



**HAL**  
open science

# A Framework for Evaluating Physical-Layer Network Coding Gains in Multi-hop Wireless Networks

Raphaël Naves, Hicham Khalife, Gentian Jakllari, Vania Conan, André-Luc Beylot

► **To cite this version:**

Raphaël Naves, Hicham Khalife, Gentian Jakllari, Vania Conan, André-Luc Beylot. A Framework for Evaluating Physical-Layer Network Coding Gains in Multi-hop Wireless Networks. IEEE International Conference on Computer Communications (INFOCOM 2017), IEEE CommSoc, May 2017, Atlanta, United States. pp.1-9, 10.1109/INFOCOM.2017.8057187 . hal-03003820

**HAL Id: hal-03003820**

**<https://hal.science/hal-03003820>**

Submitted on 17 Nov 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Framework for Evaluating Physical-Layer Network Coding Gains in Multi-Hop Wireless Networks

Raphaël Naves<sup>1</sup>, Hicham Khalifé, Gentian Jakllari<sup>1</sup>, Vania Conan, and André-Luc Beylot

**Abstract**—Physical-Layer Network Coding (PLNC) was first introduced as a solution to increase the throughput of a two-way relay channel communication. Unlike most wireless communication techniques which try to avoid collisions, PLNC allows two simultaneous transmissions to a common receiver. In basic topologies, this technique has been shown to significantly enhance the throughput performance compared to classical interference-free communications. However, quantifying the impact of PLNC in large multi-hop networks remains an open question. In this work, we introduce the first theoretical framework that, given a particular network topology and traffic matrix, can evaluate the optimal network throughput subject to a fairness constraint relative to the initial demand of each flow, when PLNC is adopted. Based on linear programming, our solution ensures to respect the particularities imposed by PLNC. We use this framework to evaluate three state-of-the-art PLNC schemes across a variety of topologies and traffic matrices. Our numerical analysis reveals that while in simple toy topologies PLNC can significantly increase the network throughput, in large topologies the verdict is mixed. For certain topologies and traffic patterns adopting PLNC can double the throughput while in others, depending on the scheme used for implementing PLNC, the gain can be as high as 60 percent or as low as 0 percent when compared to traditional interference-free transmissions.

**Index Terms**—Physical-layer network coding, interference management, scheduling

## 1 INTRODUCTION

THE constant increase in the number of devices as well as the exponential growth in the exchanged data volumes render the future networks highly dense in terms of nodes and conveyed traffic. In a wireless environment, this density challenges today's wireless communication techniques and requires new interference mitigation strategies.

Physical-Layer Network Coding (PLNC) [2], [3] can reduce the interference by allowing multiple concurrent transmissions simultaneously. Consider the two-way relay channel (TWRC) depicted in Fig. 1 where nodes  $A$  and  $B$  want to send a packet to each other, say  $p_1, p_2$ , through a relay  $R$ . Traditionally, we assume that nodes  $A$  and  $B$  cannot transmit simultaneously to  $R$ . Hence, 4 interference-free transmissions are necessary for the packets to reach their respective destinations, which orthogonalized in time would translate to 4 time slots (Fig. 1a). The traditional

packet-based Network Coding (NC) [4] already showed how to reduce the number of slots to 3: The relay, instead of transmitting  $p_1$  and  $p_2$  to their respective destinations in two orthogonal time slots, could send a single packet,  $p_1 \oplus p_2$ . Node  $A$  (similarly  $B$ ), having kept  $p_1$  in its memory, can XOR it with the coded packet it receives to recover its intended packet,  $p_2$  (Fig. 1b). However, NC was designed for wired networks and, as has been the custom, its first adaptation to wireless networks continued with the tradition of wireless interference-free transmissions [5].

Then, Zhang et al. [2] realized that the wireless medium could actually accelerate Network Coding (Fig. 1c). They introduced Physical-Layer Network Coding in which nodes  $A$  and  $B$  can transmit simultaneously their packets to the relay. By processing the superimposed received signals,  $R$  receives a linear combination of the two transmitted packets. In particular, at the bit level,  $R$  retrieves the XOR-ed packet  $p_1 \oplus p_2$ , which it forwards to nodes  $A$  and  $B$  in the second time slot. Upon reception, they can extract their respective message by removing their contribution from the received coded packet. As a result, PLNC improves throughput by 100 and 50 percent compared to interference-free transmissions and the basic Network Coding, respectively. This gain, however, comes at the price of strict synchronization to guarantee the reception of perfectly aligned signals at the relay.

Despite its novelty, PLNC does not fundamentally alter the nature of wireless communications. Even if nodes  $A$  and  $B$  may transmit exactly at the same time to  $R$ , when  $R$  is

- R. Naves is with Thales Communications & Security, Gennevilliers CEDEX 92622, France, and the Department of Telecommunications and Networks, ENSEEIHT, University of Toulouse, Toulouse 31000, France. E-mail: raphael.naves@thalesgroup.com.
- H. Khalifé and V. Conan are with Thales Communications & Security, Gennevilliers CEDEX 92622, France. E-mail: hicham.khalifethalesgroup.com, vania.conanthalesgroup.com
- G. Jakllari and A.-L. Beylot are with the Department of Telecommunications and Networks, ENSEEIHT, University of Toulouse, Toulouse 31000, France. E-mail: {jakllari, beylot}@enseeiht.fr.

(Corresponding author: Raphaël Naves)

1.  $\oplus$  designates the XOR operation.

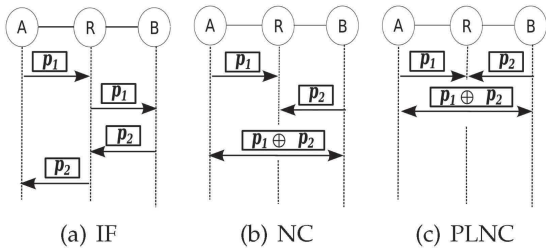


Fig. 1. Two-way relay channel (TWRC) scenario with (a) traditional interference-free transmissions, (b) packet-based network coding, and (c) physical-layer network coding.

transmitting, no one else in the interference range of  $A$  and  $B$  should transmit or it will result in the usual collision. Thus, PLNC is to be applied opportunistically and requires coordination between the nodes involved. In this paper, we refer to this ad-hoc coordination as a PLNC realization scheme. The two-way relay channel, we used for illustrating PLNC, is the canonical example of a PLNC realization scheme but as we show in Section 6 not necessarily the only one. Using toy topologies built around these schemes, PLNC is shown to multiply the network throughput. What is less clear, however, is how does this benefit translate in larger networks, which do not necessarily exhibit a particular topology or traffic pattern. Some research initiatives have tried to investigate the performance of Physical-Layer Network Coding in multi-hop networks, still, the focus was on small topologies, using basic PLNC modes [6]. To our knowledge, no previous work has provided a theoretical approach for quantifying the gains PLNC can produce in large multi-hop networks.

In this paper, we present the first theoretical framework that, given a particular network topology and traffic matrix, can compute lower and upper bounds on the optimal network throughput subject to fairness constraints when Physical-Layer Network Coding is implemented. Our framework implements the widely studied XOR-based decoding PLNC form and integrates 3 different schemes for combining received signals. In designing this framework, we addressed three major challenges: First, we introduced a conflict graph that captures the particularities PLNC introduces when it comes to modeling wireless interference (Section 4.2). Second, we integrated the constraints resulting from the PLNC-aware conflict graph into a linear program inspired by the max-flow formulation (Section 4.3). Third, with the problem of computing the optimal throughput being NP-Hard, we introduced an approach for estimating the optimal value by computing lower (Section 4.4) and upper (Section 4.5) bounds.

We implemented our framework in MATLAB and evaluated the performance of three state-of-the-art PLNC realization schemes on a variety of traffic matrices and network topologies, ranging from typically-used synthetic topologies to those based on real-life traces. Our results show that, while Physical-Layer Network Coding can multiply throughput when compared to interference-free transmissions in simple topologies, in large networks the verdict is mixed. Depending on the specific PLNC realization scheme, the network topology and the traffic matrix, the gain in throughput over traditional interference-free transmission can be as high as 60 percent or as low as 0 percent. Our work can, thus, guide protocol designers when conceiving and evaluating MAC and routing solutions for PLNC.

The remainder of the paper is structured as follows: In Section 2, we give an overview on previous work on throughput computation and Physical-Layer Network Coding. We then detail in Section 3 our system model and considered assumptions that are used in Section 4 to define the linear programs constituting our framework. Section 5 shows the throughput gains achieved with the TWRC model whereas an extension of this model to two advanced Physical-Layer Network Coding schemes is highlighted in Section 6. Section 7 presents a thorough performance evaluation of the three state-of-the-art PLNC realization schemes followed by a description of the key lessons learned in Section 8. Section 9 concludes this paper and describes future research perspectives.

## 2 RELATED WORK

### 2.1 Scheduling and Throughput Calculation

During the last decade, scheduling concurrent transmissions in wireless networks have attracted a lot of attention from the research community. Indeed, a smart medium sharing between nodes is essential to ensure an efficient bandwidth utilization. But, this problem is known to be NP-complete with traditional models of transmissions [7]. However, Jain et al., work in [8] is considered as an important breakthrough in this area. In particular, the paper proposes a framework to precisely evaluate the end-to-end throughput of any concurrent unicast transmissions in any topology, using implicitly an optimal scheduling. Even if the solution does not define the scheduling to adopt, the work constitutes a strong basis to determine the theoretical achievable performance of any wireless network. In their approach, the authors exploit the well-known protocol model (based only on distance) and physical model (based on distance and received signal strength) as interference models which ensure that each destination of a unicast transmission receives distinctly its intended packet. They thus model classical access techniques whose role is to avoid collision of packets. Integrating to an optimization problem the set of transmissions that can be scheduled in the same slot without interfering, an achievable lower bound on the throughput can then be computed. When all these different sets of links are found, this lower bound converges to the upper bound therefore defining the maximum achievable throughput. Clearly, the main challenge is to account for all these possible set of links what constitutes an NP-hard problem. However, authors claim that only few minutes are required to compute the achievable maximum throughput of any reasonable size networks (more than 20 nodes).

More recently, authors in [9] extended the previous optimization problem with the capability to illustrate the classical (packet-based) Network Coding gains. To this aim, the model authorizes broadcast communication schemes. The protocol model is also used to ensure a sufficient Signal-to-Noise-Ratio (SNR) at all receivers of broadcast communications. Two ways of using Network Coding are considered: with and without opportunistic listening. Interestingly, authors highlight that more than 30 percent gain in throughput can be achieved in certain conditions compared to simple unicast transmissions. These results confirm the intuitive idea that Network Coding, which was originally introduced for small topologies, offers promising performance in large ad-hoc networks. More importantly, the described framework offers a tool to precisely compute the

maximum throughput per unicast flow, in any considered topology in the presence of network coding.

## 2.2 Physical-Layer Network Coding in Large Multi-Hop Networks

Although it has been proved that Physical-Layer Network Coding does not significantly impact the complexity of the scheduling problem [10], the gains it offers especially in large multi-hop and realistic topologies is often unknown. Indeed, if the performance of PLNC no longer has to be proved in small topologies [6], the efficiency of this technique in large multi-hop networks remains an open question. To the best of our knowledge, no study has been conducted to accurately evaluate and quantify potential gains of PLNC in realistic ad-hoc networks. It has been showed in [11] and [12] that PLNC does not change the scaling law derived by Gupta and Kumar in [13]. More precisely, with an infinity of flows in the network, the per node throughput is improved with a fixed factor but converges also asymptotically to zero with PLNC. Undoubtedly, this result is interesting however, it does not characterize the theoretical benefits of PLNC in realistic radio deployments. Clearly, this information is crucial to identify the topologies where PLNC can be applied and where it cannot.

More importantly, PLNC techniques have not been generalized to realistic multi-hop networks. Most of the studies focus on the TWRC scheme previously described. This observation can be made on both theoretical papers and implementation solutions [14], [15], [16], [17]. The impressive gains that PLNC allows in this configuration encouraged researchers to integrate this two-hop scenario in large ad-hoc networks. In [18] and [19] for instance, access methods are especially designed to detect favorable conditions to set up a PLNC transmission when three nodes are selected in a way to form a TWRC scheme. Even if a global throughput increase can be observed with some particular traffic patterns, one can wonder whether the Physical-Layer Network Coding is exploited to its maximum potential. Indeed, the TWRC scenario fits perfectly to two-hops networks with bi-directional traffic, but is it still the case for large ad-hoc networks with concurrent multi-hop unicast flows?

## 2.3 Beyond the TWRC Scenario

As highlighted in [3], other more ambitious PLNC models are possible. In fact, very recently, the work proposed in [20] and [21] integrates new PLNC schemes. In particular, authors combine PLNC and the opportunistic listening technique that wireless medium allows, to improve the performance of the network. What is more, the obtained results in the realistic deployments of [20], confirm the intuitive idea that different PLNC strategies can be more efficient in certain conditions than the basic TWRC model. Similarly, the multi-way relay channel in which multiple users exchange information with the help of a relay terminal leverages the PLNC concept and may offer better performance than the TWRC scheme [22], [23]. However, few higher layer protocols try to integrate this PLNC exploitation in real deployments.

In addition to these PLNC schemes, only efficient in presence of crossing flows, other ways to benefit from the PLNC concept in other traffic conditions were proposed. First, Katti et al. exploit the nodes ability to receive two simultaneous

signals to increase the transmission rate in any multi-hop line topology longer than 2 hops [5]. In the same vein, in [24] and [25] the authors leverage the PLNC concept for a scenario in which 2 source nodes aim to transmit 1 packet to each of the 2 destinations through 2 potential relays. The proposed scheduling policy allows the sources to send simultaneously their packets to the relays. Thanks to a collaborative mapping selection based on game theory, the relays then forward successively to the destinations two independent linear combinations of the native packets. More generally, a classical multi-sources multi-relays scenario is investigated in [26] and [27] whereby  $N$  sources send  $N$  different packets to the same destinations through  $N$  different relays. In this many-to-many communication pattern, it is shown that PLNC requires  $N + 1$  time slots to transmit the  $N$  packets to the destinations compared to the  $2N$  slots required with traditional interference-free transmissions. However, this PLNC mechanism is different from the ones investigated in this paper since it relies on compute-and-forward relaying [26]. In fact, this approach exhibits numerous differences with the XOR-based decoding technique we refer to in this work. First, compute-and-forward requires the utilization of lattice codes like LDLC (Low-Density Lattice Codes) [28] with special modulation alphabet and is hence incompatible with most of modulations used today (BPSK, QPSK, . . .). Second, as shown in [29], compute-and-forward is generally associated to lattice shaping to control the power of transmitted lattice codewords which is difficult to model being very inherent to each implementation. Last but not least, the decoding process complexity at relays cannot be ignored, and may explain the relatively small number of compute-and-forward PLNC implementations over real radio devices.

Consequently, in this work, we aim to evaluate the gains of 3 representative PLNC schemes based on a XOR decoding at relays: the original TWRC scheme, the Butterfly scheme which is a generalization of the TWRC scenario and the Intra-Flow PLNC which does not need any crossing flows to be applied. Nevertheless, we believe that our framework constitutes a practical tool for other PLNC schemes evaluation.

## 3 MODEL & PRELIMINARIES

We consider a multi-hop wireless topology where unicast packets are exchanged between a source and a destination. All nodes are assumed to be identical, serving as source, relay, or destination. If the source and the destination are not direct neighbors, the frame is routed through other nodes; the details of the routing protocol are orthogonal to our work. Finally, we assume a TDMA-based access control.

### 3.1 PLNC Interference Model

There are two widely used models for characterizing wireless interference: the protocol model, assuming wireless interference is a 0-1 step function of distance, and the more realistic physical model, which relies on the SNR [8], [9], [13].

Despite their differences, both models are based on the assumption that a transmission between two nodes cannot be successful if other nodes close enough to the receiver are also transmitting. Obviously, this assumption is not necessarily true with Physical-Layer Network Coding, which as depicted in Fig. 1c, allows two simultaneous transmissions

to a same receiver. Therefore, in the following we extend the physical model so as to characterize interference in wireless networks implementing PLNC.

Let  $SNR_S(D)$  be the SNR at node  $D$  of a signal sent by node  $S$ . Using PLNC, node  $D$  can handle the simultaneous reception of two signals from two different nodes,  $S_1$  and  $S_2$ , if the two following conditions are satisfied:

$$\begin{cases} SNR_{S_1}(D) = \frac{P(S_1, D)}{\sum_{\substack{k \neq 1 \\ k \neq 2}} P(S_k, D) + N} > \beta \\ SNR_{S_2}(D) = \frac{P(S_2, D)}{\sum_{\substack{k \neq 1 \\ k \neq 2}} P(S_k, D) + N} > \beta, \end{cases} \quad (1)$$

where  $\beta$  is the minimum SNR for a successful reception,  $P(S, D)$  the received power at node  $D$  of signal sent by  $S$ , and  $N$  the ambient noise. We assume that the transmitted signal is attenuated by a constant factor  $\frac{1}{d^\alpha}$ , where  $d$  is the distance between the sender and the receiver and  $\alpha$  is the path-loss exponent. Typical values for the path-loss exponent vary between 2 and 6, depending on the propagation environment (obstructions, atmospheric conditions, ...). According to this model, for  $D$  to decode the superimposed signals of  $S_1$  and  $S_2$  both have to be individually received with a sufficient SNR.

Note, however, that the destination is not able to decode individually the packets from nodes  $S_1$  and  $S_2$ . When a PLNC reception is performed, the receiver only retrieves the XOR-ed packet of the two transmitted messages. This constraint is a key point in the rest of the paper.

### 3.2 Notation and Modeling Parameters

We model the wireless network as a graph  $G = (N, E)$ , where  $N$  is the set of nodes and  $E$  the set of links. A node  $n_i$  can send a packet to a node  $n_j$  if they are linked with a directed edge  $e_{ij} \in E$ .

We consider in the studied networks a set of unicast flows  $\mathcal{F}$ . Each flow  $f \in \mathcal{F}$  is defined by four parameters:

- (1) A source  $s(f)$ .
- (2) A destination  $d(f)$ .
- (3) A set of paths between  $s(f)$  and  $d(f)$  denoted  $\mathcal{P}(f)$ .
- (4) A traffic demand  $r(f)$ .

We assume multi-path routing and, thus, assign to each link  $e_{ij} \in E$  a "flow-path" identifier. We denote by  $l_{ij}^{f,p}$  the amount of data of flow  $f$ , on link  $e_{ij}$  of path  $p$ . Obviously,  $l_{ij}^{f,p}$  cannot exceed  $C_{ij}$ , the capacity of link  $e_{ij}$ .

The traffic demand  $r(f)$  represents the amount of data that a source wants to send for flow  $f$ . As for  $l_{ij}^{f,p}$ , this value is expressed in frames per slot. In this work, we evaluate the PLNC performance under saturation conditions, therefore we consider in the rest of this paper that each node tries to send as many packets as possible, meaning that:

$$\forall f \in \mathcal{F}, r(f) = 1. \quad (2)$$

Nevertheless, this parameter may easily be changed in our framework.

## 4 BASIC FRAMEWORK DESCRIPTION

In this section, we introduce a theoretical framework that can compute the maximum achievable rate in a multi-hop wireless networks where stations implement the physical-layer network coding.

In a wired network using multi-path routing, finding the maximum achievable throughput between a source and destination can be easily formulated as a linear program corresponding to the max-flow problem. The challenge, however, in extending this basic framework to wireless networks resides in incorporating interference. A commonly used approach [8] is to make use of a conflict graph—a virtual graph where two vertices, corresponding to two communication links in the real world, are connected by an edge if and only if the respective links mutually interfere. The problem, then, essentially reduces to computing the maximum independent set—the set of vertices such that no two of which share an edge—of the conflict graph. A well-known NP-hard problem even to approximate.

To make matters worse, with PLNC, the nature of interference changes, making the definition of a conflict graph non-trivial. Therefore, using an approach based on the max-flow problem necessitates: 1) introducing a new transmission graph (Section 4.1) and a new conflict graph (Section 4.2) that captures the particularities PLNC introduces in wireless networks, 2) modifying the linear program accordingly (Section 4.3), and 3) introducing a heuristic for computing lower (Section 4.4) and upper (Section 4.5) bounds on the achievable throughput.

In the following, we rely on the TWRC scheme to introduce our framework, which, with simple modifications, can be applied to the Butterfly (Section 6.1) and Intra-flow PLNC schemes (Section 6.2).

### 4.1 PLNC-Aware Transmission Graph

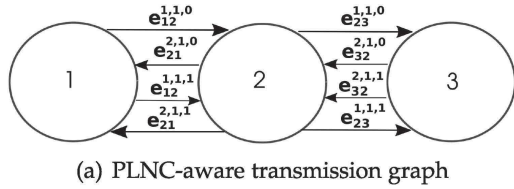
To illustrate why the nature of interference does change when using PLNC, recall the two-way relay channel depicted in Fig. 1c. The edge nodes,  $A$  and  $B$ , can transmit simultaneously to the relay,  $R$ , in contrast with traditional wireless networks, leading to the reception of a XOR-ed coded packet. However, when  $R$  transmits the coded packet, no other station in its interference domain should transmit, similar to non-PLNC networks. So, to the question of if two stations interfere with each other, we can only respond, it depends.

To take this effect into account, we define a new PLNC aware *transmission* graph, from which we can then derive the conflict graph. In the new transmission graph, every link is first classified by whether it belongs to a TWRC scheme or not, second based on the path and flow it belongs to. More formally, a TWRC element is characterized by the following:

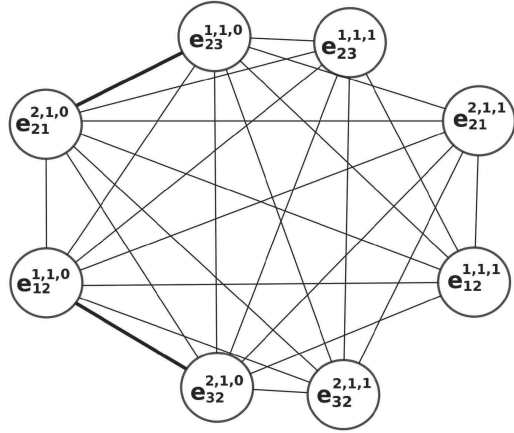
- (1) Two (flow, path) components
- (2) A source node for each (flow, path) component which is also the destination of the other component.
- (3) A relay node

Thus, while we are accustomed to writing  $e_{ij}$  for a link between stations  $i$  and  $j$ , with the new transmission graph we write  $e_{ij}^{f,p,T}$ . Where  $T$  denotes the TWRC scheme to which the link belongs (0 if it belongs to none),  $f$  denotes the traffic flow and  $p$  the path.

Fig. 2a shows the transmission graph associated with the TWRC. Flow 1 (resp. flow 2) represents the traffic between nodes 1 and 3 (resp. nodes 3 and 1). The 4 arrows with  $T = 0$  identify "traditional" transmissions of non-coded packets. The 4 other arrows designate transmissions involved in the TWRC scheme i.e., having  $T = 1$ . Typically, the scheduling rule prevents links  $e_{12}^{1,1,0}$  and  $e_{32}^{2,1,0}$  to be active simultaneously because the corresponding transmissions have to respect the traditional physical model. However,  $e_{12}^{1,1,1}$  and  $e_{32}^{1,1,1}$  can



(a) PLNC-aware transmission graph



(b) PLNC-aware conflict graph - Binary version

Fig. 2. (a) PLNC-aware transmission graph and the (b) resulting two-part conflict graph.

be scheduled at the same time as they belong to the same TWRC element.

## 4.2 PLNC-Aware Conflict Graph

The PLNC-aware conflict graph is built by extending the classical conflict graph with vertices given by the links of the new transmission graph  $G$ . Indeed, it includes both traditional, interference-free links and those belonging to TWRC elements, as depicted in Fig. 2b. As highlighted in this simplified binary version of the conflict graph, the two pairs of vertices  $(e_{12}^{1,1,0}, e_{32}^{2,1,0})$  and  $(e_{21}^{1,1,0}, e_{23}^{2,1,0})$  are connected with edges since the corresponding transmissions cannot be scheduled simultaneously. On the contrary,  $(e_{12}^{1,1,1}, e_{32}^{1,1,1})$  and  $(e_{21}^{1,1,1}, e_{23}^{1,1,1})$ , belonging to a TWRC scheme, may be active at the same time respectively and therefore the corresponding vertices in the conflict graph are not connected.

It should be noticed that the PLNC-aware conflict graph is built from the physical model for interference-free transmissions and from the PLNC interference model (Eq. (1)) defined for PLNC transmissions. Except for transmissions sharing a common node as in the small topology of Fig. 2a, the used conflict graph is a weighted conflict graph rather than a binary conflict graph. The weight of a directed edge between two vertices  $e_{ij}^{f,p,T}$  and  $e_{kl}^{f',p',T'}$  represents the amount of noise induced by the transmission  $e_{ij}^{f,p,T}$  at node  $l$ . Then, a transmission may be active only if the corresponding SNR computed with the sum of the incoming weights from all other active links satisfies the physical model or the PLNC interference model, depending on the nature of the transmission.

## 4.3 Constrained Optimization LP Formulation

To evaluate the achievable gain of Physical-Layer Network Coding in large multi-hop networks, we formulate it as a constrained optimization problem. In particular, we introduce the reduction factor  $\lambda$  ( $\lambda \in [0, 1]$ ) as the maximum value by which demands can be multiplied with the guarantee that

TABLE 1  
Summary of Used Notations

Framework input	
$\mathcal{N}$	Set of nodes of the network
$\mathcal{F}$	Set of established flows
$\mathcal{P}(f)$	Set of paths of flow $f$
$s(f), d(f)$	Source and destination of flow $f$
$r(f)$	Traffic demand of flow $f$ (frame/slot)
$C_{ij}$	Maximum amount of data that node $n_i$ can send to node $n_j$ (frame/slot)
Independent sets/Cliques discovery output	
$\mathcal{T}$	Set of TWRC elements in the network
$\mathcal{T}(f, p)$	Set of TWRC elements implicating flow $f$ and path $p$
$T$	TWRC identifier
$IN(T, n_i),$	Boolean functions which returns 1 or 0 depending on node $n_i$ is the $IN, OUT, RELAY$ of $T$ or not
$OUT(T, n_i),$	
$RELAY(T, n_i)$	
$e_{i,j}^{f,p,T}$	Link associated to the transmission of flow $f$ , path $p$ , involved in TWRC element $T$ , from node $n_i$ to node $n_j$
$I_i$	Set of links which can be scheduled simultaneously
$K$	Number of different sets $I_i$
Linear Program output	
$\lambda$	Reduction factor of the traffic demands
$q_i$	Activity period of set $I_i$
$l_{i,j}^{f,p,T}$	Amount of data through link $e_{i,j}^{f,p,T}$

all the demands are satisfied. In other words, we aim to maximize  $\lambda$  in such a way that exists a scheduling policy for which each flow may transmit  $\lambda \times r(f)$  frame per slot at minimum.

With this formulation, unlike the objective function proposed in [8] which aims at maximizing the sum of the throughputs of all the flows at the risk of starving some of them, we ensure that each source transmits a non-zero amount of data. In addition, since all flows suffer the same throughput reduction relative to their initial demand  $r(f)$ , this constraint may be seen as a fairness constraint in the sense that no flow is privileged over the others. However, we do not aim for a strict fairness between flows, in which each source would transmit at the same rate, but rather derive a fairness constraint that considers the initial needs of each flow. Finally, assuming that all the demands are equal (Eq. (2)), the proposed objective function is equivalent to a max-min fairness solution since our linear program aims to maximize the minimum throughput reduction factor.

Denoting with  $I_i$  a set of links that can be scheduled simultaneously (independent set)—computed on the PLNC-aware conflict graph— $q_i$  represents the fraction of time during which  $I_i$  is active.  $K$  is the number of all  $I_i$  sets in the conflict graph. Table 1 summarizes all the notations used in the formulation that follows.

$$\begin{aligned} & \text{maximize } \lambda \\ & \text{subject to} \\ & \text{Constraint 1: Throughput computing based on injected packets} \end{aligned}$$

$$\begin{aligned} & \forall f \in \mathcal{F} \\ & \sum_{p \in \mathcal{P}(f)} l_{s(f),j}^{f,p,0} + \sum_{p \in \mathcal{P}(f)} \sum_{T \in \mathcal{T}(f,p)} IN(T, s(f)) \times l_{s(f),j}^{f,p,T} \\ & = \lambda \times r(f) \end{aligned}$$

**Constraint 2:** Flow conservation for native packets

$$\begin{aligned} \forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall n_i \in \mathcal{N}(f, p) \setminus \{s(f), d(f)\} \\ l_{i,j}^{f,p,0} + \sum_{T \in \mathcal{T}(f,p)} IN(T, n_i) \times l_{i,j}^{f,p,T} \\ = l_{h,i}^{f,p,0} + \sum_{T \in \mathcal{T}(f,p)} OUT(T, n_i) \times l_{h,i}^{f,p,T} \end{aligned}$$

**Constraint 3:** Flow conservation for coded packets

$$\begin{aligned} \forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall T \in \mathcal{T}(f, p), \\ \forall n_i \in \mathcal{N}(f, p) \setminus \{s(f), d(f)\} \\ RELAY(T, n_i) \times l_{i,j}^{f,p,T} = RELAY(T, n_i) \times l_{h,i}^{f,p,T} \end{aligned}$$

**Constraint 4:** Bounds on the sent amount of data

$$\begin{aligned} \forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall T \in \mathcal{T}(f, p) \cup 0, \forall n_i \in \mathcal{N}(f, p) \\ \begin{cases} l_{i,j}^{f,p,T} \leq C_{ij}^T \\ l_{i,j}^{f,p,T} \geq 0 \end{cases} \end{aligned}$$

**Constraint 5:** Activity periods constraint

$$\sum_{i=1}^K q_i \leq 1$$

**Constraint 6:** Sent amount of data constraint

$$\forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall T \in \mathcal{T}(f, p) \cup 0, \forall n_i \in \mathcal{N}(f, p) \\ l_{i,j}^{f,p,T} \leq \sum_{k | l_{i,j}^{f,p,T} \in I_k} q_k C_{ij}^T$$

The first term of constraint 1 represents the amount of data sent by each traffic-flow source along the different paths. We aim to maximize this value ensuring fairness. A source can transmit a packet to the next hop in two different ways. A packet can be transmitted avoiding interference at the receiver ( $T = 0$ ). The amount of data sent this way is expressed in the first term of the sum ( $l_{s(f),j}^{f,p,0}$ ). Or, a packet can be involved in a PLNC transmission of TWRC element  $T$ , if  $s(f)$  is the source of this element. The sum  $\sum_{T \in \mathcal{T}(f,p)} IN(T, s(f)) \times l_{s(f),j}^{f,p,T}$  represents the amount of data involved in PLNC transmissions of all TWRC elements whose  $s(f)$  is the source.

Constraint 2 represents the flow conservation for native packets of each flow. Each relay has to receive or decode the same number of native packets it transmits. There are two ways of decoding native packets. The relay is the destination of traditional transmissions of native packets. The received amount of data of traditional transmissions is  $l_{h,i}^{f,p,0}$ . Or, the relay can be the ‘‘Out node’’ of a TWRC element. The amount of coded data received and decoded, by node  $n_i$ , after the TWRC element  $T$  is  $OUT(T, n_i) \times l_{h,i}^{f,p,T}$ . Then, as in constraint 1, each decoded native packet has to be forwarded in a classical interference free transmission ( $l_{i,j}^{f,p,0}$ ) or in a PLNC transmission in  $T$ , if the relay is the source of this TWRC element ( $IN(T, n_i) \times l_{i,j}^{f,p,T}$ ).

Constraint 3 ensures flow conservation for each coded packet, at each TWRC relay. Each TWRC relay has to forward every packet obtained from a PLNC reception to ensure that the two ‘‘Out nodes’’ receive and decode their intended packet. Briefly speaking, constraints 2, 3, combined with scheduling rules, ensure the end-to-end flow

conservation on one hand and that each flow destination decodes native packets sent by the source on the other hand.

Constraints 4, 5 and 6 are more standard. Constraint 4 ensures that the amount of data through each link is positive and lower than the capacity of the link. Extra care must be taken here due to the subtleties introduced by PLNC. When a node transmits a coded packet, intended for two destinations, the capacity is bounded by the lowest capacity of the two links involved. For instance, if nodes  $n_j$  and  $n_k$  are part of the same TWRC unit  $T$  with node  $n_i$  as a relay, we have  $C_{ij}^T = C_{ik}^T = \min(C_{ij}, C_{ik})$ .

Each independent set,  $I_i$ , is active during a period  $q_i$ . Obviously, the sum of these fractions of time cannot exceed one, since sets are active sequentially one after the other (constraint 5). Finally, the amount of data on each link  $l_{i,j}^{f,p,T}$  is limited by the fraction of time during which the link is active and its capacity (constraint 6). This value corresponds to the sum of periods during which a set of links  $I_i$  containing the link  $l_{i,j}^{f,p,T}$  is active.

#### 4.4 Lower Bound on Throughput

With the problem of finding the maximum independent set  $I_i$  NP-hard and, consequentially, the problem of computing the optimal throughput being NP-hard [8], we have to resort to heuristics for obtaining lower and upper bounds on the optimal throughput.

To compute a lower bound, we compute a number of maximal independent sets, the value of  $K$  in constraint 5 and then run the linear program to get a value for the throughput. Starting with  $K = 1$ , we repeat the process for increasing values of  $K$  until the throughput value we get is close enough to the upper bound.

To compute a maximal independent set, all vertices of the PLNC-aware conflict graph are ranked in a random order. Starting with the empty set, a vertex is added using this order if it does not interfere with any of the vertices already in the set. Note that, the notion of interference is based on the physical model (see Eq. (1)). This is repeated until no more vertices can be added.

#### 4.5 Upper Bound on Throughput

To derive an upper bound, we rely on the concept of clique (the reverse of the independent set)—the subset of the conflict graph such that any two vertices are connected by an edge. Let  $\mathcal{C}$  be the set of cliques on our PLNC-aware conflict graph and  $q_{ij}^{f,p,T}$  the period of activity of link  $l_{ij}^{f,p,T}$ . As two vertices being on the same clique corresponds to the respective physical links not being able to be active simultaneously, we obtain the following constraint:

**Constraint 7:** Cliques constraint

$$\forall \mathcal{C} \in \mathcal{C}, \quad \sum_{e_{ij}^{f,p,T} \in \mathcal{C}} q_{ij}^{f,p,T} \leq 1.$$

**Constraint 8:** Sent amount of data constraint

$$\begin{aligned} \forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall T \in \mathcal{T}(f, p) \cup 0, \forall n_i \in \mathcal{N}(f, p) \\ l_{i,j}^{f,p,T} \leq q_{ij}^{f,p,T} C_{ij}^T \end{aligned}$$

Replacing constraints 5 and 6 by 7 and 8, we can derive an upper bound on the throughput. We tighten this bound

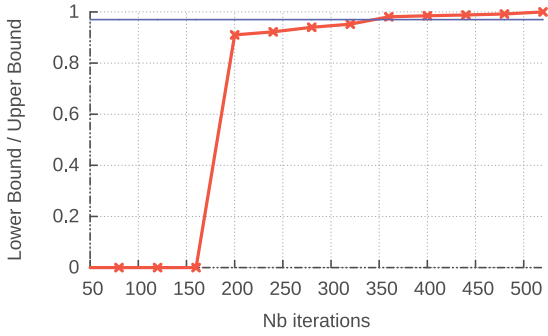


Fig. 3. Lower/upper bound ratio evolution during a simulation.

by finding all the cliques of the conflict graph, with an algorithm similar to the one for finding independent sets.

Added to constraints 2 and 3, the particular ways of finding cliques and independent sets in the PLNC-aware conflict graph highlight the influence of applying PLNC to the optimization problem.

#### 4.6 Convergence Discussion

In Practice, the lower and upper bounds computed respectively thanks to the iterative independent sets and cliques discovery with linear programs may take long time to converge to the same value. This observation is even more obvious when the number of links of the graph increases (note that they may not converge at all even if all the independent sets and cliques are found). For this reason, we track the  $\frac{\text{lower bound}}{\text{upper bound}}$  ratio evolution during a PLNC throughput computation. As an illustrative example, we consider the chain topology of Fig. 5 in the presence of 10 random flows. Results are exhibited in Fig. 3, where the  $x$ -axis depicts the number of iterations of the linear program after an independent set and clique discovery, while the  $y$ -axis shows the upper to lower bound ratio. One can see that these two bounds take several time/iterations to converge to the same value ( $\frac{\text{lower bound}}{\text{upper bound}} = 1$ ). Nevertheless, almost one third of the bounds computation duration is for closing the very last 5 percent of the computed ratio.

In the rest of the paper, for performance and efficiency considerations, we do not necessary wait until the lower bound equals the upper bound to consider that the optimal throughput is reached. More precisely, we stop the independent sets and cliques discovery once the ratio  $\frac{\text{lower bound}}{\text{upper bound}}$  is more than a decided threshold fixed at 0.97. Finally, in the very few cases where this value is not attained even after a large number of iterations (2000 in our framework), obtained results are simply removed from our evaluation.

### 5 EVALUATION OF TWRC-BASED PHYSICAL-LAYER NETWORK CODING

In this section, we start demonstrating the utility of our framework by evaluating the performance gains of implementing PLNC in multi-hop wireless networks using the TWRC scheme.

#### 5.1 Performance Gain

The PLNC performance gain is computed relative to interference-free transmissions. To avoid overestimating the PLNC gains, we use the throughput lower-bound values (Section 4.4) for PLNC and the throughput upper-bound

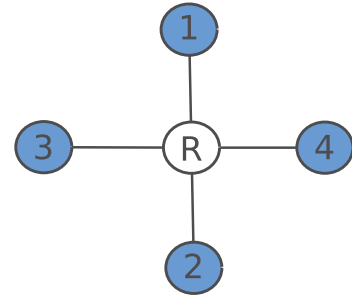


Fig. 4. Topology used for framework validation—four established flows (1→2, 2→1, 3→4, and 4→3).

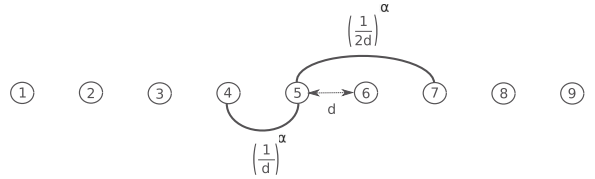


Fig. 5. Line topology - nine nodes.

values (Section 4.5) for interference-free transmissions. As a result, by choosing 0.97 as the convergence threshold, some of the results we show in the following may underestimate the PLNC gains by as much as 6 percent. For additional context, we also show the performance gains achieved with packet-based network coding (also relative to interference-free transmissions).

#### 5.2 Framework Validation—TWRC Scheme

To validate our framework, we select a toy topology (Fig. 4) with two bi-directional flows: (1, 2) and (3, 4).

With traditional interference-free transmissions only one node can be active at a time. A source can only send a frame every 8 slots – an optimal flow-preserving scheduling assigns 4 slots to R and 1 slot to each of the four source nodes. Thus, the maximum per-flow throughput is 0.125 frames per slot.

With packet-based network coding, each pair of bidirectional flows needs 3 slots to exchange one packet each (Fig. 1b). Thus, the maximum per-flow throughput is 1/6 frames per slot.

With PLNC, each source node can send a frame every 4 slots. In slot 1, nodes 1 and 2 transmit simultaneously a packet to R, which broadcasts it in slot 2. Similarly, in slot 3, nodes 3 and 4 can transmit simultaneously, followed by R. It is straightforward to see that the maximum throughput is doubled compared to interference-free transmissions with a (normalized) flow rate of 0.25 frame per slot. We applied our framework to this scenario, the computed lower bound and upper bound on throughput converge immediately and we obtained the exact value of 0.25.

#### 5.3 Gain Evaluation in a Multi-Hop Line Topology

The Fig. 4 topology, composed of 2 TWRC elements, undoubtedly benefits Physical-Layer Network Coding, explaining the factor-2 improvement. The difficult question, however, is, what gains are achievable in larger topologies with different traffic patterns. We start addressing this question using the 9-node chain topology depicted in Fig. 5, representative of a vehicle convoy, for instance. The distance between two consecutive nodes is constant. We first set the path loss exponent  $\alpha$  to 2 and choose a SNR threshold  $\beta$



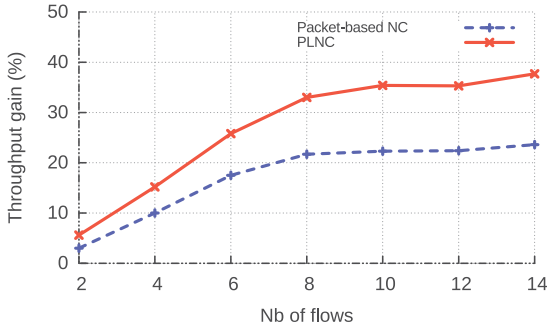


Fig. 6. Evaluation of PLNC gains in a multihop line topology.

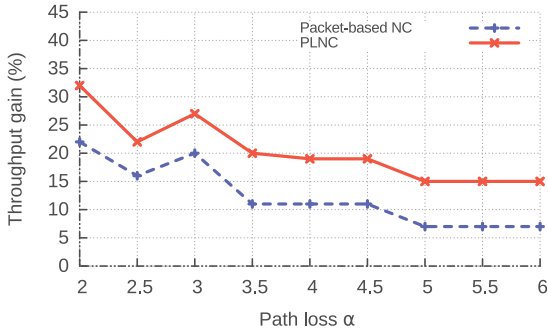


Fig. 7. Path loss exponent impact on PLNC gains.

equal to 15 dB. The link capacity between nodes  $i$  and  $j$  is equal to  $(\frac{1}{d_{ij}})^\alpha$ , where  $d_{ij}$  is the distance between the two nodes as illustrated in Fig. 5. The network traffic load is varied by changing the number of the traffic flows.

Fig. 6 shows the average gain of packet-based network coding and physical-layer network coding. For each number of traffic flows, results are obtained over 50 runs with randomly selected sets of established flows. The data points to two conclusions. First, PLNC clearly outperforms packet-based network coding. Second, while the gains are not as impressive as in the TWRC scenarios, PLNC is shown to improve throughput by as much as 35 percent when compared to interference-free transmissions.

We then investigate the impact of the path loss exponent  $\alpha$  on the achieved throughput gains. To do so, for the same 8 instantiated flows, we evaluate the gains of the 2 coding techniques while modifying the path loss  $\alpha$  value. Results depicted in Fig. 7 lead to many observations. First, the general trend is that the throughput gains decrease with  $\alpha$ . This is due to a better performance of interference-free transmissions with high  $\alpha$  values rather than a throughput decrease of the PLNC mode. Indeed, when  $\alpha = 6$  for instance, a better spatial reuse becomes possible with interference-free transmissions, meaning that nodes 1 and 4 can transmit simultaneously to nodes 2 and 3 respectively without interfering. Such configuration is not possible when  $\alpha = 2$ . Therefore, although PLNC throughput remains almost constant while varying  $\alpha$ , the interference-free throughput increases leading to lower relative PLNC gains. These gains stabilize in Fig. 7 when the maximal spatial reuse pattern is reached, however stay slightly in favor of PLNC. Second, PLNC gains remain almost stable compared to packet-based Network Coding. In other words, varying the path loss exponent affects evenly the 2 coding techniques.

#### 5.4 Communication with an Access Point

In this experiment, we consider a more evolved scenario, representative of mesh network deployments. As depicted

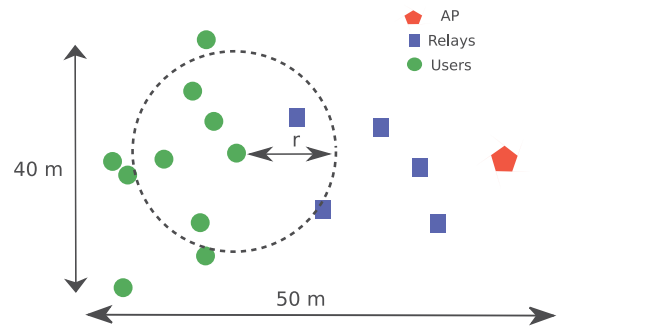


Fig. 8. Users attempting to communicate with an access point (AP) through potential relays.

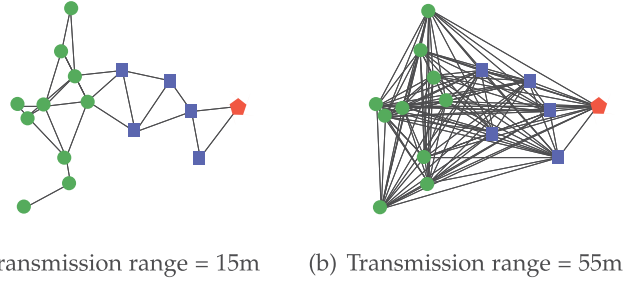


Fig. 9. Graph connectivity for different transmitted signal power.

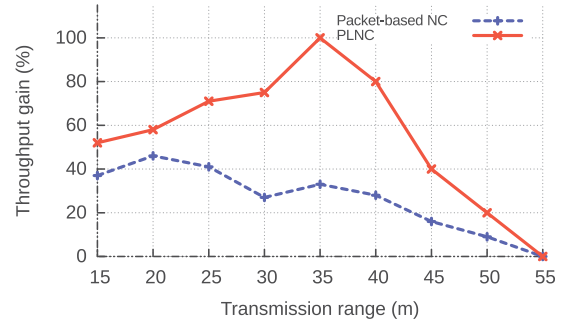


Fig. 10. Potential PLNC gains depending on the transmitted power.

in Figs. 8 and 10 user nodes exchange packets with the access point (AP) through 5 potential relays. The path loss and SNR threshold values are the same as in the initial line-topology experiment (Section 5.3). In the first round of simulations, we establish bidirectional traffic between each user node and the AP.

While ensuring the graph is connected, we vary the transmission range of each node (users, relays and AP) and compute the throughput gain of packet-based network coding and PLNC. For each transmission range, the connectivity between nodes is different leading to different graphs as shown in Fig. 9. We found that in our setting 15 m is the minimum range value for which the graph remains connected. Note that, a node may be out of the range of all the relays, in such case its packets are routed through other user nodes.

Fig. 10 shows that the PLNC gains are significant in this scenario. In particular, when the transmission range does not exceed a certain threshold, leading to 2, 3 or 4-hop flows, gains are always above 40 percent, reaching 100 percent when the transmission range is 35 m. In the latter case, all users are two hops away from the AP, leading to the creation of 10 TWRC elements. Finally, once the range becomes big enough, as shown in Fig. 9b, the topology turns into a clique, in which PLNC is not applicable, explaining the zero gain.

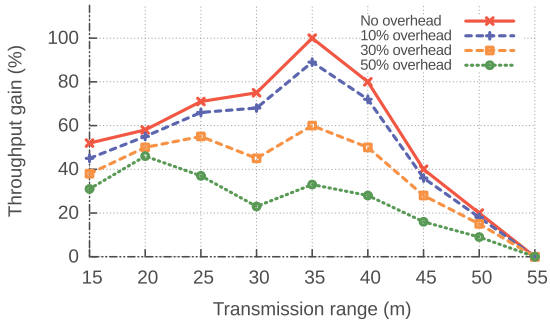


Fig. 11. Phy-overhead impact on the PLNC performance.

### 5.5 PHY-Layer Particularities Modelling

Until now, the proposed framework has mainly characterized MAC-layer aspects without taking into account the overhead introduced in order to support PLNC transmissions. Since many issues, as signal misalignment at destination, may seriously affect the PLNC performance, assuming that PLNC feasibility only depends on the received SNR conceals the real challenges encountered to perform such transmissions in practice.

As evoked in [15] and [16], in asynchronous networks, control information need to be exchanged before the TWRC sequence for achieving a tight synchronization between nodes. Moreover, some precoding techniques are performed in order to reduce the signal misalignment impact at destination. Even when nodes are synchronized in time thanks to a common clock (e.g., GPS), pilot sequences are sent successively by each source to enable channel estimation at the receiver [30]. All these PHY-layer techniques constitute an overhead for PLNC transmissions, reducing the theoretical gains promised at MAC level. Even if this overhead tends to be reduced in future PLNC implementations, its impact has to be integrated into our framework for the sake of fair comparison.

To this aim, we introduce a new parameter  $\gamma$  representing the percentage of additional control information required to allow PLNC transmissions. We then derive the capacity  $C_{ij}^T$  of a link involved in the first step of a TWRC sequence from the capacity  $C_{ij}$  of the same link used for an interference-free transmission as follows:

$$C_{ij}^T = (1 - \gamma)C_{ij}. \quad (3)$$

In the ideal case, where no overhead is added the link capacity for PLNC would equal interference-free capacity. In Fig. 11, we evaluate the PLNC performance for the same AP scenario, while taking account the PHY-layer overhead. Obviously, the results show that the PLNC gains decrease when the overhead increases. However, even with an important overhead amount, the PLNC performance remains quite compelling. In particular, when the PHY-overhead is 30 percent, the gains reach almost 60 percent compared to interference-free transmissions with a transmission range of 35 m. Even, with  $\gamma = 0.5$ , meaning that half of the first step of a TWRC sequence is dedicated to control information, when the transmission range is less than 45 m, the PLNC achieves more than 30 percent gains compared to interference-free transmissions, the same performance as for the traditional packet-based Network Coding whose overhead is not accounted.

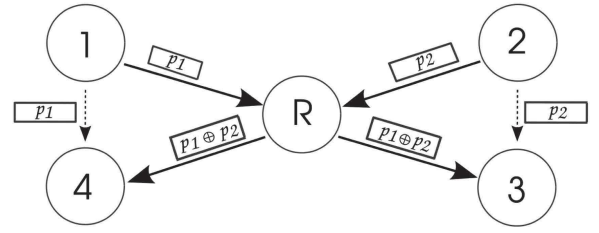


Fig. 12. The Butterfly PLNC (1→3 and 2→4)—two steps (time-slots) illustration.

## 6 ADVANCED PLNC SCHEMES

The two-way relay channel is the canonical scheme for applying PLNC in wireless networks but not the only one. In this section, we consider two additional schemes, referred to as Butterfly and Intra-flow, and show how our framework can be adapted for evaluating their performance in multi-hop wireless networks.

### 6.1 The Butterfly Scheme

While the two-way relay channel is an elegant way to implement PLNC, it requires bi-directional flows that share 3 consecutive nodes, a requirement that can be difficult to meet even in a large networks [3]. To relax this requirement, a new way of implementing PLNC has been introduced in [20] and [21]. Called “Butterfly”, it is a simplified version of the Butterfly introduced for the traditional network coding [5]. As illustrated in Fig. 12, 5 nodes are involved in this scenario: 2 sources, 2 destinations and 1 relay node. Each source (nodes 1 and 2) wants to send a packet to its destination (nodes 3 and 4) through the common relay (node R). Each destination is close enough to the other flow’s source to overhear its packet. The opportunistic listening technique is then applied. Unlike traditional protocols, where nodes capture only packets that are addressed to them, the destinations of the Butterfly scheme buffer all packets. In the first slot, sources simultaneously transmit their data to the relay, leading to a PLNC reception. In the second slot, the relay broadcasts the obtained XOR-ed packet to the two destinations. Thanks to the opportunistic listening in the first slot, each destination has enough information to retrieve its intended packet. In this scenario, the gain of the TWRC scheme is zero because there is no TWRC opportunity (no bi-directional traffic). With the Butterfly, on the other hand, the gain is 100 percent (2 slots instead of 4 with traditional communications).

Looking now at a larger topology with more concurrent unicast flows, one can see that estimating the gains with the Butterfly model becomes intricate. Indeed, the interference area of the Butterfly is larger than that of the TWRC scheme since more nodes are involved (5 instead of 3) what reduces the number of transmissions which can be scheduled simultaneously.

By replacing the TWRC elements (that are special cases of the Butterfly model) by Butterfly elements, our framework can also be used to evaluate the gain of the latter scheme. The scheduling rules here are slightly modified since transmissions in the first part of the PLNC sequence are now multicast (wireless broadcast). The received signal at each opportunistic listener also has to satisfy the physical model condition to ensure decoding. In addition, the linear program has to verify that the amount of data of new multicast transmissions is bounded by the lowest capacity of the two links.

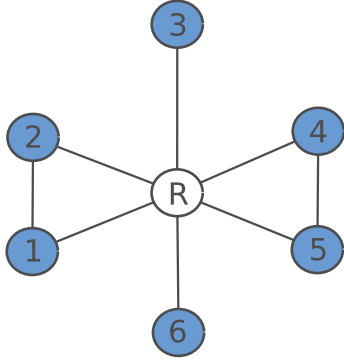


Fig. 13. Topology used for Butterfly framework validation—four established flows (1→4, 5→2, 3→6, and 6→3).

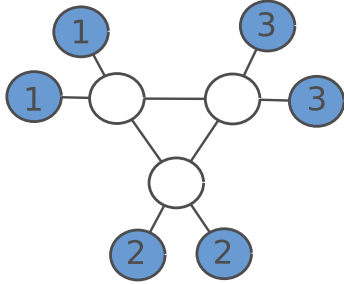


Fig. 14. Topology with three clusters for gains evaluation of Butterfly PLNC.

### 6.1.1 Framework Validation—Butterfly Scheme

In order to validate the correctness of the Butterfly scheme integration into our framework, we use the toy topology depicted in (Fig. 13). We instantiate 4 two-hop long flows transiting through the same relay  $R$ : 1→4 (flow 1), 5→2 (flow 2), 3→6 (flow 3) and 6→3 (flow 4).

Since the relay  $R$  serves as source or as destination for all the point-to-point transmissions, with traditional interference-free transmissions none of the 8 links can be active simultaneously. Then, with an optimal time sharing, the maximum throughput for the 4 flows is 1 frame every 8 slots, i.e., 0.125 frame/slot.

By including only the TWRC PLNC, nodes 3 and 6 can send simultaneously their packet to the relay  $R$  following this step  $R$  can broadcast the received coded packet to these two nodes. Then, instead of 8 independent sets obtained with interference-free transmissions, we distinguish only 6 independent sets of links with TWRC PLNC:  $I_0 = \{e_{1,R}^{1,1,0}\}$ ,  $I_1 = \{e_{R,4}^{1,1,0}\}$ ,  $I_2 = \{e_{5,R}^{2,1,0}\}$ ,  $I_3 = \{e_{R,2}^{2,1,0}\}$ ,  $I_4 = \{e_{3,R}^{3,1,1}, e_{6,R}^{4,1,1}\}$  and  $I_5 = \{e_{R,3}^{3,1,1}, e_{R,6}^{4,1,1}\}$ . Then, the optimal throughput achieves 0.1667 frame/slot with an optimal scheduling.

Leveraging the opportunistic listening at node 2 (resp. node 4) when node 1 is transmitting (resp. node 5), the transmissions from nodes 1 and 5 to  $R$  may be scheduled simultaneously and the two destinations can then decode the coded packet sent by the relay. Therefore, the number of independent sets is reduced from 6 to 4 thanks to the Butterfly PLNC. By allocating a quarter of the time to each of them, the per flow throughput reaches 0.25 frame/slot.

For the sake of validation, we applied our framework to this scenario, and we have obtained, as expected, the 3 maximum per flow throughput values (0.125, 0.1667 and 0.25 frame/slot) depending on the considered transmission mode (Interference-free transmissions, TWRC PLNC or Butterfly PLNC).

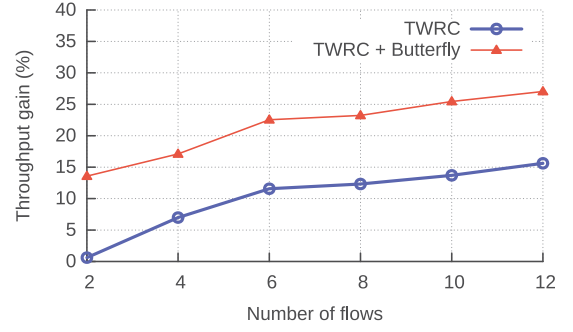


Fig. 15. Gains of butterfly PLNC.

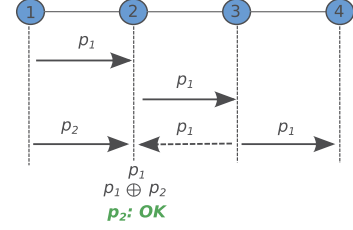


Fig. 16. Intra-flow PLNC concept.

### 6.1.2 Preliminary Results

We use our modified framework for a quick illustration of the potential of the Butterfly scheme. We apply the same approach as in Section 5 on the topology shown in Fig. 14 in which nodes from 3 different clusters send data to each other through the relays. Fig. 15 shows that the Butterfly scheme outperforms TWRC by almost a factor of 2, regardless of the traffic load. This is due to the fact that the Butterfly scheme offers more opportunities for performing PLNC transmissions, leading to an increase in throughput.

### 6.2 Intra-Flow PLNC

Even if the Butterfly does not require exact bi-directional traffic as in the TWRC scheme, it is clearly inefficient when flows follow the same direction. A solution to take advantage of PLNC in such situations was introduced in [3]. In a multi-hop flow, a particular relay, by definition, knows packets sent by the next relay. This enables the realization of the so-called Intra-flow PLNC scheme. As illustrated in Fig. 16, nodes 1 and 3 can transmit simultaneously packets  $p_1$  and  $p_2$ , respectively. Node 2 receives a physically-encoded packet of  $p_1$  and  $p_2$  and, having kept  $p_1$  in its buffer, it can decode and retrieve packet  $p_2$ . With this, in a chain topology, the flow rate is 0.5 frame per slot compared to 0.33 (1 frame every 3 slots) with traditional interference-free transmissions.

In contrast to TWRC and Butterfly schemes, Intra-flow PLNC does not require any particular traffic pattern but, lacking a theoretical tool, it has not been studied in large ad-hoc networks.

We adapt our framework to accommodate Intra-flow PLNC. This is necessary due to the fact that Intra-flow PLNC is based on the assumption that each relay already knows the packets sent by the following node. This condition is not satisfied in the TWRC scheme. In particular, the relay node of TWRC elements does not know the native packets sent by the two sources – it just decodes and relays the XOR-ed packet.

To account for this new condition, we assign a new boolean parameter  $a$  to each link of the interference graph. We enforce  $a = 1$  or  $a = 0$  depending on whether the corresponding packet transmission follows (comes immediately

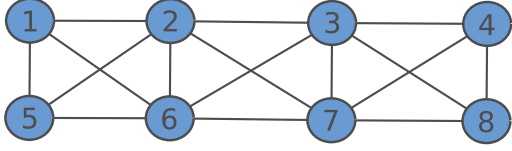


Fig. 17. Topology used for Intra-flow PLNC validation—two established flows (1→4 and 5→8).

after) a TWRC element or not respectively. The amount of data through each link is then denoted  $l_{i,j}^{f,p,T,a}$ . Preventing a link with  $a = 1$  to be involved in an Intra-flow PLNC transmission at its previous node, we ensure that each node can decode its intended packets after each Intra-flow PLNC reception. To preserve flow conservation, we derive a new linear program extending the previous one. Constraints 1, 4, 5 and 6 remain unchanged. However, constraints 2 and 3 are replaced by constraints 8, 9 and 10.

**Constraint 8:** Flow conservation for packets following a TWRC sequence

$$\begin{aligned} & \forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall n_i \\ & \in \mathcal{N}(f, p) \setminus \{s(f), d(f)\} \sum_{T \in \mathcal{T}(f, p)} IN(T, n_i) \times l_{i,j}^{f,p,T,1} + l_{i,j}^{f,p,0,1} \\ & = \sum_{T \in \mathcal{T}(f, p)} OUT(T, n_i) \times l_{h,i}^{f,p,T,0} \end{aligned}$$

**Constraint 9:** Flow conservation for native packets

$$\begin{aligned} & \forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall n_i \\ & \in \mathcal{N}(f, p) \setminus \{s(f), d(f)\} \quad l_{i,j}^{f,p,0,0} + \sum_{T \in \mathcal{T}(f, p)} IN(T, n_i) \times l_{i,j}^{f,p,T,0} \\ & = l_{h,i}^{f,p,0,0/1} \end{aligned}$$

**Constraint 10:** Flow conservation for coded packets

$$\begin{aligned} & \forall f \in \mathcal{F}, \forall p \in \mathcal{P}(f), \forall T \in \mathcal{T}(f, p), \quad \forall n_i \\ & \in \mathcal{N}(f, p) \setminus \{s(f), d(f)\} \quad RELAY(T, n_i) \times l_{i,j}^{f,p,T,0} \\ & = RELAY(T, n_i) \times l_{h,i}^{f,p,T,0/1} \end{aligned}$$

Constraint 8 ensures conservation of each packet received in the broadcast transmission of a TWRC sequence. Indeed, with our definition, all of them are relayed in what we identify as transmissions following a TWRC sequence ( $a = 1$ ). Intra-flow PLNC cannot be processed with these transmissions.

Constraint 9 ensures flow conservation for packets received in interference free transmissions. These transmissions can follow a TWRC sequence or not ( $a = 0/1$ ). They can be forwarded by each relay in another interference free transmission or in a TWRC sequence (if the concerned node is the source of a TWRC element).

Constraint 10 is similar to constraint 3 of the previous linear program. The only difference is that the relay of a TWRC element can receive packets identified as following another TWRC sequence (i.e., with  $a = 0/1$ ).

### 6.2.1 Framework Validation—Intra-Flow PLNC

We apply our framework to the scenario depicted in Fig. 17 in which nodes nodes 1 and 5 transmit packets to nodes 4 and 8 respectively. When allowing interference-free transmissions, the TWRC or the Butterfly scheme, the computed

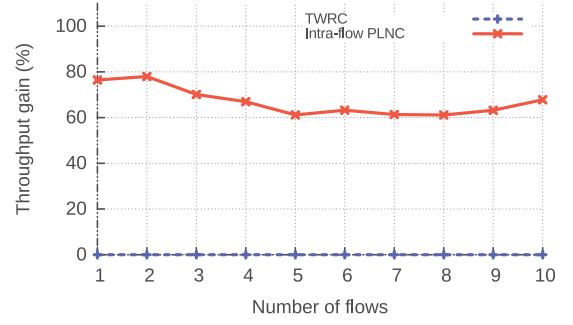


Fig. 18. Intra-flow PLNC gains evaluation.

maximum throughput is 0.1667 frame per slot. Indeed, with interference-free transmissions none of the 6 necessary transmissions to route the 2 flows may be active simultaneously. Even the transmission from node 7 to 8 cannot be scheduled in the same time with the transmission from 1 to 2 due to the interference produced at node 2. Then, by adopting an equal and fair time sharing between the two flows, the maximum throughput per flow is 1 frame every 6 slots. As there is no crossing flows, the TWRC and Butterfly PLNC schemes do not improve the throughput compared to interference-free transmissions.

By integrating the Intra-flow PLNC feature, the maximum throughput computed thanks to our framework is 0.25 frame per slot. Indeed, transmissions from nodes 1 and 3 can be scheduled simultaneously since node 2 can handle the reception of the two signals. Similarly for the flow 5→8, nodes 5 and 7 may be active in the same time. Consequently, the number of independent sets decreases from 6 to 4 and the obtained throughput with an optimal scheduling becomes 1 frame every 4 slots. This corroborates the obtained value in our framework.

### 6.2.2 Preliminary Results

After the empirical validation on tractable topologies, we can now illustrate by solving the new optimization problem, the potential gains of Intra-flow PLNC using the topology of Fig. 8 with the transmission range set to 15 m. We instantiate traffic only from the AP to the user nodes and vary the number of flows. As expected, Fig. 18 shows that TWRC is ineffective in this configuration as all packets flow are in the same direction. For the same setting, however, Intra-flow PLNC is shown to be remarkably effective, improving throughput by over 60 percent when compared to interference-free transmissions.

## 7 GENERAL EVALUATION OF PLNC SCHEMES

In this section, we use a MATLAB implementation of our theoretical framework to evaluate the performance of the TWRC, Butterfly and Intra-flow PLNC schemes using three representative scenarios.

### 7.1 Performance Metric

In all the following scenarios, we evaluate the PLNC schemes in terms of the gain in throughput realized when compared to the traditional interference-free transmission. As we did in Section 5, for the PLNC schemes we show the throughput lower-bound values while for interference-free transmissions we show the upper-bound values. As a result, the gains shown in the following are conservative.

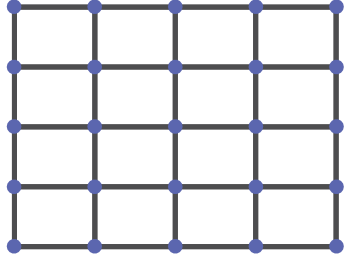


Fig. 19. Grid topology.

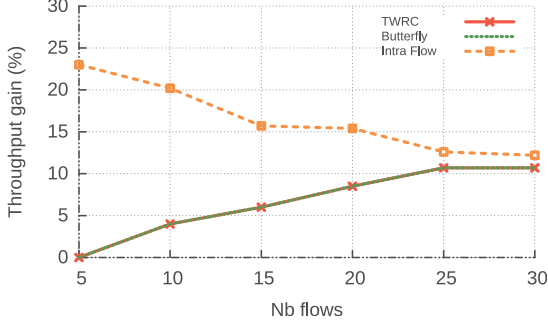


Fig. 20. Performance evaluation of the three PLNC schemes in a grid topology.

## 7.2 The Popular Grid Topology

We start the performance evaluation with the popular  $5 \times 5$  grid topology depicted in Fig. 19. We vary the number of traffic flows in the network and for each flow the source and destination are selected uniformly at random among all the network nodes.

Fig. 20 shows the throughput gains for all PLNC schemes. The data point to a few interesting observation: First, whereas in the base topologies the Intra-flow PLNC gain (66 percent) are lower than both the TWRC and Butterfly gain (100 percent), in the grid topology the roles are reversed. Intra-flow PLNC clearly outperforms both TWRC and Butterfly, especially when the number of flows is low. Second, Butterfly and TWRC perform exactly the same. This is due to the fact that the grid topology does not offer any opportunities to trigger PLNC transmissions via opportunistic listening. Thus, the Butterfly scheme falls back to TWRC.

In a second experiment, we investigate the impact of routing decisions on the performance of Intra-flow and TWRC (Butterfly performs exactly the same in this topology). We select 10 flows at random and over 100 simulation runs evaluate the gains of the two PLNC schemes by changing the routes these flows use. For each simulation run and for each flow, the actual data path is chosen at random among all the available shortest paths (in terms of hops). Fig. 21 shows that the standard deviation of the Intra-flow gains is lower than the standard deviation of the TWRC gains. In practice, this means that the routing decisions have less influence on the Intra-flow gain than they have on the TWRC gain. This is due to the fact that Intra-flow PLNC is applied between the packets of the same path, making its performance independent of routing. On the contrary, the number of TWRC elements, and by extension the throughput gain realized, closely depends on the routing decisions. To validate our intuition, for different values of throughput gains observed during the simulation we collect the number of TWRC elements established. Unsurprisingly, as shown in Fig. 22, the gain increases linearly with the number of TWRC elements.

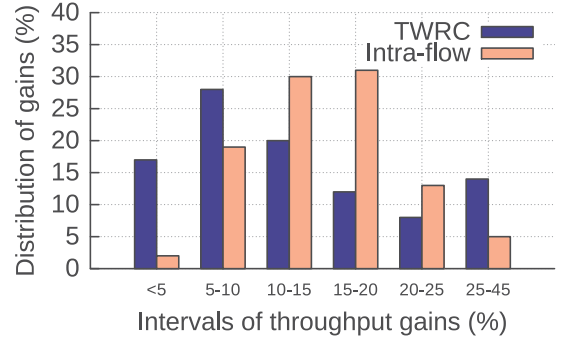


Fig. 21. TWRC and Intra-flow gain distribution with 100 different route combinations.

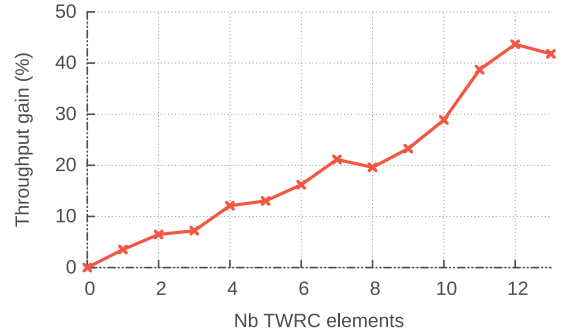


Fig. 22. TWRC gains depending on the number of TWRC elements in the network.

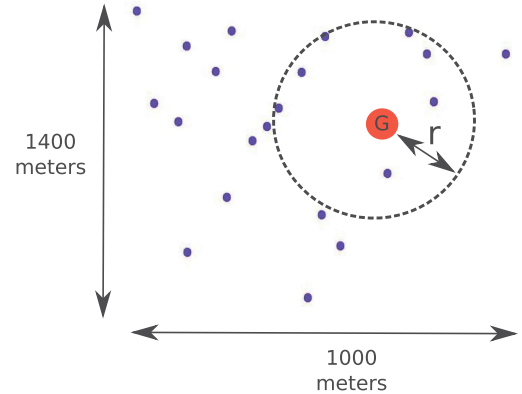


Fig. 23. A municipal mesh network deployed in Portland, Oregon.

## 7.3 A Municipal Wireless Mesh Network

In this experiment, we evaluate the performance of the PLNC schemes in a scenario inspired from a real-life setting. Toward this, we use the topology depicted in Fig. 23. It is built using the node locations of a mesh APs network deployed in Portland, Oregon in 2007 [31], when the transmission range of each node is 250 meters. As described in [32], this network is originally composed of 70 nodes. However, it seems to be divided in 3 geographical clusters separated by the Willamette River and we only focus on one of these clusters.

Since no communication pattern is provided, we have constructed the following scenario: Only one node is linked to the Internet and all the communications from the other APs are routed through this gateway node. This scenario is similar to the coverage extension studied in [8]. We choose node G in Fig. 23 to serve as the gateway to the Internet and vary the number of flows in the network. Obviously, each flow has for source or destination the gateway, node G. The path loss threshold is fixed

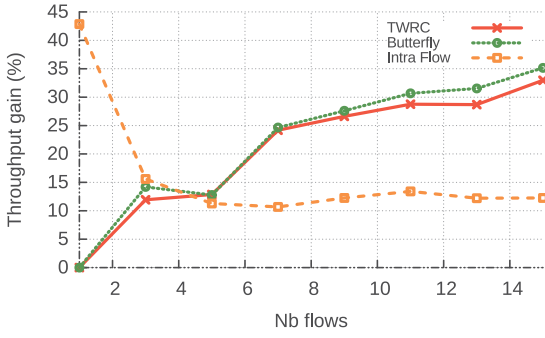


Fig. 24. PLNC gains evaluation in Portland municipal mesh network.

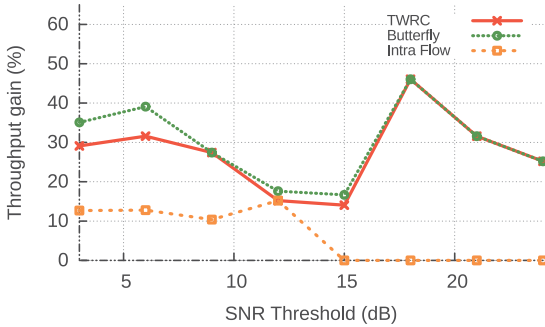


Fig. 25. SNR threshold impact.

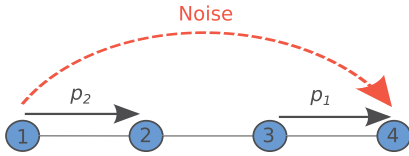


Fig. 26. Noise inside the Intra-flow element.

at 4 and the SNR threshold is set to 10 dB. We do not consider any overhead at PHY-layer.

Fig. 24 shows that the TWRC and Butterfly schemes clearly outperform Intra-flow PLNC. This can be explained by the fact that the mesh setting enables the formation of many TWRC elements, in particular around the gateway node. Indeed, in the presence of a high number crossing traffic (high number of flows), the TWRC and Butterfly schemes are able to better cope with the generated interference since they combine together bi-directional traffic.

It is also interesting to evaluate the impact of the minimum SNR that a node requires to decode a packet, on the performance of the 3 PLNC schemes. To this end, we select a given number of traffic flows, 8, and calculate the performance gain for all schemes as we vary the SNR threshold,  $\beta$ . Fig. 25 shows that the SNR threshold has a significant influence on the performance of each scheme. In particular, two interesting phenomena emerge for high SNR threshold values: First, the Butterfly scheme gains converge to those of TWRC. Second, and more surprisingly, the Intra-flow gain collapses to zero. To explain the latter, let us consider the simple example depicted in Fig. 26. The benefit of the Intra-flow PLNC is predicted on node 4 being able to receive packet  $p_1$  while node 1 is transmitting  $p_2$ —the assumption being that the node 1 transmission is too weak to cause destructive interference. While this is a generally reasonable assumption, it does not hold once node 4 is equipped with a transceiver requiring a very high SNR. In this case, node 4 cannot receive while node 1 is transmitting, preventing the execution of Intra-flow PLNC. Similarly,

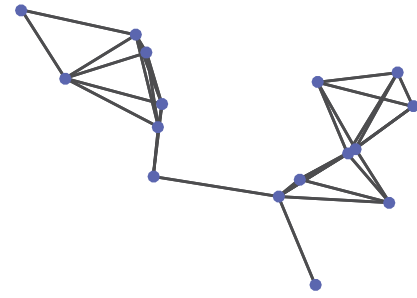


Fig. 27. Parking topology—Range = 35m.

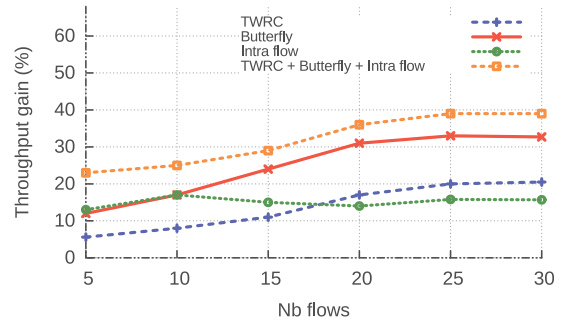


Fig. 28. Combination of the 3 PLNC schemes.

with a high  $\beta$  value, the opportunistic transmissions in the Butterfly scenario cannot be scheduled simultaneously.

#### 7.4 A Cluster of Fire Engines

Multi-hop wireless communications are often set up to replace infrastructure-based networks when they are unavailable due to natural disasters, for example. As it is difficult to get real traces in such situations, we decide to evaluate the PLNC performance in a less critical environment with the data set provided in [33] and studied in [34]. It retraces one year of GPS location of fire vehicles in the Oviedo area (Spain). We focus on GPS location on day 7 of 16 nodes in a parking lot i.e., close enough to communicate (refer to Fig. 27 for an illustration of the topology)—connectivity between 2 radio nodes is established when they are separated by less than 35 meters. Once again, it is hard to determine a precise pattern of communication between the nodes. Therefore, we evaluate the potential gains of the different PLNC schemes by creating traffic flows between nodes selected at random.

Fig. 28 shows that in this setting it is the Butterfly PLNC that offers the best gain. Clearly, when analyzing the data of all experiments the main observation that emerges is that there is no single scheme that is superior in all settings. Therefore, we adapted our theoretical framework to be able to evaluate the performance of all schemes when they are used together. As Fig. 28 shows, combining all schemes pays off, resulting in a 40 percent performance gain, significantly better than any other scheme deployed alone.

Even while considering the potential overhead introduced in order to support PLNC transmissions, the PLNC performance remains quite good. As shown in Fig. 29, by dedicating 30 percent of the PLNC transmission to solve physical layer issues, PLNC gains still achieve more than 20 percent compared to traditional interference-free transmissions.

## 8 OBSERVATIONS FROM THE PERFORMANCE EVALUATION

The performance evaluation presented in Section 7 leads to several important conclusions:

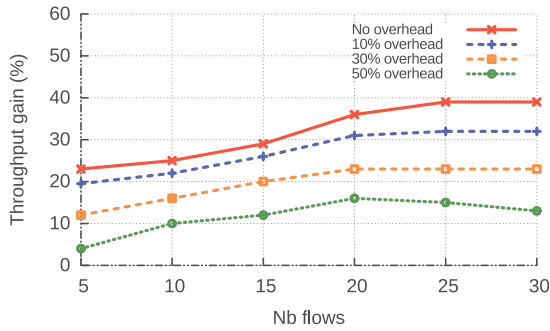


Fig. 29. Phy-overhead impact on the PLNC performance.

- (1) The 3 studied PLNC schemes present very different performance depending on the topology and the traffic pattern. The gains compared to interference-free transmissions vary from 10 to 100 percent, reaching between 25-40 percent in general conditions.
- (2) Gains of PLNC are more important when traffic converges to a single node than in a fully-distributed environment of communications.
- (3) The TWRC scheme is sufficient when the flows are exactly bidirectional. The routing decisions have an important impact on the performance of this PLNC scheme. Routing protocols need to be adapted to this scheme.
- (4) The Butterfly PLNC relaxes the constraint of bidirectional traffic. Since the Butterfly scheme is a generalization of the TWRC one, it always leads to larger gains than the basic TWRC. The routing decisions have less influence on the performance of this scheme. However, the Butterfly scheme suffers from a high SNR threshold. In this case, applying the TWRC scheme is sufficient.
- (5) The Intra-flow PLNC is very different and complementary to the two other schemes. In particular, it is effective when the flows follow the same direction and for light-loaded traffic. However, the three-hop long communications necessary for this scheme to perform is a real obstacle. Finally, as for the Butterfly scheme, its performance decreases with a high SNR threshold value.

## 9 CONCLUSION AND FUTURE WORK

In the coming years, wireless networks are expected to experience an exponential growth in the number of devices, severely challenging the access techniques known today. Physical-Layer Network Coding (PLNC) can be seen as a candidate technique to increase the efficiency and reduce the interference in these dense networks. In this work, we have investigated the capability of PLNC to enhance throughput in large multi-hop networks. To do so, we have formalized the scheduling problem in multi-hop wireless network with PLNC as an optimization problem. We have introduced a novel interference graph that captures how PLNC changes the definition of interference in multi-hop wireless networks.

Starting with the widely used two-way relay channel model, we have computed the achievable throughput in large topologies. Then, we have empowered our framework with the capability of accounting for additional and more complex PLNC schemes such as the Butterfly and the Intra-flow

models. Our results show that, with these models, PLNC increases throughput by 30 percent in realistic topologies.

In the future, we aim to enrich our framework with other scheduling policies. For now, our framework has adapted optimal flow-preserving scheduling to be implemented with TDMA-based access. What will the performance gains be with other scheduling policies such as the one produced by the IEEE 802.11 MAC? Additionally, incorporating other PLNC forms (such as compute-and-forward and multi-way relay channel) in our modelling framework can be of a great interest to the community. This would allow comparing the throughput achieved by each scheme over the same topology and in the same conditions.

## ACKNOWLEDGMENTS

This work was supported in part by the Agence Nationale de la Recherche under the ANR JCJC CiTADEL grant. An early version appeared in the *Proceedings of IEEE INFOCOM 2017*, Atlanta, GA, USA [1].

## REFERENCES

- [1] R. Naves, H. Khalifé, G. Jakllari, V. Conan, and A.-L. Beylot, "A framework for evaluating physical-layer network coding gains in multi-hop wireless networks," in *Proc. IEEE Conf. Comput. Commun.*, May 2017, pp. 1–9.
- [2] S. Zhang, S. Liew, and P. Lam, "Hot topic: Physical-layer network coding," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw.*, 2006, pp. 358–365.
- [3] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: Analog network coding," in *Proc. Conf. Appl. Technol. Architectures Protocols Comput. Commun.*, 2007, pp. 397–408.
- [4] 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, Jul. 2000.
- [5] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XOR's in the air: Practical wireless network coding," in *Proc. ACM Conf. Appl. Technol. Architectures Protocols Comput. Commun.*, 2006, pp. 243–254.
- [6] A. Argyriou, "Mac protocol for wireless cooperative physical layer network coding," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2012, pp. 1596–1601.
- [7] O. Goussevskaia, Y. Oswald, and R. Wattenhofer, "Complexity in geometric SINR," in *Proc. ACM 8th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2007, pp. 100–109.
- [8] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proc. ACM 9th Annu. Int. Conf. Mobile Comput. Netw.*, 2003, pp. 66–80.
- [9] S. Sengupta, S. Rayanchu, and S. Banerjee, "An analysis of wireless network coding for unicast sessions: The case for coding-aware routing," in *Proc. 26th IEEE Int. Conf. Comput. Commun.*, 2007, pp. 1028–1036.
- [10] O. Goussevskaia and R. Wattenhofer, "Complexity of scheduling with analog network coding," in *Proc. 1st ACM Int. Workshop Foundations Wireless Ad Hoc Sensor Netw. Comput.*, 2008, pp. 77–84.
- [11] K. Lu, S. Fu, Y. Qian, and H. Shen, "On capacity of random wireless networks with physical-layer network coding," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 5, pp. 763–777, Jun. 2009.
- [12] T. Zhang, K. Lu, A. Jafari, and S. Fu, "On the capacity bounds of large-scale wireless network with physical-layer network coding under the generalized physical model," in *Proc. IEEE Int. Conf. Commun. Workshops*, 2010, pp. 1–5.
- [13] P. Gupta and P. Kumar, "The capacity of wireless networks," *Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.
- [14] L. Lu and S. Liew, "Asynchronous physical-layer network coding," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 819–831, Feb. 2012.
- [15] L. Lu, L. You, Q. Yang, T. Wang, M. Zhang, S. Zhang, and S. Liew, "Real-time implementation of physical-layer network coding," in *Proc. ACM 2nd Workshop Softw. Radio Implementation Forum*, 2013, pp. 71–76.

- [16] L. Lu, T. Wang, S. Liew, and S. Zhang, "Implementation of physical-layer network coding," *Elsevier Phys. Commun.*, vol. 6, no. 1, pp. 74–87, 2013.
- [17] L. Xie, I. Ho, S. Liew, L. Lu, and F. Lau, "The feasibility of mobile physical-layer network coding with BPSK modulation," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 3976–3990, May 2017.
- [18] S. Wang, Q. Song, X. Wang, and A. Jamalipour, "Distributed MAC protocol supporting physical-layer network coding," *IEEE Trans. Mobile Comput.*, vol. 12, no. 5, pp. 1023–1036, May 2013.
- [19] R. Naves, G. Jakllari, H. Khalife, V. Conan, and A.-L. Beylot, "A physical-layer network coding aware distributed scheduling policy," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf.*, 2017, pp. 1576–1581.
- [20] W. Mao, X. Wang, A. Tang, and H. Qian, "Anc-era: Random access for analog network coding in wireless networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 1, pp. 45–59, Jan. 2016.
- [21] X. Wang and W. Mao, "Analog network coding without restrictions on superimposed frames," *IEEE Trans. Netw.*, vol. 24, no. 3, pp. 788–805, Apr. 2016.
- [22] D. Gunduz, A. Yener, A. Goldsmith, and H. Poor, "The multiway relay channel," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 51–63, Jan. 2013.
- [23] L. Ong, S. J. Johnson, and C. M. Kellett, "An optimal coding strategy for the binary multi-way relay channel," *IEEE Commun. Lett.*, vol. 14, no. 4, pp. 330–332, Apr. 2010.
- [24] T. Hynes, D. Halls, and J. Sykora, "Hardware implementation of distributed learning algorithm for mapping selection for wireless physical layer network coding," in *Proc. IEEE Int. Commun. Workshop*, 2015, pp. 2127–2132.
- [25] T. Hynes, D. Halls, and J. Sykora, "Practical implementation of cloud initialization procedure for wireless physical layer network coding clouds," in *Proc. 20th Eur. Wireless Conf.*, 2014, pp. 1–6.
- [26] B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6463–6486, Oct. 2011.
- [27] L. Wei and W. Chen, "Compute-and-forward network coding design over multi-source multi-relay channels," *IEEE Trans. Wireless Commun.*, vol. 11, no. 9, pp. 3348–3357, Sep. 2012.
- [28] N. Sommer, M. Feder, and O. Shalvi, "Low-density lattice codes," *IEEE Trans. Inf. Theory*, vol. 54, no. 4, pp. 1561–1585, Apr. 2008.
- [29] Y. Tan, S. Liew, and T. Huang, "Mobile lattice-coded physical-layer network coding with practical channel alignment," *IEEE Trans. Mobile Comput.*, vol. 17, no. 8, pp. 1908–1923, Aug. 2018.
- [30] R. Naves, G. Jakllari, H. Khalife, V. Conan, and A.-L. Beylot, "When analog meets digital: Source-encoded physical-layer network coding," in *Proc. IEEE WOWMOM Conf.*, 2018, pp. 1–9.
- [31] R. Senior and C. Phillips, "CRAWDAD dataset pdx/metrofi (v. 2011–10-24)," Oct. 2011. [Online]. Available: <https://crawdad.org/pdx/metrofi/20111024>
- [32] C. Phillips, R. Senior, D. Sicker, and D. Grunwald, "Robust coverage and performance testing for large-area wireless networks," *AccessNets*, vol. 6, pp. 457–469, 2009.
- [33] S. Cabrero, R. Garca, X. G. Garca, and D. Melendi, "CRAWDAD dataset oviedo/asturies-er (v. 2016–08-08)," Aug. 2016. [Online]. Available: <https://crawdad.org/oviedo/asturies-er/20160808>
- [34] S. Cabrero, R. Garca, X. Garca, and D. Melendi, "Understanding opportunistic networking for emergency services: Analysis of one year of GPS traces," in *Proc. 10th ACM MobiCom Workshop Challenged Netw.*, 2015, pp. 31–36.



**Raphaël Naves** received the master's degree from the University of Toulouse, Toulouse, France, in 2015. He is currently working toward the PhD degree at Thales Communications & Security, Gennevilliers, France, and the University of Toulouse, Toulouse, France. His research interests include the field of wireless networks, physical layer techniques, access layer protocols, and cooperative networks. More specifically, he works on the integration of the new techniques of the physical layer part in large multihop networks.



**Hicham Khalifé** received the MSc degree from the University Pierre et Marie Curie, in 2005, and the PhD degree from the University Pierre et Marie Curie, in 2008. From 2009 to 2011, he served as an associate professor with the Computer Engineering Department of IPB ENSEIRB-MATMECA in Bordeaux, France, and a member of the LaBRI research laboratory enrolled in the COMET Networking team. He joined Thales Communications & Security in December 2011. His main research contributions are essentially in the area of wireless networks and multi-hop collaborative radio networks. He has served as a PC member of many international conferences and published and reviewed for international journals and conferences.



**Gentian Jakllari** received the bachelor's degree in computer science from the University of Ioannina, in 2002, and the MSc and PhD degrees in computer science from the University of California, Riverside, in 2005 and 2007, respectively. From 2007 to 2011, he was a network scientist with the Internetworking Research Department at BBN Technologies in Cambridge, MA. In 2011, he joined INP-ENSEEIH with the University of Toulouse, France, where he is now an associate professor. He holds a joint appointment with the Toulouse Science Research (IRIT) and is a member of the RMES team. His research interests are in computer networks with emphasis on wireless networking and mobile systems, including cross-layer and cross-technology design for Gigabit wireless, smartphone-based localization and navigation, cyber-physical systems, IoT, and energy-efficiency.



**Vania Conan** received the engineering degree and the PhD degree in computer science from Mines ParisTech, in 1990 and 1996, respectively, and the habilitation degree from Université Pierre et Marie Curie, Paris, in 2012. He is a senior research expert in networking and communications at Thales Communications & Security (TCS), in Gennevilliers, France. He is presently head of the networking laboratory in the Advanced Studies Department at TCS. In the past years, he has been conducting research in the field of wireless & Ad-Hoc networking, including cross-layer, opportunistic protocols, and network coding. He has published more than 60 international conference and journal papers and filed several patents in networking technologies. His current research topics include cross-layer design of mobile network protocols and virtualized network design.



**André-Luc Beylot** received the PhD degree in computer science from the University of Paris 6, in 1993. In January 2000, he received the habilitation diriger des recherches from the University of Versailles. Since September 2000, he has been a professor in the Telecommunication and Network Department of the ENSEEIHT. From 2008 until 2011, he led the IRT Team of IRIT Lab. Since January 2011, he has been the head of the ENSEEIHT site of IRIT. Since January 2015, he has been the head of GDR CNRS - Réseaux et Systèmes Distribués, RSD. His research interests focus on performance and design of wireless networks.