A Network Coded ARQ Protocol for Broadcast Streaming over Hybrid Satellite Systems

Sameh Sorour and Shahrokh Valaee
The Edward S. Rogers Sr. Department of Electrical and Computer Engineering
University of Toronto
Toronto, ON, M5S 3G4, Canada
Email:{samehsorour, valaee}@comm.utoronto.ca

***Abstract*—Due to the high round trip delay in satellite systems, the retransmission of lost packets using conventional ARQ schemes is performed in a very rigid manner and after a very long time of the initial packet transmission. This results in a high average packet delay and packet drop rate in broadcast streaming applications. Moreover, conventional ARQ schemes are generally inefficient in broadcast scenarios.**

In this paper, we propose a network coded ARQ protocol that performs both proactive and reactive packet retransmissions in hybrid satellite systems. The proposed protocol employs a network coding approach to generate efficient proactive retransmission packets without the knowledge of lost packets. This not only allows the transmission of these coded retransmissions before the arrival of the initial packets to their destinations but also achieves more efficient packet recovery compared to conventional ARQ. Reactive retransmissions in response to packet acknowledgments are then employed if one or more packets are still lost. Simulation results show considerable gains for our proposed protocol over the selective repeat ARQ protocol in terms of average packet delay, packet drop rate and goodput.

I. Introduction

The increasing demand on high-speed streaming applications by fixed and mobile receivers, spread over geographically large areas, urge the need to employ geostationary (GEO) satellite systems as potential means to provide such services. For broadcast streaming applications, a reliable delivery of packets is required to occur at all receivers before a certain deadline beyond which these packets become useless. Consequently, the high altitude of GEO satellites represents a critical challenge against the practical use of these systems. Indeed, the long propagation delay over both the forward and reverse GEO satellite links greatly delays packet retransmissions using conventional ARQ schemes (such as Go-Back-N and selective repeat), which causes large packet delays and thus a high packet drop rate when some of these packets are not received before their deadlines.

To overcome the acknowledgement delays over reverse links, several works have considered different solutions [1]–[5]. In [3], a hybrid satellite system that employs terrestrial links in both packet acknowledgments and retransmissions was suggested. This solution may be good for fixed receivers where an established wired connection can be employed. However, it may not be quite adequate over terrestrial wireless networks for mobile receivers as it will greatly overload these networks with retransmission packets especially in broadcast scenarios with large number of receivers. [4], [5] suggested a modification of the previous solution that employs low bit rate terrestrial reverse links only for packet acknowledgments. We will consider this topology in our paper.

In all the previous solutions, it was assumed that ARQ is performed using the conventional selective repeat ARQ (SR-ARQ) protocol with accumulated, selective or negative packet acknowledgements. In SR-ARQ, the packet retransmission phase starts only when the receivers determine whether they correctly detected the packets or not. This results in a full round trip time (RTT) delay between a packet transmission and its retransmission if lost. To the best of our knowledge, the use of proactive retransmissions without prior knowledge of the lost packets in satellite systems was proposed in [6] using block coded packet level forward error correction (FEC). However, this work did not present a structured retransmission protocol for streaming applications in satellite systems.

On the other hand, network coding was proposed for packet retransmission in several works. In [7] and [8], network coded ARQ (NC-ARQ) schemes using opportunistic and random network coding, respectively, were proposed for wireless broadcast. However, these works assumed rateless network coding which is not suitable for streaming applications. Moreover, they were not tailored to cope with the long round trip time encountered in satellite systems.

In this paper, *we propose an NC-ARQ protocol for broadcast streaming applications over hybrid satellite systems*. The proposed protocol employs full deterministic network coding to generate proactive retransmissions without prior knowledge of the lost packets. Consequently, it allows the transmission of these coded packets before the reception of the original packets by the receivers. This is expected to reduce both the average delay of received packets and the number of dropped packets due to deadline violation, compared to SR-ARQ. Moreover, the use of NC-ARQ provides more efficient packet recovery opportunities compared to original packet repetition, which could result in significant average goodput gains. For further packet drop rate reduction, the protocol allows the use of reactive network coded retransmissions, in response to negative acknowledgements, if the proactive retransmissions

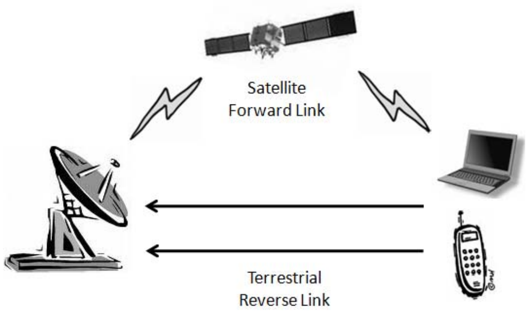

Fig. 1. Hybrid satellite system model

fail in helping all receivers to recover their lost packets.

The rest of the paper is organized as follows. In Section II, the hybrid satellite system model and parameters are introduced. In Section III, we describe the satellite based SR-ARQ protocol we employ for comparison. Our proposed NC-ARQ protocol for broadcast streaming applications is introduced in Section IV. Simulation results comparing our proposed protocol to SR-ARQ are depicted in Section V. Section VI concludes the paper.

II. System Model and Parameters

A. System Model

In this paper, we consider the hybrid satellite system depicted in Figure 1. It consists of an earth station, a GEO satellite, multiple dual-mode receivers and terrestrial reverse links. The earth station is connected to an internet backbone and/or media broadcast centers. When receivers request a service, the earth station transmits the requested information to the satellite which relays it to the receivers. We assume that different receivers experience different physical shadowing and fading levels. From the data link layer perspective, this translates into different packet loss rates (PLR). The PLR of receiver m is denoted by p_m. Upon packet loss by any of the receivers, it reports a negative acknowledgment through the low-bit rate terrestrial link to the earth station, which performs packet retransmission accordingly. The terrestrial links could be wired in case of fixed receivers or wireless (through a cellular or WiMAX network) in case of mobile receivers.

B. System Parameters

We define the round trip time (RTT) of the hybrid satellite system as the time from the departure of a packet from the earth station until the reception of its acknowledgement (either positive or negative) at the earth station. In this paper, we assume an RTT of 300 ms for all receivers: 250 ms for the propagation of the packet on the GEO satellite forward link plus 50 ms for the propagation of the acknowledgment on the terrestrial link. We also consider streaming applications with deadlines in the range of several hundreds of milliseconds which corresponds to 2 or 3 RTTs. This means that each lost packet could be retransmitted only once using the conventional ARQ schemes since any subsequent retransmissions will occur after the packet's deadline.

Let R_b be the bit rate of the forward channel and L be the packet length. Also define the bandwidth-delay product (BDP) of the system as the number of packets that are already sent by the earth station but their acknowledgements are not yet received back at the earth station. For GEO systems, this parameter is considered fixed for all receivers as the distances from the earth station to each of the receivers, through the satellite forward link, are almost equal. The BDP can be expressed as:

$$\text{BDP} = \frac{\text{RTT} \times R_b}{L} \tag{1}$$

III. Satellite Based SR-ARQ Protocol

In this section, we illustrate the satellite based SR-ARQ protocol we employ for comparison in this paper. In general SR-ARQ protocols, the stream of packets is partitioned into frames in which packets are numbered using a header field called sequence number. The SR-ARQ protocol employs the sliding window mechanism, where any packet in the window can be transmitted over the channel without waiting for the positive acknowledgements of its preceding packets. When the leading packet in the window is acknowledged/discarded, the start and end edges of the window slide by one packet. The sliding window size must be kept smaller than or equal to the frame size to avoid the un-acknowledged transmission of two packets having the same sequence number, which will cause a confusion in their acknowledgments. As it is understood from its name, the SR-ARQ protocol retransmits only the negatively acknowledged packets.

To efficiently utilize the satellite link, the window size of a satellite based SR-ARQ protocol should be greater than or equal to the BDP of the hybrid satellite system. The reason for this choice of window size is to prevent idle times on the satellite link in awaiting for packet acknowledgments. To minimize the use of terrestrial links, we assume a negative acknowledgment scheme in which receivers report only lost packets. Consequently, the earth station assumes that all packets, transmitted more than one RTT ago and not negatively acknowledged, are correctly received by all receivers.

Since we consider streaming applications with deadlines in the order of 2 to 3 RTTs, we assume that the earth station retransmits a packet once it receives its negative acknowledgement. Afterwards, this packet is discarded since any further retransmissions of this packet will violate its deadline. Consequently, when a packet is retransmitted, it gets out of the sliding window of the SR-ARQ protocol.

ARQ schemes are known to adopt the ordered packet delivery principle, where the packets following a missing packet are stored at the data link (DL) layer and are not passed to upper layers until the missing packet is correctly received. Since a packet can be discarded in streaming applications, we assume that packets following a missing packet are not passed to the upper layer until this missing packet is either correctly received or discarded. This principle is justifiable for streaming

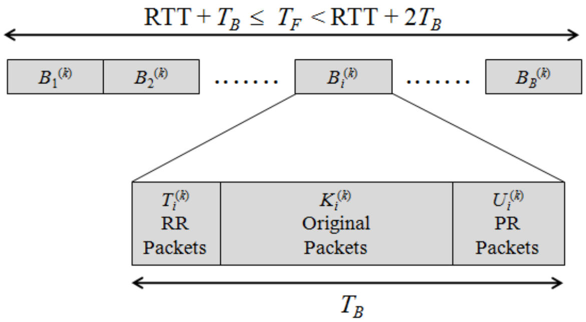

Fig. 2. Frame structure of the proposed protocols

applications since a packet cannot be employed before another packet that was generated and transmitted at an earlier instant unless it is discarded. Consequently, we will also consider the same principle in our proposed NC-ARQ protocol.

IV. Proposed NC-ARQ Protocol

In this section, we introduce our NC-ARQ protocol for broadcast streaming applications in hybrid satellite systems. The protocol design aims to minimize the average delay of received packets and reduce the number of dropped packets due to deadline violation.

A. Frame Structure

From the DL layer viewpoint, the temporal scale is partitioned into large frames, each divided into B blocks as depicted in Figure 2. Each block has a fixed duration T_B in which a fixed number of packets (denoted by N) is transmitted. The value of T_B can be set to any desired value considerably smaller than RTT and can be adjusted to fit the requirements of the upper layer protocols. The value of N depends on the block duration T_B, the packet length L and the employed modulation and coding scheme (i.e. the channel bit rate R_b). Let $B_i^{(k)}$ denote the i-th block in the k-th frame.

In our proposed NC-ARQ protocol, each block $B_i^{(k)}$ includes three types of packets as depicted in Figure 2:

- *Original Packets*: These are the packets of the streaming application requested by the receivers served by this satellite channel. These packets are transmitted uncoded. We denote the number of original packets in the i-th block of the k-th frame by $K_i^{(k)}$.
- *Proactive Retransmission (PR) Packets*: These packets are generated from the $K_i^{(k)}$ original packets of block $B_i^{(k)}$ using network coding. They are appended in the block after the original packets as a proactive measure to recover lost original packets. We denote the number of PR packets in the i-th block of the k-th frame by $U_i^{(k)}$.
- *Reactive Retransmission (RR) Packets*: These packets are network coded packets generated from the $K_i^{(k-1)}$ original packets of $B_i^{(k-1)}$. They are sent in reaction to the packet acknowledgments sent by one or multiple receivers for block $B_i^{(k-1)}$. We denote the number of RR packets in the i-th block of the k-th frame by $T_i^{(k)}$.

In the next sections, we will describe how these packets are generated and how their numbers are determined for each block.

Since we follow the ordered packet delivery principle, the RR packets are located at the beginning of each block $B_i^{(k)}$ to reduce the delay of the packets lost in block $B_i^{(k-1)}$ as well as correctly received packets in all subsequent blocks until $B_{i-1}^{(k)}$. For further delay reduction, the frame duration T_F should be designed such that the required retransmissions in $B_i^{(k)}$ occur shortly after the reception of $B_i^{(k-1)}$ acknowledgments. Consequently, T_F must be between RTT+T_B and RTT+2T_B.

For example, let the beginning of frame k be at t = 0. If T_B = 50 ms, the determination of lost packets in $B_1^{(k)}$ is done by the receivers at t = 300 ms (250 ms for the GEO propagation delay and 50 ms for the receivers to receive all the packets of the block). It will take an additional 50 ms to report this information to the earth station. Since there is a block that starts at t = 350 ms, the frame duration should be set to 350 ms = RTT+T_B. On the other hand, if T_B = 40 ms, the determination instant for $B_1^{(k)}$ is t = 290 ms, and thus the acknowledgments will reach the earth station at t = 340 ms. Since the beginning of the next block is at t = 360 ms, the frame duration is equal to 360 ms $<$ RTT+2T_B (380 ms).

B. Original and Proactive Retransmission (PR) Packets

Since the PR packets are sent from the earth station without prior knowledge of the lost original packets, they are generated from all $K_i^{(k)}$ original packets $\left[P_1, \ldots, P_{K_i^{(k)}}\right]$ using linear network coding. Let $\left[a_{u,1}, \ldots, a_{u,K_i^{(k)}}\right]$ be a coefficient vector whose elements are chosen from a Galois field of appropriate dimension. The u-th PR coded packet (P_u^c) can be generated as follows:

$$P_u^c = \sum_{v=1}^{K_i^{(k)}} a_{u,v} \, P_v \tag{2}$$

The coefficient vectors employed to generate the different PR packets should be chosen such that any $K_i^{(k)}$ packets from the original and PR packets are linearly independent. We refer to this property as the decodability property. In [7] and [8], these coefficient vectors are generated randomly from a large enough Galois field to guarantee the decodability property almost surely. The network coding technique employing such approach is called random network coding. In our centralized system, we can pre-generate a large number of coefficient vectors satisfying the decodability property and pre-store them in both the earth station and the receivers. This deterministic network coding approach not only guarantees the decodability property but also eliminates the need to append the coefficient vectors (or their generating seeds) to the PR packet headers thus removing this undesired overhead.

The use of PR packets, satisfying the decodability property, allows the receivers to decode the original packets by correctly receiving any $K_i^{(k)}$ packets from the original and PR packets. Consequently, the PR packets give the receivers the opportunity to retrieve their lost original packets at a much earlier time and with much higher efficiency compared to conventional ARQ schemes. This results in a lower packet drop rate and lower delay not only for these packets but several subsequent ones according to the ordered packet delivery principle.

The success probability of this proactive approach depends on the number of coded packets employed for proactive retransmission. Obviously, the greater this number the greater the probability of success. However, increasing the number of PR packets will reduce the number of original packets in the block thus reducing the original packet transmission rate (that we will refer to as *goodput*), which is not desirable. In our proposed protocol, we adopt an average approach as follows. It has been shown in [7] that the expected number of trials (n) for all the receivers of a broadcast system to correctly detect exactly $K_i^{(k)}$ packets can be approximated by:

$$n = \frac{K_i^{(k)}}{1 - \max_{m \in \mathcal{R}} p_m} , \tag{3}$$

where $\mathcal{R}$ is the set of all receivers. Since we have only $N - T_i^{(k)}$ slots for the transmission of original and PR packets in block $B_i^{(k)}$, the number of original packets the protocol transmits in this block, to satisfy (3), is:

$$K_i^{(k)} = \left\lfloor \left(N - T_i^{(k)} \right) \left(1 - \max_{m \in \mathcal{R}} p_m \right) \right\rfloor . \tag{4}$$

Thus, the number of PR packets the protocol sends in block $B_i^{(k)}$ is:

$$U_i^{(k)} = \left\lceil \left(N - T_i^{(k)} \right) \cdot \max_{m \in \mathcal{R}} p_m \right\rceil . \tag{5}$$

The value of $T_i^{(k)}$ in (4) and (5) is determined as will be explained in the next section.

C. Reactive Retransmission (RR) Packets

Let $R_{i,m}^{(k-1)}$ be the number of original and PR packets correctly received by the m-th receiver in block $B_i^{(k-1)}$. If $R_{i,m}^{(k-1)} \geq K_i^{(k-1)}$, the m-th receiver has enough packets to detect the original packets of $B_i^{(k-1)}$ and thus does not need RR packets in $B_i^{(k)}$. On the other hand, if $R_{i,m}^{(k-1)} < K_i^{(k-1)}$, the m-th receiver needs to correctly receive $K_i^{(k-1)} - R_{i,m}^{(k-1)}$ additional packets to detect all the original packets of $B_i^{(k-1)}$. These packets are retransmitted as RR packets in block $B_i^{(k)}$.

In our protocol, we set $T_i^{(k)}$ to be:

$$T_i^{(k)} = \max_{m \in \mathcal{R}} \left\{ K_i^{(k-1)} - R_{i,m}^{(k-1)} \right\} . \tag{6}$$

Thus, the protocol transmits $T_i^{(k)}$ network-coded combinations of the original packets of $B_i^{(k-1)}$. The reason behind this choice is that the retransmission of individual original packets in the broadcast case will benefit only the receivers that lost these packets whereas the retransmission of coded packets will benefit all the receivers as long as they get enough packets to decode their missing original packets.

Parameter	Value
Channel bit rate (R_b)	10 Mbps
Packet length (L)	1.25 kB
Block duration (T_B)	50 ms
Frame duration (T_F)	350 ms
No. of packets/block (N)	50 packets

TABLE I
SIMULATION PARAMETERS

V. PERFORMANCE EVALUATION

A. Performance Metrics

In this paper, we employ the following metrics to evaluate the performance of our proposed protocol and compare it with the SR-ARQ protocol:

- Average packet delay: defined for any receiver as the average time spent by the correctly received packets from their transmission instant at the earth station until their ordered delivery to the upper layers at that receiver.
- Packet Drop Rate: defined for any receiver as the ratio of the number of packets that suffered from deadline violation at that receiver to the total number of packets that arrived to the earth station for transmission.
- Average Goodput: defined as the rate at which original bits are transmitted over the channel.

We define the percentage improvement of metric X as being:

$$\% \text{ Improvement} = \frac{|X_{SR} - X_{NC}|}{X_{SR}} \times 100 . \tag{7}$$

B. Simulation Results

Table I illustrates the system parameters employed in our simulations. Based on the channel bit rate and packet length, the packet transmission time is 1 ms and thus the BDP can be computed from (1) to be 350 packets. A block duration of 50 ms will carry 50 packets. The frame duration is computed as illustrated in the example of Section IV-A. The simulations are done with MATLAB over a horizon of $1,000,000$ frames (i.e. 350,000,000 packets) broadcasted to M receivers having different PLR. Figures 3 and 4 depict the performance comparisons between the SR-ARQ and NC-ARQ protocols in terms of average packet delay and packet drop rate, respectively, against the receivers' PLR. The simulation is done for 100 receivers and the performances of all receivers having the same PLR are aggregated in one point for each ARQ protocol. Figure 5 depicts the comparison of goodput against the number of receivers between SR-ARQ and NC-ARQ.

It can be seen from Figure 3 that our proposed protocol achieves 14% to 47% percentage improvement in average packet delay. We can also observe that the receivers with low packet loss rates (10^{-3} to 10^{-2}) receive the packets in their frames without need for reactive retransmissions. The reason behind this result is the transmission of a number of PR packets, in each block, that provides an average immunity against packet loss for the receivers with the highest packet loss rate as shown in (5). Consequently, the number of PR packets is higher than needed for the receivers with lower

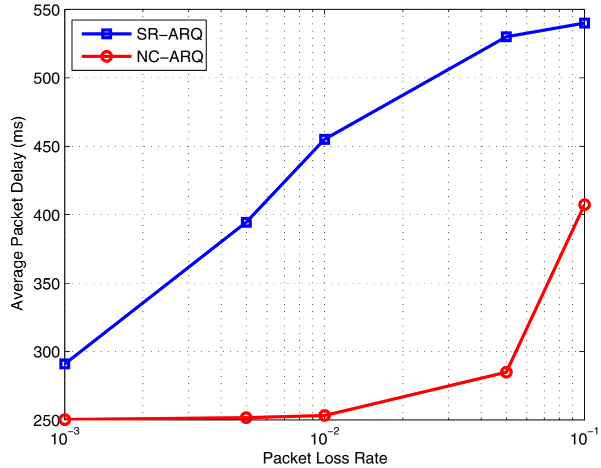

Fig. 3. Average packet delay vs packet loss rate

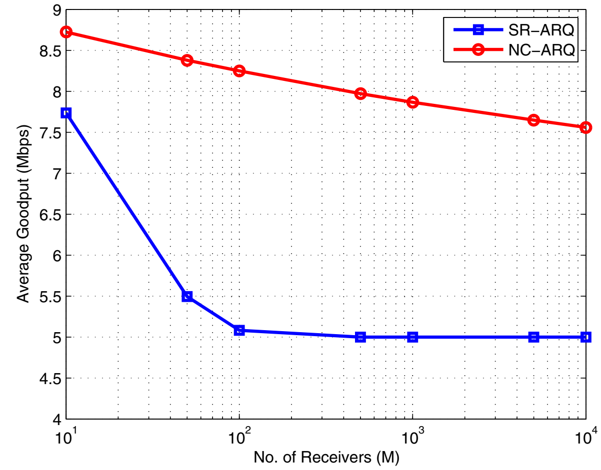

Fig. 5. Average goodput vs No. of receivers (M)

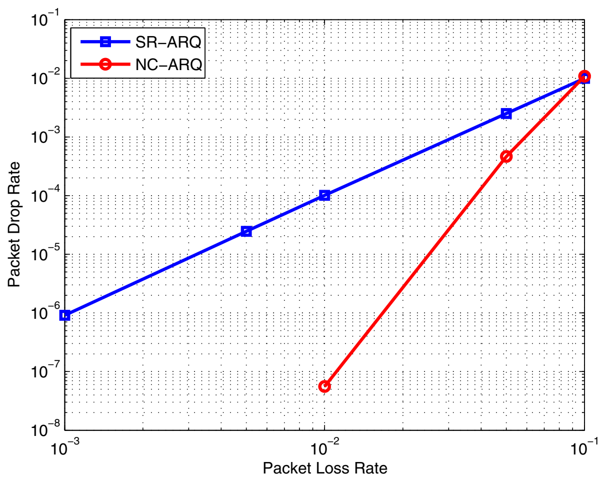

Fig. 4. Packet drop rate vs packet loss rate

packet loss rates. This gives them a higher chance of receiving the original packets in their blocks and eliminates their need for RR packets, thus reducing their average packet delay. This also reflects on the packet drop rate as depicted in Figure 4. We can see that our proposed protocol achieves no packet drops at all for receivers with packet loss rates below 10^{-2} due to the transmission of more PR packets than their needs. Figure 5 shows a 13% to 63% percentage improvement achieved by our proposed protocol in terms of average goodput. This gain is achieved due to the high efficiency of network coded packet retransmissions in broadcast scenarios compared to ARQ. This efficiency results in a lower number of retransmissions in our proposed NC-ARQ protocol compared to SR-ARQ that retransmits all lost packets by all receivers.

VI. Conclusion

In this paper, we proposed a network coded ARQ protocol for broadcast streaming applications over hybrid satellite systems. The proposed protocol exploits the abilities of full deterministic network coding in generating efficient proactive retransmission packets. These packets are transmitted right after their original packets without prior knowledge of their loss status. This greatly improves the average packet delay performance in such high RTT systems. Moreover, along with the reactive retransmission packets, they provide higher immunity to packet drops caused by deadline violations. Simulation results showed that our proposed NC-ARQ protocol indeed achieves considerable gains over the conventional SR-ARQ protocol in terms of average packet delay and packet drop rate. It also achieved large goodput gains as it exploits the network coding abilities to achieve lower number of retransmissions compared to the SR-ARQ protocol.

References

[1] G. Cao and Y. Wu, "Reliable multicast via satellites," *International Conference on Information Technology: Coding and Computing*, pp. 183–188, Apr 2001.

[2] E. Weigle, M. Hiltunen, R. Schlichting, V. Vaishampayan, and A. Chien, "Peer-to-peer error recovery for hybrid satellite-terrestrial networks," *Sixth IEEE International Conference on Peer-to-Peer Computing (P2P'06)*, pp. 153–160, Sept. 2006.

[3] D. Friedman and A. Ephremides, "A scheme to improve throughput for ARQ-protected satellite communication," *Fifth International Mobile Satellite Conference*, pp. 39–43, Jun. 1997.

[4] S. Cho, "Adaptive error control for hybrid (satellite-terrestrial) networks," *IEEE Wireless Communications and Networking Conference (WCNC'99)*, vol. 2, pp. 1013–1017, 1999.

[5] N. Celandroni, E. Ferro, G. Giambene, and M. Marandola, "Sat01-3: TCP performance in a hybrid satellite network by using ACM and ARQ," *IEEE Global Telecommunications Conference (GLOBECOM'06)*, pp. 1–6, December 2006.

[6] U. Mudugamuwa, M. Karaliopoulos, R. Tafazolli, and B. Evans, "Reliable multicast transport and power scheduling for MBMS delivery over 3G mobile satellite systems," *IEEE 59th Vehicular Technology Conference (VTC'04-Spring)*, vol. 5, pp. 2836–2841, May 2004.

[7] D. Nguyen, T. Tran, T. Nguyen, and B. Bose, "Wireless broadcast using network coding," *Third Workshop on Network Coding, Theory and Applications (NetCod'07)*, pp. 1–11, January 2007.

[8] ——, "Hybrid ARQ-random network coding for wireless media streaming," *Second International Conference on Communications and Electronics (ICCE'08)*, pp. 115–120, June 2008.