# Enhanced Network Coding Scheme for Efficient Multicasting in Ad-Hoc Networks

MOHAMED OSAMA, AHMED SHAWISH
Faculty of Computer and Information Sciences
Ain shams University
Cairo, Egypt
mohamed.osama@fcis.asu.edu.eg, ahmed.gawish@fcis.asu.edu.eg

*Abstract:* - Increasing the throughput is an important objective for wireless ad-hoc networks. Many methods have been innovated for this purpose and on top of them is the network coding. The existing network coding schemes, such as COPE and its updated versions, have succeed to provide a remarkable throughput gain in case of unicast flows, while they failed to provide the same performance in case of the multicast scenario. With the notable flourish of conference-based and multimedia streaming applications that are mainly depending on multicast flows, it becomes crucial to find a method that is able to deal efficiently with both unicast and multicast flows. In this paper, we provide a novel enhanced network coding scheme, which we call Graph-Based Network Coding "GBNC" that is able to handle both unicast and multicast flows simultaneously with the same performance. The proposed scheme incorporates the graphic theory and the elimination technique to efficiently discover all possible coding opportunity and avoid the draw backs of the previous coding methods. The extensive simulation results reports the ability of the proposed scheme to achieve similar throughput gain to that of COPE in unicast flows and nearly double the gain in case of multicasting.

*Key-Words:* - Network Coding, Ad-Hoc networks, Multicast.

## 1 Introduction

Minimal configuration and quick deployment makes wireless networks (WLAN) suitable for last-mile Internet coverage. However, WLAN suffers from limited resources and bandwidth that attract many researchers. Numerous methods have been proposed to increase the throughput and to efficiently use the characteristics of such environment [1].

Network Coding (NC) proposed in [2], is introduced as the most promising and innovative technique to increase the WLAN throughput. COPE [3,4] which received warm reception from the research community and was considered the first practical scheme for NC demonstrated an efficient throughput gain in case of unicast traffic, while it didn’t succeeded to provide a similar gain in the multicast case. Many follow up work like [5]-[9] all trying to improve the throughput in wireless networks. However, most of them focused on unicast without a similar attention to multicast. Even the work done targeting multicast only like [22] and [23] didn’t report a considerable enhancement as it should be.

Recently, with the increasing demand on applications like all-informed voice, group push-to-talk, situational information sharing etc, supporting one-to-all and many-to-all (i.e., multicast) communication patterns in multi-hop wireless networks posed a problem that needed to be efficiently addressed. The need for an efficient scheme that is able to enhance the WLAN throughput in both multicast and unicast cases simultaneously became crucial.

In this paper, We propose a new enhanced network coding scheme, which we refer to as Graph-Based Network Coding "GBNC" that can handle both unicast and multicast flows simultaneously with the same throughput gain. The proposed scheme efficiently discovers all possible coding chances using a Graph theory and then the chance with the highest gain is selected. Thanks to the new graphical-based method, the proposed scheme succeeded to avoid the drawbacks of previous techniques and deals with both multicast and unicast flows with the same efficiency. Extensive simulation studies shows the ability of the proposed GBNC scheme to achieve similar high throughput gain as the COPE does in unicast flows, while it significantly outperforms the weak performance of the COPE in case of multicasting flow by achieving almost double of the COPE throughput gain.

The rest of this paper is organized as follows. Section 2, introduces the background on the available network coding schemes. In section 3, the proposed scheme is addressed in details. In section 4, the simulations results are presented and the achieved throughput gain of is discussed. Finally, the paper is concluded and future work is listed in Section 5.

## 2 Background

The pioneering work on network coding started with a paper by Ahlswede et al. [2], who demonstrated that having routers encode different messages allows the communication to achieve multicast capacity. It was soon followed by the work of Li et al., who showed that, for multicast traffic (e.g., the butterfly scenario), linear codes are sufficient to achieve the maximum capacity bounds [10]. Koetter and M´edard [11] presented polynomial time algorithms for encoding and decoding, and these results were extended to random codes by Ho et al. [12]. However, all this work was primarily theoretical and assumed multicast traffic only. COPE [2,3], which attracted a lot of research interest, proposed the first practical scheme for one-hop NC across unicast sessions in wireless mesh networks [2]. Following papers tried to model and analyze COPE [13], [14], [15].

Others proposed new coded wireless systems, based on the idea of COPE [16], [17]. In [18], the performance of COPE is improved by investigating its interaction with MAC fairness. Optimal scheduling and routing for COPE are considered in [13] and [15], respectively. $I^2$NC [19] built upon such work but did not handle multicast flow and focused on loss rate only. Use of network coding along with cooperative communication was found to provide throughput gain for TCP flow as in [20] but multicast flow was not considered and work was primarily for TCP and forced more complexity to incorporate cooperative communication.

MORE [21] is the first intra-flow NC-based protocol for reliable unicast and multicast over WMNs, in which nodes that overhear the transmission and are closer to the destination may participate in network coding and forwarding of the coded packets, forming forwarding belts toward the destinations. However belt forwarding can be inefficient, especially for multicast in which multiple overlapped belts are formed and many nodes intend to forward. Pacifier [22] improved upon MORE by using a multicast tree instead of multiple belts. Only nodes on the multicast tree are allowed to perform random NC. It is reported in [22] that Pacifier performs better than MORE for reliable multicast in WMNs. Both MORE and Pacifier relied of acknowledgments from the set of receivers and applied classic NC that is not suited for multicast flows. HoPCaster [23] outperformed Pacifier by integrating network coding and receiver-driven hop-to-hop transport to achieve high-throughput reliable multicast, yet it did not modify the coding scheme and did not handle unicast.

The experimental evaluations reported in both [24] and [25] show that RLNC provides substantial coding gains/performance improvements in a real network. All these protocols are proposed for single-source scenarios. A source groups a set of consecutive packets into blocks called generations or batches (we will use generations as the common term). Coding operations are confined to packets belonging to the same generation. In case a simulation scenario includes multiple sources, the basic idea is then simply replicated – each source creates, independently, generations of coded packets. To the best of our knowledge, only few papers [26][27][28] explicitly addressed multi-source wireless broadcast. None of these investigated in depth whether cross-source coding (i.e., combining packets from potentially different sources) provides performance improvements, compared to repeating the single source solution multiple times, once per source. In contrast to all the previous work, this paper proposes a new novel network coding scheme that handles unicast flow as efficient as COPE and outperforms it in case of multicast.

## 3 The Proposed network coding scheme

To explain the proposed network coding scheme, let’s consider the same case study addressed in the main Network coding scheme COPE [1], illustrated

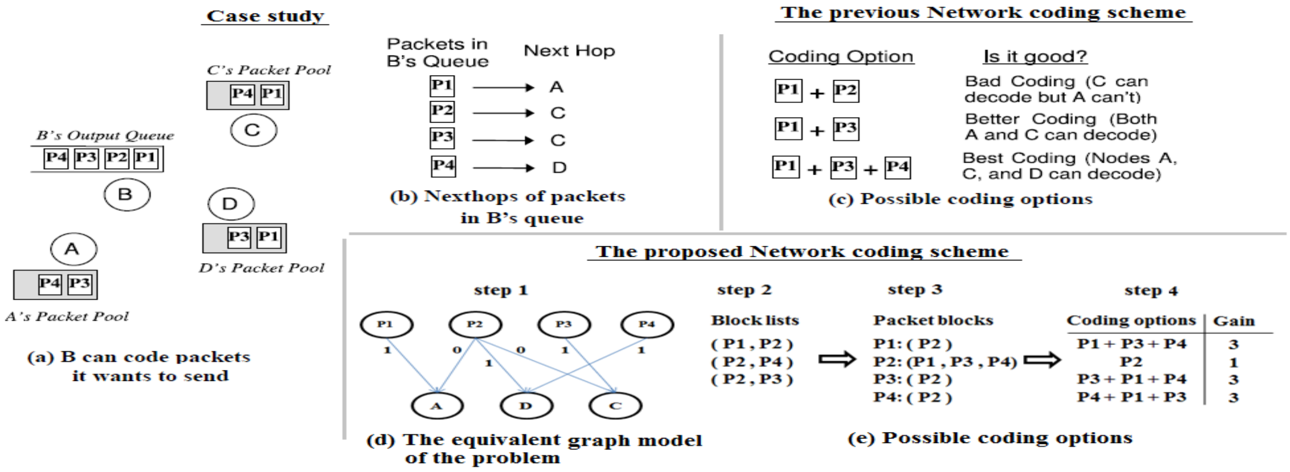

Fig. 1, Coding opportunity example under both previous and proposed Network Coding schemes

in fig. 1 part (a, b, c). In such case, the source node *B* having a set of *N* packets (e.g., 4 packets: *P*1, *P*2, *P*3, and *P*4) and a set of neighbour nodes randomly scattered around *B* (e.g., nodes: *A*, *D*, and *C*) that may be the next hope of the buffered packets as shown in fig. X part (a). The list of next hops of packets in *B*'s queue is illustrated also in the same figure part (b).

**Step 1: graphical model**

First the relation between the packets and the next hop network nodes is graphically modeled as illustrated in fig. X part (d). In the resulted graph includes both packets & network nodes represented as graph nodes, in addition to directed weighted edges based of the following rules:

1) Each packet $P_i$ in the output queue of the source node *B* is represented as a graph a node if and only if it is a new packet to at least one of the neighbour hop. Accordingly, all the packets (i.e., *P*1, *P*2, *P*3, and *P*4) are graphically represented as each of them is new for at least one of the neighbour hop.
2) Each neighbour network node is represented as graph node if and only if there exists at least one packet in the source node's output queue that is new to it. Accordingly, all neighbour nodes (i.e., *A*, *D*, and *C*) are represented as they confine this rule.
3) A directed weighted edge is drawn from each packet to the network nodes is such packet is new to such node. This is why there is no edge between P1 and both nodes C & D as both already have P1 in their buffers. The weight assigned to this edge is 1 if the packet $P_i$ (where i=1,2,..4) needs to be routed to such node as either a destination or a relay hop, otherwise the weight of this edge is 0. Accordingly, the weight of the edge between P1 and node A is 1 as it is next hop of such packet, while the weight of the edge between P2 and node A is 0 as it is not the next hop of such packet.

**Step 2: Block list**

Toward the objective of finding feasible coding options, a block list should be constructed. Such list identifies the group of packets that can not be coded together as they can be both new for the next hop. To creat such list, each neighbor node is addresed apart if the count of edges reaching such node is greater than one. Accordingly, nodes *A*, *C*, and *D* will be addressed apart. The grouping is then created by combining the sources of edges (i.e., the packets) incident to such node according to the graph created in step 1. For example, *P*1 and *P*2 are grouped in the block list as they are the source of edges directed to node *A*, while *P2* and *P*4 are grouped in the block list as they are the source of edges directed to node *D*, as illustrated in fig. 1.

A reduction phase is then conducted over the final block lists. this reduction process iterates upon the block lists with the goal of removing groups that may be completely contained in another block list. For example if the block list contain both (P1,P2) & (P1,P2,P4) the frist group (P1,P2) will be removed from the final list as both packets are contained in (P1,P2,P4).

**Graph Nodes and edges construction procedure**

```
GraphNodes = {}
for Packet $i$=1 to $M$ do
    Pick packet $p_i$
    for Neighbor $j$=1 to $N$ do
        Pick neighbor $n_j$
        if $p_i \notin$ packet pool of $n_j$ then
            if $p_i$ routed to $n_j$ then
                Add edge from $p_i$ to $n_j$ with gain 1
            else
                Add edge from $p_i$ to $n_j$ with gain 0
            end if
            if $n_j \in$ GraphNodes then
                Add edge between $p_i$ and $n_j$
            else
                GraphNodes = GraphNodes $\cup$ {$n_j$}
                Add edge between $p_i$ and $n_j$
            end if
        end if
    end for
    if $\exists j$: $p_i \notin$ packet pool of $n_j$ then
        GraphNodes = GraphNodes $\cup$ {$p_i$}
    end if
end for
```

**Step 3: Packet Blocks**

This step aims at creating per packet block list named as packets blocks. Here, each packet is carefully addressed with the help of the block list to identify the list of other packets that can not be coded with it. For example, $P2$ can not be coded with $P1$, $P3$, and $P4$ as the block list includes ($P1$,$P2$), ($P2$,$P3$) and ($P2$,$P4$). Hence the packet blocks report this fact as P2: (P1, P3, P4).

**BlockingList construction procedure**

```
BlockingList = $\Phi$
for NeighborGraphNode $i$=1 to $N$ do
    Pick neighbor $n_i$
    Blocking = $\Phi$
    if $n_i$ count of edges >1 then
        for Edge $j$=1 to $M$ do
            Pick edge $e_j$
            Pick packetnode $p$ source of $e_j$
            Blocking = Blocking $\cup$ $p$
        end for
        BlockingList = BlockingList $\cup$ Blocking
    end if
end for
```

**Step 4: Coding option & gain**

In this step, each packet is carefully addressed a part as a candidate for selection to be the first packet in the coding option, the packet is picked if and only if there does not exist a packet in its packet blocking that has higher gain than it. if the packet can not be selected the algorithm simply considers another packet, if it can be selected the remaining packets are examined for selection as long as they are not blocked according to the block list and they do not block a packet with higher gain

the gain of each computed coding option is calculated as the total gains of the packets selected in such coding option. The best coding option is selected based on the highest gain supplied

Fig. 1 shows a graph representation based on this model.

**Codingways computing procedure**

```
codingways = $\Phi$
for PacketNodes $i$=1 to $M$ do
    codingway = $\Phi$
    currentblocked = $\Phi$
    Pick packetnode $p_i$
    if $p_i$ has highest gain in its blocklist then
        codingway = codingway $\cup$ $p_i$
        currentblocked = currentblocked $\cup$ packets in
         $p_i$ blocklist
        for PacketNodes $j$=1 to $N$ do
            if $j \neq i$ & $p_j \notin$ currentblocked & $\nexists$ packet $p$
            in $p_j$ blocklist: ($p \notin$ currentblocked & gain of
             $p$ > gain of $p_j$) then
                codingway = codingway $\cup$ $p_j$
                currentblocked = currentblocked $\cup$
                packets in $p_j$ blocklist
            end if
        end for
        codingways = codingways $\cup$ codinway
    end if
end for
```

## 4 Results and Discussion

In this section, we demonstrate our enhanced coding scheme and compare the results with COPE scheme.

The topology used for simulation consists of 17 randomly placed static Ad-hoc nodes with randomly picked source-destinations pairs, uniform arrival and normal distribution for packet arrival. Wireless medium channel transmission rates of 6, 8,..., 22 was used. UDP packet size was set to 80 bytes conforming to the G711 voice codec. We verified our implementation by simulating unicast flows using both no-network-coding and COPE. As depicted in fig 2 the results are similar to those obtained in [2, 3]

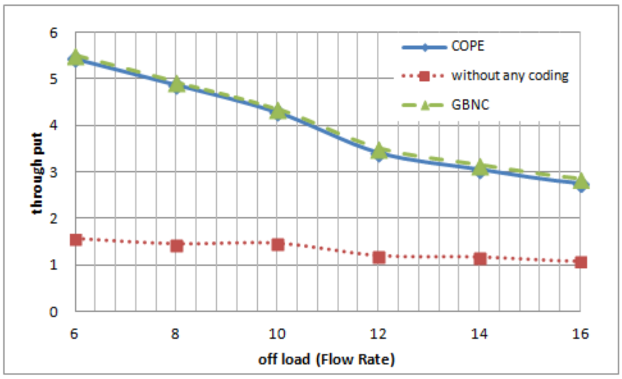

**Fig. 2 Throughput gain against flow rate for unicast case**

Fig. 2 shows that the throughput gain obtained by our scheme is almost identical to the gain obtained by COPE in case of unicast flow. The figure plots the aggregate end-to-end throughput as a function of the demands, with GBNC, with COPE and without any scheme. Without any coding applied, throughput starts to deteriorate as the demands increase because of the effect of higher contention levels and consequent loss of packets induced by collisions. Applying coding reduces the number of packet transmissions resulting in higher level of throughput.

As depicted in fig. 3, simulating multicast flow along with unicast flow shows that the scheme introduced by COPE yields almost the same throughput as with no coding since multicasted packets will block the selection of other packets. The proposed scheme manages to provide nearly similar throughput gain as in the unicast flow case. This is due to selecting the best coding way that results in delivering the maximum number of new packets to their intended hops based on the gain of each coding as illustrated in section 3.

The proposed multicast GBNC scheme achieves nearly double the throughput gain compared to COPE. Due to marking the edges in our modeled graph by 0 or 1 according to whether the packet needs to go to a specific node or not, GBNC efficiently comes up with the best coding selection of packets to deliver the maximum number of new packets in every transmission where COPE scheme fails as new packets blocks all other packets if selected and no better selection is considered.

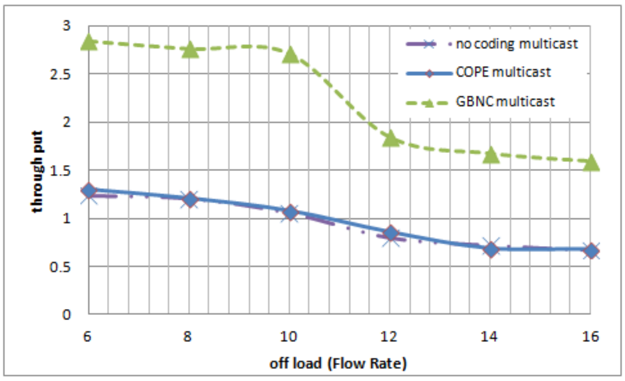

**Fig. 3 Throughput gain against flow rate for multicast case**

## 5 Conclusion & Future Work

This paper addressed the problem of selecting best coding option of packets. A new scheme has been introduced to efficiently handle both cases of unicast and multicast flows simultaneously. The conducted simulation study reported the ability of the proposed scheme to achieve the same UDP throughput gain as COPE in case of unicast flow, while achieving double the throughput gain in case of multicast. These results are considered a significant enhancement of the network coding schemes and hence the network's throughput improvement in general. It is also an important step toward a scheme that is independent of the flow type.

*References:*

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, no. 4, pp. 445 – 487, March 2005.

[2] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, pp. 1204–1216, Jul. 2000.

[3] 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

[4] S. Katti, H. Rahul, W. Hu, D. Katabi, M.Medard, and J.Crowcroft, "Xors in the air: Practical wireless network coding*," IEEE/ACM Transactions on Networking*, vol. 16, pp. 497–510, Jun. 2008.

[5] P. Chaporkar and A. Proutiere, "Adaptive Network Coding and Scheduling for Maximizing Throughput in Wireless Networks," in *ACM MOBICOM*, 2007.

[6] S. Rayanchu, S. Sen, J. Wu, S. Banerjee, and S. Sengupta, "Loss-Aware Network Coding for Unicast Wireless Sessions: Design, Implementation, and Performance Evaluation," in *ACM SIGMETRICS*, 2008.

[7] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading Structure for Randomness in Wireless Opportunistic Routing," in *ACM SIGCOMM*, 2007.

[8] B. Scheuermann, W. Hu, and J. Crowcroft, "Near-Optimal Co-ordinated Coding in Wireless Multihop Networks ," in *ACM CONEXT*, 2007.

[9] Georgios S. Paschos, Constantinos Fragiadakis, Leonidas Georgiadis and Leandros Tassiulas, "Wireless network coding with partial overhearing information, " in *IEEE INFOCOM*, 2013.

[10] S. R. Li, R. W. Yeung, and N. Cai. Linear network coding. In *IEEE Transactions on Information Theory*, 2003.

[11] R. Koetter and M. M´edard. An algebraic approach to network coding *IEEE/ACM Transactions on Networking, 2003*.

[12] T. Ho, R. Koetter, M. M´edard, D. Karger, and M. Effros. The Benefits of Coding over Routing in a Randomized Setting. In *ISIT, 2003*.

[13] P. Chaporkar and A. Proutiere, "Adaptive network coding and scheduling for maximizing througput in wireless networks," *in Proc. of ACM Mobicom*, Montreal, Canada, Sep. 2007.

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

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

[16] Q. Dong, J. Wu, W. Hu, and J. Crowcroft, "Practical network coding in wireless networks," *in Proc. of MobiCom*, Montreal, Canada, Sept. 2007.

[17] S. Omiwade, R. Zheng, and C. Hua. "Butterflies in the mesh: lightweight localized wireless network coding," *in Proc. of NetCod*, Lausanne, Switzerland, Jan. 2008.

[18] F. Zhao and M. Medard, "On analyzing and improving COPE performance," *in Proc. of ITA*, San Diego, CA, Feb. 2010.

[19] Hulya Seferoglu, Athina Markopoulou, K. K. Ramakrishnan, Intra- and Inter-Session Network Coding in Wireless Networks, *Infocom* 2011, pp 1035-1043

[20] Yifei Wei, Mei Song and F. Richard Yu, TCP Performance Improvement in Wireless Networks with Cooperative Communications and Network Coding, *ICC* 2012

[21] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing", *ACM SIGCOMM* 2007.

[22] D. Koutsonikolas, Y. Liu, C. Wang, "Pacifier: high-through, reliable multicast without crying babies in wireless mesh networks*," INFOCOM* 2009.

[23] Rami Halloush, Hang Liu, Lijun Dong, Mingquan Wu and Hayder Radha, HopCaster: A Network Coding-Based Hop-by-Hop Reliable Multicast Protocol, *Globecom* 2012

[24] G. Lauer, and D.S. Morris, "Network coding performance: An emulation experiment," 2008 *Military Comm. Conference,*pp.1-7, Nov. 2008.

[25] V.Firoiu et al., "Experiences with Network Coding within MANET Field Experiments," *2010 Military Comm. Conference*, pp. 1239-1244, Nov. 2010.

[26] C. Fragouli, J. Widmer, and J.-Y. L. Boudec, "A network coding approach to energy efficient broadcasting: from theory to practice", *IEEE Conference on Computer Communications*, pages 1-11, Apr. 2006

[27] C. Fragouli, J. Widmer, and J. Le Boudec. 2008. "Efficient broadcasting using network coding", *IEEE/ACM Trans. Netw.*, Apr. 2008, pp. 450-463.

[28] Y. Kondo, H. Yomo, S. Yamaguchi, P. Davis, R. Miura, S. Obana, "Reliable Wireless Broadcast with Random Network Coding for Real-Time Applications", *Conference on Wireless Communications and Networking*, pp. 1-6, 2009