Broadcasting XORs: On the Application of Network Coding in Access Point-to-Multipoint Networks

The MIT Faculty has made this article openly available. Please share how this access benefits you. Your story matters.

| Citation       | Fouli, Kerim, Jerome Casse, Ivan Sergeev, Muriel Medard, and Martin Maier. “Broadcasting XORs: On the Application of Network Coding in Access Point-to-Multipoint Networks.” Lecture Notes in Computer Science (2012): 25–36. |
|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| As Published   | http://dx.doi.org/10.1007/978-3-642-34976-8_3                                                                                                                                                          |
| Publisher      | Springer-Verlag                                                                                                                                                                                            |
| Version        | Author's final manuscript                                                                                                                                                                                 |
| Accessed       | Wed Mar 16 03:46:02 EDT 2016                                                                                                                                                                             |
| Citable Link   | http://hdl.handle.net/1721.1/90404                                                                                                                                                                         |
| Terms of Use   | Creative Commons Attribution-Noncommercial-Share Alike                                                                                                                                                   |
| Detailed Terms | http://creativecommons.org/licenses/by-nc-sa/4.0/                                                                                                                                                       |
Broadcasting XORs: On the Application of Network Coding in Access Point-to-Multipoint Networks

Kerim Fouli\(^1\), Jérôme Cassé\(^2\), Ivan Sergeev\(^1\), Muriel Méard\(^1\), and Martin Maier\(^3\)

\(^1\) Research Laboratory of Electronics (RLE), Massachusetts Institute of Technology (MIT), Cambridge, MA 02139, USA \{fouli, vsergeev, medard\}@mit.edu
\(^2\) Department of Mathematics and Applications, École Normale Supérieure (ENS), Paris, France 75005 \jerome.casse@ens.fr
\(^3\) Optical Zeitgeist Laboratory, INRS, Montréal, Canada H5A 1K6 \maier@emt.inrs.ca

Abstract. We investigate network coding (NC) in access point-to-multipoint (PMP) broadcast networks. Characterized by a shared unicast upstream channel and a time-shared broadcast downstream channel, PMP networks are widely deployed in optical and wireless access networks. We develop a queuing-theoretic model of NC at the medium access control (MAC) sublayer and analyze the impact of NC on packet delay. Our analysis is validated through discrete-event simulation and demonstrates significant delay advantages for NC under high loads and localized traffic.

Keywords: Access Networks, Network Coding, Packet Delay, Point-to-Multipoint, Polling.

1 Introduction

Network coding (NC) is an innovative technology that has been shown to improve throughput, simplify routing, and provide robustness against transmission errors and failures in various packet networks [1, 2]. In addition to their traditional forwarding and routing functions, NC-enabled nodes may combine or separate transient bits, packets, or flows through coding and decoding operations without loss of information. A growing number of promising NC applications have been proposed in diverse areas such as wireless mobile networks, video transmission, peer-to-peer networks, security, monitoring, and sensor networks [1, 3, 4]. In a recent study, significant throughput gains were demonstrated experimentally in NC-enabled WiFi-based mesh networks [5].

In this work, we focus on the application of NC within a particular class of point-to-multipoint (PMP) broadcast networks. Figs. 1 (a) and (b) depict their fundamental characteristics. These are centralized networks with a number of user nodes (U) communicating exclusively through a central access point (AP) as follows. The uplink from any user towards the AP is a point-to-point link
while the downlink from the AP back to the users is a broadcast PMP link. Although the uplink and downlink may use more than one frequency channel or transmission medium (e.g., separate fibers, wireless spatial diversity), we restrict our study to the use of one separate channel in each of the upstream and downstream directions. The AP arbitrates the transmissions of the terminals over the uplink channel dynamically, whereas the downlink uses time-division multiplexing (TDM).

In such PMP architectures, NC can exploit the underlying broadcast architecture to convert unicast transmissions into more efficient broadcast transmissions, as depicted in Figs. 1 (c) and (d). In this illustrative scenario, two packets are exchanged between two users, $U_1$ and $U_2$. Without NC, such an exchange may be performed in four separate packet transmissions, with the hub receiving and then broadcasting each packet individually, as shown in Fig. 1 (c). With NC, the hub may code the received packets into a single packet using a simple bitwise exclusive-OR (XOR) operation, denoted by $\oplus$ (see Fig. 1 (d)). Upon receiving the coded packet, the terminals decode the packets destined to them using a copy of their previously transmitted packets. NC hence achieves the packet exchange in only three packet transmissions, using 50% less downstream bandwidth.

Fig. 1. Network coding in Point-to-Multipoint (PMP) networks.

The example of Figs. 1 (c) and (d) is a particular case of NC where the receiver nodes ($U_1$ and $U_2$) use copies of their own previously transmitted packets to decode received packets. The concept has been explored in the context of wireless communications, where it is denoted reverse carpooling [6], piggybacking, or pairwise XOR coding.

The centralized form of reverse carpooling illustrated in Fig. 1 is applicable not only to PONs, but to a number of access-metro network architectures such as wireless local area networks (e.g., infrastructure-mode WiFi), cellular access networks (e.g., long term evolution (LTE) and WiMAX), broadband satellite
networks, and cable networks (e.g., DOCSIS). The study of localized traffic in PMP networks is motivated by the tremendous growth of traffic generated by applications that stand to benefit from local packet exchanges, including user-generated high-definition video, peer-to-peer, video gaming, as well as voice and video conferencing [10]. Furthermore, content caching at user nodes may open up further opportunities for local exchanges.

The application of NC in a centralized PMP setting for local traffic was recently studied in the context of passive optical networks (PONs) through simulation [8, 7, 9]. However, previous studies of NC in PONs offer limited analyses of performance gains. Our work differs from previous performance analyses of NC in PONs [7] in the following ways. First, we do not consider normally distributed traffic rates. Second, we focus on the use of NC at the MAC sublayer. Most importantly, we are concerned with the queueing effects on delay within a switch, particularly at high loads.

In this article, we present a generic queuing-theoretic framework of a PMP network. To simplify the analysis, we ignore upstream channel arbitration and focus on queuing delay at the AP. We derive expressions for the average throughput and queuing delay under local traffic when NC is applied at the MAC sublayer. Our analytical results are then validated through OPNET [11] discrete-event simulations. In our discussion, we emphasize the high-load regime, where NC gains are visible.

The remainder of this article is organized as follows. In Section 2, we describe our analytical model and assumptions. In Section 3, we show analytically the potential advantages of the use of NC in PMP networks through deriving or bounding the average throughput, queue size, and queuing delay at the AP. In Section 4, we verify the conducted analysis and further investigate NC through simulation. Finally, we conclude in Section 5.

2 Network Model

Let the PMP network have $n$ users exchanging traffic locally, as depicted in Fig. 2. As mentioned in Section 1, since we focus on queuing delay at the AP, we assume a contention-free upstream channel where packets transmitted by users arrive instantaneously at the AP.

At the AP, two queuing configurations are studied. In the first, incoming packets are simply stored in an infinite first-in-first-out (FIFO) queue, as shown in Fig. 2 (a), with a fixed service time $T$. In the second, incoming packets are stored in a two-dimensional buffer matrix according to their source-destination pair, as depicted in Fig. 2 (b). In this configuration, $Q_{i \rightarrow j}$ denotes the queue from user $i$ to user $j$, where $i$ and $j$ are distinct indices in $(1..n)$. The buffer matrix has $N = n(n-1)$ infinite queues. This queuing configuration was proposed in [7] to allow the NC-enabled server (NC server) to process source-destination queue pairs jointly, thus eliminating any required search for coding opportunities. Since the coding consists of low-complexity XOR operations, we assume that coding delays are negligible compared to queueing delays.
Fig. 2. PMP network model with (a) a single first-in-first-out (FIFO) queue or (b) a two-dimensional buffer matrix. NC’s bidirectional relaying implies that the symmetrical entries of the upper and lower triangulars are visited simultaneously by the NC server.

Each user is modeled as the source of a stream of packets with arrivals that are independent and identically distributed (i.i.d.) over a period $\frac{N}{T}$. Furthermore, each packet is destined to one of the remaining $(n-1)$ users in an i.i.d. fashion.

A NC-enhanced AP uses exclusively the buffer matrix configuration of Fig. 2 (b). For any distinct indices $i$ and $j$ in $(1..n)$, the NC server is capable of processing the pair of queues $Q_{i\rightarrow j}$ and $Q_{j\rightarrow i}$, denoted $(i \rightarrow j, j \rightarrow i)$, simultaneously. Alg. 1 gives a formal description of NC server operation. The NC server iterates through all queue pairs following a round-robin discipline, allocating a fixed time-slot $T$ to each queue pair irrespective of the queue contents. This allows it to code packets opportunistically [5]. Queuing delay in such a strict TDM scheme constitutes an upperbound for the dynamic case where the NC server does not wait when no packets are present in $(i \rightarrow j, j \rightarrow i)$.

**Algorithm 1 NC server**

```
for i = 1 → (n - 1) do
  for j = (i + 1) → n do
    Process $(i \rightarrow j, j \rightarrow i)$ in time $T$
    1. Queues non-empty ⇒ transmit one coded packet
    2. One queue empty ⇒ transmit one uncoded packet
    3. Both queues empty ⇒ wait $T$ (only TDM scheme)
  end for
end for
```
3 Analysis

In this section, we derive an expression for the average delay in the system for the two configurations of Fig. 2 under Poisson user-traffic arrivals. We hence assume that packet arrivals are Poisson with parameter $\lambda$, where $\lambda$ is the rate of arrival of packets originating from user $i$ and destined to user $j$.

3.1 No coding: the single-queue configuration

Under aggregate Poisson arrivals, the configuration of Fig. 2 (a) satisfies the Markov property [12], and is therefore a classical $M/D/1$ Markov chain where the fixed service time is $T$ and the total packet arrival rate is $N\lambda$.

The condition for stability is $N\lambda < \frac{1}{T}$. Therefore, the maximum system throughput satisfies

$$T_{\text{max}} < \frac{1}{T}.$$  \hspace{1cm} (1)

We use the Pollaczek-Khinchine (P-K) mean-value formula [12] to derive the average number of packets in the queue

$$N_Q = \frac{(N\lambda)^2}{2\{1 - (N\lambda)E(T)\}} = \frac{(N\lambda)^2T^2}{2\{1 - (N\lambda)T\}},$$  \hspace{1cm} (2)

and the average queuing delay

$$W = \frac{N\lambda T^2}{2\{1 - (N\lambda)T\}}.$$  \hspace{1cm} (3)

3.2 NC: the buffer matrix configuration

Definitions In the architecture of Fig. 2 (b), the time between any two visits from the NC server to any $Q_{i\rightarrow j}$ is $\frac{N}{2}T$. We define the following variables:

- $N_{t,i\rightarrow j}$: number of packets arriving in $Q_{i\rightarrow j}$ over the time interval $[0,t]$.
- $p_k$: probability of $k$ arrivals in an interval $[t,t + \frac{N}{2}T]$, or
  $$p_k = P\left(N_{t + \frac{N}{2}T,i\rightarrow j} - N_{t,i\rightarrow j} = k\right),$$  \hspace{1cm} (4)

where $i \neq j$. Our assumptions on incoming traffic ensure that $p_k$ is independent from $t$, $i$, and $j$. Under Poisson arrivals, $p_k$ is given by

$$p_k = e^{-\lambda \frac{N}{2}T} \frac{(\lambda \frac{N}{2}T)^k}{k!},$$  \hspace{1cm} (5)

where $\lambda$ is the packet arrival rate into any queue $Q_{i\rightarrow j}$ in our configuration. 
- $X_{t,i\rightarrow j}$: number of packets in $Q_{i\rightarrow j}$ at time $t$. 
– \(t_k(i \rightarrow j)\): the instant preceding the \(k^{th}\) visit of the NC server to \(Q_{i \rightarrow j}\).

Assuming that \(Q_{1 \rightarrow 2}\) is processed at time \(t = 0\), and that the server follows the procedure of Alg. 1, \(t_k(i \rightarrow j)\) is given by
\[
t_k(i \rightarrow j) = \left( (m - 1)(n - m) + (M - m - 1) \right) T + k \frac{N}{2} T,
\]
where \(m = \min\{i, j\}\), \(M = \max\{i, j\}\), and \(i \neq j\).

For example, for \(Q_{1 \rightarrow 2}\), the expression reduces to
\[
t_k(1 \rightarrow 2) = k \frac{N}{2} T,
\]
(7)

– \(A_{k+1,i \rightarrow j}\): number of packets arriving at \(Q_{i \rightarrow j}\) between the \(k^{th}\) and the \((k + 1)^{th}\) visit of \(Q_{i \rightarrow j}\). It follows that
\[
A_{k+1,i \rightarrow j} = N_k(i \rightarrow j),i \rightarrow j - N_k(i \rightarrow j),i \rightarrow j.
\]
(8)

– \(A\): A random variable representing the number of arrivals in any queue \(Q_{i \rightarrow j}\) over any period of length \(\frac{N}{2} T\). \(A\) has probability mass function \(\{p_k\}_{k \in \mathbb{N}}\), hence
\[
\mathcal{P}(A = k) = p_k.
\]
(9)

– \(A(x)\): the probability-generating function of \(A\), given by
\[
A(x) = \sum_{k=0}^{\infty} p_k x^k.
\]
(10)

– \((S_{k,i \rightarrow j})_{k \in \mathbb{N}}\): infinite series representing the number of packets (i.e., state) of \(Q_{i \rightarrow j}\) just before the \(k^{th}\) visit of the NC server. \((S_{k,i \rightarrow j})_{k \in \mathbb{N}}\) is defined by
\[
(S_{k,i \rightarrow j})_{k \in \mathbb{N}} = (X_k(i \rightarrow j),i \rightarrow j)_{k \in \mathbb{N}}.
\]
(11)

For instance, \(Q_{1 \rightarrow 2}\) is associated to the series
\[
(S_{k,1 \rightarrow 2})_{k \in \mathbb{N}} = (X_k \frac{N}{2} T,1 \rightarrow 2)_{k \in \mathbb{N}}.
\]
(12)

**Stability Condition and Throughput** We assume that any \(Q_{i \rightarrow j}\) receives on average less than one packet in any period of length \(\frac{N}{2} T\). Such a condition is reasonable as the NC server processes at most one packet during that period. This stability condition can be expressed as
\[
\mathbb{E}(A) = A'(1) < 1,
\]
(13)

where \(\mathbb{E}(\cdot)\) denotes the expected value.

With Poisson input traffic, stability requires
\[
\lambda \frac{N}{2} T < 1
\]
(14)

Therefore, the maximum system throughput satisfies
\[
T_{\max}^{NC} < N(\frac{1}{\frac{N}{2} T}) = \frac{2}{T}
\]
(15)
Markov Chain Under the procedure of Alg. 1, any queue $Q_{i\rightarrow j}$ perceives a single service from the NC server each $\frac{N}{2}T$ period, hence the TDM characterization. If the $(k+1)^{th}$ server visit finds $Q_{i\rightarrow j}$ empty (i.e., $S_{k,i\rightarrow j} = 0$), its next state $S_{k+1,i\rightarrow j}$ will be the number of new packet arrivals ($A_{k+1,i\rightarrow j}$). Otherwise, it will be the sum of previous state and arrivals, minus one served packet.

In other words, for any queue $Q_{i\rightarrow j}$, the series $(S_{k,i\rightarrow j})$ satisfies

$$S_{k+1,i\rightarrow j} = f(S_{k,i\rightarrow j}, A_{k+1,i\rightarrow j})$$

where

$$f(s, a) = \begin{cases} s + a & \text{if } s = 0, \\ s + a - 1 & \text{otherwise,} \end{cases}$$

and $A_{k+1,i\rightarrow j}$ is independent from $S_{k,i\rightarrow j}$ for any $k$. It follows that the series $(S_{k,i\rightarrow j})$ is a Markov chain with a transition matrix $A$ given by

$$A = \begin{pmatrix} p_0 & p_1 & p_2 & p_3 & \cdots \\ p_0 & p_1 & p_2 & p_3 & \cdots \\ 0 & p_0 & p_1 & 0 & \cdots \\ 0 & 0 & p_0 & p_1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

Fig. 3 illustrates the Markov chain described by $A$, where each state represents the number of packets waiting in $Q_{1\rightarrow 2}$. In Fig. 3, the transitions departing from state $S_k = 0$ are dashed whereas those departing from $S_k = 1$ are solid. The transitions departing from all states $S_k \geq 2$ (not shown) are identical to those departing from $S_k = 1$.

Under the stability condition of eqn. (13) and Poisson arrivals (eqn. (5)), it can be shown that the Markov chain $(S_{k,i\rightarrow j})_{k\in\mathbb{N}}$ associated to any queue $Q_{i\rightarrow j}$ is irreducible, aperiodic, and positive-recurrent [12]. Furthermore, since queues $Q_{i\rightarrow j}$ differ only by their initial conditions (i.e., $S_{0,i\rightarrow j}$), we use queue $Q_{1\rightarrow 2}$ as a representative of source-destination queue statistics in the rest of the study.

**Total delay in $Q_{1\rightarrow 2}$** We denote $(\pi_s)_{s\in\mathbb{N}}$ the stationary distribution of the Markov chain $S_{k,1\rightarrow 2}$, and $\Pi(.)$ its probability-generating function, given by

$$\Pi(x) = \sum_{s=0}^{\infty} \pi_s x^s.$$
We first derive the average number of packets in $Q_{1 \rightarrow 2}$ by calculating $\Pi(x)$ using the P-K transform equation then determining $\Pi'(1)$ [12]. The P-K formula yields

$$\Pi(x) = (1 - \rho) \frac{A(x)(x - 1)}{x - A(x)}$$  \hspace{1cm} \text{(20)}$$

where $\rho$ is the utilization factor, given by

$$\rho = E(A) = A'(1).$$  \hspace{1cm} \text{(21)}$$

Under the stability and traffic conditions of Section 3.2, we have

$$A(x) = \sum_{k=0}^{\infty} e^{-\lambda \frac{N}{2} T} \left( \frac{\lambda \frac{N}{2} T}{k!} \right)^i e^{\lambda \frac{N}{2} T (x-1)}. $$  \hspace{1cm} \text{(22)}$$

Therefore, equ. (20) yields

$$\Pi(x) = (1 - \lambda \frac{N}{2} T) \frac{(x-1)e^{\lambda \frac{N}{2} T (x-1)}}{e^{\lambda \frac{N}{2} T (x-1)}}.$$  \hspace{1cm} \text{(23)}$$

From equ. (23), we derive the following expressions for $\Pi(x)$ and $\Pi'(x)$:

$$\Pi(x) = (1 - \lambda \frac{N}{2} T) \frac{1}{1 + x \sum_{i=0}^{\infty} \left( \frac{-\lambda \frac{N}{2} T (i+1)}{(i+1)!} \right)^i (x-1)^i};$$  \hspace{1cm} \text{(24)}$$

$$\Pi'(x) = -(1 - \lambda \frac{N}{2} T) \sum_{i=0}^{\infty} \left( 1 - \frac{\lambda \frac{N}{2} T (i+1)}{(i+1)!} \right) \frac{(x-1)^i}{(1 + x \sum_{i=0}^{\infty} \left( \frac{-\lambda \frac{N}{2} T (i+1)}{(i+1)!} \right)^i (x-1)^i)^2}. $$ \hspace{1cm} \text{(25)}$$

Evaluating $\Pi'(x)$ for $x = 1$, we obtain

$$\Pi'(1) = \frac{(1 - \lambda \frac{N}{2} T) \lambda \frac{N}{2} T}{1 - \lambda \frac{N}{2} T}. $$ \hspace{1cm} \text{(26)}$$

Hence, for the NC configuration of Fig. 2 (b), the average queuing time is given by

$$W = \Pi'(1) = \frac{(1 - \lambda \frac{N}{2} T) \frac{N}{2} T}{1 - \lambda \frac{N}{2} T}. $$ \hspace{1cm} \text{(27)}$$

**Total number of packets in the system** Let $X_t$ denote the total number of queued packets in the system at time $t$, hence

$$X_t = \sum_{i,j} X_{t,i \rightarrow j}. $$ \hspace{1cm} \text{(28)}$$

It can be shown that

$$\lim_{t \rightarrow \infty} \max_{[t,t+\frac{N}{2} T]} E(X_t) \leq N \Pi'(1). $$ \hspace{1cm} \text{(29)}$$
Therefore, we can use $N(\Pi')$ as an upperbound for the number of packets in the buffer matrix:

$$N_Q \leq \frac{(1 - \lambda T) \lambda N^2 T}{1 - \lambda^2 T}$$

(30)

4 Numerical Results

4.1 Infinite Queues

In this section, we carry out OPNET [11] simulations in order to validate the results of Section 3 for both configurations of Fig. 2, where all queues are infinite. As for the analysis, we simulate uniform local traffic where input traffic is uniformly distributed over the sources and the destinations. Table 1 shows the main simulation parameters. We call the configurations of Fig. 2 (a) and (b) native and NC-TDM, respectively. Recall that in NC-TDM, the NC server waits for one packet duration even when both source queues are empty, as per Alg. 1.

| Table 1. Simulation Parameters |
|--------------------------------|
| Number of users | 8 |
| Packet size ($S_p$) | 1500 Bytes |
| Downstream data rate | 1 Gb/s |
| Warmup period | 2 s |
| Data collection period | 5 s |

Although the simulations illustrated in this section use a warmup period of 2 s, a warmup of 10 s was also implemented for all simulations, with identical results, hence verifying steady-state conditions at all load points. For all simulation plots in this section, we include 95% confidence intervals.

![Fig. 4. Average packet delay in native and NC-TDM configurations, for both analysis (dashed) and simulation (solid).](image-url)
Fig. 4 shows the superposed analysis and simulation plots for average queuing delay (i.e., waiting time) in both configurations. The analytical expressions used for the native and NC-TDM cases are given by equations (3) and (27), respectively. The offered load is increased from 0.1 Gb/s to the maximum stable load, given for the native and NC-TDM cases by equations (1) and (15), respectively. In Fig. 4, the simulation plots match the analysis for both the native and NC-TDM cases, particularly for the more relevant high loads. In the native case, the average difference between analysis and simulation for the highest load quarter (0.75 Gb/s to 1 Gb/s) is 5 µs, a relative difference of 3.6%. In the NC-TDM scheme, the average difference in the highest load quarter (1.5 Gb/s to 2 Gb/s) is 190 µs, a relative difference of 10%.

The NC-TDM plots show that the steady-state queuing delay remains stable up to 2 Gb/s. This indicates that network coding keeps the average queue size within the buffer matrix (i.e., the size of $Q_{i \rightarrow j}$) stable for loads above the available downstream data rate. Both analysis and simulation results confirm that network coding is therefore able to support loads up to twice the available downstream data rate under uniform local traffic.

Both analysis and simulation also show that there is a delay penalty incurred by the NC-TDM scheme at all loads below 1 Gb/s. This load penalty remains below 0.5 ms and is due to the TDM process described in Alg. 1: Whereas the native scheme is dynamic in nature, the NC-TDM scheme results in a time-slot without packet transmission for each empty source-destination queue pair. Such inefficiency may be removed using a dynamic NC scheme, as shown in the next section.

4.2 Finite Queues

In this section, we further run OPNET [11] simulations for queues that are limited in size. In addition to applying the parameters of table 1, we limit the total buffer size to $N_{\text{max}} = 1$ MByte. Moreover, we implement an additional scheme whereby NC is performed in a dynamic fashion (i.e., no waiting time in Alg. 1). We label this scheme $NC$-dynamic. In the NC configuration of Fig. 2 (b) which applies to both the NC-TDM and NC-dynamic schemes, we limit each source-destination queue ($Q_{i \rightarrow j}$) to $N_{\text{max}}/N$ in order to have equal total buffer resources compared to the native scheme. Note that the smaller size of the source-destination queues does not affect the maximum attainable throughput in the NC schemes, as is shown below.

Fig. 5 (a) depicts the average throughput for the native, NC-TDM, and NC-dynamic schemes when the total offered load ranges from 0.1 Gb/s to 2.5 Gb/s. As expected from throughput inequalities of equ. (1) and (15), throughput is equal to the offered load until it reaches the stability limit. That limit is equal to the downstream data rate ($T_{\text{max}} = 1$ Gb/s) for the native scheme and reaches twice the downstream rate ($T_{\text{NC}}^{\text{max}} = 2$ Gb/s) using network coding (see Fig. 5(a)).

Fig. 5 (b), we plot the average queuing delay in the three simulated schemes. We observe that the average delay curves reach saturation values in all implemented schemes, a consequence of the limits in queue sizes.
At very high loads, incoming packets that are not dropped are likely to see a nearly full queue. In the native scheme, their waiting time is upper-bound by

\[
W_{\text{max}} = \left( \frac{N_{\text{max}}}{S_p} \right) \times T = 8.3886 \text{ ms},
\]

where \( S_p \) is the packet size and \( T \) is the service time (i.e., packet transmission time). In the NC schemes, however, that figure becomes

\[
W_{\text{max}}^{\text{NC}} = \left( \frac{N_{\text{max}}}{NS_p} \right) \times \left( \frac{TN}{2} \right) = \frac{W_{\text{max}}}{2} = 4.1943 \text{ ms},
\]

accounting for the reduced queue size and the increased service cycle time. Fig. 5 (b) depicts the two computed delay upper-bounds. This theoretical 50% reduction in saturation delay is verified in the simulation results. The softer saturation seen in the NC schemes is due to the ability of network coding to drain queues faster, hence preventing the queue size from reaching its maximum level even at very high loads.

More importantly, compared to the NC-TDM scheme, the NC-dynamic scheme effectively eliminates any low-load penalty inherent to the TDM process. This is clear from the low-load inset of Fig. 5 (b). In addition to its improved throughput, the NC-dynamic scheme thus exhibits packet delays that are lower than the native scheme at all loads. At loads slightly above 1 Gb/s, the delay gains of network coding relative to the native scheme attain two orders of magnitude.
5 Conclusions

We have shown through analysis and discrete-event simulation that the application of network coding in Point-to-Multipoint (PMP) broadcast networks achieves potentially a two-fold throughput increase as well as queuing delay gains reaching two orders of magnitude at high loads and under uniform local traffic. In access PMP networks, such performance gains are a strong argument in favor of allocating more bandwidth to upstream channels at the expense of downstream channels, under highly localized traffic. Future investigation includes unbalanced pair loads, varying number of users, multicast traffic, new NC server algorithms, and dual-stage PMP models.

Acknowledgment

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. M. Médard and A. Sprintson (Editors), Network Coding: Fundamentals and Applications. Academic Press, Nov. 2011.
2. T. Ho and D. Lun, Network Coding: An Introduction. Cambridge University Press, Apr. 2008.
3. C. Fragouli, D. Katabi, A. Markopoulou, M. Médard, and H. Rahul, “Wireless Network Coding: Opportunities & Challenges,” in Proc., IEEE MILCOM, Orlando, FL, USA, Oct. 2007, pp. 1–8.
4. C. Fragouli and E. Soljanin, “Network Coding Applications,” Foundations and Trends in Networking, vol. 2, no. 2, pp. 135–269, Jan. 2007.
5. S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, “XORs in the Air: Practical Wireless Network Coding,” IEEE/ACM Transactions on Networking, vol. 16, no. 3, pp. 497–510, June 2008.
6. M. Effros, T. Ho, and S. Kim, “A Tiling Approach to Network Code Design for Wireless Networks,” in Proc., Information Theory Workshop, Punta del Este, Uruguay, Mar. 2006, pp. 62–66.
7. K. Miller, T. Biermann, H. Woesner, and H. Karl, “Network Coding in Passive Optical Networks,” in Proc., IEEE International Symposium on Network Coding, Toronto, ON, Canada, June 2010, pp. 1–6.
8. M. Belzner and H. Haunstein, “Network Coding in Passive Optical Networks,” in Proc., ECOC, Vienna, Austria, Sep. 2009, pp. 1–2.
9. K. Fouli, M. Maier, and M. Méard, “Network Coding in Next-Generation Passive Optical Networks,” IEEE Communications Magazine, vol. 49, no. 9, pp. 38–46, Sept. 2011.
10. Cisco Systems, Cisco Visual Networking Index: Forecast and Methodology, 2009-2014, June 2010.
11. OPNET Modeler 17.1 – http://www.opnet.com
12. L. Kleinrock, Queueing Systems, Volume I: Theory. John Wiley & Sons, 1975.