdoctoral fellow at the Institute of Network Coding, CUHK. At CUHK, he received the Outstanding Ph.D. Thesis Award and the Postgraduate Research Output Award both in 2013. His current research interests include wireless communication in 802.11 networks, multi-user detection, physical-layer network coding, and software-defined radios.

Lizhao You received his B.S. and M.E. degrees from Nanjing University in 2009 and 2013, respectively. He is currently a Ph.D. student at the Department of Information Engineering, The Chinese University of Hong Kong. His research interests include wireless networks, computer networks and distributed systems.

Soung Chang Liew received his S.B., S.M., E.E., and Ph.D. degrees from the Massachusetts Institute of Technology. From 1984 to 1988, he was at the MIT Laboratory for Information and Decision Systems, where he investigated Fiber-Optic Communications Networks. From March 1988 to July 1993, he was at Bellcore (now Telcordia), New Jersey, where he engaged in Broadband Network Research. He has been a Professor at the Department of Information Engineering, the Chinese University of Hong Kong, since 1993. He is an Adjunct Professor at Peking University and Southeast University, China.

Prof. Liew's research interests include wireless networks, Internet protocols, multimedia communications, and packet switch design. Prof. Liew's research group won the best paper awards in *IEEE MASS* 2004 and *IEEE WLN* 2004. Separately, TCP Veno, a version of TCP to improve its performance over wireless networks proposed by Prof. Liew's research group, has been incorporated into a recent release of Linux OS. In addition, Prof. Liew initiated and built the first inter-university ATM network testbed in Hong Kong in 1993. More recently, Prof. Liew's research group pioneers the concept of *Physical-layer Network Coding* (PNC).

Besides academic activities, Prof. Liew is active in the industry. He co-founded two technology start-ups in Internet Software and has been serving as a consultant to many companies and industrial organizations.

Prof. Liew is the holder of nine U.S. patents and a Fellow of IEEE, IET and HKIE. He currently serves as Editor for IEEE Transactions on Wireless Communications and Ad Hoc and Sensor Wireless Networks. He is the recipient of the first Vice-Chancellor Exemplary Teaching Award in 2000 and the Research Excellence Award in 2013 at the Chinese University of Hong Kong. Publications of Prof. Liew can be found in www.ie.cuhk.edu.hk/soung.

1536-1233 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.