Network Coding Report 2

{Yang Yang, Ma Rong, Shang Shang, Wang Chen}@sjtu

I. INTRODUCTION

NETWORK coding became the spot light in the research field of wireless communication right after its birth given by Ahlswede [1]. The basic attraction showed by [1] is that allowing immediate node to mix information form different flows, the network as a whole can achieve the broadcast capacity. The most typical butterfly-pattern network given in our report 1 demonstrate that the network coding can make the multicast session works more efficiently with a 10/9 performance enhancement [1]. Although network coding has provoked some fresh thinkings, the theory actually has some differences with our practical network, especially with wireless network environment. What's more, very few network coding implementation exist and few for wireless environment.

A. Wired v.s. Wireless

Most performance analysis of network coding are based on several important assumptions, which, to a great extent, limit the application of network coding.

1) Network Modeling
Currently, when we want to translate a real network environment into a math model, we just translate the network into a graph where an edge between two nodes means that the two nodes are physically connected(or the radio range allows the two nodes to communicate).
2) Traffic Pattern
Multicast communication is studied with network coding of multiple unicast flows remains a largely unknown territory. Moreover, the sender and receiver are fixed and given.
3) Traffic Rate
The traffic rate (or distribution) is predetermined and do not change.

Under these assumptions, we can derive wonderful performance enhancement. However, wireless mesh network do not coordinate with these assumptions.

1) Broadcast is a nature characteristic for wireless communications, which means when the nodes are transmitting, all the nodes that have direct edge linking with it are affected. The channel of one particular edge is actually shared by other neigbhoring edge. So the math model of wireless network have more restriction than its wired counterpart.
2) Unicast communication is the dominate traffic pattern. In a random wireless environment, there are actually multiple unicast session with different senders and receivers. They do not signal their desire to communicate, but just start sending packets.
3) Traffic rate are bursty and varies over time rather than constant.
4) Connectivity in a wireless network is highly variable due to changing channel and medium conditions. This unpredictable nature makes the wireless network even harder for modeling.

B. Existing Network Coding Algorithm

- Random network coding

  In distributed file systems, suitably designed file-distribution strategies can find application in content-distribution networks, peer-to-peer networks and also distributed libraries. We have both client-server model as well as peer-to-peer network model. Network coding lead us to a new file-distribution network strategy and the new strategy is called Random network coding.

  In this algorithm, we consider a large file which is broken into m pieces. There are multiple nodes or distributed memory elements each of which can store k of the m pieces. The distributed memory elements are referred to as peers. No peer has any knowledge about what the other peers have stored and the peers does not coordinate for storing pieces of the file. A downloader completes its download once it gathers each of the m elements or has enough information to recover the m pieces that constitute the entire file. In this algorithm, we consider what is the probability that r peers can provide sufficient information for the down-loader to complete at leat x fraction of the download. With Random network coding, the file-distribution network can show better performance.

- COPE

  COPE, an opportunistic approach to network coding, is oriented at wireless mesh networks. Each node snoops on the medium, learns the status of its neighbors, detects coding opportunities and codes as long as the recipients can decode. The flexible design allows COPE to efficiently support multiple unicast flows, even when traffic demands are unknown and bursty and the senders and receivers are dynamic.

COPE greatly utilize the broadcast nature of wireless network (Figure) and easy to be implemented. in the following section, we will mainly talk about COPE algorithm. Section II will explore the requirement to apply network coding in wireless network. These requirement has lead to the COPE theory, which will be generally mentioned in section III. Following that, section IV gives a detailed implementation for COPE. On our schedule,there will be a performance analysis in some particular network environment in **Report 3**.

II. Problem To Be Solved

1) Collision in broadcast
In wireless environments, packet transfer relies on the nature of the medium. Network coding broadcasts a single encoded packet to multiple receives on the same broadcast nature of the medium. However, the current 802.11 broadcast protocol has no mechanism such as ack or RTS/CTS to avoid collision. As a result, broadcast performs unsatisfactorily when in the busy environment. High possibility of collision leads to the loss of the packet. Network coding can improve the capability for sending information and saving the bandwidth, hence to reduce the occurrence of collision. There are also another two way to avoid collision. One of them is to change the MAC layer completely, and the other one is using off-the-shelf hardware/drives on 802.11.

2) Cooperate with TCP
It is critical for network coding to work properly with TCP, since network coding runs on top of TCP. Network coding should consider both loss recovery and packet reordering. First, while TCP halving the transmission rate when packet loss, the 802.11 MAC retransmits the lost package locally at each hop. Moreover, network coding uses broadcast to transmit package, it is unclear for receivers to determine the reception and loss of the package. Advanced mechanisms are needed to assure the error rate of wireless network coding is within the range that TCP can handle. Second, network coding will cause packets from the same flow to reorder. Since TCP uses the packet sequence number to detect losses, this coding scheme still needs to be redesigned.

3) Working properly in bursty and dynamic environment
Time is needed to find the optimal coding strategy though the senders and the receivers are predetermined. But in reality, requirement for simultaneity does not allow large time delay. The situation will be worse when in the bursty and dynamic environment.

4) Applications in unicast
Unfortunately, most of the theoretical results for applications of network coding on unicast are negative. In unicast, packets from multiple flows may be encoded together at some intermediate node, but later they should be decoded because their paths will diverge. Scheme is needed to determine where and whether a particular packet should be decoded. If not, transferring unneeded data will take the valuable bandwidth.

5) Low complexity for encoding and decoding
Network coding needs operation for encoding and decoding. Operation of high complexity makes the network coding using in high throughput applications questionable. For practical implementation, the algorithms of encoding and decoding should have linear complexity.

III. COPE Theory

Cope is a scheme of network coding used in the wireless situation, exploiting the broadcast nature of medium to improve the performance of the data transmission. The key idea of the network coding is to insert a coding layer between the IP and MAC layers, which detects coding opportunities and use them to forward multiple packets in a single transmission.

Cope contains three main techniques:

A. *Opportunistic listening*

Due to the broadcast medium, nodes have opportunities to hear the packets which are not intended for them. Cope sets all the nodes in promiscuous mode, makes them be able to hear all the packets in the medium and store them for a limited time T. A particular buffer will be used to store the packets which are involved in encoding and decoding. In addition, each node broadcast reception report to its neighbors. The information of received packet will be recorded in the reception report area in the cope header. Reception reports are sent by annotating the data packets the node transmits.

B. *Opportunistic coding*

The main issue the protocol has to solve is that what and how packets to be coded. Each node has to answer this question relied on local information and without asking other nodes. Each node should exploit the reception report to find out what packets its neighbors have and thus generates the best coding scheme to make the best result within the least transmission times.

In fact, a node may have multiple options to choose which packets to XOR together to maximize the throughput. An example is displayed to illustrate the opportunistic coding.

The topology graph of node A,B,C,D is shown in Fig. 1 The table below shows the next-hop of each of these packets.

Packets	NextHop
P1	A
P2	C
P3	C
P4	C

Table below shows the packets in its forwarding queue in each nodes.

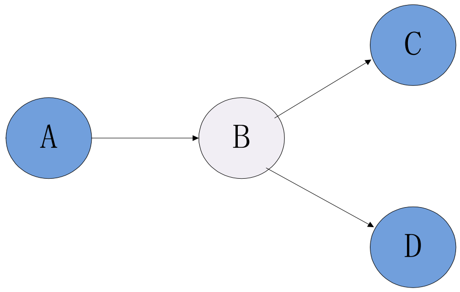

Fig. 1.

Node	PacketInQueue
A	P3, P4
B	P1, P2, P3, P4
C	P1, P4
D	P3, P1

When the MAC signals to B to transmit, B picks packet p1 from the head of the queue to transmit it. Assume B knows which neighbor has heard which packets. Now node B has a few coding options; the first option, it could send p1 XOR p2 to node A and C. Since C has p1 in its storage space, it can XOR it again with p1 XOR p2 to obtain the packet sent to it, i.e., p2. But, node A does not have p2, and thus cannot decode the encoded packet. Thus, if B sends p1 XOR p2, it will be a bad coding decision because only one neighbor can benefit from this transmission.. The second option is to send p1 p3allows both neighbors C and A to decode and obtain their intended packets, delivering two packets in a single transmission. This option shows a better coding decision for B. But the best coding decision for B would be to send p1 XOR p3 XOR p4 which allows all three neighbors to receive their intended packets. In general, a relay node should check various packet combinations to find the largest number of packets that can be delivered in one transmission while still allowing each of the intended recipients to decode its packet. The above example indicates a simple rule for choosing which packets to code together.

Theorem 1 To transmit n packets, $p_1, ..., p_n$, to n recipients, $r_1, ..., r_n$, a node can XOR the n packets together only if each intended recipient r_i has all n-1 packets p_j for j $\neq$ i.

This rule ensures that each next hop can decode the XOR-ed version to extract its native packet. Whenever a node has a chance to transmit a packet, it chooses the largest n that satisfies the above rule to maximize the benefit of coding.

C. Learning Neighbor State

During the transmission, the reception report may lose because of collision. In this situation, the sender node can not know what packets the receiver node has. Therefore, nodes need to guess whether a neighbor has a particular packet. The cope only allow one form of guessing: if a node receives a packet transmitted by neighbor A, it assumes that nodes closer to A than itself have also received the packet. This scheme enables nodes to make smarter encoding decisions. On the other side, if the nodes make some mistake when guessing, it only causes that the wrong packet to be transferred on top of the queue which is not a big deal.

IV. COPE Implementation

Previous section explained the three technique parts of COPE. In this section, we try to define more practical physical implementation to fulfill these techniques.

A. Opportunistic Listening

In opportunistic listening, each node cache temporarily the packet which treat the node as intended receiver. We make all nodes in the network store suitable packets for a limited amount of time T. In time T, the node should cache enough amount of packets which serves as coding candidates. In real wireless environment, taking 802.11b as a example, the transmitting rate is 11Mbps and the packet delay is usually on the order of tens of milliseconds. Just take T as 0.5 seconds, then at most we can cache hundreds of packets and the total amount of storage required is less than 750 kilobytes (11Mbps * 0.5). This is highly applicable.

In order to derive a reception report based on the packets in the cache, we can just give each packet a unique ID. Since the packets are generally randomly at each source node, we cannot guarantee the ID of each packet is unique if the ID is given by the source and appended to the packet. An efficient way to identify the packet may be fulfilled by representing the packet by its own hash code, MD5, for example. A more clever way would be using the destination address and sequence number of the packets to generate MD5 code. In this way, we can guarantee the uniqueness of packet ID and at the same time, reduce the computation complexity.

Reception Report should be updated in time to provide the sender with optimized coding depiction. Reception Report must be exchanged at a time interval much less than T since the cache of the node is limited and out of date packet will soon be dropped if new packets continues to arrive. There are two way that COPE exchange reception report.

1) Broadcast
   In this way reception report is exchanged periodically, but since broadcast doesnot have a ACK mechanism, so the reception report can be easily collided with each other without any alarm, the sender will have no assurance that the reception report is correctly received.
2) Piggy-back
   In this way reception report is appended to normal packets and doesn't affect the transmission of normal packets since it doesn't cause extra collision. Although piggy-back mechanism will not update reception report in time if the traffic is few, in these condition we actually doesnot rely much on network coding since the channel is not congested as assumed.
   So we can see that Piggy-back-based reception report is a preferable one. But we can also mix these two techniques together.

B. Opportunistic Coding

In theorem 1, optimized coding is found by deriving the largest n available for transmission. To get the largest n, we must know the reception report of neighboring node as will as current output queue. Here we give a pseudo code to explain how to get this largest n.

```
{PackateToBeXor}=NULL;
MainNextHop=NULL;
{SecondaryNextHop}=NULL;
packet=Popout{OutputQueue};
MainNextHop=packet.nexthop;
{PacketToBeXor} <-- packet;

for N=1 to Length{OutputQueue}
packetN=Nth element in the output
queue;
if(packetN.nexthop != MainNextHop)
if(packetN belongs to
{ReceptionReport of{SecondaryNextHop}})
if({PacketToBeXor} is a
subset of {ReceptionReport
of{packetN.nexthop}})
{SecondaryNextHop}<-- packetN.nexthop;
{PacketToBeXor} <--packetN;
end if;
end if;
end if;
end for;
```

Then

```
    frame=MainNextHop
        + {SecondaryNextHop}
        + XOR{PacketToBeXor}
        + MD5{PacketToBeXor};
    send(frame);
```

C. Learning Neighbor State

Due to the unstableness of reception report, the node may guess what the neighbors have heard. One way to guess the neighbor state is by the help of Network Layer Routing Table. In some Routing Protocol, each node can maintain a weighted metric denoting the connection performance between each pair of node in the network. In this way, we can assume that two node are close to each other and can be heard mutually if their connection performance is good. In other words, a node learns the location of other nodes in the network by having all nodes flood link state information. Alternatively, the nodes my use GPS to locate themselves. Then they flood their location to other nodes in the network.

D. packet sending strategy

COPE utilize the broadcast nature of wireless network and the packet send is received by more than one node. Under current TCP/IP protocol stack, this can only be fulfilled by broadcast (the MAC destination address will be set to all F), 802.11 broadcast packet has many intended receiver and it is unclear who should ack. In fact, 802.11 broadcast packets are not ack-ed. This means that the sender of a broadcast cannot infer the occurrence of collisions, and thus does not back off. Each of the node will try to send the packets as fast as possible, which lead to severe collision and performance deteriorate. This problem is solved if we add a new slim layer between Data link layer and network layer. Just name it Network coding layer.

After the packet from the network layer is properly encapsulated, it will go through Network coding layer, which will add a list of Intended receiver.The final MAC layer frame is showed in figure 3. Upon receiving a packet, the node will check if it is the main receiver of the packet by checking the Data link layer header. If it is not the main receiver, it will further check the network layer header to see whether it is an intended receiver. If so, the packet will go through further process. Notice that only the main receiver of the packet will send ACK to the sender. In this way we can benefit from the TCP/IP congestion control mechanism and at the same time, fulfill the COPE broadcast. This method is called Seudo Broadcast.

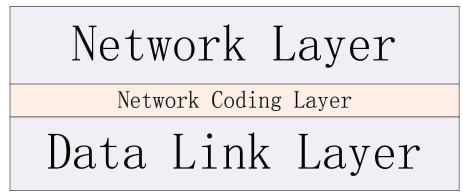

Fig. 2. Network coding layer

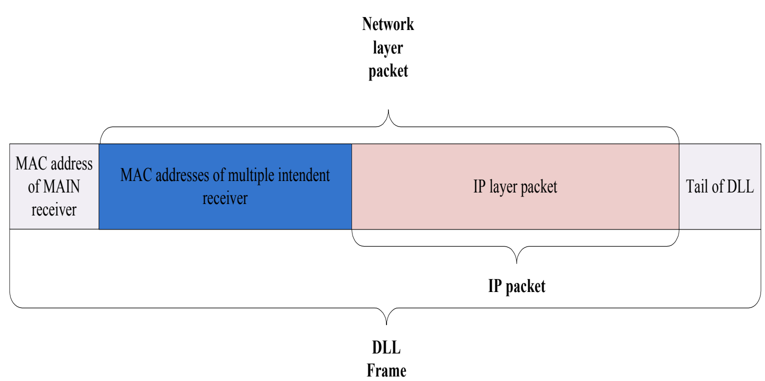

Fig. 3. MAC frame (after network layer header(blue part) is added)

V. Performance In Particular Network

On report 3

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, ***Network Information Flow***,IEEE Transactions on Information Theory, IT-46, pp. 1204-1216, 2000

[2] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, ***XORs in The Air: Practical Wireless Network Coding***,In ACM SIGCOMM 2006, Pisa, Italy, September 11 -15

[3] S. Katti, D. Katabi, Wenjun Hu, and Rahul Hariharan . ***The Importance of Being Opportunistic: Practical Network Coding For Wireless Environments***, In Proc. 43rd Allerton Conference on Communication, Control, and Computing, Monticello, IL, Sep. 2005.

[4] C. Gkantsidis, P. R. Rodriguez ***Network Coding for Large Scale Content Distribution***, Microsoft Research Technical Report (MSR-TR-2004-80)

[5] M. Mdard, S. Acedaski, S. Deb, and R. Koetter ***How good is random linear coding based distributed networked storage?***, Netcod 2005, Italy, April 2005

[6] P. A. Chou, Y. Wu, and K. Jain, ***Practical network coding***,Allerton Conference on Communication, Control, and Computing, Monticello, IL, October 20

[7] Z.Li and B.Li. ***Network coding:The case for multiple unicst sessions*** In Allerton Conference on Communications,2004