On analyzing and improving COPE performance

Fang Zhao, Muriel Médard
Research Laboratory of Electronics
Massachusetts Institute of Technology, Cambridge, MA 02139, USA
E-mail: {zhaof, medard}@mit.edu

***Abstract*—COPE is a new architecture for unicasts in wireless mesh networks that employs opportunistic network coding to improve total throughput. Katti *et al.* showed through experiments that this system significantly improves the throughput of wireless networks with UDP traffic, and several attempts have been made to analyze the COPE performance. However, they are not completely satisfactory. In this paper, we give a new analysis of COPE, and argue that the key to COPE's success lies in the interaction between COPE and the MAC protocol. The local fairness enforced by the MAC protocol among competing nodes play an important role here. Based on the analysis, we also propose a simple modification to the COPE system that can further improve the network throughput.**

I. Introduction

The introduction of network coding in 2000 [1] has the potential to revolutionize the way we operate networks. The broadcast nature of the wireless medium renders network coding particularly useful. One of the first practical network coding systems for wireless networks is COPE, introduced by Katti *et al.* [2], [3]. COPE is a new forwarding architecture for wireless network that inserts a coding shim between the IP and MAC layers, which identifies coding opportunities and benefits from them by forwarding multiple packets in a single transmission.

The outstanding performance of COPE has generated a lot of research interests. Some researchers tried to model the COPE system and analyze its performance [4], [5], and we will discuss these in more details in Section II. Others proposed new coded wireless network systems based on the idea of COPE. Dong *et al.* [6] proposed loop coding, which allows receivers to temporarily store coded packets for future decoding. Omiwade *et al.* [7] proposed BFLY, a localized network coding protocol that allows intermediate nodes not only to XOR packets together (as in COPE), but also to forward coded packets. Chaporkar *et al.* [8] presented a joint network coding and scheduling schemes to optimize network throughput.

So far, the attempts to model and analyze COPE's performance are not completely satisfactory. In this paper, we try to explain the COPE performance curve by closely examining one coding structure, and we show that the throughput gain is a result of the interaction between COPE and the MAC protocol. The local fairness enforced by the MAC protocol among competing nodes when distributing bandwidth plays an important role here. In addition, based on our observations in the analysis, we propose a simple modification to the COPE system that can further improve the network throughput.

The rest of this paper is organized as follows. We first introduce the basic idea of COPE and existing work on COPE performance analysis in Section II. Section III presents our analysis of COPE's performance under UDP traffic. In Section IV, we propose a simple improvement to COPE, and the paper is concluded in Section V.

II. Background

A. The COPE system

We first explain the basic idea of COPE by using the Alice-and-Bob network shown in Fig. 1. Here, Alice and Bob want to exchange a pair of packets via a router, R. In a traditional routing network, Alice and Bob would first send their packets to R, and then R forwards the two packets to their respective destinations in two time slots. This process takes 4 transmissions. However, if network coding is allowed in the router, after R has received the two packets from Alice and Bob, it can XOR the two packets together and broadcast this new packet. When Alice and Bob receive the XOR-ed packet, they can obtain each other's packet by XOR-ing again with their own packet. In this way, we utilize the broadcast nature of the medium and save one transmission, which can be used to send additional data, increasing the network throughput.

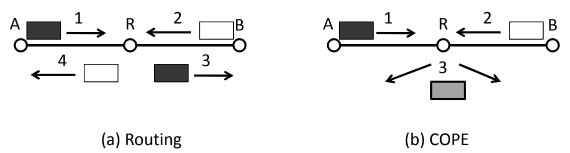

Fig. 1. Example of how COPE increases the throughput in the Alice-and-Bob wireless network.

Even larger coding gain can be obtained when more packets are coded together. For example, consider the cross network shown in Fig. 2. Here, nodes 1, 2, 4, 5 each has a packet to be sent to the opposite node via node 3 in the middle. In addition, when node 1 sends, nodes 2 and 4 can overhear the transmission. Same goes for nodes 2, 4, and 5. It is easy to see that in conventional routing network, it takes 8 transmissions for the 4 packets to be delivered. However, in COPE, we can first let the four source nodes send their packets to node 3, and then node 3 XORs all of them together and broadcast the coded packet. Since every node has its own packet and the two packets overheard from the transmissions of their neighbors, it can derive the packet destined to it from the XOR-ed packet.

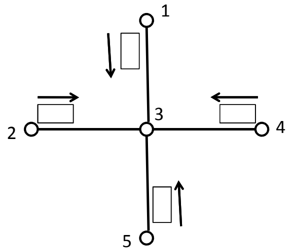

Fig. 2. Cross network.

Therefore, in COPE, the process only takes 5 transmissions and we save 3/8 of the bandwidth as compared to the routing case.

In summary, COPE employs network coding to utilize the broadcast nature of the wireless channel. The coding here is simple XOR, and the decoding is done at the next hop, *i.e.*, there is no forwarding of the coded packets. Implementation of the COPE system involves many practical issues, as explained in [2], [3]. Here, we only summarize the three main techniques incorporated in COPE:

1) **Opportunistic Listening:** Since the wireless channel is a broadcast medium, and the nodes are equipped with omni-directional antennae, COPE makes the nodes snoop on all communications over the wireless medium and store the overheard packets for a limited period. In addition, each node broadcasts reception report to its neighbors about which packets it has stored, to enable their neighbors to find coding opportunities.
2) **Opportunistic Coding:** Based on its knowledge of what the neighbors have, a node decides on what packets to code together. The rule it follows is to maximize the number of native packets delivered in a single transmission, while ensuring that each intended nexthop has enough information to decode its native packet.
3) **Learning Neighbor State:** In addition to using the reception reports to find out what packets a neighbor has, a node may also need to guess whether a neighbor has a particular packet. This is done intelligently by leveraging the routing computation. In the absence of deterministic information, COPE estimates the probability that a particular neighbor has a packet as the delivery probability of the link between the packet's previous hop and the neighbor.

COPE has been implemented by Katti *et al.* [3] in a 20-node wireless mesh network, and tested with both TCP and UDP traffic. In their tests, TCP does not show any significant improvement with coding, and this is due to TCP's reaction to collision-related losses. Due to collisions at the bottleneck nodes, the TCP flows suffer timeouts and excessive back-off. Thus, the bottleneck nodes never see enough traffic to make use of coding. Few coding opportunities arise, and hence the throughput performance is the same with and without coding.

On the other hand, with UDP traffic, COPE can provide a several-fold increase in the throughput of wireless ad hoc networks. Fig. 3, taken from [3], demonstrates the throughput gain of the COPE system for the 20-node testbed for UDP traffic with randomly picked source-destination pairs, Poisson arrivals, and heavy-tail size distribution.

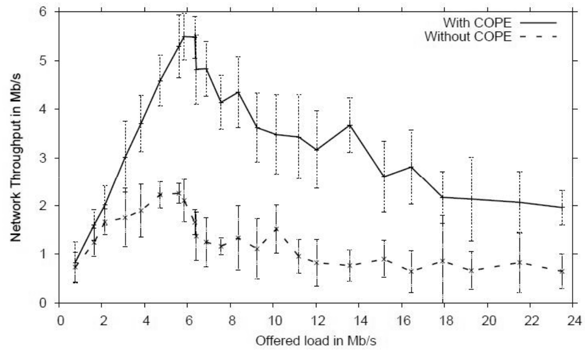

Fig. 3. COPE can provide a several-fold (3-4x) increase in the throughput of wireless ad hoc networks with UDP flows. This figure is taken from [3].

B. Existing analysis of COPE

The good performance of COPE with UDP traffic has attracted the attention of many researchers, and several attempts have been made to model COPE and explain the huge throughput gain.

Sengupta *et al.* formulated the throughput computation in a wireless network coding system into a linear programming (LP) problem [4]. Their formulation only considers the coding cases involving two packets, the scenario illustrated in Fig. 1. However, from the statistics in [3], we see that coded packets consisting of more than two native packets plays an important role in the throughput gain. In addition, the LP formulation does not capture the interaction between COPE and the MAC protocol very well. The LP problem enforces fairness among the overall flows, whereas in reality, fairness is enforced by the MAC protocol on a local scale. These differences lead to discrepancies between the experimental results and that predicted by the theoretical formulation.

Le *et al.* [5] tries to understand COPE by focusing on one *coding structure* at a time. A *coding structure* includes one coding node as well as the one-hop predecessor nodes and the one-hop successor nodes of the associated coding flows. The networks in Fig. 1 and Fig. 2 are both examples of single coding structures. The key performance measure they use is the encoding number, i.e., the number of packets that can be encoded by a coding node in each transmission. They upper bound the throughput gain in COPE by $2n/(n+1)$ for a general wireless network, where n is the maximum encoding number in one of its coding structures. Clearly, this upper bound is less than 2, which is much smaller than the throughput gain observed in Fig. 3. This is due to the fact that the analysis in [5] only deals with coding gain, but does not take into consideration the coding+MAC gain.

III. COPE PERFORMANCE ANALYSIS

The existing analysis of COPE fails to address two important aspects of the performance curves.

1) The magnitude of the throughput gain in COPE experiments is much larger than that predicted by the analysis;
2) As shown in Fig. 3, for both the COPE and the non-COPE systems, the throughput first increases linearly with the offered load. After reaching a *peak* point, the throughput decreases with increased load, and finally settles down to a *saturation* level. On the contrary, the performance curves derived from the existing theoretical formulations have a different shape. The throughput rises with increased load until it reaches a saturation level, and further increase in load does not affect the total throughput by much.

These discrepancies motivate us to take a closer look at the COPE system. We find out that the key to explain the COPE performance curves lies in the interaction between coding and the MAC protocol, and the local fairness enforced by the MAC protocol when it assigns bandwidth to competing nodes.

To understand this, we first look at the simple Alice-and-Bob network shown in Fig. 1. Here, we assume that the three nodes in the network share the wireless channel, and the total bandwidth is 1. Flows of size 0.01 are originated from node A/B, and are to be sent to node B/A, respectively. The relay node, R, does not generate any traffic. By increasing the number of flows, the total offered load to the network is increased. Also, the probability that a flow is generated by A or B is equal. We denote the bandwidth allocated to nodes A, B, and R as BW_a, BW_b, and BW_r, respectively. The wireless channel is lossless.

- **Routing (non-COPE) case:** When the offered load is very small, every node can get enough bandwidth to transfer what they have, and the total throughput grows linearly with the offered load. The bandwidth demand for the relay node is equal to the sum of the sending rates at A and B, and the total throughput of the system is always equal to BW_r. The throughput reaches its peak when the channel bandwidth is completely used up, i.e.,

$$BW_a = 0.25, BW_b = 0.25, BW_r = 0.5.$$

In this case, the total throughput of the system is 0.5. As the offered load increase beyond 0.5, the system will not be able to handle all the traffic. Queues at some of the nodes will grow, and packets are going to be dropped. Consider the *saturation* case, when the offered load is very large, all the nodes have backlogs. They are constantly competing for the channel. In this case, the MAC protocol will allocate the channel fairly among the three nodes, i.e.,

$$BW_a = BW_b = BW_r = 1/3.$$

Note that the total throughput of this system is always equal to BW_r, thus, the saturation throughput is 1/3.

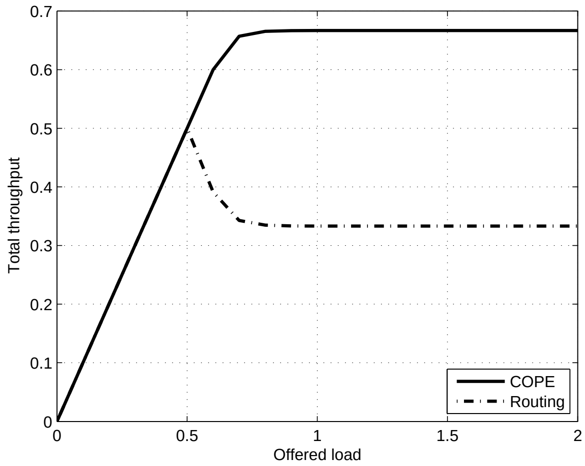

Fig. 4. Throughput for COPE and non-COPE systems in an Alice-and-Bob network with cross traffic only.

Now, we look at the transition stage where the offered load is between 1/2 and 2/3. For simplicity, assume symmetric load for nodes A and B, and we denote it by l. If the assigned bandwidth for A and B is less than that required to clear their queues, they would be constantly requesting for the channel. This situation is the same as that in the saturation stage, and they will be allocated 1/3 bandwidth each. However, since their demand is less than 1/3, they won't have a backlog in this case. This is a contradiction, therefore, nodes A and B will be assigned bandwidths l, and the bandwidth assigned to R is then equal to $1 - 2l$. The complete throughput curve is shown in Fig. 4.

- **Coding (COPE) case:** When coding is allowed at the relay node, if the loads at A and B are perfectly symmetric, every packet delivered by R generates a throughput of 2 packets. Similar to the routing case, when the offered load is small, every node gets its required bandwidth, and the throughput grows linearly with offered load until the total bandwidth is used up, i.e.,

$$BW_a = BW_b = BW_r = 1/3.$$

Since every transmission by R delivers two packets, the peak throughput of the COPE system is 2/3. When the offered load is increased further, packets starts to get dropped by A and B, however, the bandwidth allocation remains the same in the saturation stage, and the total throughput stays at 2/3. This throughput curve is also plotted in Fig. 4.

We next consider the cross network shown in Fig. 2. Assume symmetric traffic is generated at the four side nodes, 1, 2, 4, and 5, and is to be delivered to the opposite nodes. Also, when node 1 transmits, nodes 2 and 4 can overhear the transmission, and they store the overheard packets for future decoding. Same goes for the other three nodes. The curves in Fig. 5 show

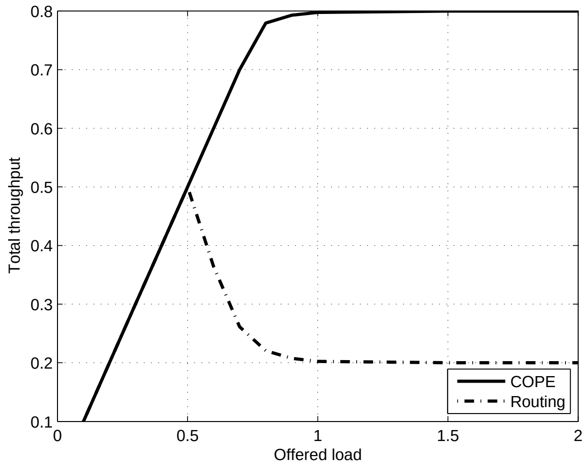

Fig. 5. Throughput for COPE and non-COPE systems in a cross network with cross traffic only.

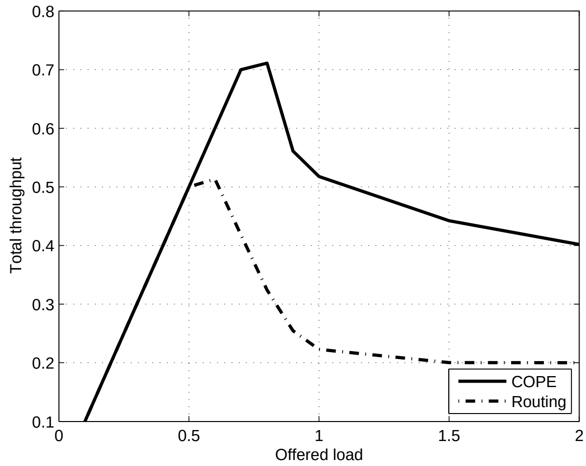

Fig. 6. Throughput for COPE and non-COPE systems in a cross network with cross traffic and traffic generated at the center node.

the throughput performance for this network with and without coding.

- **Routing (non-COPE) case:** Similar to the Alice-and-Bob network, the total throughput first increases linearly with the offered load until it reaches the peak where $BW_1 = BW_2 = BW_4 = BW_5 = 1/8$, $BW_3 = 1/2$, and total throughput is 1/2. The throughput then drops when offered load is increased further until it reaches the saturation stage where each node is allocated a bandwidth of 1/5, and the total throughput is also equal to 1/5.
- **Coding (COPE) case:** When the load at the four side nodes are perfectly symmetric, the relay node 3 can code four packets together every time it transmits. The throughput of this system peaks at 4/5 when the offered load at each side node is equal to 1/5. The throughput then remains at this level when offered load is further increased.

As we can see, when more flows are involved in the coding structure, the throughput gain becomes larger. This gain is not just due to coding, but also due to the fact that the MAC protocol allocates bandwidth among competing node fairly. The throughput of the system is limited by the bandwidth at the bottleneck node. With coding, the bottleneck node drains it queue multiple times faster than that in the non-coding case, thus resulting in the significant throughput gain. Even larger throughput gain can be obtained if the coding structure involves more than four flows, but they rarely happen in a practical system.

In the above simple models, we only considered cross traffic and all flows can be coded together. What happens when there exist unicast flows that cannot be coded with any other flow? To answer this question, we consider the cross network where in addition to the traffic generated by the side nodes, there are also flows generated by the center node, node 3, to be sent to one of the side nodes. The throughput performance of the COPE and non-COPE systems for this scenario is plotted in Fig. 6. As we can see, in the coded system, the total throughput drops after reaching the peak. This is because the traffic generated by the center node cannot be coded with any other packets, and the bandwidth used to send these 'unicast' packets are less efficiently used as compared to that used for sending the coded packets. As more and more flows are generated by node 3, these 'unicast' packets take up more and more bandwidth at the bottleneck node, reducing the total throughput.

The curves in Fig. 6 resembles that in Fig. 3 both in shape and in the magnitude of gain. As in the experiments, the largest gains are observed when the non-COPE curve has started dropping from the peak, and the COPE curve has yet to drop. Although our analysis only focuses on one coding structure, we believe the performance of COPE in a general network follows the same trend. This is because in a practical network, the throughput is limited by a few bottleneck nodes (coding structures). Therefore, by closely examining one coding structure, we can understand what really causes the COPE system behave in such a way. As mentioned previously, the key factor here is the interaction between COPE and the MAC and also the the local fairness enforced by the MAC protocol.

IV. Improvements on COPE

Our observation in the previous section leads to a simple improvement of COPE that can further increase the network throughput. Recall that in the case when there are both cross traffic and traffic originated from the center node (Fig. 6), the reason why the COPE curve drops is because the center node has to use some of its bandwidth to take care of the 'unicast' packets, which are less efficient in terms of throughput. To improve the total throughput, we would like to give higher priority to coded packets, as they help to drain the queue

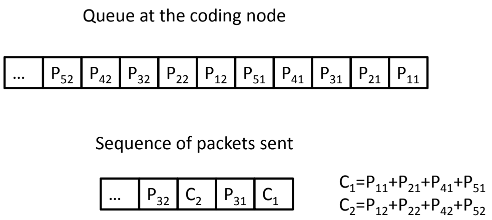

Fig. 7. An example of queue status and packets sent in a cross network with COPE.

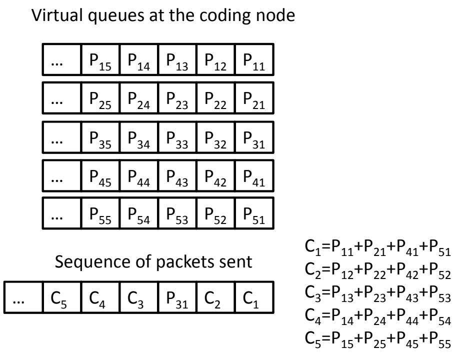

Fig. 8. An example of queue status and packets sent in a cross network with modified COPE.

at the bottleneck node at a faster rate. A simple way to do this is to have virtual queues for each input-output pair at the coding node, and packets are sent from these virtual queues in a round-robin manner.

In the current COPE system, only one queue is maintained at a node. Every time there is a transmission opportunity, the node dequeues the first packet, and checks if it can be coded with any other packets currently in its queue. If yes, the packets would be coded together and sent out; otherwise, the native packet will be sent alone. Fig. 7 illustrates the sequence of packets sent by the center node in a cross network with COPE. Here, P_{ij} denotes the j-th packet from node i. As we can see, in this case, 'coded' and 'uncoded' packets share the bandwidth equally, which is very inefficient and unfair, as the coded packets serve more users than the uncoded one.

If we keep separate virtual queues for each input-output pair and serve them in a round robin manner, what would happen in the cross network case is illustrated in Fig. 8. Here, the uncoded packets take up a much smaller fraction of the bandwidth, and the total throughput of the system improves.

We simulated the cross network with this simple modification, and the results are shown in Fig. 9. This modification leads to about 50% gain in the network throughput as compared to the original COPE system.

V. Conclusion

In this paper, we analyzed the performance of COPE with UDP traffic. We showed that the local fairness enforced by

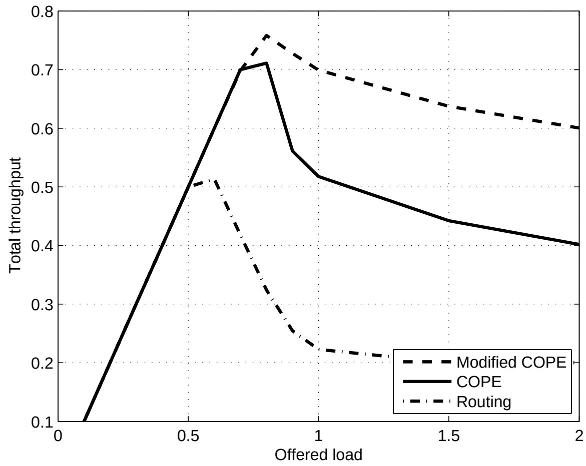

Fig. 9. Throughput for COPE, modified COPE, and non-COPE systems in a cross network with cross traffic and traffic generated at the center node.

the MAC protocol among competing nodes and the effect of COPE on this play an important role in COPE's performance. Our analysis successfully explains both the magnitude of gain in throughput with COPE and the shape of the performance curve. Furthermore, we proposed to use one virtual queue for each input-output pair in the coding node to give higher priority to coded packets. This modification to COPE is very simple, and it leads to significant gain in throughput when the offered load is high.

Acknowledgment

This material is based upon work under subcontracts #069145/060786 issued by BAE Systems National Security Solutions, Inc. and supported by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare System Center (SPAWARSYSCEN), San Diego under Contract No. N66001-08-C-2013/N66001-06-C-2020, subcontract #S0176938 issued by UC Santa Cruz supported by the United States Army under Award No. W911NF-05-1-0246, and supported by the National Science Foundation (NSF) under Grant No. CNS-0627021.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[2] S. Katti, D. Katabi, W. Hu, H. Rahul, and M. Médard, "The importance of being opportunistic: practical network coding for wireless environments," in *Proc. 43th Annual Allerton Conference on Communication, Control, and Computing*, Oct. 2005.

[3] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "Xors in the air: practical wireless network coding," in *Proc. ACM SIGCOMM '06*, 2006.

[4] S. Sengupta, S. Rayanchu, and S. Banerjee, "An analysis of wireless network coding for unicast sessions: the case for coding-aware routing," in *Proc. IEEE Infocom*, May 2007.

[5] J. Le, J. Lui, and D. M. Chiu, "How many packets can we encode? - an analysis of practical wireless network coding," in *Proc. IEEE Infocom*, April 2008.

[6] Q. Dong, J. Wu, W. Hu, and J. Crowcroft, "Practical network coding in wireless networks," in *Proc. the 13th Annual International Conference on Mobile Computing and Networking (MobiCom '07)*, Sept. 2007.
[7] S. Omiwade, R. Zheng, and C. Hua, "Practical localized network coding in wireless mesh networks," in *Proc. 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '08)*, June 2008.
[8] P. Chaporkar and A. Proutiere, "Adaptive network coding and scheduling for maximizing throughput in wireless networks," in *Proc. the 13th Annual International Conference on Mobile Computing and Networking (MobiCom '07)*, Sept. 2007.