

A Hybrid Network Coding Technique for Single-Hop Wireless Networks

Tuan Tran, Thinh Nguyen, *Member, IEEE*, Bella Bose, *Fellow, IEEE* and Vinodh Gopal

Abstract—In this paper, we investigate a hybrid network coding technique to be used at a wireless base station (BS) or access point (AP) to increase the throughput efficiency of single-hop wireless networks. Traditionally, to provide reliability, lost packets from different flows (applications) are retransmitted separately, leading to inefficient use of wireless bandwidth. Using the proposed hybrid network coding approach, the BS encodes these lost packets, possibly from different flows together before broadcasting them to all wireless users. In this way, multiple wireless receivers can recover their lost packets simultaneously with a single transmission from the BS. Furthermore, simulations and theoretical analysis showed that when used in conjunction with an appropriate channel coding technique under typical channel conditions, this approach can increase the throughput efficiency up to 3.5 times over the Automatic Repeat reQuest (ARQ), and up to 1.5 times over the HARQ techniques.

Index Terms—Network Coding, Channel Coding, Wireless LAN, WiMAX.

I. INTRODUCTION

IN TODAY communication networks such as the Internet and wireless ad hoc networks, data delivery is performed via store-and-forward routing. That is, intermediate routers do not alter the content of the packets as they traverse hop-by-hop from a source to a destination. In contrast, network coding (NC) [1] is the generalized approach to packet routing that allows an intermediate router to encode an outgoing packet by mixing multiple incoming packets appropriately. In this way, it is theoretically possible to achieve the throughput capacity of an arbitrary multicast session, while this is not possible with the traditional store-and-forward routing techniques.

However, supporting sophisticated functionalities at intermediate routers goes against the end-to-end design principle by Saltzer *et al.* [2] which argues for simple routers to increase performance and scalability. On the other hand, it is possible to employ NC at places where additional complexity can be justified, e.g., wireless base stations (BS) in WiMAX networks or access points (AP) in Wi-Fi networks. That said, in this paper, we consider the scenarios where the BS/AP has the ability to intercept and mix packets belonging to different flows from the Internet to multiple wireless users.

Manuscript received 1 August 2008; revised 10 January 2009. The work of T. Nguyen was supported in part by CAREER CNS-0845476. The work of B. Bose was supported in part by CCF-0728810 and CCF-0701452. This paper was presented in part at the Fourth Workshop on Network Coding, Theory and Applications (NetCod), Hong Kong, January 2008.

T. Tran, T. Nguyen and B. Bose are with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, 97331 USA (e-mail: trantu, thinhq, bose@eecs.oregonstate.edu).

V. Gopal is with Intel Corporation, USA (e-mail: vinodh.gopal@intel.com).
Digital Object Identifier 10.1109/JSAC.2009.090610.

Let us consider a TCP flow originates from a source in the Internet and terminates at a wireless receiver. If a packet is lost at the last mile wireless link, this packet is automatically retransmitted from the source, not from the BS. This design follows the end-to-end argument in keeping the functionality of the BS simple. On the other hand, this approach has been shown to be bandwidth inefficient due to the adverse affect it has on TCP [3]. In this paper, we also argue for breaking the end-to-end principle, but from a coding perspective to increase the wireless throughput efficiency. Specifically, we show that the wireless bandwidth can be efficiently utilized by allowing retransmissions to be performed at the BS, and *more importantly, by proper mixing of lost packets from multiple flows*. This is in stark contrast to the existing techniques such as the Automatic Repeat (ARQ) or Hybrid-ARQ (HARQ) protocols where lost packets from different flows are retransmitted individually.

That said, existing approaches to transmit information reliably and effectively over an error-prone network employ either the Auto Repeat reQuest (ARQ), Forward Error Correction (FEC), or Hybrid ARQ (HARQ) techniques [4]. Using the retransmission approach, the source simply retransmits the lost data. This approach assumes that the receivers can somehow communicate to the source whether or not it receives the correct data. On the other hand, using the FEC approach, the source encodes additional information together with the original data before broadcasting them to the receivers. If the amount of lost data is sufficiently small, a receiver can recover the lost data using some decoding techniques. The HARQ approach combines both of those techniques.

The HARQ techniques have been shown to be quite effective in many wireless transmission scenarios. As such, our proposed technique employs both the NC and HARQ approaches (NC-HARQ) to increase the throughput efficiency in single-hop wireless networks such as Wi-Fi or WiMAX. In particular, the BS or AP does not retransmit a lost packet belonging to a particular flow immediately. Rather, it maintains a queue of lost packets from all the flows, and periodically retransmits the appropriately *coded* packets to all the wireless users. A *coded* packet is formed by performing bit-wise exclusive-or of multiple lost packets in the queue. Assuming that a receiver can hear and cache all the transmissions, including transmissions for other receivers, using this method, one transmission from the BS enables multiple receivers to recover their lost packets simultaneously. Furthermore, we show that, adding the right amount of Forward Error Correction (FEC) can result in much higher throughput efficiency. Specifically, our contributions include some analytical results on the throughput

efficiencies of the proposed and existing techniques, together with a heuristic algorithm that dynamically selects the optimal amount of FEC for the given channel conditions.

The organization of our paper is as follows. We first discuss some related work in Section II. In Section III, we describe the problem formulation in the context of Wi-Fi/WiMAX networks. In Section IV, we provide some theoretical analysis on the performance of ARQ, HARQ, the proposed NC and NC-HARQ techniques under different channel conditions. Based on these analysis, we describe a heuristic algorithm that dynamically chooses the optimal amount of redundancy to be used with NC in Section IV-C. In Section V, we present the jointly achievable throughput region for the NC technique. Simulation results and discussions are provided in Section VI. Finally, we conclude with a few remarks and future work in Section VII.

II. RELATED WORK

Our work is rooted in the recent development of NC for wireless ad hoc networks [5]–[8]. In [5], Wu *et al.* proposed the basic technique that uses XOR of packets to increase the throughput efficiency of a wireless mesh network. In [6], Katti *et al.* implemented an XOR-based technique in a wireless mesh network and showed a substantial bandwidth improvement over the current approach.

Incidentally, our problem is most similar to the index coding with side information problem first proposed by Birk and Kol [9], and Bar-Yossef *et al.* [10]. Subsequently, the connection between the index coding problem and matroid theory has been investigated by Rouayheb *et al.* [11]. In both our problem and the index coding problem, the sender wants to broadcast a message $x_i \in X$ to receiver R_i . Each receiver is assumed to have some side information on the subset of X . The goal is to find an encoding method that minimizes the number of transmissions so that every receiver can correctly receive its message. On the other hand, majority of literature on index coding assumes a noiseless communication channel between the receivers and the sender, while dealing with noisy communication is essential to our problem. Therefore, the analysis and focus of the two problems are quite different. Specifically, our solution gears towards designing a transmission protocol that can be implemented in future Wi-Fi and WiMAX networks.

Our work is also related to the wireless broadcast model proposed by Eryilmaz *et al.* [12]. In this work, Eryilmaz *et al.* proposed a *random network coding* technique for multiple users downloading a single file or multiple files from a wireless base station. Rather than using XOR operations, their technique encodes every packet using coefficients taken randomly from a sufficiently large finite field [13], [14]. This technique guarantees that the receivers can decode the original data with high probability. Another work is somewhat related to ours is that of Ghaderi *et al.* [15]. In [15], the authors analyzed the reliability benefit of NC for reliable multicast by computing the expected number of transmissions using the link-by-link ARQ technique compared to that of NC technique. Additionally, Rouayheb *et al.* [11] show the relation between index coding problem and network coding

and matroid representation problems. Especially, the authors have shown that vector linear codes outperform scalar linear codes but they are insufficient for achieving the optimum number of transmissions.

There are other works on multi-hop wireless networks with multiple unicast sessions. Li *et al.* [16], [17] have shown that NC can provide marginal benefits over the approaches that do not use NC. Also, Lun *et al.* [18] shows a capacity-approaching coding technique for unicast or multicast over lossy packet networks in which all nodes perform opportunistic coding by constructing encoded packets with random linear combinations of previously received packets. There is also a rich literature on ARQ, FEC, and HARQ techniques for wireless networks [19]–[21].

III. PROBLEM DESCRIPTION

In a typical data transmission from the Internet to a wireless user in a Wi-Fi or WiMAX network, packets first traverse through a wireless base station (BS) or an access point before arriving at the users. Since multiple flows (applications) traversing the BS, it has the opportunity to apply NC techniques to improve the overall throughput efficiency of the last wireless link. That said, our paper focuses on the transmissions between the BS and the receivers. In particular, we assume that the BS employs a buffer to avoid excessive packet drop due to burst traffic from the Internet. Thus, at any time, the BS has a set of packets Ω , to be delivered to a number of receivers. Each receiver may request a different subset of Ω , which from the BS's viewpoint, corresponds to supporting different *unicast* sessions. A special case arises when all receivers request all packets in Ω , which corresponds to a *broadcast* session. Although, a typical scenario is a mixture of unicast and broadcast in which more than one receiver request the same subset of packets, in this paper, we consider the unicast and broadcast sessions separately. That said, we make the following assumptions about the wireless channel model and the transmission mechanisms.

- 1) There are $K > 1$ receivers.
- 2) Data is assumed to be sent in packets, and each packet is sent in a time slot of a fixed duration.
- 3) The BS knows which packet from which receiver is lost. This can be accomplished through the use of positive and negative acknowledgments (ACK/NAKs).
- 4) All ACKs/NAKs are instantaneous and reliable. This assumption is not critical to our approach, and is used to simplify the analysis.
- 5) Every packet is protected with a sufficiently large number of Cyclic Redundancy Check (CRC) bits r to ensure that the probability of an undetectable bit error within a packet is virtually zero.
- 6) Bit error at a receiver R_i (due to unrecoverable bit errors) follows the Bernoulli trial with parameter p_i . Furthermore, the bit errors at the receivers are uncorrelated. This model is clearly insufficient to describe many real-world scenarios. One can develop a more accurate model, albeit complicate analysis.

Given the assumptions above, we analyze the performance of the proposed and existing techniques in the unicast and

broadcast scenarios. For example in the unicast scenario consisting of K receivers, if each receiver requests M distinct packets. Each packet contains N bits with L_i original information bits and $N - L_i$ parity bits if FEC is employed. Thus if we assume that $L_i = L$, the BS needs to deliver a total of $\sigma = M \times K \times L$ information bits successfully to all the receivers. Because of the addition of parity bits and/or the retransmitted bits due to channel errors, the expected number of transmitted bits δ , required to successfully deliver all original information bits is larger than σ . Similarly, for the broadcast scenario, since all K receivers request the same set of M packets, the total information bits $\sigma = M \times L$. That leads to the following definition for throughput efficiency that will be used as the evaluating metric for various transmission techniques.

Definition 3.1: The throughput efficiency of a transmission technique is defined as $\eta = \frac{\sigma}{\delta}$, the ratio of the total number of information bits to the expected number of transmitted bits.

Using this definition, a technique A is better than technique B if it results in higher throughput efficiency. Furthermore, no technique can have a throughput efficiency that is greater than 1. Next, we provide some theoretical analysis on the throughput efficiencies of the proposed and of the existing retransmission-based techniques, especially, the plain ARQ and HARQ protocols.

IV. ANALYSIS OF TRANSMISSION TECHNIQUES

In this section, we provide some theoretical analysis on throughput efficiencies of the ARQ, HARQ, and the proposed NC-HARQ techniques for both unicast and broadcast scenarios.

For the sake of simplicity, we first present the analysis for the case of two receivers, then extending our analysis to the general case of $K > 2$ receivers. Note that part of this analysis have been introduced previously in a conference paper [22]. Also, we emphasize that there are a number of parameters associated with each technique. The values of these parameters affect the throughput efficiency of a particular technique. For example, the throughput efficiency of the retransmission technique is greatly influenced by the packet size being used, while the performance of the HARQ technique depends on the amount of redundancy used. Although one can find the optimal parameters to obtain the highest throughput efficiency for each technique under the given network conditions, and use these parameters for comparison among different techniques, doing so may not be practical in other aspects. For example, the optimal packet size to achieve the highest throughput efficiency for the ARQ technique might be too small or too large to be efficiently realized in hardware. Therefore, the aim of this section is to provide the analytical expressions for the throughput efficiencies of different transmission techniques as a function of their parameters, and omit the optimal selection of these parameters. When comparing the performance of two techniques, we will provide the justification for choosing the ranges of the parameters that make the most sense.

To aid the analysis, we use the following notations:

- p_i : The bit error rate at receiver R_i (recall that the bit error follows a Bernoulli trial).

- P_i : The packet loss rate at receiver R_i when FEC is *not* employed. P_i is a function of p_i and the packet size.
- Pf_i : The packet loss rate at receiver R_i when FEC is employed. It is a function of p_i , the packet size, and the FEC protection level.
- N : The number of bits in a packet, including all data and parity bits. All packets have the same size.
- L_i : The number of data bits in a packet intended for receiver R_i . For the simplicity, we assume $L_i = L$.
- $RS(n, k)$: Reed-Solomon code with k data symbols and $n - k$ redundant symbols.
- m : The number of bits per a FEC symbol.
- r : The number of CRC bits used to detect bit errors in every packet. Every technique uses the same number of CRC bits.

A. Some Existing Retransmission-based Techniques

In this section, we provide some analysis on throughput efficiency for some retransmission-based techniques for both unicast and broadcast scenarios. We first begin with the well-known Automatic Repeat reQuest protocol.

1) *Automatic Repeat reQuest (ARQ) Technique*: ARQ is the simplest retransmission-based protocol between a sender and a receiver. Here, the sender first sends a packet to the receiver and waits for an ACK or NAK message from the receiver. Each packet contains a number of check bits that allow the receiver to detect whether bit errors have occurred during transit. If an error is detected, the receiver will send a NAK message to the sender. If the sender receives a NAK, it retransmits the packet in error (lost packet). On the other hand, if the sender receives an ACK, it transmits the next packet. Of course, the ACK and NAK messages themselves can be lost. In this case, the sender can set a maximum waiting time for the ACK and NAK messages. If these messages do not arrive before the deadline, the sender retransmits the lost packet. For ease of analysis, in this paper, we assume that ACK and NAK messages are never lost, but we note that the analysis can be easily modified to incorporate these lost ACK/NAK messages.

That said, in a unicast scenario involving multiple receivers, the BS sends packets intended for different receivers in a round robin fashion. That is, the BS ensures that a particular receiver successfully receives its packet before sending a different packet to another receiver. In a broadcast scenario, the BS ensures that the current packet is received successfully at all the receivers before sending the next packet. We now present the analysis on the throughput efficiency of the ARQ for these scenarios.

First, we assume that a packet loss occurs when there is at least one bit error within a packet. Thus, the packet loss rate P_i of the receiver R_i can be computed as

$$P_i = 1 - (1 - p_i)^N, \quad (1)$$

where N denotes the packet size in bits, and p_i denotes the bit error rate. Our first result is that, for the two-receiver broadcast scenario, the throughput efficiency (defined in Definition 3.1) when using an ARQ technique is:

$$\eta_{BA} = \frac{L}{N \left(\frac{1}{1-P_1} + \frac{1}{1-P_2} - \frac{1}{1-P_1 P_2} \right)}, \quad (2)$$

and for the two-receiver unicast scenario, the throughput efficiency is:

$$\eta_{UA} = \frac{2L}{N \left(\frac{1}{1-P_1} + \frac{1}{1-P_2} \right)}. \quad (3)$$

Proof: We start with the broadcast scenario. Let X_1 and X_2 be the random variables denoting the number of attempts to successfully deliver a packet to R_1 and R_2 , respectively. Thus, the expected number of transmissions needed to deliver a packet successfully to all receivers is a random variable $Y = \max_{i \in \{1,2\}} \{X_i\}$. The probability of using at most k required transmissions is

$$\begin{aligned} P[Y \leq k] &= P \left[\max_{i \in \{1,2\}} \{X_i\} \leq k \right] \\ &= \prod_{i=1}^2 P[X_i \leq k] = \prod_{i=1}^2 (1 - P_i^k). \end{aligned}$$

Therefore,

$$P[Y = k] = \prod_{i=1}^2 (1 - P_i^k) - \prod_{i=1}^2 (1 - P_i^{k-1}). \quad (4)$$

The expected number of transmissions to successfully deliver a packet to all the receivers can then be computed as:

$$\begin{aligned} E[Y] &= \sum_{k=1}^{\infty} k \left(\prod_{i=1}^2 (1 - P_i^k) - \prod_{i=1}^2 (1 - P_i^{k-1}) \right) \\ &= \sum_{k=1}^{\infty} k (P_1^{k-1} - P_1^k) + \sum_{k=1}^{\infty} k (P_2^{k-1} - P_2^k) \\ &\quad + \sum_{k=1}^{\infty} k (P_1^k P_2^k - P_1^{k-1} P_2^{k-1}) \\ &= \frac{1}{1-P_1} + \frac{1}{1-P_2} - \frac{1}{1-P_1 P_2} \end{aligned} \quad (5)$$

Since every transmitted packet contains L information bits, converting the average number of transmissions to bits and use the definition of throughput efficiency, we obtain (2).

Let us now consider the unicast scenario. Here, each receiver wants to receive distinct packets. The distribution on the number of transmissions before a successful reception at a receiver follows a geometric distribution, thus the average number of transmissions per a successful packet at receiver R_i is $\frac{1}{1-P_i}$. Adding the average number of transmissions of the two receivers and converting this to bits, yielding the average number of transmitted bits to successfully deliver two distinct packets to two receivers. Translating packets to bits yields (3). ■

Using the same arguments, one can generalize the above results to the case of K receivers. We have the following theorem.

Theorem 4.1: Using the ARQ protocol, the throughput efficiency of the K -receiver broadcast scenario is

$$\eta_{BA} = \frac{L}{N} \sum_{i_1, i_2, \dots, i_K} \frac{(-1)^{i_1+i_2+\dots+i_K-1}}{1 - P_1^{i_1} P_2^{i_2} \dots P_K^{i_K}}, \quad (6)$$

where $i_1, i_2, \dots, i_K \in \{0, 1\}$, $\exists i_j \neq 0$. And for the K -receiver unicast scenario, the throughput efficiency is

$$\eta_{UA} = \frac{K.L}{N \left(\sum_{i=1}^K \frac{1}{1-P_i} \right)}. \quad (7)$$

2) *Hybrid ARQ (HARQ) Technique:* Hybrid ARQ technique is a simple modification to the basic ARQ technique. Here, additional error-correcting bits are inserted into each packet. If the number of bit errors is sufficiently small, and can be corrected, then no retransmission is necessary. Otherwise, when it is not possible to correct the errors, the entire packet is retransmitted. From the performance's viewpoint, an HARQ technique is equivalent to that an ARQ technique where the channel has been improved via the use of error-correcting bits. Therefore, the throughput efficiency for pure ARQ technique (Theorem 4.1) can be translated directly to the HARQ technique. The only difference is that the packet loss rates and the number of information bits have been reduced, due to the addition of error-correcting bits. Thus, our task is simply to compute the new packet loss rates and the number of information bits per packet, and use Theorem 4.1 to determine the throughput efficiency for the HARQ technique.

We analyze a simple Type-I HARQ technique [23] where Reed Solomon code $RS(n, k)$ is used for error correcting and r CRC bits for error detection. We recall that the symbol length is m bits and each packet consists of X code blocks. Upon receiving a packet, the receiver first performs the error correction using $RS(n, k)$ then error checking (detection) using CRC bits. At the receiver, we omit combining technique, e.g., Chase Combining (CC) [23] in decoding for ease of analysis. We now begin with the 2-receiver broadcast scenario.

Given that the symbol length is m bits, the Symbol Error Rate (SER), i.e., the probability of one or more bits corrupted within a symbol for a receiver R_i is given by

$$SER_i = 1 - (1 - p_i)^m. \quad (8)$$

Therefore, the irrecoverable packet loss rate Pf_i for receiver R_i after using $RS(n, k)$ is

$$Pf_i = 1 - \left[\sum_{j=0}^t \binom{n}{j} (1 - SER_i)^{n-j} (SER_i)^j \right]^X, \quad (9)$$

where $t = \lfloor \frac{n-k}{2} \rfloor$ and X denotes the number of code blocks within a packet.

Now, based on Theorem 4.1 and the fact that adding error-correcting bits effectively change the packet loss rate, we have the following theorem regarding the HARQ technique.

Theorem 4.2: Using the HARQ protocol, the throughput efficiency of the K -receiver broadcast scenario is

$$\eta_{BF} = \frac{L}{N} \sum_{i_1, i_2, \dots, i_K} \frac{(-1)^{i_1+i_2+\dots+i_K-1}}{1 - Pf_1^{i_1} Pf_2^{i_2} \dots Pf_K^{i_K}}, \quad (10)$$

where $i_1, i_2, \dots, i_K \in \{0, 1\}$, $\exists i_j \neq 0$. And for the K -receiver unicast scenario, the throughput efficiency is

$$\eta_{UF} = \frac{\sum_{i=1}^K L_i}{N \left(\sum_{i=1}^K \frac{1}{1-Pf_i} \right)}. \quad (11)$$

Receiver	Packet								
	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9
R1	X	O	O	X	O	O	X	O	X
R2	O	O	X	O	X	O	X	O	O

Fig. 1. Combined packets for time-based retransmission for a two-receiver wireless broadcast scenario: $a_1 \oplus a_3, a_4 \oplus a_5, a_7, a_9$; $M = 9$. Here we denote “x” and “o” as lost and successful packets, respectively.

B. Proposed Network Coding Technique

In this section, we investigate NC techniques that combine lost packets from multiple flows to reduce the number of retransmissions.

1) *Basic Network Coding Technique*: We first investigate the basic NC technique in which error correcting bits are not included in a packet. Incorporating error-correcting bits will be considered in the next subsection. The receiver’s protocol is similar to that of the receiver in the ARQ technique. That is, the receiver sends a NAK immediately if it does not receive a packet correctly. However, the sender does not retransmit the lost packet immediately when it receives a NAK. Instead, the sender maintains a list of lost packets and the corresponding receivers for which their packets are lost. The retransmission phase starts at a fixed interval of time in terms of number of time slots. During the retransmission phase, the sender forms a new packet by XORing a maximum set of the lost packets from different receivers before retransmitting this *coded* packet to all the receivers. Specifically, if there are K receivers, then the maximum number of lost packets from different receivers is K , one from each receiver, will be combined. When there are no more K distinct lost packets from K receivers to be combined, this implies that the receiver with the lowest packet loss rate have successfully received all its packets. Therefore, the maximum number of lost packets from different receivers is now $K - 1$. The process repeats until there remains only one receiver with lost packets. These lost packets will be retransmitted alone. Note that each time the maximum number of distinct lost packets from different receivers to be combined is reduced by one, this implies that a receiver with next higher packet loss rate, has received all its packets successfully. The last receiver is the one with the highest packet loss rate. As shown in the proof of Theorem 4.3, it is possible to follow this procedure, if the number of packets M to be sent by the sender to each receiver, is large. More precisely, the proof of Theorem 4.3 shows that with probability 1, this procedure is possible.

Even though a receiver successfully receives the coded packet, it must be able to recover the lost packet, and it does so by XORing the coded packet with appropriate set of previously successful packets. The information on choosing this appropriate set of packets is included in the packets sent by the BS.

For example, Fig. 1 shows a pattern of lost packets (denoted by the crosses) and successful packets (denoted by the circles) for the broadcast scenario with two receivers R_1 and R_2 . The combined packets are $a_1 \oplus a_3, a_4 \oplus a_5, a_7, a_9$, where a_i denotes the i -th packet.

Receiver R_1 recovers packet a_1 as $a_3 \oplus (a_1 \oplus a_3)$. Similarly, receiver R_2 recovers packet a_3 as $a_1 \oplus (a_1 \oplus a_3)$. When

the same packet loss occurs at both receivers R_1 and R_2 , the encoding process is not needed and the BS just has to retransmit that packet alone. Note that the sender has to include some bits to indicate to a receiver which set of packets it should use for XORing. Here, we assume that all packets have the same size for all the receivers, thus can be conveniently XORed together. The same approach can be used for the unicast scenario. The only difference is that a receiver may have to cache packets intended for all other receivers as well. This enables it to decode its own lost packets subsequently. We have the following results on the broadcast and unicast scenarios.

Theorem 4.3: Using the basic NC technique, when the number of packets to be sent $M \rightarrow \infty$, the throughput efficiency for K -receiver broadcast scenario is

$$\eta_{BN} \sim \frac{L(1 - \max_{i \in \{1, 2, \dots, K\}} \{P_i\})}{N}, \quad (12)$$

and for K -receiver unicast scenario is

$$\eta_{UN} \sim \frac{K.L}{N} \left(\frac{1}{K + \sum_{i=1}^K \frac{\prod_{j=i}^K P_j}{1 - P_i}} \right) \quad (13)$$

Proof: We first consider the broadcast scenario. Without loss of generality, assuming that $P_i \leq P_j$ if $i \leq j$, $\{i, j\} \in \{1, 2, \dots, K\}$. Let random variable X_i denote the number of lost packets at receiver R_i after M transmissions. As discussed, the combined packets in the NC technique are dynamically formed based on the feedbacks from the receivers. If a combined packet is correctly received at some receivers, but not at others, a new combined packet is generated to ensure that the receivers with the correct packet will be able to obtain the new data using the new combined packet. This implies that after a long run, the number of retransmissions will be dominated by the receiver which has the largest error probability. To prove this, let us consider two receivers R_i and R_j whose packet loss rates respectively are P_i and P_j where $P_i \leq P_j$. Furthermore, let a random variable $X = X_j - X_i$, then the claim is equivalent to proving $Pr(X < 0) \rightarrow 0$ as $M \rightarrow \infty$. Since each transmission follows a Bernoulli trial, X_i and X_j are Binomial random variables. Especially, when $M \rightarrow \infty$, based on the central limit theorem, distributions of X_i and X_j approach that of a Gaussian random variable; consequently, distribution of X approaches that of a Gaussian random variable too. Note that X_i and X_j are independent, we have

$$\begin{aligned} \mu_X &= E[X_j] - E[X_i] \\ &= M(P_j - P_i) \end{aligned} \quad (14)$$

$$\begin{aligned} \sigma_X^2 &= var(X_j) + var(X_i) \\ &= M[(P_j(1 - P_j) + P_i(1 - P_i))] \end{aligned} \quad (15)$$

Thus, the probability density function of X can be written as

$$Pr(X) = \frac{1}{\sqrt{2\pi\sigma_X}} e^{-\frac{(X - \mu_X)^2}{2\sigma_X^2}} \quad (16)$$

Obviously, when $M \rightarrow \infty$, both μ_X and σ_X increase. In particular, μ_X increases with an order of M while σ_X

increases with an order of \sqrt{M} . Hence, the tail area, i.e., $Pr(X < 0)$, asymptotically goes to 0 as $M \rightarrow \infty$. Carrying out the same argument, we can prove that $Pr(X_i \leq X_K) \rightarrow 1$ as $M \rightarrow \infty$ for $\forall i$. Thus, let a random variable Y denote the number of retransmissions needed to deliver all lost packets. The expected value of Y is

$$\begin{aligned} E[Y] &= E\left[\max_{i \in \{1, 2, \dots, K\}} \{X_i\}\right] \\ &\sim E[X_K]. \end{aligned} \quad (17)$$

Therefore, the expected number of transmissions to successfully deliver a set of M packets to K receivers is given by

$$\begin{aligned} T_{BN} &= M + E[X_K] \\ &\sim M + \frac{M \times \max_{i \in \{1, 2, \dots, K\}} \{P_i\}}{1 - \max_{i \in \{1, 2, \dots, K\}} \{P_i\}}. \end{aligned} \quad (18)$$

To obtain the throughput efficiency, we first divide T_{UN} by M to get the average number of bits per transmission (packet). Next, since each packet contains only L information bits out of N transmission bits. Hence, throughput efficiency is calculated by $L / \left(\frac{N}{(1 - \max_{i \in \{1, 2, \dots, K\}} \{P_i\})}\right)$, yielding (12).

For the unicast scenario case, we use induction method to prove the theorem. Interested readers can find details of the proof in the Appendix. ■

2) Network Coding-Hybrid ARQ (NC-HARQ) Technique:

In this section, we investigate the NC technique in conjunction with existing HARQ protocol for the broadcast and unicast scenarios. Intuitively, when transmitting packets over a bad channel, a stronger FEC code should be used to correct bit errors within a packet. If a weak FEC code is used in the HARQ protocol, a few bit errors may require the sender to retransmit the entire packet (possibly on the order of thousands bits), resulting in lower throughput efficiency. On the other hand, when the channel is good, a strong FEC code results in too much redundancy that also lowers the throughput efficiency. Thus, the ratio of the number of redundant bits to the number of information bits should be a function of channel condition to increase the throughput efficiency.

That said, we first start with the broadcast scenario where all the receivers want to receive identical information. Here, it is convenient to use the same FEC protection level for all the packets, regardless of the various channel conditions for different receivers. This means that, when too much redundancy is used, it would over-protect the receivers with good reception, while too little redundancy would hurt the receivers with bad reception. Thus, balancing the right amount of FEC is the key to improve the throughput efficiency. We have the following theorem.

Theorem 4.4: Using the NC-HARQ technique, when the number of packets to be sent is sufficiently large, the throughput efficiency for the K -receiver broadcast scenario is

$$\eta_{BNF} \sim \frac{L(1 - \max_{i \in \{1, 2, \dots, K\}} \{P f_i\})}{N}, \quad (19)$$

and the throughput efficiency of the K -receiver unicast scenario is

$$\eta_{UNF} \sim \frac{\sum_{i=1}^K L_i}{N} \left(\frac{1}{K + \sum_{i=1}^K \frac{P f_i}{1 - P f_i}} \right). \quad (20)$$

Proof: The proof is directly obtained from Theorem 4.3 by replacing the packet loss rate P_i with the irrecoverable error probability $P f_i$. The reason for this simple replacement is that the irrecoverable error probability of a packet for a certain receiver R_i is the same regardless whether that packet is a regular packet or a coded packet. Thus, the same argument in the proof of Theorem 4.3 holds. Intuitively, adding redundancy to the packets simply changes the packet loss rates and the bandwidth overhead, which then affects the throughput efficiency. ■

C. Optimal Redundancy

In Section IV-B2, we show how to compute the throughput efficiencies for the broadcast and unicast scenarios given the packet loss rates which in turn are functions of the amount of redundancy, i.e., the FEC for each packet. Now, we seek the optimal $RS(n, k)$ code to result in highest throughput efficiency. In what follows, we assume that the bit error rates at different receivers are known. Thus, (9) can be used to compute irrecoverable packet lost rate for each receiver, given a particular $RS(n, k)$ code. That said, a straightforward approach is to use an exhaustive search. Assuming that n is fixed, since the same $RS(n, k)$ is used to transmit packets to all the receivers, only a search through all the possible values of $k = 1, 2, \dots, n$ (hence $n-k$ redundant symbols) is necessary to choose the value of k that maximizes the throughput efficiency (Equation (19)). Note that the throughput efficiency of the broadcast scenario depends only on the maximum packet loss rate, hence the exhaustive method is feasible.

On the other hand, for the K -receiver unicast scenario, using the exhaustive search may not be feasible when the number of receivers is large. Specifically, one has to find an optimal coding level so that (20) is maximized. Since a coding level k_i can take on the values from 1 to n , the time complexity of the searching method is quite expensive, i.e., $\mathcal{O}(nK^2)$. Especially, when the channel condition changes, one needs a fast algorithm to adjust the amount of redundancy in time. We propose the following approximate algorithm to compute the optimal coding level.

We note that the throughput efficiency mostly depends on the largest packet loss rate P_K (we assume that the packet lost rates are ordered from the smallest to the largest) and the associated overhead. Thus, our algorithm attempts to increase the throughput efficiency by reducing the largest packet loss rate with an appropriate increase in the overhead. Specifically, our algorithm first initializes all $k_i = n$ for the transmission packets. In the second step, the algorithm computes the corresponding packet loss rates $P f_i$'s for all the receivers. In the third step, it chooses the receiver with largest packet lost rate and reduces the data within a code block k_i by 1 symbol and increases the redundancy by 1 symbol, thus keeping n fixed. In the fourth step, it computes the new throughput efficiency. If the new throughput efficiency increases, the algorithm repeats the steps two and three, until the new throughput efficiency no longer increases. The optimal value k_i^* is the one found in the immediate previous iteration. Note that by considering only the largest packet loss rate, the complexity of the proposed algorithm is reduced to $\mathcal{O}(nK)$. The pseudo-code for the algorithm is shown in Algorithm 1.

Algorithm 1 : Finding the optimal redundancy for the K -receiver unicast scenario

Inputs: K, X, m, n, p_i .

Outputs: k_i 's

```

1: for  $i = 1$  to  $K$  do
2:    $k_i = n$  {Initialize  $k_i$ }
3:    $k_i^* = k_i$  {Initialize optimal values of  $k_i$ }
4:    $SER_i = 1 - (1 - p_i)^m$ 
5:    $t_i = \lfloor \frac{n-k_i}{2} \rfloor$ 
6:    $Pf_i = 1 - \left[ \sum_{j=0}^{t_i} \binom{n}{j} (1 - SER_i)^{n-j} SER_i^j \right]^X$ 
      {Compute irrecoverable packet loss rates}
7: end for
8:  $prev\_eff = 0$  {Setting the previous throughput efficiency
to zero}
9:  $curr\_eff = \frac{\sum_{i=1}^K k_i}{n} \left( \frac{1}{K + \frac{\max_{j \in \{1, \dots, K\}} \{Pf_j\}}{1 - \max_{j \in \{1, \dots, K\}} \{Pf_j\}}}} \right)$ 
      {Compute the current throughput efficiency}
10: while  $curr\_eff > prev\_eff$  do
11:   Choose  $l$  such that for  $k > 2, l = \arg \max_i \{Pf_i\}$ 
12:    $k_l = k_l - 1$  {Add 1 more redundant symbol to the
receiver with largest packet loss rate. Make sure that
 $k_i > 0$  for all  $i$ }
13:    $prev\_eff = curr\_eff$ 
14:    $Pf_l = 1 - \left[ \sum_{j=0}^{t_l} \binom{n}{j} (1 - SER_l)^{n-j} SER_l^j \right]^X$ 
15:    $curr\_eff = \frac{\sum_{i=1}^K k_i}{n} \left( \frac{1}{K + \frac{\max_{j \in \{1, \dots, K\}} \{Pf_j\}}{1 - \max_{j \in \{1, \dots, K\}} \{Pf_j\}}}} \right)$ 
      {Compute new throughput efficiency}
16:    $k_j^* = k_j$ 
17: end while
    
```

V. ACHIEVABLE THROUGHPUT REGION

In the previous sections, the definition of throughput efficiency for the K -receiver unicast scenario is computed based on the throughput fairness for all the receivers. That is, every receiver is to receive all their packets in same time duration. Thus, using this definition, maximizing the throughput efficiency really implies maximizing the total rate with the constraint that every receiver must have the same rate as computed at the end of same duration. In many real world situations, for a given total wireless bandwidth, it may be useful to characterize the simultaneous achievable throughputs for all receivers. In other words, if one receiver is allowed to receive information at a faster rate than that of another, what are the throughput regions of these receivers?

Let us consider a scenario consisting of one BS and two receivers R_1 and R_2 . The packet loss rates of R_1 and R_2 are 0.1 and 0.2, respectively. If all the time slots of the BS are used to transmit packets for R_1 , then the throughput of R_1 would be 90% of the BS capacity since the R_1 error rate is 10%. Similarly, the throughput of R_2 is 80% if all the time slots are used to transmit R_2 's packets. Therefore, if a time-sharing technique is used, i.e., the BS sends packets to R_1 and R_2 at α and $(1-\alpha)$ fractions of the time, respectively, for $\alpha \in [0, 1]$, then the achievable throughput pair is a linear interpolation of the two end points (0.9,0) and (0,0.8) as shown in Fig. 2. If N

denotes the total number of available time slots, M_1 and M_2 denote the expected number of successful packets sent to R_1 and R_2 , respectively, then it is straightforward to show that M_1 and M_2 must satisfy

$$\frac{M_1}{1 - P_1} + \frac{M_2}{1 - P_2} \leq N \quad (21)$$

Now, for the same scenario, using NC technique, we have the following theorem.

Theorem 5.1: Assuming that N is sufficiently large, for $M_1 P_1 (1 - P_2) \leq M_2 P_2 (1 - P_1)$, M_1 and M_2 must satisfy

$$M_1 + M_2 + \frac{m}{1 - \max\{P_1, P_2\}} + \frac{M_1 P_1 P_2}{1 - P_1} + \frac{M_2 P_2 - m}{1 - P_2} \leq N, \quad (22)$$

and for $M_1 P_1 (1 - P_2) > M_2 P_2 (1 - P_1)$, M_1 and M_2 must satisfy

$$M_1 + M_2 + \frac{m}{1 - \max\{P_1, P_2\}} + \frac{M_1 P_1 - m}{1 - P_1} + \frac{M_2 P_1 P_2}{1 - P_2} \leq N, \quad (23)$$

where $m = \min\{M_1 P_1 (1 - P_2), M_2 P_2 (1 - P_1)\}$

Proof: To obtain the Inequality (22), we note that the expected number of time slots to successfully transmit M_1 and M_2 packets to R_1 and R_2 must be at least $M_1 + M_2$. During these transmissions, there will be lost packets, specifically, on average, $M_1 P_1$ from R_1 and $M_2 P_2$ packets from R_2 . Now, the first term $\frac{m}{1 - \max\{P_1, P_2\}} = \frac{M_1 P_1 (1 - P_2)}{1 - \max\{P_1, P_2\}}$ represents the expected number of time slots required to successfully transmit combined packets to both receivers. The last two terms, $\frac{M_1 P_1 P_2}{1 - P_1}$ and $\frac{M_2 P_2 - M_1 P_1 (1 - P_2)}{1 - P_2}$ represent the expected number of time slots required to successfully retransmit the remaining lost packets of R_1 and R_2 , respectively. The summation of these time slots must be less than the total number of available time slots N , thus the Inequality (22) must hold. Similar argument can be applied to obtain the Inequality (23), and that completes the proof. ■

Fig. 2 shows the achievable throughput of R_1 versus R_2 using the NC technique. Interestingly, from an information theoretic viewpoint, our proposed NC technique can be viewed in light of the broadcast channel problem first proposed by Cover [24], [25]. In his celebrated superposition coding, Cover was the first to show that one can achieve a larger capacity region than that of the time-sharing technique. Our proposed technique is less efficient than the superposition coding technique, however, we note that, the superposition coding technique is an information theoretic argument, and not practical in today wireless networks.

We now argue that our approach is asymptotically optimal when the number of receivers is large. Specifically, when the number of receivers approaches infinity, and the number of packets to be sent approaches infinity at a much faster rate than the number of receivers, then the throughput efficiency is 1 (if $L = N$, i.e., no error correcting bits is used) as shown in (13) of Theorem 4.3. This is the best efficiency one can hope for. The intuition is that when there is a sufficiently large number of receivers, for every transmission, at least one of the receivers will correctly receive a packet. Even if that packet is not intended for a receiver that receives it correctly,

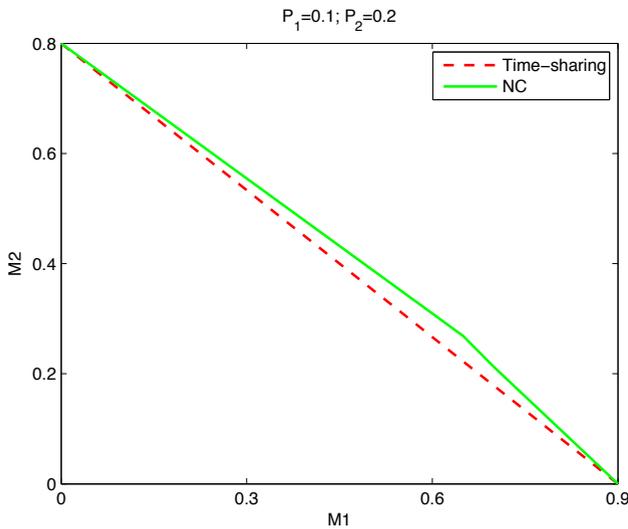


Fig. 2. Achievable rate of pure time-sharing and the network coding techniques.

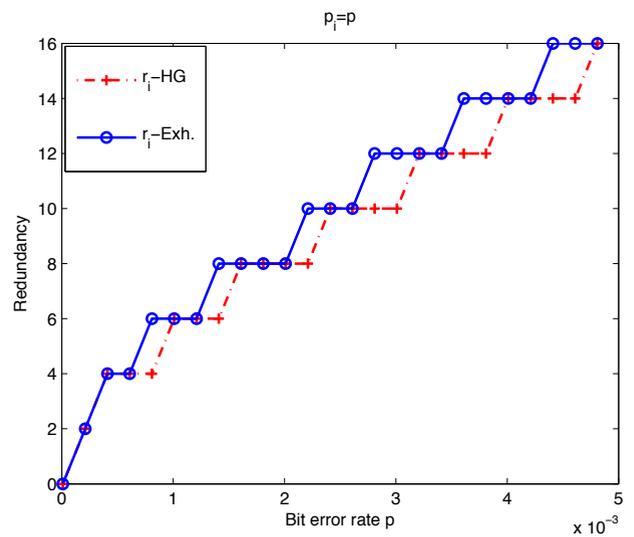


Fig. 4. Optimal redundancies for a 50-receiver wireless unicast scenario obtained by Heuristic-Greedy (HG) and Exhaustive search (Exh.) techniques when all p_i 's are set to p , and p varies from 10^{-6} to 4.5×10^{-3} .

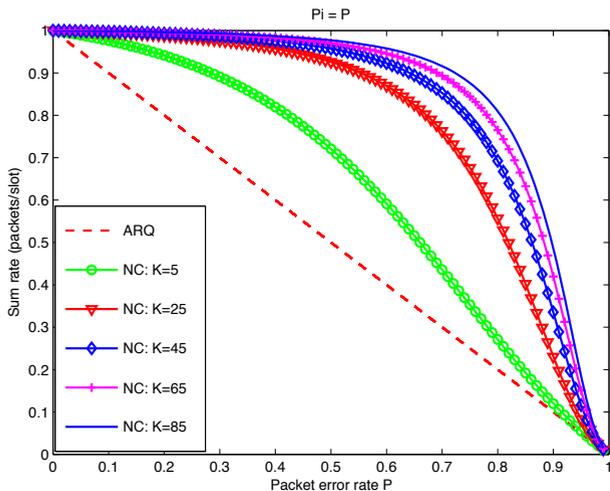


Fig. 3. Achievable sum-rate of pure time-sharing and the network coding techniques.

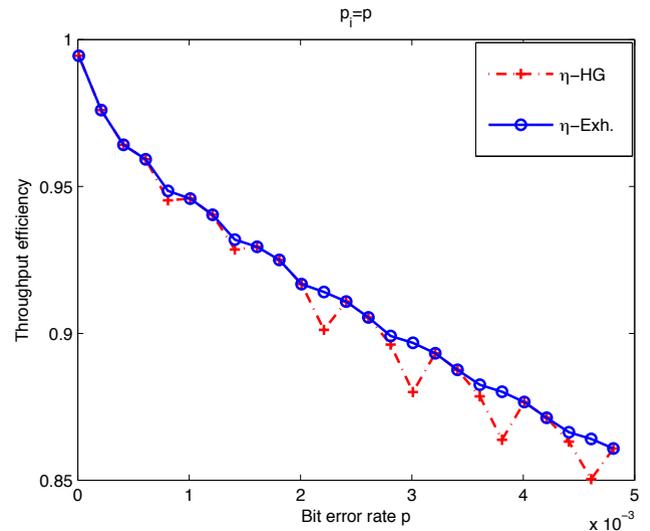


Fig. 5. Throughput efficiency for a 50-receiver wireless unicast scenario using heuristic-greedy (HG) and exhaustive search (Exh.) techniques when all p_i 's are set to p , and p varies from 10^{-6} to 4.5×10^{-3} .

using our approach, this packet can still be used to recover a lost packet for that receiver in the future. Essentially, every packet is useful at least for one receiver in this setting. Thus one should expect the throughput efficiency approaching 1.

To illustrate our point, let us consider a unicast scenario. Here, the sum rate is defined as the sum of all expected successful received packets at all receivers. For the simplicity let us assume that all receivers have the same packet loss rate, $P_i = P$, then the sum rate normalized by the number of used time slots versus the packet error rate is plotted in Fig. 3. The dash line represents the achievable rate for pure time-sharing technique $1 - P$, while the curves represent the achievable rates for the network coding technique for different number of receivers. As shown, the achievable sum rate of NC technique when $P > 0$, is extended to one when the number of receivers increases to infinity. When $P_i = 1$, the sum rate is 0. Keep in mind that, for our proof to go through, the number of packets

to be sent M has to increase at a much faster rate than the number of receivers K .

VI. SIMULATIONS AND DISCUSSIONS

In this section, we present simulation results on the throughput efficiency and throughput gain in different network scenarios. To simulate the transmissions in a Wi-Fi network, the packet size should be set around 1500 bytes. However, when using such a large packet size under a large bit error rate, e.g. on the order of 10^{-3} , the throughput efficiencies of the ARQ and NC techniques are much worse than those of the HARQ and NC-HARQ techniques. To be fair, we use a smaller packet size, i.e., 665 bytes for ARQ and NC techniques, and also incorporate a very light protection using $RS(63, 59)$. For HARQ and NC-HARQ techniques, the packet size is set at 1559 bytes (Wi-Fi packet size) and data is encoded with

$RS(127, 117)$. We use CRC-32 for error detection in all the simulations.

We also note that there is an overhead associated with the NC techniques. Specifically, one needs to specify which packets in the combined packets. Typically, if there are M packets in the queue, then the number of bits to represent these packets is $\log M$. Therefore, in most cases, when the packet size is large, on the order of KBytes, such as those of IEEE 802.11, this overhead is negligible.

Also, since the NC technique uses only exclusive-bit-wise XOR, thus, encoding and decoding can be done fast, especially if implemented in hardware. On the other hand, the BS needs to have enough memory to store a sufficiently large number of lost packets from all receivers in order to have throughput gain. The algorithm used for choosing packets to combine is quite simple as one just needs to examine the queues, then combining the maximum number of lost packets. That said, when using NC, one has to consider the packet delay introduced by buffering of lost packets. For some time-sensitive applications, this can be problematic. We will address this in future work.

We first compare the optimal redundancies estimated by the greedy-heuristic algorithm, described in IV-C, and by exhaustive search method (exhaustive search method is only feasible for a smaller number of receivers). As described above, the broadcast wireless scenario is simple, therefore we consider only unicast wireless scenario. In particular, a 50-receiver unicast wireless scenario is under investigation. Fig. 4 represents the obtained optimal redundancies r_i using exhaustive and greedy methods when p varies from 10^{-6} to 4.5×10^{-3} . As seen, the optimal redundancy estimated by greedy algorithm is very close to that of exhaustive search, especially when the bit error rate is small. These differences are due to the fact that by looking only one step ahead and taking into account the largest error packet, the greedy algorithm may produce local optimal value. The throughput efficiencies obtained by these methods are shown in Fig. 5. As shown, the exhaustive search method is optimal, thus achieves higher throughput efficiency compared to that of the greedy method. However, because of its high complexity, its use might be limited. On the other hand, the throughput efficiency of the greedy algorithm is slightly less, but its low complexity makes it an effective technique for real-world scenarios with many receivers.

We next compare the throughput efficiencies and throughput gains among the techniques. Figs. 6(a) and Fig. 6(b) show the simulation and theoretical throughput efficiencies as a function of bit error rate for broadcast and unicast scenarios with one sender and two receivers. The bit error rates of two receivers are set equal to each other, and varied from 10^{-6} to 4.5×10^{-3} . As seen, the simulation results verify our theoretical derivations. Furthermore, we note that the NC-HARQ technique always outperforms the HARQ technique and the NC technique always outperforms the ARQ technique for the given identical set of parameters. This is because NC approach has the identical method in the transmission phase with that of the ARQ or HARQ, but has a more effective retransmission method. In small bit error rate regions, the NC technique performs the best which is intuitively plausible since

redundancy introduced by the NC-HARQ technique would just increase the bandwidth overhead unnecessarily. Similarly, Fig. 6(b) shows the throughput efficiency versus bit error rate for the wireless unicast scenario. As shown, the throughput efficiency of NC-HARQ technique always outperforms other techniques.

Figs. 7(a) and 7(b) show the throughput gains of HARQ, NC, NC-HARQ techniques over the ARQ technique for broadcast and unicast scenarios. The throughput gain of technique A over B is defined as the ratio of the throughput efficiency of A over that of B . As seen, for some bit error rate regions, the proposed NC-HARQ technique can be more than three and two times efficient than ARQ technique for both the broadcast and unicast scenarios, respectively.

We now compare the performance of the proposed dynamic NC-HARQ algorithm against other techniques. In this technique, the sender is able to adjust the amount of FEC in real time to adapt to the channel conditions. In our simulation we assume slow fading channels; they are stable for a while before changing to another state. In particular, p_1 and p_2 vary from 10^{-6} to 4×10^{-3} with a step size of 4×10^{-4} . All other parameters are identical to the previous simulations for all the non-adaptive techniques. Figs. 8(a) and (b) show the throughput gains over ARQ technique as a function of p_1 and p_2 for different techniques in the broadcast and unicast scenarios, respectively. As seen, the dynamic NC-HARQ algorithm has the best performance as it can adapt the amount of redundancy appropriately. Especially, in the range of high bit error rate, the throughput gain by using dynamic NC-HARQ can be more than 12 and 5.5 times better than ARQ technique for broadcast and unicast scenarios, respectively. An interesting observation is that in both scenarios, the heuristic-greedy algorithm can achieve a throughput gain almost the same as that of the exhaustive search at a much lower complexity.

Figs. 9(a) and (b), respectively, show the throughput efficiencies of NC and ARQ techniques versus the number of receivers in broadcast and unicast wireless scenarios. Packet loss rates of all receivers are equal to 20%. For the broadcast scenario in Fig. 9(a), when the number of receivers increases, the throughput efficiency of the NC technique remains constant while that of the ARQ technique decreases significantly. This is because using NC, the throughput efficiency depends only on the receiver with the largest packet loss rate; while in the ARQ technique, every receiver's channel condition affects to the throughput efficiency.

Next, the throughput efficiency versus the number of receivers of the unicast scenario is shown in Fig. 9(b). An interesting observation can be seen is that when the number of receivers increases, the throughput efficiency of NC technique asymptotically approaches to one. This is intuitively matched with the achieved sum rate shown in Fig. 3. This is because, when there is a large number of receivers, every transmitted packet will be received correctly at least at one receiver with a probability closed to one. To illustrate this, let us consider a scenario in which all receivers have the same packet loss rate P . Let $P(\kappa)$ denote the probability that a transmitted packet is intended for one receiver, and it is successfully received at

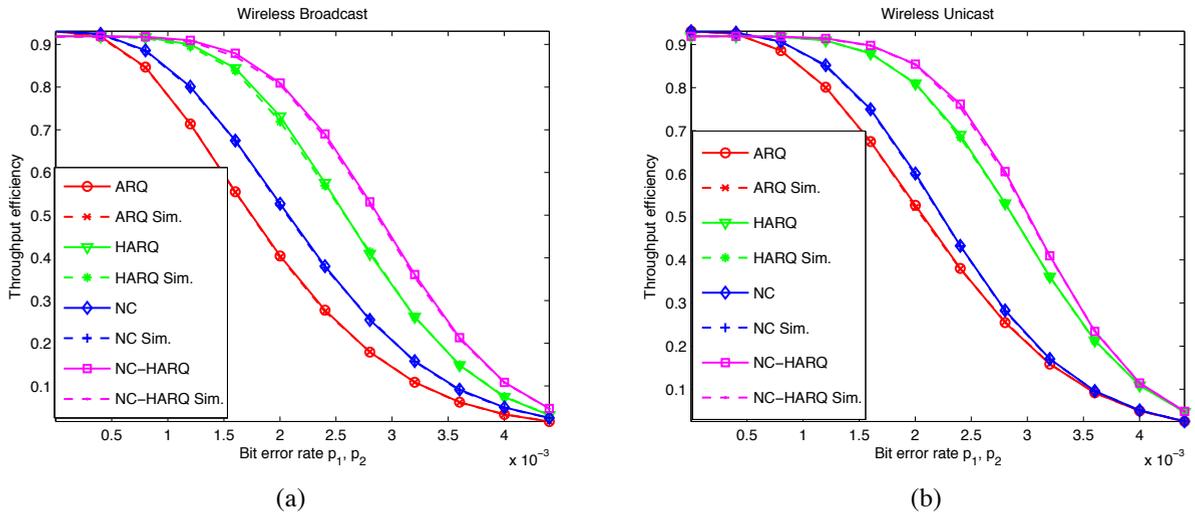


Fig. 6. Throughput efficiency versus bit error rate for theory and simulation (a) Broadcast and (b) Unicast.

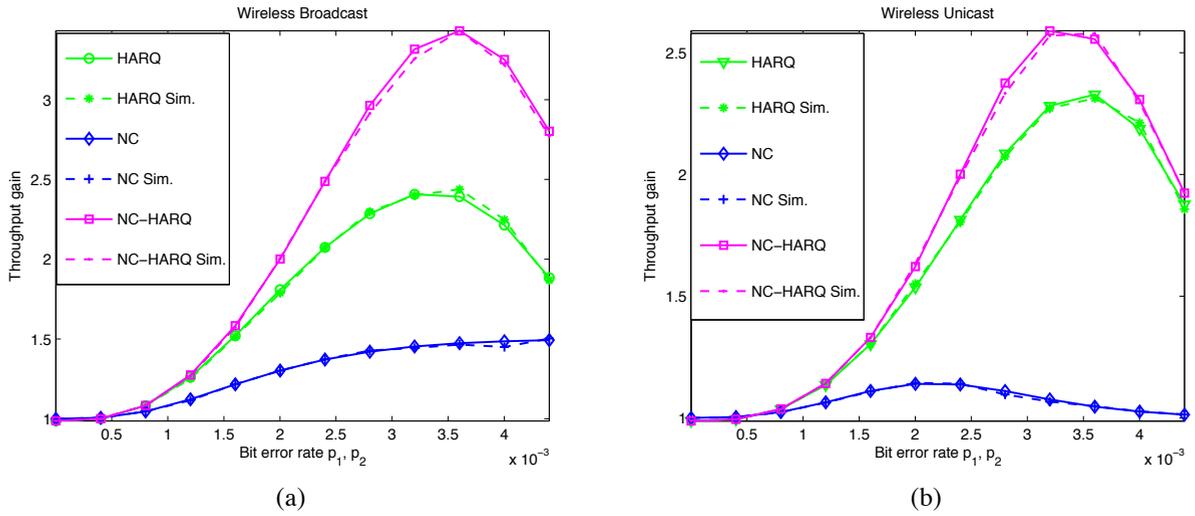


Fig. 7. Throughput gain over ARQ technique versus bit error rate for theory and simulation for (a) Broadcast and (b) Unicast.

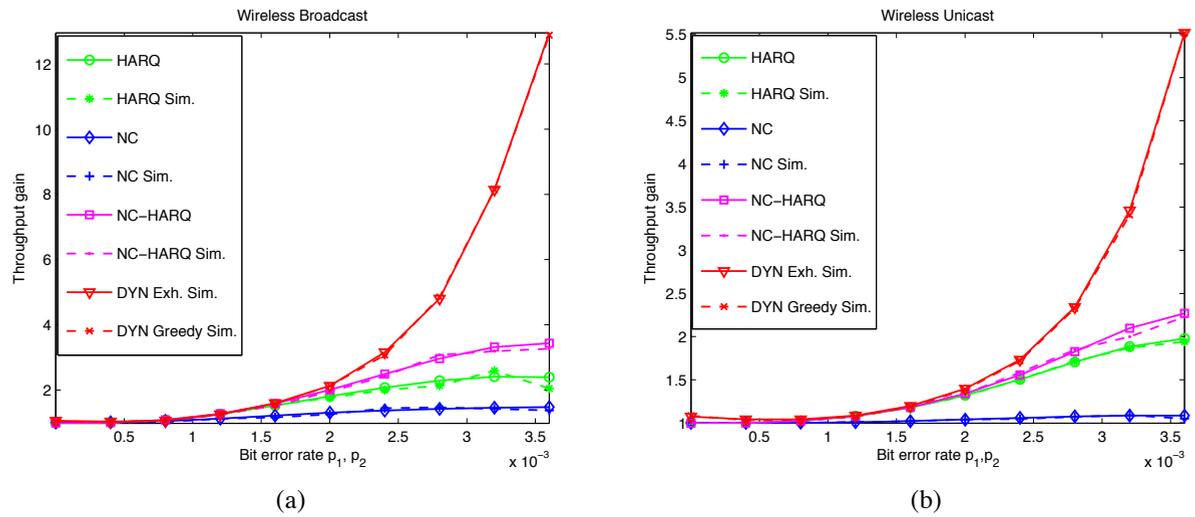


Fig. 8. Throughput gain of different techniques under changing network conditions for (a) Broadcast and (b) Unicast.

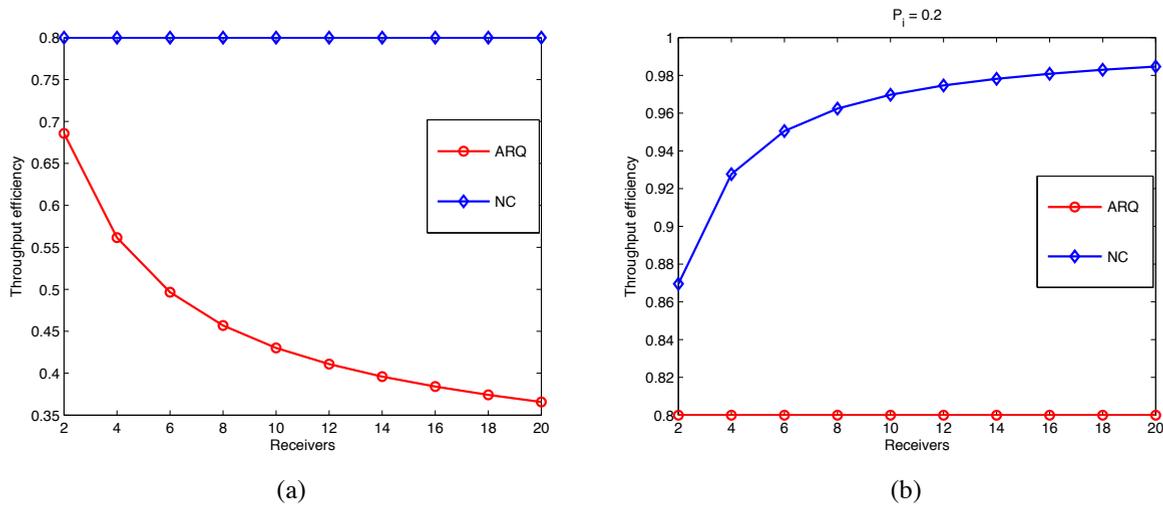


Fig. 9. Throughput efficiency versus the number of receivers for (a) Broadcast and (b) Unicast.

at least one other receivers. We have

$$P(\kappa) = \sum_{i=1}^{K-1} \binom{K-1}{i} P^{K-i-1} (1-P)^i \quad (24)$$

Even when $P = 90\%$, if there are $K = 50$ receivers, the probability that there exists at least one receiver receives a packet successfully is equal to 0.9943. This value is very close to one. Obviously, when the number of receivers goes to infinity, this probability goes to 1. Using the NC technique, even when a packet is intended for a certain receiver, other receivers still store this packet in its buffer. Subsequently, other receivers will use this packet to recover their lost packets simultaneously by XORing with the combined packet sent out by the BS. Effectively, every transmission carries useful information to the receivers. Therefore, one should expect the throughput efficiency approaching 1. Note that this argument holds true only if the number of packets to be sent M goes to infinity at a faster rate than that of the number of receivers K as implied in the proof of Theorem 4.3.

VII. CONCLUSIONS AND FUTURE WORK

We have proposed a hybrid network coding technique to increase throughput efficiency of single-hop wireless networks for both the broadcast and unicast scenarios. The theoretical and simulation results showed that our proposed technique can efficiently utilize high throughput over those of traditional techniques for a typical range of channel conditions. We also proposed a heuristic method for dynamically changing the amount of redundancy for each transmitted packet to adapt the channel conditions. The simulation has shown that the proposed technique can outperform traditional techniques several-fold in terms of throughput efficiency. Our ongoing work is to characterize how the buffer size affects to the network performance, especially, when the transmission flows include time-sensitive applications. How to use NC for unbalance channels, i.e., the channels have different transmission rates and carry different types of applications, is also an interesting topic for investigation.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments, which have helped improve the clarity of the paper.

APPENDIX

Proposition A.1: The throughput efficiency of a wireless unicast scenario using network coding technique for two receivers with packet loss rates P_1 and P_2 is:

$$\eta_{UN} \sim \frac{2L}{N} \frac{1}{2 + \frac{P_1 P_2}{1-P_1} + \frac{P_2}{1-P_2}}, \quad (A.1)$$

where $P_1 \leq P_2$ and the number of packets destined for each receiver $M \rightarrow \infty$.

Proof: Without loss of generality, assume that the receivers R_1 and R_2 want to receive the M odd and M even packets, respectively. The bandwidth gain of the network coding technique depends on how many pairs of lost packets among the two receivers that one can find in order to generate the combined packets. Let $e_1 = [\times|o]$ denote a transmission received unsuccessfully at receiver R_1 and successfully at receiver R_2 . Similarly, we denote erasure patterns $e_2 = [o|\times]$ and $e_3 = [\times|\times]$. Let random variables X_1 and X_2 , respectively, denote the number of erasure patterns e_1 at odd time slots and the number of erasure patterns e_2 at even time slots. Furthermore, let random variables Y_1 and Y_2 denote the number of erasure patterns e_3 at odd and even time slots respectively. Based on the central limit theorem we have $Pr(X_1 \leq X_2) \rightarrow 1$ as $M \rightarrow \infty$. This is because by assumption $P_1 \leq P_2$, consequently, $P_{e_1} = P_1(1-P_2) \leq P_{e_2} = P_2(1-P_1)$. Thus, the combined packets are dominated by X_2 , the number of erasure pattern e_2 at the receiver which has higher packet loss rate. Retransmitted packets can be classified into two groups: the combined and non-combined packets. Hence, the total number of transmissions expected to deliver M packets to each receiver successfully is

$$T = 2M + E[X_2] \cdot E[Z_2] + E[Y_1] \cdot E[Z_1] + E[Y_2] \cdot E[Z_2], \quad (A.2)$$

where Z_1 and Z_2 are the random variables denoting the numbers of attempts before a successful transmission for

R_1 and R_2 , respectively; Z_1 and Z_2 follow the geometric distribution, $E[Z_1] = \frac{1}{1-P_1}$ and $E[Z_2] = \frac{1}{1-P_2}$. Note that $E[X_2] + E[Y_2] = MP_2$ is the expected number of lost packets at receiver R_2 . Substituting $E[Z_1]$, $E[Z_2]$ into (A.2), the expected number of transmissions to successfully deliver M packets for R_1 and R_2 is given by

$$T \sim 2M + \frac{MP_2}{1-P_2} + \frac{MP_1P_2}{1-P_1}, \quad (\text{A.3})$$

and dividing by M we obtain

$$T_{UN} \sim 2 + \frac{P_1P_2}{1-P_1} + \frac{P_2}{1-P_2}. \quad (\text{A.4})$$

Note that each packet contains L information bits out of N bits, consequently, the throughput efficiency for NC unicast is

$$\eta_{UN} \sim \frac{2L}{N} \frac{1}{2 + \frac{P_1P_2}{1-P_1} + \frac{P_2}{1-P_2}}.$$

■

Proof: (Theorem 4.3 for the unicast wireless scenario)

We prove by induction. Without loss of generality we assume that $P_i \leq P_j$ if $i \leq j$, $\{i, j\} \in \{1, 2, \dots, K\}$. First, let us consider the base case $K = 2$. We have

$$\begin{aligned} \eta_{UN} &\sim \frac{2L}{N} \frac{1}{2 + \sum_{i=1}^2 \frac{\prod_{j=i}^2 P_j}{1-P_i}} \\ &\sim \frac{2L}{N} \frac{1}{2 + \frac{P_1P_2}{1-P_1} + \frac{P_2}{1-P_2}}. \end{aligned} \quad (\text{A.5})$$

The theorem holds for $K = 2$ since (A.5) follows directly from the Proposition A.1.

We now prove that the theorem holds for $K = 3$. Fig. 10(a) and Figs. 10(b), (c) and (d), respectively, present all possible erasure patterns and its decompositions. Let us first consider the erasure patterns shown in Fig. 10(b), that represents a scenario in which the packets are intended to R_1 or R_2 , and lost at R_3 . Hence, in the retransmission phase, the most efficiency technique that the BS can do is to consider combining error packets, if possible, for R_1 and R_2 only and some non-combined packets will be retransmitted alone. In other words, the BS uses the same combining strategy as that of the 2-receiver unicast scenario. Therefore, the expected number of transmissions required to deliver the lost packets shown in Fig. 10(b) is

$$T_{UN}^3(1) = T_{UN}^2P_3, \quad (\text{A.6})$$

where $T_{UN}^2 \sim \frac{MP_1P_2}{1-P_1} + \frac{MP_2}{1-P_2}$ denotes the expected number of retransmissions required to deliver the lost packets for two receivers R_1 and R_2 .

For the second and the third decompositions in Figs. 10(c) and (d), the BS combines the error packets as $1 \oplus 2 \oplus 3$, $1 \oplus 3$ and $2 \oplus 3$. The number of available ingredient packets for each type of the coded packets is dominated by R_3 , the receiver has the largest packet loss rate. For example, in the combination for all receivers $1 \oplus 2 \oplus 3$, the average number of available packets at R_1 , R_2 and R_3 respectively are $m_1 = MP_1(1-P_2)(1-P_3)$, $m_2 = MP_2(1-P_1)(1-P_3)$ and $m_3 = MP_3(1-P_1)(1-P_2)$. This implies that the ingredient packet constructing the coded packets for all receivers is dominated by the receiver with the highest packet loss rate,

$\max_{i \in \{1,2,3\}} \{m_i\}$. Then some lost packets that can not be combined will be retransmitted alone. The combinations are illustrated in Figs. 10(c) and (d). Let a random variable X_i denote the number of lost packets at receiver R_i in Figs. 10(c) and (d). Using the same argument as that of the broadcast scenario, we then can prove $X_3 = \max_{i=1,2,3} \{X_i\}$ with probability 1. Hence, the expected number of retransmissions required for the erasure patterns in Figs. 10(c) and (d) is given by

$$\begin{aligned} T_{UN}^3(2) &= E \left[\max_{i \in \{1,2,3\}} \{X_i\} \right] \\ &\sim E[X_3] \\ &\sim \frac{MP_3}{1-P_3}. \end{aligned} \quad (\text{A.7})$$

Adding up $3M$ transmissions used for transmitting original packets with (A.6) and (A.7) we obtain the expected number of transmissions needed to deliver all intended data. That is

$$\begin{aligned} T_{UN}^3 &= 3M + T_{UN}^3(1) + T_{UN}^3(2) \\ &\sim 3M + \frac{MP_1P_2P_3}{1-P_1} + \frac{MP_2P_3}{1-P_2} + \frac{MP_3}{1-P_3} \end{aligned} \quad (\text{A.8})$$

T_{UN}^3 divided by $3M$ which is the total number of useful data packets we prove the theorem for $K = 3$.

Now, suppose the theorem holds for $K = n-1$, $n \geq 3$. This implies that the expected number of transmissions required to deliver M packets for each receiver is

$$T_{UN}^{n-1} \sim (n-1)M + M \sum_{i=1}^{n-1} \frac{\prod_{j=i}^{n-1} P_j}{1-P_i}. \quad (\text{A.9})$$

We then prove that the theorem holds for $K = n$. Let T_{UN}^n denote the expected number of transmissions required to deliver M packets for each receiver. There are n receivers, therefore, the BS needs to use nM transmissions to deliver the original packets for the receivers. In the retransmission phase, the BS considers using network coding to combine lost packets. The erasure pattern is decomposed into three subsets S_1 , S_2 and S_3 . The set S_1 represents erasure patterns of packets intended to $\{R_1, R_2, \dots, R_{n-1}\}$ and lost at R_n , while the set S_2 represents erasure patterns of packets intended to $\{R_1, R_2, \dots, R_{n-1}\}$ and successful at R_n (one can refer to Fig. 10(b) and (c) for the case $K = 3$); and the set S_3 represents erasure patterns of packets intended to R_n . Obviously, in the set S_1 , the BS considers combining lost packets for receivers $\{R_1, R_2, \dots, R_{n-1}\}$ only since these packets are lost at R_n . Hence, the expected number of retransmissions required for delivery the lost packets in the set S_1 is the same as that of the expected number of retransmissions required for retransmitting lost packets of $(n-1)$ -receiver scenario $\{R_1, R_2, \dots, R_{n-1}\}$. That is

$$\begin{aligned} T_{UN}^n(1) &\sim M \left(\sum_{i=1}^{n-1} \frac{\prod_{j=i}^{n-1} P_j}{1-P_i} \right) P_n \\ &\sim M \sum_{i=1}^{n-1} \frac{\prod_{j=i}^n P_j}{1-P_i}. \end{aligned} \quad (\text{A.10})$$

An arbitrary erasure pattern of the set S_2 can be paired up with an erasure pattern in S_3 to generate a coded packet. Note

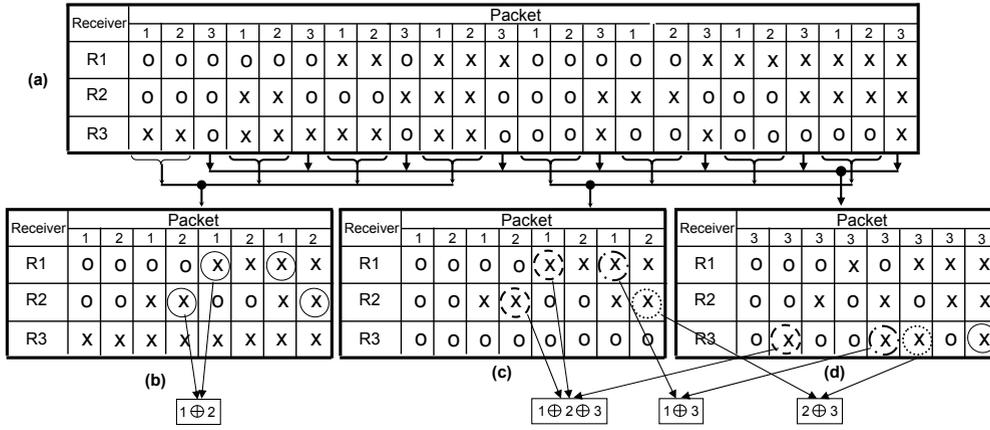


Fig. 10. Erasure pattern for 3-receiver unicast wireless scenario. Packets numbered as 1, 2 and 3 denote time slots used for transmitting data for receivers R_1 , R_2 and R_3 , respectively. The circle patterns imply the errors that need to be retransmitted (either in combined packets or non-combined packets).

that in these combinations, every coded packet contains the information of packets intended to R_n . Let a random variable Y_i denote the number of lost packets of receiver R_i in the sets S_2 and S_3 . Since $P_n = \max_{i \in \{1,2,\dots,n\}} \{P_i\}$, therefore, the expected number of retransmissions required to deliver all lost packets for the erasures in the set S_2 and S_3 is

$$\begin{aligned}
 T_{UN}^n(2) &= E \left[\max_{i \in \{1,2,\dots,n\}} \{Y_i\} \right] \\
 &\sim E[Y_n] \\
 &\sim \frac{MP_n}{1 - P_n}.
 \end{aligned} \tag{A.11}$$

Adding up nM , the transmissions for original packets, with (A.10) and (A.11), the retransmissions for lost packets, we obtain the expected number of transmissions required to deliver M packets for each receiver.

$$\begin{aligned}
 T_{UN}^n &\sim nM + M \sum_{i=1}^{n-1} \frac{\prod_{j=i}^n P_j}{1 - P_i} + \frac{MP_n}{1 - P_n} \\
 &\sim nM + M \sum_{i=1}^n \frac{\prod_{j=i}^n P_j}{1 - P_i}.
 \end{aligned} \tag{A.12}$$

Dividing M by T_{UN}^n , and multiplying the result by $K.L/N$, the ratio of information data and packet size, then by induction gives the theorem. ■

REFERENCES

- [1] R. Ahlswede, N. Cai, R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, pp. 1204–1216, July 2000.
- [2] J. Saltzer, D. Reed, and D. Clark, "End-to-end arguments in system design," *ACM Transaction on Computer System*, vol. 2, no. 4, pp. 277–288, November 1984.
- [3] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. Katz, "A comparison of mechanisms for improving tcp performance over wireless links," *IEEE/ACM Transactions on Networking (TON)*, vol. 5, no. 6, pp. 756–769, December 1997.
- [4] J. Clark Jr. and J. Cain, *Error-Correction Coding for Digital Communications*, New York: Plenum, 1982.
- [5] Y. Wu, P. A. Chou, and S.-Y. Kung, "Information exchange in wireless networks with network coding and physical-layer broadcast," in *Technical Report MSR-TR-2004-78, Microsoft Research*, Aug. 2004.
- [6] S. Katti, D. Katabi, W. Hu, H. Rahul, and M. Medard, "The importance of being opportunistic: Practical network coding for wireless environments," in *Proc. 43rd Annual Allerton Conference on Communication*, 2005.
- [7] C. Fragouli, J. Le Boudec, and J. Widmer, "Network coding: An instant primer," in *ACM SIGCOMM Computer Communication Review, Vol. 36, Issue 1*, Jan. 2006.
- [8] S. Deb, M. Effros, T. Ho, D. R. Karger, R. Koetter, D. S. Lun, M. Medard, and N. Ratnakar, "Network coding for wireless applications: A brief tutorial," in *IWWAN*, 2005.
- [9] Y. Birk and T. Kol, "Coding-on-demand by an informed source for efficient broadcast of different supplemental data to caching clients," *IEEE Transactions on Information Theory*, vol. 52, pp. 2825–2830, 2006.
- [10] Z. Bar-Yossef, Y. Birk, T. S. Jayram, and T. Kol, "Index coding with side information," in *The 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2006.
- [11] Salim Y. El Rouayheb, Alex Sprintson, and Costas N. Georghiades, "On the index coding problem and its relation to network coding and matroid theory," in <http://arxiv.org/abs/0810.0068>, Oct. 2008.
- [12] Atilla Eryilmaz, Asuman Ozdaglar, and Muriel Medard, "On delay performance gains from network coding," in *40th Annual Conference on Information Sciences and Systems*, 2006.
- [13] T. Ho, M. Medard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," in *Proc. 41st Annual Allerton Conference on Communication, Control, and Computing*, Oct. 2003.
- [14] T. Ho, M. Medard, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inform. Theory*, 2004.
- [15] M. Ghaderi, D. Towsley, and Jim Kurose, "Reliability benefit of network coding," in *Tech. Report 07-08, Computer Science Department, University of Massachusetts Amherst*, Feb. 2007.
- [16] B. Li Z. Li, "On increasing end-to-end throughput in wireless ad hoc networks," in *Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine)*, 2005.
- [17] Z. Li and B. Li, "Network coding: the case for multiple unicast sessions," in *Allerton Conference on Communications*, 2004.
- [18] D. Lun, M. Medard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," in *Proc. 42nd Annual e Allerton Conference on Communication, Control, and Computing*, , Sept./Oct. 2004.
- [19] A. Shiozaki, "Adaptive type-ii hybrid broadcast arq system," *IEEE Transactions on Communications*, vol. 44, pp. 420–422, April 1996.
- [20] S.R. Chandran and S. Lin, "Selective-repeat-arq schemes for broadcast links," *IEEE Transactions on Communications*, vol. 40, pp. 12–19, Jan. 1992.
- [21] S. Kallel and D. Haccoun, "Generalized type ii hybrid arq scheme using punctured convolutional codes," *IEEE Transactions on Communications*, vol. 38, pp. 1938 – 1946, Nov. 1990.
- [22] T. Tran, T. Nguyen, and B. Bose, "A joint network-channel coding technique for single-hop wireless networks," in *Fourth Workshop on Network Coding, Theory, and Applications*, Jan. 2008.
- [23] Stephen Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice-Hall, 1995.
- [24] T. Cover, "Broadcast channels," *IEEE Transactions on Information Theory*, vol. IT-18, pp. 2–14, January 1972.
- [25] T. Cover and J. Thomas, *Elements of Information Theory (second edition)*, Wiley-interscience, 2006.



Tuan Tran received his B.S. degree in Electronics and Telecommunications from Hanoi University of Technology (HUT), Vietnam, in 2000. From 2001 to 2004 he was with HUT as a lecturer. He got his first M.S. degree in Electronics and Telecommunications from HUT and second M.S. degree in Navigation and Related Applications from Polytechnic University of Turin, Italy, in 2004 and 2006, respectively. In 2006, he visited Istituto Superiore Mario Boella (ISMB), Italy. He is working towards his Ph.D. degree in Electrical and Computer Engineering at

Oregon State University, USA. His research interests include networking, channel coding, wireless communications, and multimedia communications.



Thinh Nguyen is an Assistant Professor at the School of Electrical Engineering and Computer Science of the Oregon State University. He received his Ph.D. from the University of California, Berkeley in 2003 and his B.S. degree from the University of Washington in 1995. He has many years of experience working as an engineer for a variety of high tech companies. He has served in many technical program committees. He is an associate editor of the IEEE Transactions on Circuits and Systems for Video Technology, the IEEE Transactions

on Multimedia, the Peer-to-Peer Networking and Applications. His research interests include Multimedia Networking and Processing, Wireless Networks, and Network Coding.



Bella Bose received the B.E. degree in electrical engineering from Madras University, Madras, India in 1973, the M.E. degree in electrical engineering from Indian Institute of Science, Bangalore, in 1975, and the M.S. and Ph.D. degrees in computer science and engineering from Southern Methodist University, Dallas, TX, in 1979 and 1980, respectively.

Since 1980, he has been with Oregon State University, Corvallis, Oregon, where he is a Professor and the Associate Director for the School of EECS. His current research interests include error control

codes, fault-tolerant computing, parallel processing, and computer networks. Bose is a Fellow of both ACM and IEEE.



Vinodh Gopal is a Senior Staff Architect at Intel Corporation working on content-processing algorithms and acceleration of compute-intensive applications. His areas of expertise are microprocessor and System-on-chip (SoC) architecture, cryptography, compression and network processing. In his current role in Intel's Embedded and Communications group, he is responsible for leading technical research, driving product development and technologies at high performance for various market segments. He received a Bachelors degree in Computer

Science from the Indian Institute of technology (IIT-Bombay) in 95 and a Masters degree in CS from SUNY-Buffalo in 97.

Vinodh joined Digital Equipment Corporation (DEC) in 97 and worked in the Alpha processor group, developing the world's fastest high-end RISC processors. He worked on multiple generations of Alpha micro-processors, most notably the ev7 processor, as the Project-Lead for DEC's logic synthesis optimizer. He joined Intel in 2002 as a Senior Engineer and worked as a key architect on the Floating-point execution unit of an IA64 Itanium high-performance RISC processor. In 2005, he joined the Embedded and Communications Group as the principal hardware architect for a cryptographic math processor for public key cryptography. He then led the hardware architecture for a Compression engine for the next-generation product. Vinodh has an extensive track-record of innovations with numerous pending patents and publications. He has received many key recognition awards at Intel, including two innovator awards for outstanding patent filings. He is an IEEE Member and has served as reviewer for many IEEE conferences.