Network Coding in IEEE 802.11 Wireless LANs with an Enhanced Channel Access Scheme

Antonios Argyriou
Philips Research
Eindhoven, The Netherlands
Email: antonios.argyriou@philips.com

***Abstract*—In this paper, we present a set of simple extensions to the IEEE 802.11 MAC that allow the efficient realization of wireless network coding. The main characteristic of the proposed protocol is that it improves the efficiency of packet coding decisions and allows verifying the decodability of packets before they are transmitted over the network. The previous goals are achieved by adopting two techniques. First, we introduce the concept of opportunistic acknowledgments (OACKs), and second, we adopt the use of virtual buffers that are used for selectively storing overheard packets by a particular node. We performed extensive simulations for several ad hoc network topologies that test the proposed protocol. Our results indicate that significant throughput enhancements can be achieved when compared with network coding schemes that are realized with the legacy 802.11 MAC.**

***Index Terms*—Network coding, wireless ad hoc networks, MAC protocol, distributed channel access, IEEE 802.11.**

I. Introduction

The last few years, the concept of network coding (NC) is being promoted as an extension to the classical routing principle in packet networks [1]. NC suggests that a network should not only route packets between hosts but also intelligently process/code packets to maximize network throughput and achieve multicast capacity. The most important benefits that can be obtained by employing network coding include higher throughput and reliability [2]. In the case of wireless networks, the basic idea is to combine network coding with wireless broadcast to increase the information content per transmission. Fig. 1 demonstrates the basic idea of wireless network coding. In this simple topology, nodes N_4 and N_5 can overhear the transmissions of packets b and a respectively. This means that if N_2 can broadcast the XOR-coded packet $a \oplus b$, this information is enough for decoding both packets a and b at N_4 and N_5 respectively. Without network coding and broadcasting, four transmissions would be required instead of three. This idea has been applied in 802.11-based multi-hop wireless networks and significant throughput improvements have been demonstrated [3], [4].

However, one topic that has not been studied thoroughly in the literature, is the interaction between opportunistic wireless network coding and MAC protocols. Most of the existing studies are theoretical and make several assumptions either about the structure of the network or the channel access scheme [5], [6], [7]. Even though of great value, these works do not shed light into practical aspects that

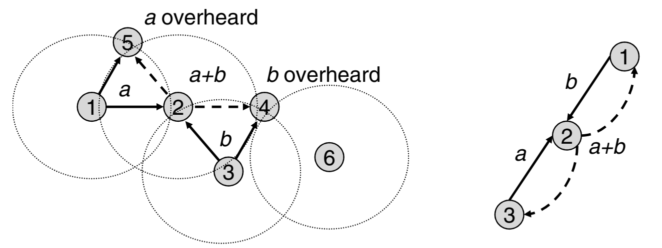

Fig. 1. Examples of wireless network coding with opportunistic packet overhearing (left side), and without opportunistic overhearing (right side). Dashed lines indicate the single transmission of a coded packet.

are related with the design of distributed MAC protocols. More recently, Chaporkar and Proutiere investigated adaptive network coding and scheduling for wireless ad-hoc networks while they excluded opportunistic listening [8]. Simulation studies have also demonstrated the sub-optimal performance of the IEEE 802.11 MAC protocol and NC [9]. Recently, Le *et. al.* claimed that no particular link scheduling mechanism is needed for ensuring high performance of NC over 802.11 but instead proper buffer sizing is enough [10]. No formal proof is provided though while limited topologies and traffic patterns are considered. The most closely related work with this paper is the COPE system that is the first practical implementation of the wireless network coding principle [4]. Although several limitations of existing MAC protocols for NC have been identified, none of the aforementioned works introduced any mechanisms at the MAC layer that address these problems and result in substantial performance improvements of NC. In this paper, we design a distributed MAC protocol that is based on extensions to the IEEE 802.11 MAC and adopts mechanisms that target the efficient realization of network coding.

II. Opportunistic Listening

The broadcast property of the wireless channel is the basic characteristic of wireless ad hoc networks that is exploited by network coding. However, opportunistically overhearing packet transmissions does not directly lead to improved performance of the NC algorithm. Let us revisit Fig. 1 for explaining the problem. Assume N_1 transmits a unicast packet a that is intended for node N_2, but it is also overheard by N_5 who is within the radio range of both N_1 and N_2. With existing MAC protocols nodes that are not the intended receivers (in this

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

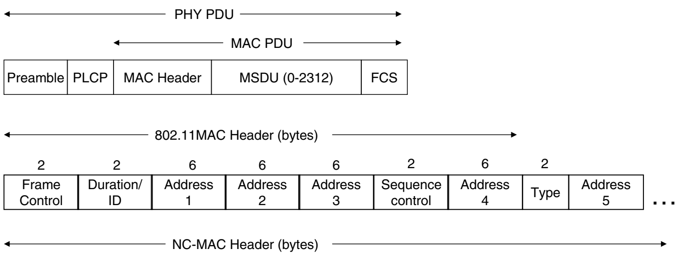

Fig. 2. Proposed header structure that extends the 802.11 header.

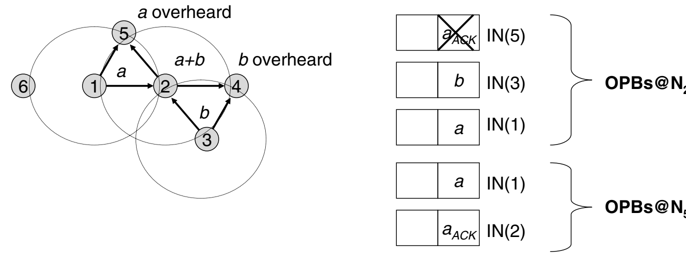

Fig. 3. Nodes use the last successfully received packet as an indication of each receiver's OPB state.

figure N_5) cannot acknowledge correct reception of overheard packets. This means that the node that employs network coding (in this figure N_2), should either not code packets for which it is not certain that they were overheard, or it can code despite the absence of this knowledge. However, the node cannot be certain if a coding decision is correct unless it has a precise knowledge of the packets stored at each node. In existing works, the problem is addressed with the use of asynchronous cumulative out-of-band acknowledgments [4]. The disadvantages of this approach are first the stochastic estimation of the fate of already transmitted packets, increased packet delay, and the need for separate medium contention rounds for transmission of out-of-band acknowledgments.

III. The Proposed Protocol

In this section we present the network-coding aware MAC (NC-MAC) that is based on minor modifications to the IEEE 802.11 MAC. The most important feature of our protocol is that it allows the packet coding algorithm at each node to determine the optimality of coding decisions depending on the knowledge of "who has what". Our protocol improves this knowledge by introducing a lightweight extension to the channel access scheme.

A. Packet Types and Header Structure

With the proposed protocol a node can transmit three types of data packets: 1) a regular unicast uncoded data packet (D), 2) a unicast uncoded data packet targeted to multiple opportunistic receivers (DO), and 3) a coded packet targeted to multiple innovative receivers (DC). Furthermore, a new type of acknowledgment packets is introduced but its purpose is explained in the next section. For these three types of data packets the extended MAC headers are created by adding additional fields for destination addresses and a field that indicates the packet type (see Fig. 2).

B. Overhearing Algorithm

Contrary to existing schemes, we are making use of an additional form of acknowledgments for opportunistically overheard packets. This is a mechanism of receiving in-band notifications that ensures correct coding decisions while it renders unnecessary the explicit cross-layer communication between higher protocol layers and the MAC protocol. We call this type of acknowledgements, *opportunistic acknowledgments (OACKs)* to distinguish from the ACKs that are required for the reliable data delivery. OACKs do not ensure full reliability since no retransmission timer is associated with opportunistically overheard packets. Nevertheless, OACKs offer a fairly simple and powerful technique that allows a node to have a better indication regarding the status of the overheard packets at each neighboring/receiving node. Since OACKs are control packets, they are transmitted at the lowest possible PHY rate to ensure high probability of successful delivery. In conjunction with OACKs, multiple virtual *opportunistic packet buffers (OPBs)* are maintained for each transmitting neighbor of a particular node. The OPBs are mainly responsible for tracking "who sent what" and "who acknowledged what". Therefore, they store all the transmissions of either data or ACK packets from each neighbor. The existence of OPBs does not necessary correspond to increased storage requirements. For example for our simple example in Fig. 1, N_1 should require from N_5 to OACK packet a since it has detected transmissions from N_2 to N_5. This OACK will provide N_2 the opportunity to use packet a for coding. Furthermore, N_5 will OACK packet a but N_2 will send a regular ACK since it is the target destination of a.

To determine the group of nodes that should OACK the transmission of a data packet from a specific node, we introduce as simple algorithm that we describe below. Each node checks the following condition based on the packets that it overhears:

Rule 1: A node N_x should request from another node N_y to OACK a packet l currently considered for transmission, iff within the last K transmissions detected from $l \rightarrow nexthop$ at least one was destined to node N_y. The sender makes a transition to $OACK\_requested$ state.

If within the last K transmitted packets from $l \rightarrow nexthop$ one was transmitted to N_y then N_x uses this information as a *soft indication* that there is potentially a packet flow from node $l \rightarrow nexthop$ to node N_y that could be coded. However, we want to avoid that a node receives and stores opportunistically overheard packets that are not actually used for coding. Since *Rule 1* does not actually ensure that there is a coding opportunity and the request of an OACK was the appropriate choice, we define a second rule that can actually do that:

Rule 2: If a node is in the $OACK\_requested$ state and within K consecutive transmissions detected from $l \rightarrow$

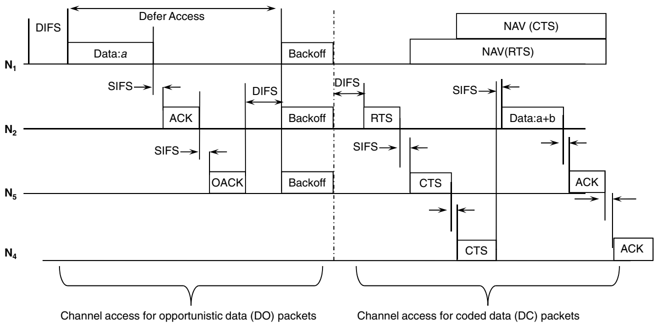

Fig. 4. Proposed channel access scheme with transmission of OACKs and ACKs. The SIFS interval between the successive acknowledgments is used for protecting from other nodes' contention.

nexthop one is a coded packet destined to node N_y, then subsequent OACKs can be sent by following Rule 1. The sender makes a transition to $OACK\_validated$ state

These rules consist together an algorithm that is important for ensuring dynamic adaptation due to changing channel or packet flow conditions. Furthermore, rules 1 and 2 ensure that at most one opportunistically overheard packet is unnecessarily stored. For understanding better the state of a node, we present in Fig. 3 and next to the simple toy topology, the OPBs of nodes N_2 and N_5. This figure basically demonstrates the loss of an OACK. In this case N_2 estimates that N_5 did not receive packet a. However, N_5 actually has received the packet while it has also overheard the ACK sent by N_2. To overcome problems of inconsistent buffers, and therefore incorrect coding decisions, nodes simply "flush" their OPBs when their is a lack of ACKs. In this way the protocol ensures that it has an up-to-date picture of the OPBs of its neighbors before any coding decisions are made.

C. Channel Access

At this point we have introduced the concept of OACKs and we explained the algorithm for manipulating the multiple virtual OPBs that are mapped to a single incoming MAC queue. Now the transmission of DC and DO-type packets differs with respect to the channel access mechanism in a way we describe next.

Fig. 4 demonstrates the proposed channel access scheme when nodes transmit OACKs and coded packets. In the left part of this figure, N_2 acknowledges a packet since it is the next hop of a transmission from N_1. According to our network topology, this also means that N_5 can overhear transmissions both from N_1 and N_2. According to *Rule 1*, N_1 must have requested an OACK from N_5 which is sent after SIFS. After N_1 receives an OACK from N_5 it infers that N_5 can hear also N_2's transmissions. N_1 waits for DIFS so that it is certain that no other nodes have ACKed a, and then it proceeds with resetting the backoff counter normally. Note that the duration of a successful or not transmission of a DO-type packet is not constant since the number of opportunistic receivers is also not constant.

The proposed channel access scheme is using the RTS/CTS mechanism only for transmitting coded packets. A transmitting node sends a single RTS with multiple nexthop addresses in the RTS header (i.e., to the nodes that can decode an innovative packet). CTS packets are sent after SIFS with the order being determined by the order of the receivers addresses as they appear in the RTS packet (Fig. 4). After a CTS is received from the nodes that can actually decode an innovative packet, the transmitter sends with a single transmission the coded packet. The received packet is decoded and the appropriate OPBs are updated. Finally an ACK is sent after SIFS similar to the CTS control packets. Note that whether RTS or data packets are re-transmitted to those neighbors from which CTS or ACK was not received, depends on whether reliable multicast is employed or not. For our protocol we decided to avoid this option since we do not want to affect fundamentally the basic channel access scheme in 802.11 that is the distributed coordination function (DCF). In case a number of the receivers do not ACK a coded packet, the coding algorithm at the sender is responsible for identifying new coding opportunities. Note however that when the above event happens, a retransmission can be useful for more than one receivers.

D. Network Coding Algorithm

The MAC SDUs that are passed from the upper layers can be coded by any network coding algorithm since it is orthogonal to the previously described protocol. In our work we adopt a simple XOR-based coding scheme. The main packet coding procedure is executed for the packet l_1 that resides at the head of the outgoing MAC FIFO queue. Based on our previous discussion, recall that the OPBs maintain the most updated information regarding the nodes that have received particular packets. The decodability of different options is examined next. The first packet l_1 in the outgoing MAC queue is coded only if it has been overheard or transmitted by a neighboring node. Then the node exhaustively checks which node in the neighborhood had received l_1 and the candidate packet for coding l_2, and can therefore decode a coded version of these two packets. For these nodes, a transmission contributes to an innovative packet as we described previously. When a valid coding combination is found, the packet l_1 is coded while the destination nodes that must receive it are updated accordingly. This procedure is repeated until the optimal number of packets that can be coded is identified. It is possible that the algorithm may not code a packet, and transmit it either as a D packet, or if there are opportunistic receivers as a DO packet. Finally, recall that the decoding rule is fairly simple and requires that encoded packets are always decoded at the next hop.

IV. Performance Evaluation

We have built our own simulation tool for the proposed protocol and the IEEE 802.11a MAC protocol. We also implemented a simulation model for a coding scheme that adopts the main features of COPE [4] and includes all the basic functionality like pseudo-broadcast, separate receiver reports for coded packets layered on top of the MAC (and they are

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

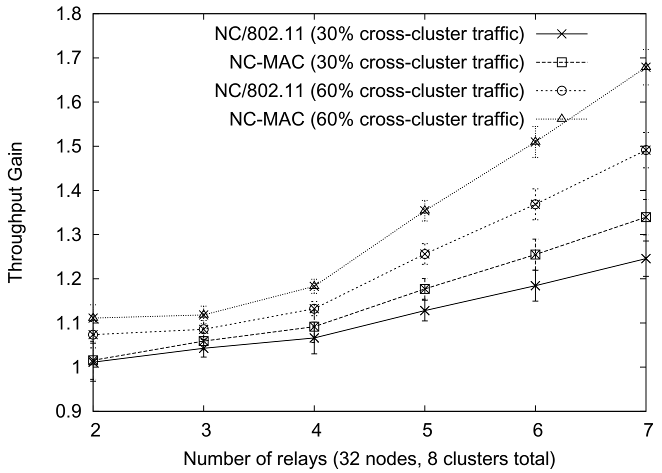

Fig. 5. Simulations with the percentage of multihop flows vs. intra-cluster flows being 30% and 60% respectively.

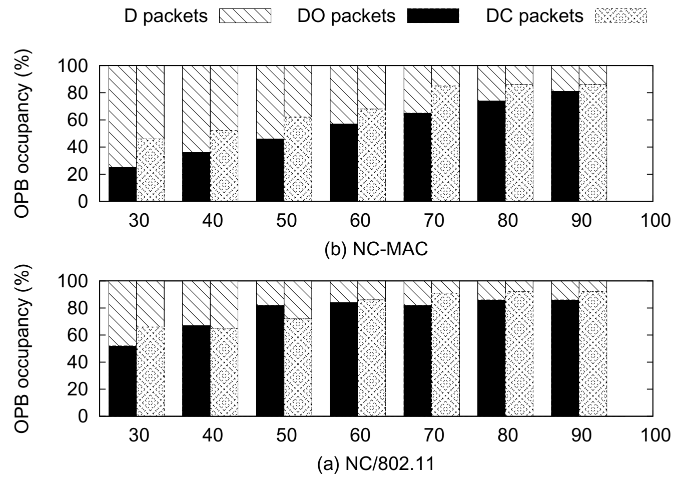

Fig. 6. Average system-wide MAC queue occupancy vs. cross-cluster traffic load for the two protocols under test and the three types of data packets. The two bars correspond to regular and relay nodes respectively. The NC/802.11 protocol also stores coded packets at regular nodes contrary to NC-MAC.

sent using separate data transmissions for which they have to contend for the channel). We call this protocol NC/802.11 to indicate that NC is implemented independently of the 802.11 MAC protocol. We experimented with the number of CBR flows and the number of hops they traverse. Also, since nodes are assumed to have backlogged traffic, they always have a packet to transmit either for their own cluster or for a cluster several hops away. The channel access timing parameters are similar with 802.11 ($SIFS$=16μsec, $DIFS$=34μsec). For all our simulations we assumed contention-related packet losses. Flows between two nodes in different clusters are generated randomly so that relay nodes are loaded asymmetrically. Shortest path routing is used for determining the end-to-end path of each flow.

A. Number of Relay Nodes between Clusters

We start with 32 nodes allocated to 8 clusters and we gradually increase the number of relays from 2 to the maximum possible of 7. Fig. 5 shows that the performance increase achieved by NC-MAC becomes more significant as the number of cross-cluster flows is increased. As we will later see, this situation can happen until a certain point that depends primarily on the size of the incoming MAC queue. Nevertheless, for a higher number of relay nodes that interconnect the clusters the importance of our protocol is even more significant as the results in Fig. 5 indicate. This is a result we expect since the cross-cluster offered load can be distributed to more relays which correspond to an increased coding gain. The proposed NC-MAC can capitalize even more on the introduction of more relay nodes since the coding opportunities are increased. The reason behind the increase in the number of coding opportunities will become clear in the next set of simulations.

Another important observation from the same set of results is that the proposed protocol is less sensitive to the number of nodes that contend for the medium within each cluster due to the low overhead of the OACKs. Even with light inter-cluster load, and many opportunistic receivers, no unnecessary buffering of overheard packets takes place. At the same time, for NC/802.11 even for light inter-cluster load, a node still suffers from contention-related losses for the out-of-band acknowledgments sent for coded packets. Furthermore, more buffer space is occupied at the opportunistic receivers. Fig. 6 presents results for the same experiment and it indicates how the available buffer space is allocated between the three types of data packets both at regular nodes and the relays. The results are very encouraging since they show that the previous performance is attained with minimum number of DO packets. A subtle point that is not shown is that even though for high inter-cluster load the percentage of DO packets in the buffer is nearly the same for both protocols, NC-MAC stores only packets that are actually needed for coding. The same holds for coded packets at the relays where several of them are unnecessarily coded by NC/802.11.

B. Multihop Flows

The effect of flows spanning multiple hops is examined next and Fig. 7 presents the related simulation results. These results correspond to a chain topology of 32 nodes allocated to 8 clusters. In this case the offered load at each relay is increased because the relays have to forward traffic from their neighboring relays. Therefore, the percentage of traffic that the relays must forward is higher while the locally generated load is decreased. At the same time, the coding opportunities are also increased. Nevertheless, there is a critical point that is different for each protocol and depends on the number of relays and the forwarding offered load. When the forwarding load is increased more than a certain value, the increased contention and buffer overflows at the relays disallows them from forwarding any further coded packets. In our simulations we observed that increasing the buffer size could only solve the problem when the number of hops is low while at the same

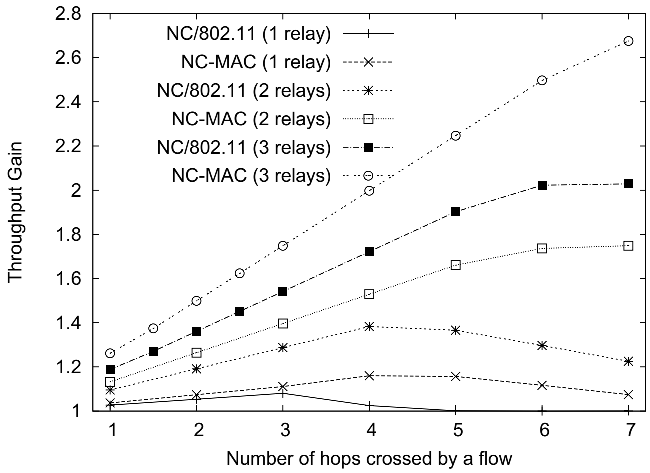

Fig. 7. Throughput gain vs. number of hops that are traversed by a flow. 32 nodes allocated to eight clusters. 50% of flows are outgoing.

time the relays code symmetric packet flows.

V. Conclusions

In this paper, we proposed a practical approach for improving the efficiency of wireless network coding at the MAC layer. The proposed protocol ensures deterministically correct coding decisions at network nodes, while decoding takes place with minimal additional protocol headers and no cross-layer interactions. These issues are solved first with the use of opportunistic and adaptive acknowledgments at the MAC layer, and second, with the use of virtual opportunistic packet buffers. The result is a set of simple extensions to the IEEE 802.11 MAC protocol and the DCF channel access scheme. The proposed protocol can work incrementally as part of IEEE 802.11 since the communication capabilities of legacy nodes are not affected. Simulations of the proposed protocol for several ad hoc network topologies demonstrate a performance improvement over existing schemes by 30-40%.

References

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

[2] C. Fragouli, D. Katabi, A. Markopoulou, M. Medard, and H. Rahul, "Wireless network coding: Opportunities and challenges," in *Allerton Conference*, 2007.

[3] S. Katti, D. Katabi, W. Hu, H. Rahul, and M. Medard, "The importance of being opportunistic: Practical network coding for wireless environments," in *Allerton Conference*, 2005.

[4] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," in *SIGCOMM*, 2006.

[5] J. Liu, D. Goeckel, and D. Towsley, "Bounds on the gain of network coding and broadcasting in wireless networks," in *IEEE INFOCOM*, 2007.

[6] S. Sengupta *et al.*, "An analysis of wireless network coding for unicast sessions: The case of coding-aware rotuing," in *IEEE INFOCOM*, 2007.

[7] Y. E. Sagduyu and A. Ephremides, "On joint MAC and network coding in wireless ad hoc networks," *IEEE Transactions on Information Theory*, vol. 53, no. 10, pp. 3697–3713, October 2007.

[8] P. Chaporkar and A. Proutiere, "Adaptive network coding and scheduling for maximizing throughput in wireless networks," in *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2007, pp. 135–146.

[9] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, "On MAC scheduling and packet combination strategies for practical random network coding," in *IEEE ICC*, 2007.

[10] J. Le, J. C. Lui, and D. M. Chiu, "Towards coding-efficient link-scheduling and coding-aware routing in wireless networks," in *International Conference on Network Protocols*, 2007.