Lecture 4 — September 19, 2007

Lecturer: John Byers *BOSTON UNIVERSITY* *Scribe: Marisa Vasconcelos*

Today we discussed applications for **Network Coding** and introduced the COPE paper[2].

4.1 Introduction

Before the introduction of network coding, it was believed that the rate at which a source can send information to a set of destinations was determined by the min-cut of the graph(Max-flow min-cut theorem). Ahlswede et al [1] showed that using intermediate nodes to encode the information that they received, this limit could be achieved for every destination.

For example, if A and B want to send a packet to each other. Assume that A can only reach B via a router. Thus, A has to send the message to the router and the router will forward it to B. Similarly for B, its message has to be forwarded to the router and then to A. In this scenario, we used four transmissions to send these 2 packets.

If we consider a network coding approach, both A and B send their packets to the router, the router will XOR the packets and broadcast the XOR-ed packet. A and B can then recover their packets using the packet they just transmitted, which will be buffered temporarily. Thus, using network coding, we need only three transmissions.

Some assumptions must be made, if we want to use network coding. One is that we need to store the packets that we sent, to be able to decode received packets. A design consideration will be how long should packets be stored for. Another point is that the packets that will be encoded must arrive almost "at the same time" (systolic transmission).

In class, scenarios where network coding would be useful were presented. A *unicast* scenario, one-point-to-point flow, would not be a good candidate for coding, since TCP can achieve reliability using a feedback channel. However, in a point-to-point communication with multiple paths (flows) to go from the source to the destination, if we use good source coding, we don't need the feedback channel. Network coding will be particulary useful if we have one source and multiple destinations (Figure 4.1).

Coding for multiple sources to multiple destinations is still an open problem. Especially, because we need scheduling among the sources. In class, it was suggested to use a virtual source and random coding (random dispersions) to de-synchronize the sources, and reduce the number of redundant packets sent. For large networks, network coding will be done at each node in the network.

4.1.1 Problems and Challenges

The goal is to minimize the amount of work required to decode the message. There are polynomial time algorithms to compute the optimal encoding, but they are expensive and if

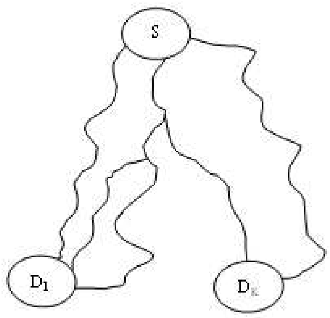

Figure 4.1. Multiple destinations

the network changes, e.g, a link goes down, the coding has to be recomputed.

Using Rabin's idea of choosing random coefficients, Equation 4.1, avoids these difficulties, but creates several problems. For instance, we will have more expensive Gaussian elimination required for decoding. Also, the per packet computation required in Equation 4.1 must be fast, to avoid overwhelming the router, and potentially causing packet losses.

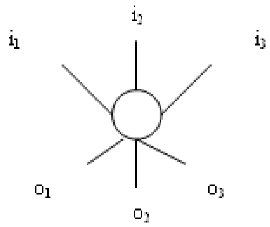

Figure 4.2. Local encoding

$$o_i = r_{Z_p} i_1 + r_{Z_p} i_2 + r_{Z_p} i_3 \tag{4.1}$$

If we have heterogeneous receiver rates, additional work needs to be done. For example, slow down the fastest receivers. Also, we have a problem with blocks that persist for a long time in the network. For example, if a node is waiting for a particular block to decode, this will be an overhead for the other receivers that are not interested in this block.

4.2 COPE

In [2], the authors present a new forwarding architecture, implemented as a thin layer between the IP and MAC layers, that utilizes network coding. COPE takes advantage of the fact that in a wireless medium each packet is broadcasted and the neighbors of the sender can hear each transmitted packet.

To maximize the chances of encoding and to make sure that each hop will have enough information to decode, each node has to know which packets its neighbors have. COPE uses reception reports and computes the delivery probability (ETX[1]) to estimate this information.

The main characteristics of *COPE* are:

- ***Opportunistic Listening***: nodes hear all the transmitted packets and store them for some period T.
- ***Opportunistic Coding***: the idea is to maximize the number of packets encoded in a single packet and guarantee that the receivers will have enough information to decode it.
- ***Learning Neighbor State***: COPE uses reception reports, where each node shares information about which packets it currently stores. One problem with these reports is if there is congestion in the network, some of those reports can be lost or delayed. Thus, a node cannot only rely on these reports. It also has to use some heuristics based on the ETX metric, to obtain a better view of the neighbor state.

[1] ETX : Expected number of transmissions required to deliver a packet to a destination.

Bibliography

[1] R. Ahlswede, N. Cai, S. Li and R. Yeung. “*Network Information Flow*“, IEEE Transactions on Information Theory, 46(4):1204-1216, July 2000.

[2] S. Katti, H. Rahul, W. Hun, D. Katabi, M. Medard and J. Crowcroft. “*XORs in the Air: Practical Wireless Network Coding*“, in Proceedings of ACM SIGCOMM ’06.