

On MAC Scheduling and Packet Combination Strategies for Practical Random Network Coding

Elena Fasolo[†], Michele Rossi[†], Jörg Widmer^{*} and Michele Zorzi[†]

[†]DEI, University of Padova, via Gradenigo 6/B – 35131, Padova, Italy

^{*}DoCoMo Euro-Labs, Landsberger Strasse 312 – 80687 Munich, Germany.

Abstract—The present paper investigates practical algorithms to efficiently exploit random network coding for data delivery in multi-hop wireless networks. In the past few years, a great deal of work has been carried out to derive analytical results about network coding. However, only recently have researchers started to utilize the theoretical findings in practical settings. Network coding is a new paradigm for data delivery which proved to be very efficient. It is particularly suitable for wireless networks due to the inherent broadcast nature of the channel. Even though previous work dealt with practical schemes exploiting these new techniques, many issues concerning the coexistence of network coding and channel access mechanisms are still unsolved. In addition, it is still unclear how packets should be combined in order to get the highest benefits in terms of throughput, delay, and energy efficiency. Our work presents an accurate investigation of these aspects. In particular, we couple several MAC and scheduling schemes together with different network coding strategies, and compare them via extensive ns2 simulation. Finally, we propose a new timing strategy for the combination of data packets in random network coding.

I. INTRODUCTION

In the context of wireless pervasive systems, we are often faced with the problem of disseminating or retrieving a large amount of data which may be partially and/or temporarily stored within mobile devices, see [1] and references therein. Standard approaches to deal with this problem consist of flooding algorithms, epidemic routing [2] and probabilistic routing [3]. These techniques, however, often incur high overhead which may not be sustainable in practical settings. Recently, network coding has been proposed as an efficient paradigm to disseminate information over distributed networks [4]. The core of this strategy was presented for the first time in [5], and subsequently developed in the last few years by giving particular emphasis on its theoretical aspects. Network coding is a packet dissemination strategy which can be used to improve the throughput, thus guaranteeing high performance. In contrast to the store and forward paradigm, network coding implements a more complex *store, code, and forward* approach where each node stores the incoming packets in its own buffer, and, at transmission time, sends a *combination* of the stored data. To successfully decode, say, K packets, a node must collect K independent combinations. This type of strategy is particularly suitable in case of one to all or all to all communication. However, the efficiency of this technique has also been proven for the unicast and multicast communication cases [6]–[11].

We note that most of the work done so far has focused on the theoretical aspects of network coding [12]–[15]. In fact, researchers only recently started to look at practical solutions to reap the full benefits of network coding techniques in actual network settings [16]. Initially, practical schemes were proposed

for wired networks, where coding strategies were applied to peer-to-peer applications [17], [18]. In [19], one of the first examples of a simple practical solution for network coding, the authors focused on how the coding matrix as well as the information related to the random combination of packets can be shared by different nodes. This is a crucial aspect for network coding algorithms to work in actual networks. Further work can be found in [9], [11], [20]. A recent paper [9] focuses on unicast transmissions exploiting the network coding paradigm. In [9], it is experimentally shown that large gains, in terms of maximum throughput, are possible even in the case of unicast transmissions. The scheme presented in [11] jointly considers packet combinations with ARQ strategies for wireless sensor networks. In [20], the authors analyze and present some heuristics to combine the packets and prove the superiority of network coding with respect to flooding schemes in multi-hop wireless networks.

We observe that, even if some valuable work has already been pursued, many practical aspects, such as the interaction between MAC schedules and network coding techniques, still need to be properly addressed. In our view, these practical aspects may limit the benefits achievable with network coding. Motivated by these needs, in this paper we present results on MAC schedules and packet combination rules. In particular, we show their impact on system performance and propose a first improvement to better cope with collisions and suboptimal schedules. We consider a CSMA-like system affected by collisions, interference, and a random scheduling of the packets. In such a scenario, we test the behavior of network coding over simple wireless network configurations in order to capture the effects of each protocol component. Results are obtained using the ns2 network simulator, appropriately extended to include network coding functions.

The rest of the paper is organized as follows. In Section II, we briefly introduce the basic concepts of network coding and present some well known schemes designed for practical networks. In Section III, we describe our performance investigation by introducing both network coding schemes and networking protocols, and present our proposal and the evaluation scenarios. In Section IV, we show and discuss the numerical results obtained. Finally, in Section V, we conclude the paper suggesting some possible lines for future research.

II. REVIEW OF LINEAR NETWORK CODING

As mentioned in Section I, network coding differs from the classical store and forward paradigm as information packets are encoded at intermediate nodes in the network and subsequently

forwarded according to a store, code, and forward approach. The strength of this strategy consists of the coding phase, which improves the dissemination efficiency and reduces the number of transmissions required to deliver data. Although network coding schemes require some data processing at intermediate nodes to code/decode packets, it has been proven that simple coding operations, e.g., linear coding, are sufficient to achieve optimal performance [21]. According to this approach, when a node receives, as input, a given number of packets, these are linear combinations of some original information packets that are to be disseminated through the network. In practice, the node receives the new data and re-transmits a (possibly) new linear combination of the received/stored packets. This mechanism can be described by a theoretical model which is summarized in the following (see [5], [21] for further details).

Let n be the number of nodes in the network and let x_1, \dots, x_n be the source symbols (or in general packets) generated at these nodes. Symbols are defined over a finite Galois field F_q of q elements. Each linear combination y over F_q can be described as the product of a vector of coefficients, \mathbf{a} (*coding vector*), and the vector of source symbols, \mathbf{x} :

$$y = \mathbf{a}^T \mathbf{x} = \begin{bmatrix} a_1 & \dots & a_n \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad (1)$$

Each device in the network produces one combination symbol. Asynchronous decoding at any given node (i.e., obtaining \mathbf{x}), requires the correct reception of at least n independent combination symbols (along with their respective coding vectors \mathbf{a}).

In a multicast scenario, linear codes considerably improve the throughput of the system [21]. Further work, by Koetter and Médard [15], derives the coefficients of linear coding/decoding functions by exploiting some of their theoretical properties as well as accounting for the time needed for their calculation. In [19], the authors propose to send the coding vector along with y in the same packet. Each node v stores the coding vectors it receives in a decoding matrix \mathbf{G}_v . A newly received packet is said to be *innovative* if its vector increases the rank of this matrix. Reception of non-innovative packets is simply ignored. A node v will transmit a linear combination that lies in the vector space of its decoding matrix \mathbf{G}_v . Once a node has received n linearly independent combinations (and hence \mathbf{G}_v has full rank), it can decode and retrieve the original symbols x_1, \dots, x_n . The authors of [19] also propose to randomly choose the coding coefficients (in F_q). The strength of random coding is that if the local coding vectors are generated randomly, and the symbols lie in a finite Galois field of sufficient size, the decoding matrix \mathbf{G}_v has full rank with high probability. This is important as the random nature of the scheme allows to implement the coding/decoding operations in a distributed and asynchronous manner. Random network coding is the main coding technique adopted in most of the current work on distributed and asynchronous wireless ad hoc networks [20], [22]. The price to pay is the overhead associated with the transmission of the coding vectors \mathbf{a} .

In [20], [22], the authors show that it is possible to reap most of the benefits of network coding by means of a proper

scheduling of packet transmissions. They provide analytical results for simple networks and a practical algorithm that they test on an ideal MAC, where collisions never occur. In particular, they consider circular topologies where each node has exactly two neighbors and generates a single packet to be transmitted to all other nodes. They show that a network coding scheme guaranteeing $T/T_{nc} = 2$ exists, where T_{nc} is the total number of transmissions required to deliver one packet to all other nodes using network coding and T is the total number of transmissions required by flooding.

One weakness of these studies is that they completely neglect the impact of the MAC protocol on network coding. On the other hand, implementing a scheme with perfect coordination among nodes may not be an easy task in practice, especially in a wireless environment. In addition, interference and collisions are expected to have a strong impact on the network coding performance (delivery ratio and latency).

III. PROBLEM DESCRIPTION

In the remainder of the paper we evaluate the impact of a realistic MAC and physical layer on random network coding in wireless ad hoc networks. In general, such networks are severely constrained by interference and channel impairments, especially in the case of broadcast communication. Consider, for instance, that each node is interested in retrieving information from all other nodes in the network. In this case, the use of traditional access mechanisms such as CSMA-like protocols would incur high contention on the wireless channel which, in turn, translates into a high number of collisions and of dropped packets. Network coding is a promising technique to increase system performance by reducing the number of transmissions and exploiting the random combination of data to increase transmission efficiency. Up to now, several practical network coding schemes have been proposed for both unicast and broadcast transmissions [9], [16]–[18], [20] and their efficiency was shown through simulation and analytical results.

Our contribution differs from previous work in the following aspects. First, we systematically analyze the impact of the MAC protocol in use on the network coding performance. We note that previous studies [9], [16] address the problem of implementing network coding over actual MAC protocols. However, they lack a thorough analysis of their impact on network coding. Second, we focus on the broadcast communications paradigm rather than applying network coding to the case of unicast flows. Finally, we look at network coding strategies which do not need any knowledge about the status of neighboring nodes, thereby requiring very little overhead.

In our opinion, two main factors are to be taken into account when using network coding as part of practical solutions for wireless ad hoc networks, namely:

- *Collisions*: collisions are a source of packet losses. It is important to understand their impact on the performance of network coding.
- *Packet Scheduling*: using random access will not create perfect transmission schedules. In fact, the number of neighbors, their traffic pattern, and their movement are not known a priori. Moreover, obtaining such information in

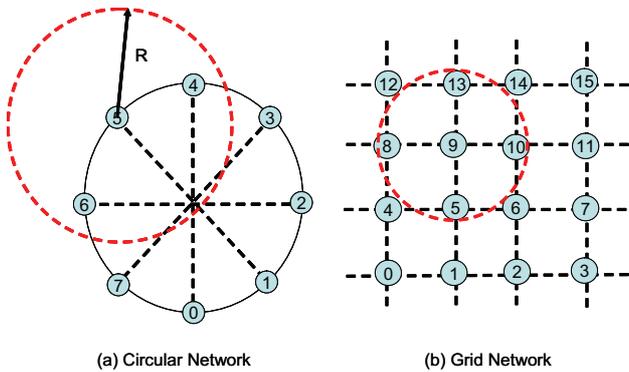


Fig. 1. Reference topologies for the performance evaluation.

order to build optimal transmission schedules is not convenient due to the large overhead involved. Understanding the impact of packet scheduling is another crucial point in the design of practical solutions.

In the following, we consider very simple network configurations to precisely understand and highlight the above issues. We start with the reference scenarios shown in Fig. 1. In the first network configuration, nodes are placed on a circular topology and each node has exactly two neighbors. In the second configuration, nodes have four neighbors and are placed on a grid. Finally, we also consider random networks, where nodes are randomly positioned within the simulation area. For the traffic pattern, each node inserts into the network a single original packet and is interested in collecting all the other inserted packets. Original packets are generated according to either a *random* or *deterministic* traffic pattern. In the former case, each node inserts its original packet independently by picking the insertion time uniformly in a fixed length interval of $\Delta_1 = 100$ ms. In the latter case, we can assume to have a simple application that inserts original packets sequentially in each node. Subsequent insertions are separated by fixed time intervals of $\Delta_2 = 1$ s. That is, the first original packet is inserted in node 0 at time 0 s, the second one in node 1 at time 1 s, and so on. The reason for this generation strategy is twofold. First, if the interval Δ_2 is sufficiently large, the collision probability is sufficiently small. Second, if the transmission schedule is $\{\text{node } 0, \text{node } 1, \dots, \text{node } n-1\}$ (see Fig. 1(a) where $n = 8$) we obtain the scheme theoretically derived in [20], which was shown to achieve the maximum throughput in circular networks.

In the following, we describe the schemes and the algorithms we analyzed. In addition, we propose a solution which improves the network coding performance in wireless ad hoc networks.

A. Implemented MAC Protocols

We consider four different MAC protocols based on CSMA, which is currently the most widely used medium access mechanisms in wireless ad hoc networks.

1) *IEEE 802.11b*: We consider IEEE 802.11b as the baseline medium access protocol. Note that the network coding strategies we examine are based on broadcast transmissions. Hence, we adopt the basic access provided by IEEE 802.11b which, in the broadcast case, does not use any acknowledgment mechanism. As a consequence, in case of a collision, no retransmission

occurs and the packet is lost, resulting in high inefficiency and low packet delivery ratio.

2) *IEEE 802.11b with pseudo broadcast [9]*: This scheme is an improvement of the basic IEEE 802.11b, where an acknowledgment mechanism is implemented. According to the idea proposed in [9], a given node first broadcasts a packet to its neighbors, by randomly picking one of them and including its address in the packet header. Only the node whose address matches the one contained in the header sends an acknowledgment to the sending device. This is done according to the basic IEEE 802.11b unicast communication (no RTS/CTS). All other neighbors overhear/decode the transmission but do not respond to the sending node. The packet is retransmitted in case there is no acknowledgment. Note that, using this mechanism, only collisions at the addressed receiver can be detected, while collisions occurring at any of the remaining neighbors are ignored. Also, this strategy does not solve the hidden node problem.

3) *IEEE 802.11 with pseudo broadcast and RTS/CTS handshaking*: To further improve the packet delivery ratio, we propose to consider the previous scheme with additional RTS/CTS handshake. These control messages are introduced to alleviate the hidden node problem. The CTS is only transmitted by the node addressed in the packet header. The delay introduced by this technique is expected to be higher, due to the additional control packets. Moreover, as for the previous schemes, we can not detect collisions at all overhearing nodes.

4) *Ideal MAC*: With the term ideal MAC we refer to a very simple mechanism where transmitted packets are only affected by the transmission delay, Δ_{tx} . That is, we can assume to have an omniscient entity which regulates the transmissions in order to avoid interference and collisions. This means that, as a node sends a packet, all its neighbors successfully receive the message after a (fixed) transmission delay. Δ_{tx} is computed using the same rate and packet size of the above MAC protocols. This scheme, which is not feasible in practice, is analyzed to obtain an upper bound on the achievable performance. Such an upper bound is used as a benchmark for the other solutions.

B. Network Coding Strategies

The core of the network coding strategies we use are the same as the ones proposed in [19]. These are based on random linear coding where the coefficients of the combination are included in each transmitted packet. In addition, we implement three different techniques for combining packets. The first two are inspired by the work in [20], while the last one is a new proposal.

In [20], it was shown that network coding allows to reduce the number of transmissions, with respect to pure store and forward, for a certain targeted packet delivery ratio. The achievable reduction in the number of messages generally depends on the number of neighbors. For instance, if there are 2 neighboring nodes, network coding over circular networks halves the number of transmissions needed to achieve a packet delivery ratio equal to one. A node does not need to transmit a new packet at each reception of an innovative message. This is the basic idea of the network coding algorithms proposed in [20]. In the following, we detail the packet combination strategies considered in this

paper. All the presented schemes are characterized by a design parameter, named *forwarding factor*, ρ , which is defined as the ratio between the number of packets transmitted and the number of innovative packets received, per node. It determines the average number of packets that each node can transmit.

1) *Probabilistic Network Coding*: This approach exploits random linear coding. Each node sends a random linear combination of the packets in its buffer, as discussed in Section II. Only the reception of innovative packets carries additional information. Hence, non-innovative packets can be discarded. With probabilistic network coding, when a node receives an innovative packet, it makes a decision as to whether a new random combination should be transmitted or not. Specifically, upon the reception of an innovative message, a new combination is transmitted with probability p by assigning to the *forwarding factor* ρ the value of p . For $\rho = 0.5$, a node on average sends a new message for every two innovative packets received. As per our discussion above, $\rho = 0.5$ would theoretically (ideal scheduling, no collisions) assure a packet delivery ratio equal to one when the number of neighbors is two.

2) *Semi-deterministic Network Coding*: This strategy is quite similar to the previous scheme. In this case, for a given forwarding factor ρ , each node sends out a new combination after receiving exactly $\lceil 1/\rho \rceil$ innovative packets. As an example, $\rho = 0.5$ means that each node deterministically transmits a new combination for every two received innovative packets. The forwarding factor in this case is not related to a probability, but is rather used as a threshold on the number of incoming messages.

3) *Timed Network Coding*: The two previous schemes have two major drawbacks. The first drawback is that they are particularly sensitive to packet losses due to, e.g., collisions, as shown in Section IV. In fact, if one of the transmitted packets is lost, the propagation of the information through the network could be interrupted. To better illustrate this, let $\rho < 1$ be the forwarding factor in use. In this case, for a given targeted packet delivery ratio, we can reduce the number of new combinations transmitted. The effect of such an operation is to increase the transmission efficiency at the expense of a higher sensitivity to packet losses. The second drawback is that both probabilistic and semi-deterministic network coding suffer from some inefficiencies when there is a small number of packets to combine. In such cases, new combinations are created from a small set of packets and, for this reason, are often not innovative. To alleviate these problems, we introduce a *timing strategy* into the first scheme. For each received innovative packet, a timer is activated. As the timer expires, the node decides to send out a new random combination with probability $p = \rho$. The timer, τ , is a uniform random variable in $[0, \tau_{max}]$. The main advantages of this timing approach are twofold. First, it facilitates packet mixing, thus reducing the likelihood of transmitting non-innovative packets. Without the timer, indeed, some of the nodes that receive an innovative packet might decide to simultaneously send out a new packet combination. This can lead to a non innovative transmission, especially when the buffers are almost empty. With the introduction of a waiting interval before coding, nodes have the chance of collecting other innovative packets and send out richer combinations. Moreover, the reduction of the number of transmissions and the random characteristic of

the timer help in reducing the collision probability at the MAC layer. The drawback of the timed scheme is the introduction of a short delay due to the timer. Hence, the timer value shall be chosen so as to achieve a good trade-off between extra-delay and performance improvements. In IEEE 802.11b, this value has to be long enough to allow the collection of more than one packet, which translates to selecting $\tau_{max} \approx 10 - 30$ ms. In the rest of the paper we consider $\tau_{max} = 20$ ms.

IV. SIMULATION RESULTS

In this section, we report the most relevant results obtained via ns2 simulations. All presented schemes are evaluated over the simple topologies introduced in Section III, taking into account the *random* and the *deterministic* traffic patterns. We tested the algorithms varying the *forwarding factor* ρ from 0.1 to 1 and the number of nodes in the network, n . For the circular topologies, we considered $n \in \{4, 8, 12, 16\}$ and for grid configurations $n \in \{9, 16, 64\}$. Regarding the MAC layer, we considered a transmission rate of 1 Mbps. Each packet has an extra overhead which is accounted for to transmit the coefficients of the random combinations [19]. As discussed in [19], such an overhead is tolerable for practical cases. In the following, we define the performance metrics we look at in our investigation:

Packet Delivery Ratio, PDR: is defined as the ratio between the number of successfully received (and decoded) packets and the number of packets in which a node is interested, averaged over all nodes.

Packet Delivery Delay, D: is the average time between the first transmission of a packet and the reception and successful decoding at the destination node.

Protocol Overhead: is the ratio between the number of transmitted packets (Pkt_{tx}) at the MAC layer and the number of successfully decoded packets (Pkt_{dcd}). This value depends both on the adopted MAC protocol and on the efficiency of the network coding strategy. For example, we expect that IEEE 802.11b pseudo broadcast with RTS/CTS, will show higher protocol overhead compared to other MAC schemes. On the other hand, timed network coding should decrease the protocol overhead by suppressing unnecessary transmissions. Note that this metric gives us a measure of the energy consumption as well.

Collision Ratio: is the number of collided packets at the receiver (Pkt_{col}) over the number of received packets (Pkt_{rcv}). Observe that in a broadcast wireless environment, tracking the number of collisions could be a problem. In fact, the same packet may collide only for a subset of the receiving nodes. For this reason, we evaluate the number of collisions at each receiver.

In the simulations, we compare the network coding strategies introduced in Section III-B against each other and against probabilistic flooding. This is done to point out the possible benefits of the network coding paradigm with respect to standard store-and-forward. The probabilistic flooding considered in this paper uses a forwarding factor ρ , which is simply the probability of forwarding a new incoming packet.

We organize our performance analysis in two parts. In Section IV-A, we evaluate the impact of different MAC protocols on

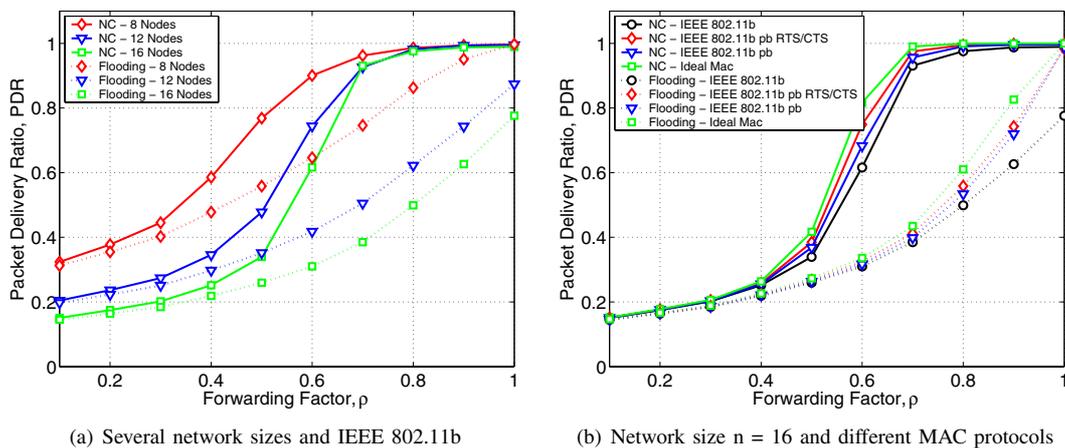


Fig. 2. Performance of *Probabilistic Network Coding* and *Probabilistic Flooding* in circular networks.

network coding whereas, in Section IV-B, we focus on different packet combination strategies.

A. The Impact of MAC Protocols

We start the performance analysis with Fig. 2, where we compare probabilistic network coding (solid lines) against probabilistic flooding (dotted lines) in a circular network topology. Fig. 2(a) shows the packet delivery ratio vs. ρ for different numbers of nodes, n . For all forwarding factors ρ , network coding outperforms probabilistic flooding. As an example, for $n = 12$ and $\rho = 0.6$, network coding achieves $PDR \approx 0.75$, whereas $PDR \approx 0.42$ for the flooding scheme. This gain increases with increasing n as well as with increasing ρ . In addition, network coding with $\rho = 1$ always results in a PDR very close to one. This is not true for flooding, which is considerably impacted by packet losses. As observed above, for this topology a PDR equal to one is theoretically achievable by setting $\rho = 0.5$ [20]. However, we observe from Fig. 2(a) that this performance level is never reached in practice and that the actual PDR depends on the network size n . These effects are due to the use of an actual MAC layer (IEEE 802.11b in this case) and to the suboptimality of random scheduling, which indicates the importance of these issues for the design of practical schemes. In Fig. 2(b) we focus on circular topology with $n = 16$ (worst case in Fig. 2(a)) and we look at the impact of the MAC layer on both probabilistic network coding and flooding. Once again, we observe the superiority of network coding. Moreover, we can evaluate the importance of the MAC scheme in use. For $\rho = 0.6$, IEEE 802.11b achieves $PDR \approx 0.6$, whereas an ideal MAC achieves $PDR \approx 0.8$. This corresponds to a 25% improvement of packet delivery ratio. On the other hand, for this value of the forwarding factor a perfect schedule leads to full packet delivery ratio. The effectiveness of pseudo broadcast (IEEE 802.11 pb in the figure) and pseudo broadcast with RTS/CTS (IEEE 802.11 pb RTS/CTS) is also clear, though the improvements are not as large as expected.

Even though circular networks are a simple reference scenario, useful to easily capture network coding behavior, we focus now on a more realistic setting where nodes are placed over a grid (see Fig. 3). Due to lack of space, we consider here only grid networks with $n = 16$ in order to directly compare

them with the circular case. However, in our simulations, we noticed the same behavior also for different network sizes. In Fig. 3(a), we show the impact of different MAC protocols on the packet delivery ratio of probabilistic network coding and flooding. As expected, the achieved performance is better than in the circular case due to the higher number of neighbors (4 instead of 2), which favors packet mixing and dissemination. Also in this scenario, the presence of realistic MAC layers reduces significantly the packet delivery ratio for a given value of ρ . In addition, Fig. 3(b) shows the protocol overhead vs. ρ for each MAC protocol. It is noted that the schemes implementing collision avoidance policies (i.e., IEEE 802.11b with pseudo broadcast and IEEE 802.11 with pseudo broadcast and RTS/CTS handshaking) improve the packet delivery ratio but also increase the protocol overhead. This is due to the MAC retransmissions in case of collisions and to the control traffic (i.e., ACK, RTS and CTS packets). In addition, we note that when we compare probabilistic network coding and flooding performance against ρ , we have a fair comparison as, given a specific ρ and a fixed MAC protocol, both network coding and flooding lead to very close protocol overhead.

Fig. 3(c) shows the effectiveness of pseudo broadcast and pseudo broadcast with RTS/CTS in decreasing the number of collisions: for a given value of the Pkt_{col}/Pkt_{recv} ratio the number of received packets (Pkt_{recv}) in pseudo broadcast with RTS/CTS is the highest. This is due to both the higher PDR of the scheme and, mostly, the additional retransmissions caused by the acknowledgments. On the downside, using additional techniques to recover from packet loss leads to longer delays, as can be seen from Fig. 3(d). The average delay increase is about one order of magnitude in the worst case (pseudo broadcast with RTS/CTS). We also note that network coding always outperforms the flooding scheme when using the same MAC and that its delay stabilizes for increasing ρ . The reason for the stabilization and even decrease in delay is that for increasing ρ (beyond a given ρ^*), PDR remains close to one but the number of innovative packets flowing in the network continues to increase. This has the effect of allowing earlier decoding.

To summarize, we observe that the presence of actual MAC protocols reduces network coding performance in terms of

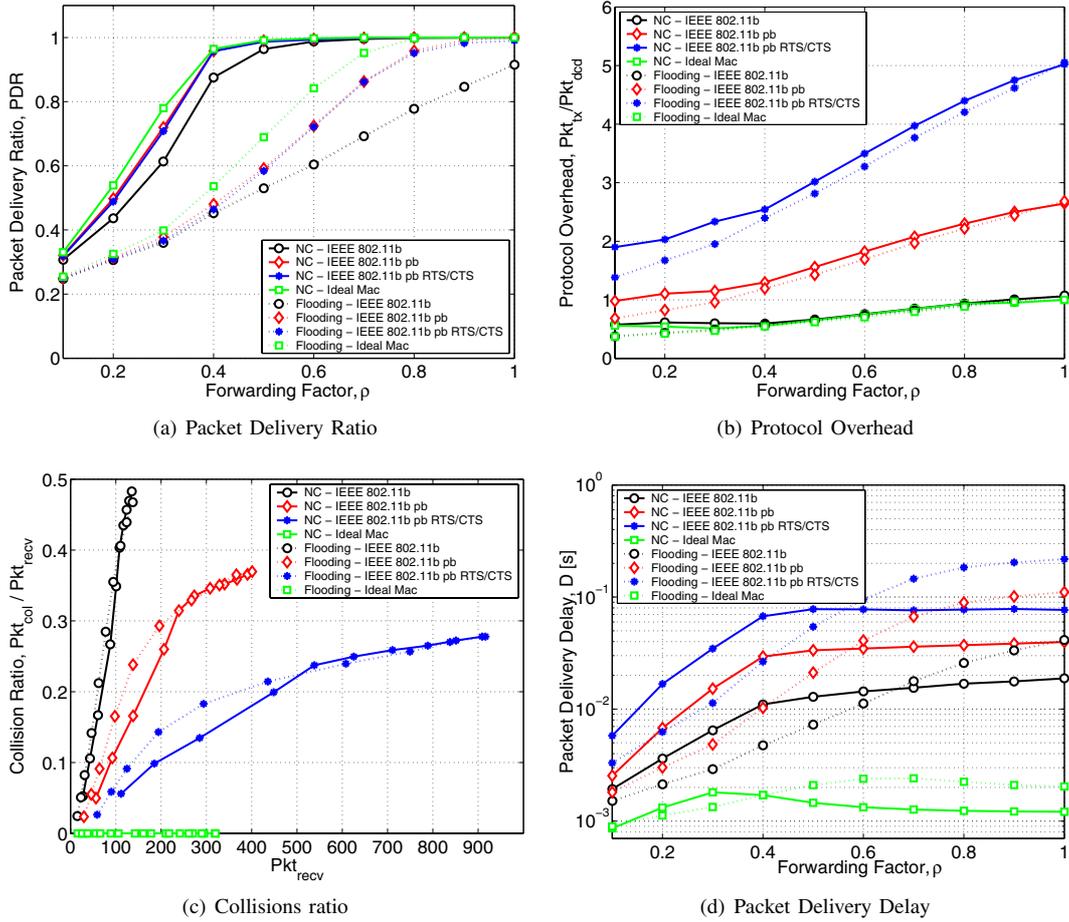


Fig. 3. Performance comparison of *Probabilistic Network Coding* and *Probabilistic Flooding* for different MAC protocols in grid networks with $n = 16$.

packet delivery ratio. This performance reduction depends on the network size, contrary to what happens for the ideal MAC case. In addition, collision avoidance policies give little improvement in terms of packet delivery ratio, while leading to poor overhead and delay performance.

B. Different Packet Combination Strategies

We now evaluate the impact of the network coding schemes described in Section III-B. Fig. 4 shows the packet delivery ratio performance for a circular network with $n = 16$ by varying the packet combination strategy (for a fixed MAC protocol). In Fig. 4(a) we use an IEEE 802.11b MAC with both the semi-deterministic and the probabilistic combination methods. The semi-deterministic schemes (dotted lines) show a sudden phase change, where PDR remains constant up to $\rho^* = 0.4$ and then suddenly increases for higher forwarding factors. This does not occur for probabilistic network coding (solid lines) whose curves are smooth. This reflects the threshold based transmission policy of semi-deterministic network coding. The exact value of the shifting point ρ^* depends on the number of neighbors. For circular networks, where each node has exactly two neighbors, $\rho < 0.5$ ($\lceil 1/\rho \rceil > 2$) never suffices to trigger the transmission of a new combination, as the initial number of innovative packets is equal to two. This flaw is not present in probabilistic and timed network coding, whose sending rules are based on

probabilities rather than on hard thresholds. Notably, timed network coding outperforms the semi-deterministic scheme with deterministic traffic pattern for $\rho \leq \rho^*$ and performs very close to this method for larger forwarding factors. In addition, the timed strategy performs better than both semi-deterministic and probabilistic network coding with random scheduling. For $\rho = 0.5$, probabilistic network coding with random scheduling achieves $PDR \approx 0.35$, whereas timed network coding leads to $PDR \approx 0.55$, which corresponds to an improvement of about 36%. The same considerations hold for Fig. 4(b) with the only difference that in this case we adopt an ideal MAC. The performance is thus rescaled accordingly. As can be seen from a direct comparison of the two figures, the impact of MAC on packet delivery ratio performance is smaller than that of the packet combination strategy in use. We also note that, in Fig. 4(b), for $\rho = 0.5$ PDR is higher than 0.8 but is still strictly lower than one (theoretical bound), even if we use an ideal MAC and a deterministic scheduling. This is due to the fact that our deterministic scheduling approach is only an approximation of the ideal scheme in [20].

Timed network coding is further evaluated in Fig. 5(a), where we plot the performance in terms of delay. We observe that the timed strategy introduces an additional delay. Also, there are some expected differences between ideal and actual MAC. For IEEE 802.11b, the delay increase is reasonably small

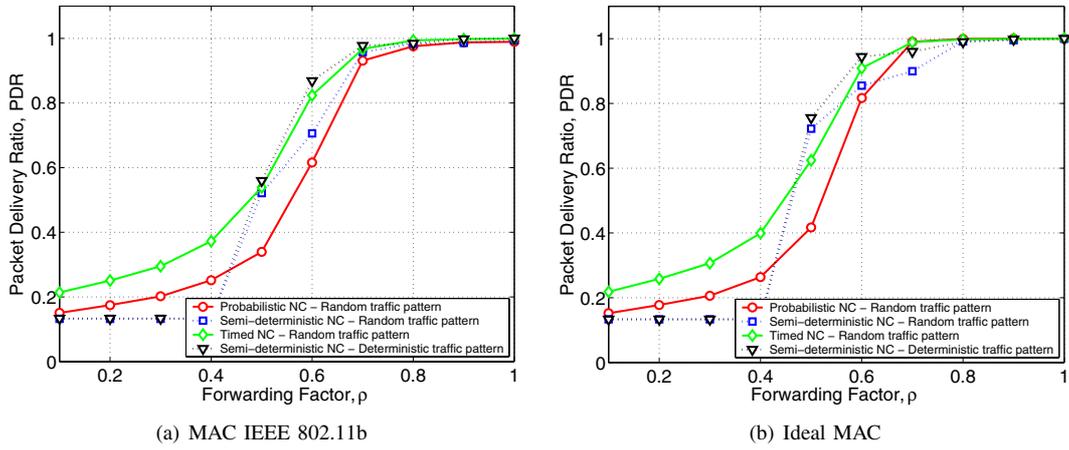


Fig. 4. Performance comparison of different combination strategies in circular networks with $n = 16$.

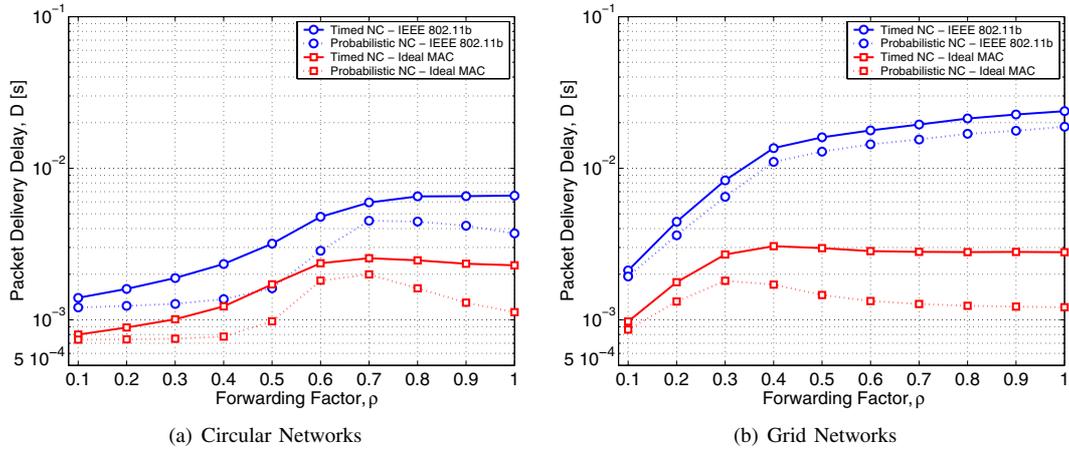


Fig. 5. Performance comparison of Packet Delivery Delay for *Probabilistic* and *Timed Network Coding* using IEEE 802.11b and Ideal MAC for $n = 16$.

(approximately equal to the average value of the timer) and is similar to that introduced by the pseudo broadcast algorithms. Hence, the timed combination provides higher benefits in terms of packet delivery ratio than pseudo broadcast, by leading to similar extra-delays. For this reason, the timed scheme may make sense when the goal is to maximize the packet delivery ratio (throughput) by accepting some delay degradation.

The delay in the grid network (four neighbors per node) scenario is plotted in Fig. 5(b): the impact of the adopted MAC is more pronounced than for circular networks. This means that the importance of MAC is higher when the number of neighbors increases (circular \rightarrow grid scenario) as said before. On the other hand, we note that the gap between timed and probabilistic schemes is smaller than in Fig. 5(a).

The performance analysis is continued in Fig. 6(a), where probabilistic network coding and timed network coding are compared for the cases of two and four neighbors. The timed combination strategy outperforms probabilistic network coding with plain IEEE 802.11b by about 30% for $\rho \approx 0.3$. Also, we observe that the gap between ideal and actual MAC is tighter when the number of neighbors is four (grid networks). This, together with the result in Fig. 5(b), suggests that the timed strategy becomes more effective with an increasing number of neighbors. In Fig. 6(b), we report PDR for ideal/actual

MACs and probabilistic/timed strategies in random networks with node densities of 7 and 15 neighbors per node (selected as representative of different density scenarios). Only connected topologies were considered to obtain this plot. Similarly to what observed earlier, PDR increases with increasing node density (7 \rightarrow 15). We stress that, in an ideal grid scenario, $PDR \rightarrow 1$ as ρ approaches the inverse of the number of neighbors. If this were true for random networks, in Fig. 6(b) for, e.g., 15 neighbors we should get a $PDR = 1$ when $\rho \approx 0.06$. However, this is not verified for two reasons: first, random networks are not uniform in the sense that some nodes have more neighbors than others; second, a probabilistic forwarding policy cannot get to the expected performance, which is instead achievable using an ad hoc deterministic scheduling (see [20]). For the 7-neighbor case, with ideal MAC we can still get $PDR = 1$ by properly tuning ρ . This is, however, not the case when an actual MAC is used, where even $\rho = 1$ does not suffice to get $PDR = 1$. Finally, we can note that the gain achieved by the timed strategy against the probabilistic scheme remains significant. In general, random networks are impacted by the very poor performance of the dissemination procedure in the proximity of the nodes with low degree (small number of neighbors). For these nodes the probabilistic approach does not work properly and the information flow (new innovative packets)

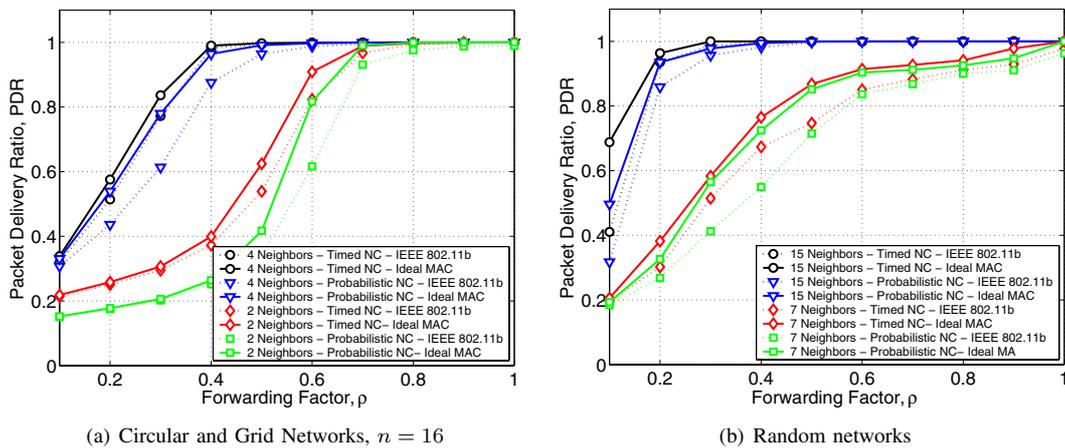


Fig. 6. Performance comparison of Packet Delivery Ratio for *Probabilistic* and *Timed Network Coding* for ideal and actual MAC.

is likely to be stopped.

Our results show that even slight modifications to the packet combination strategy may lead to considerable performance improvements. In light of this, directions for future research include strategies to exploit some knowledge about number of neighbors and coding state of nearby devices, in order to efficiently handle packet forwarding at nodes with low degree. This should help coping with the lack of regularity exhibited by random networks.

V. CONCLUSIONS AND DISCUSSION

In this paper we analyzed the impact of realistic MAC layer, transmission schedule, and packet combination strategy on random network coding. We also introduced a simple timing policy which improves network coding performance. By evaluating the packet delivery ratio of network coding strategies in the presence of different CSMA protocols, we observed that the impact of the MAC layer is not as large as expected. That is, introducing mechanisms to alleviate the collision problem only leads to limited improvements in the packet delivery ratio performance and affects the latency. On the other hand, we observed that the packet combination strategy plays a fundamental role. In fact, our proposal, in spite of its simplicity, shows promising performance.

We further note that all the schemes and protocols studied in this paper do not achieve the theoretical performance of [20], and this motivates further research, especially about the combination strategy to use. As an example, by looking at Fig. 6(a) and focusing on the timed network coding case for $\rho = 0.5$ and two neighbors, we note that the packet delivery ratio is $PDR \approx 0.6$. This means that there is still ample room for improvement in order to get closer to the theoretical performance of $PDR = 1$.

Preliminary results on more complex scenarios, such as random networks, have shown similar behavior. However, more research is needed to address these cases.

REFERENCES

- [1] E. Fasolo, C. Prehofer, M. Rossi, Q. Wei, J. Widmer, A. Zanella, and M. Zorzi, "Challenges and new approaches for efficient data gathering and dissemination in pervasive wireless networks," in *INTERSENSE*, Nice, France, Apr. 2006.
- [2] P. T. Eugster, R. Guerraoui, A. M. Kermarrec, and L. Massoulié, "Epidemic information dissemination in distributed system," *IEEE Computer Magazine*, vol. 37, no. 5, pp. 60–67, May 2004.
- [3] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic Routing in Intermittently Connected Networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, July 2003.
- [4] R. Koetter, "Network coding bibliography." [Online]. Available: <http://tesla.csl.uiuc.edu/~koetter/NWC/Bibliography.html>
- [5] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Trans. on Information Theory*, vol. 46, no. 4, July 2000.
- [6] Y. Zhu, B. Li, and J. Guo, "Multicast with Network Coding in Application-Layer Overlay Networks," *IEEE JSAC*, vol. 22, no. 1, Jan. 2004.
- [7] T. Ho, B. Leong, R. Koetter, and M. Médard, "Distributed Asynchronous Algorithms for Multicast Network Coding," in *Netcod*, Riva Del Garda, Italy, Apr. 2005.
- [8] T. Ho, M. Médard, and R. Koetter, "A coding view of network recovery and management for single receiver communication," in *CISS*, Princeton, NJ, US, Mar. 2002.
- [9] S. Katti, H. Rahul, W. Huss, D. Katabi, M. Médard, and J. Crowcroft, "XORs in The Air: Practical Wireless Network Coding," in *ACM SIGCOMM*, Pisa, Italy, Sep. 2006.
- [10] D. Lun, N. Ratnakar, R. Koetter, M. Médard, E. Ahmed, and H. Lee, "Achieving Minimum-Cost Multicast: A Decentralized Approach Based on Network Coding," in *IEEE Infocom*, Miami FL, US, Mar. 2005.
- [11] D. Dubois-Ferrere, D. Estrin, and M. Vetterli, "Packet Combining in Sensor Networks," in *ACM SenSys 2005*, San Diego, CA, USA, Nov. 2005.
- [12] C. Fragouli and E. Soljanin, "Information Flow Decomposition for Network Coding," *IEEE Trans. on Information Theory*, vol. 52, no. 3, pp. 829–848, Mar. 2006.
- [13] D. Tuninetti and C. Fragouli, "Processing along the way: forwarding vs. coding," in *ISITA*, Parma, Italy, Oct. 2004.
- [14] C. Fragouli and E. Soljanin, "On average throughput benefits of network coding," in *42st Annual Allerton Conference on Communication Control and Computing*, Monticello, IL, US, Oct. 2003.
- [15] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. on Networking*, vol. 49, no. 11, Nov. 2003.
- [16] Y. E. Sagduyu and A. Ephremides, "Crosslayer Design for Distributed MAC and Network Coding in Wireless Ad Hoc Networks," in *IEEE International Symposium on Information Theory*, Adelaide, Australia, Sep. 2005.
- [17] C. Gkantsidis and P. Rodriguez, "Network Coding for Large Scale Content Distribution," in *IEEE Infocom*, Calcutta, India, Dec. 2005.
- [18] M. Wang and B. Li, "How Practical is Network Coding?" in *IWQoS*, New Haven, CT, USA, Jun. 2006.
- [19] P. A. Chou, T. Wu, and K. Jain, "Practical network coding," in *41st Allerton Conf. Communication, Control and Computing*, Monticello, IL, US, Oct. 2003.
- [20] C. Fragouli, J. Widmer, and J.-Y. L. Boudec, "A network coding approach to energy efficient broadcasting: from theory to practice," in *IEEE Infocom*, Barcelona, Spain, Apr. 2006.
- [21] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network Coding," *IEEE Trans. on Information Theory*, vol. 49, no. 2, Feb. 2003.
- [22] J. Widmer, C. Fragouli, and J.-Y. L. Boudec, "Low-complexity energy-efficient broadcasting in wireless ad-hoc networks usign network coding," in *IEEE Information Theory Workshop*, San Antonio, TX, US, Oct. 2004.