

Robust Wireless Network Coding – An Overview

Marco Di Renzo¹, Lana Iwaza^{1,2}, Michel Kieffer¹,
Pierre Duhamel¹, and Khaldoun Al Agha²

¹ Laboratoire des Signaux et Systèmes, CNRS – SUPELEC – Univ. Paris-Sud 11
91192 Gif-sur-Yvette, Paris, France

{marco.direnzo,lana.iwaza,michel.kieffer,pierre.duhamel}@lss.supelec.fr
<http://www.lss.supelec.fr>

² Laboratoire de Recherche en Informatique, CNRS – Univ. Paris-Sud 11
91405 Orsay, Paris, France

alagha@lri.fr
<http://www.lri.fr>

Abstract. Network Coding (NC) has witnessed a tremendous upsurge in interest and activities in recent years, both in academia and industry. Indeed, since the pioneering publication of Ahlswede *et al.* in 2000, NC has rapidly emerged as a major research area in information theory due to its wide applicability to communication through real networks. The many contributions available in the literature to date, ranging from purely theoretical studies on fundamental limits to practical experimentations in real-world environments, offer a clear evidence that the shift in paradigm envisaged by NC might revolutionize the way we manage, operate, and understand the organization of networks. However, the principle of NC is not without its limitations. Initial studies on NC were mainly focused on lossless channels, which, however, might have limited applicability to a wireless context. As a matter of fact, in practical wireless environments, NC might be very susceptible to transmission errors caused by noise, fading, or interference. In particular, the algebraic operations accomplished by the intermediate nodes of the network introduce some packet dependencies in a way that the injection of even a *single* erroneous packet has the potential to corrupt *every* packet received by the destination nodes. Motivated by this consideration, recent research efforts have been devoted to the design of *robust* NC, with the main goal to circumvent the critical limitations of the NC paradigm in practical operating environments. In this paper, we aim at providing an overview of the most important and notable research directions in this emerging field.

Keywords: Network Information Flow, Network Coding, Error Control, Coding over Networks, Joint Network-Channel Decoding, Reliable Communications, Wireless Networks.

1 Introduction

Communication networks are designed to deliver information from source to destination nodes. The traditional way of delivering data employs paths for unicast connections and trees for multicast connections. When data is routed over

a unicast path, each intermediate node forwards the packets received over its incoming edges to its outgoing edges. While, in a multicast connection over a tree the intermediate nodes may duplicate packets and forward them to several outgoing edges. In other words, in today practical communication networks information delivery is accomplished through *routing*: network nodes simply store and forward data, and processing is only accomplished at the end nodes. Network Coding (NC) is a recent field in information theory that breaks with this assumption: instead of simply forwarding data, the intermediate nodes may recombine several input packets into one or several output packets. This way, NC allows the intermediate nodes to generate new packets by combining those received on their incoming edges. The potential advantages of NC over routing include resource (*e.g.*, bandwidth and power) efficiency, computational efficiency, and robustness to changes in the topology of the network [1].

Research on NC was initiated by the seminal paper [2], and has since then attracted significant interest from the research community. Many initial works on NC focused on establishing multicast connections. It was shown in [2] that the capacity of multicast networks (*i.e.*, the maximum number of packets that can be sent from the source to a set of terminals per time unit) can be achieved by coding within the network, *i.e.*, by allowing the mixing of data at the intermediate nodes of the network. A few years later, in [3] it was shown that, for multicast networks, linear coding at the intermediate nodes suffices to achieve the capacity limit, which is the max-flow from the source to each receiving node. In [4], the authors extended the results in [3] to arbitrary networks and introduced a very powerful algebraic framework for NC. The approach establishes a useful connection between a NC problem and the solution of certain systems of polynomial equations. In [5], the authors conceived a practical NC scheme without need of centralized knowledge of the network topology or of the encoding/decoding functions. The fundamental idea of [5] consists in including within each transmitted packet the global encoding vector along the edge. This way, these latter vectors, which are needed to decode the data received at any receiver, can be found in the arriving packets themselves. With the cost of a reasonable overhead, the approach can offer a totally decentralized solution to NC over networks. In [6], the authors capitalized on the analytical formulation of [4] and the practical scheme in [5] by proposing a distributed and fully randomized method to design the network codes. Moreover, it was shown in [6] that the network capacity can be achieved with probability exponentially approaching one with the code length. Finally, on a more practical side, Katti *et al.* conceived several solutions, *i.e.*, COPE, ANC, MIXIT, MORE, to efficiently exploit the NC paradigm over wireless networks [7]–[10].

However, besides the many potential advantages and applications of NC over classical routing (see, *e.g.*, [11], [12]), the NC principle is not without its drawbacks. A fundamental problem that NC needs to face with over lossy networks is the so-called *error control problem*: corrupted packets injected by some intermediate nodes might propagate through the network until the destination, and might render impossible to decode the original information. In contrast to

routing, this problem is crucial in NC due to the algebraic operations performed by the internal nodes of the network. As a matter of fact, the mixing of packets within the network makes every packet flowing through it statistically dependent on other packets: even a single erroneous packet might affect the correct detection of all other packets. On the contrary, the same error in networks using just routing would affect only a single source–destination path. Broadly speaking, possible errors in NC might arise for three main reasons [13]: i) *erasures*, which lead to insufficiently received packets at the destination to solve the NC problem and retrieve the transmitted messages, ii) *errors*, which are due to using, for complexity and practical reasons, not powerful enough link–to–link error–correcting codes or are caused by the need to avoid a retransmission of all corrupted packets, and iii) the presence of intentional *jammers*, who might introduce erroneous packets at the application layer, which might be difficult to be recovered by the destination node. In such a context, the conventional approach to drop all erroneous packets detected at the physical layer, might be very sub–optimal for several reasons, *e.g.*, i) this may lead to insufficiently received packets for decoding and may be very spectrally inefficient, ii) even packets with errors could be a source of redundancy that may help the decoding process at the destination node, and iii) even though some bits are wrong, some parts of the packets are still error–free and could be exploited via some joint source–channel decoding methods to correct the wrong–bits.

In the light of all the above, robust NC is concerned with the design of efficient methods to design codes and decoding algorithms that are robust to all kinds of errors above, and can be decoded in a computationally efficient way. In this paper, we provide an overview of two important approaches to improve the reliability of network–coded data transmission over lossy networks: i) the design of error–correcting codes in projective spaces [14], and ii) the design of joint network–channel decoding schemes [15]. The first method moves from the key observation that in random linear NC the only property of the matrix containing the transmitted packets that is preserved is its row space. Thus, the information to be transmitted should be encoded in the choice of a subspace rather than in a specific matrix. On the other hand, the second method advocates a joint decoding of network and channel codes in order to fully exploit the spatial diversity and redundancy residing in both of them. Of course, both views are two sides of the same coin: they both aim at improving the performance and robustness of network–coded wireless architectures over lossy networks.

The remainder of this paper is organized as follows. In Section 2, the concept of linear NC is introduced in a formal way, and the fundamental algebraic tool introduced in [4] for its analysis is carefully described along with the concept of Random Linear Network Coding Channel (RLNCC) model widely adopted to describe the behavior of lossy networks. In Section 3 and Section 4, recent results on the design of codes in projective spaces and on the joint decoding of network and channel codes are summarized, respectively. Finally, Section 5 concludes the paper.

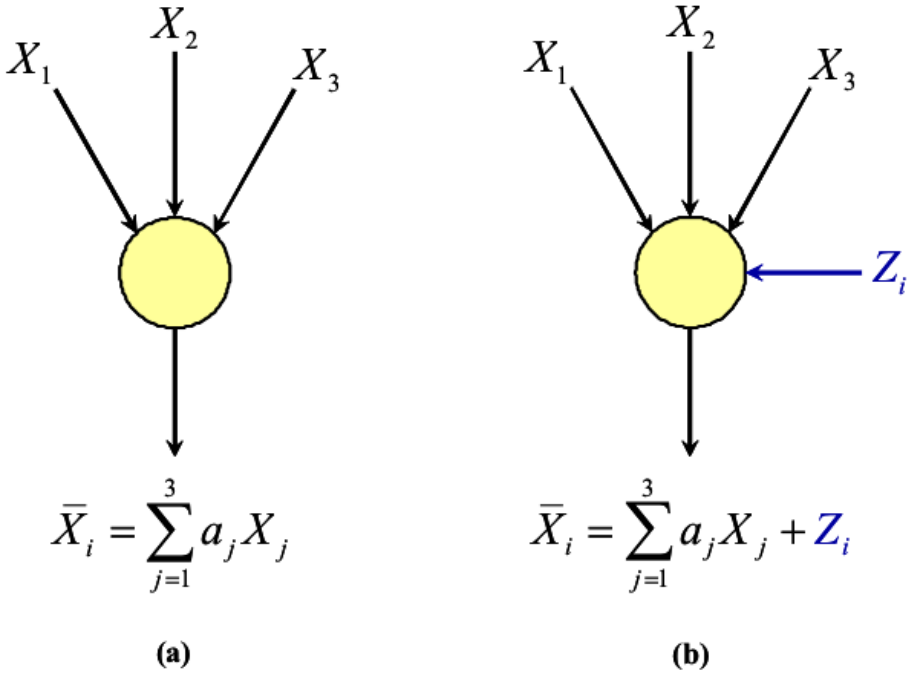


Fig. 1. Example of “injection packets” for a generic nodes of the network. (a) Error-free scenario. (b) Error-prone scenario. $\{a_j\}_{j=1}^3$ are the coefficients of the network code.

2 Fundamentals of Network Coding

Let us consider, for illustrative purposes, a point-to-point communication network with a single source node and a single destination node. Let us also assume, for the moment, that each link in the network transports free of errors a packet of M symbols in a given finite field \mathbb{F}_q . During each transmission, the source node sends n information vectors (*i.e.*, packets) X_1, X_2, \dots, X_n , each one with size $1 \times M$. Whenever a node of the network (with the inclusion of the source node) has an opportunity for transmission, it produces an outgoing packet that is obtained as a random linear combination (with values over the field \mathbb{F}_q) of all incoming packets received until then. Each node of the network performs this operation until the destination node collects N packets Y_1, Y_2, \dots, Y_N , from which it tries to infer the original data emitted by the source node, *i.e.*, X_1, X_2, \dots, X_n .

Let us now cast the transmitted X_1, X_2, \dots, X_n and received Y_1, Y_2, \dots, Y_N packets into a compact matrix representation form. Let X be a $n \times M$ matrix whose rows are the vectors X_1, X_2, \dots, X_n . Similarly, let Y be a $N \times M$ matrix whose rows are the vectors Y_1, Y_2, \dots, Y_N . Owing to the assumption of linear operations performed by each node of the network, Y and X can be related by the following simple matrix expression:

$$Y = AX \tag{1}$$

where A is a $N \times n$ matrix that corresponds to the overall linear transformation applied by the network. The interested reader is kindly referred to [4, Sec. III] for further details about the algebraic model in (1). According to the above error-free model, it follows that the receiver can obtain the set of transmitted packets X_1, X_2, \dots, X_n from the set of received packets Y_1, Y_2, \dots, Y_N by simply solving the set of linear equations in (1).

Let us now remove the assumption of error-free transmission over the wireless links. A widely adopted channel model for NC is the so-called RLNCC model. It assumes that packet errors might occur in any link of the network and are modeled as “injection packets”. In other words, a corrupted packet is modeled as the addition of an error packet to a genuine packet. According to this model, let us denote by Z_i the error packet applied at link $i \in \{1, 2, \dots, L\}$, with L denoting the number of links of the network (see Fig. 1 for a simple example). By casting the L error packets into a $L \times M$ matrix denoted by Z , the signal received at the destination node can be re-written as follows:

$$Y = AX + BZ \quad (2)$$

where B is a $N \times L$ matrix that corresponds to the overall linear transformation applied, from the link they are injected to the destination node, to Z_1, Z_2, \dots, Z_L . If $Z_i = 0$, there is no injected error to link i . Furthermore, if $Z_i = 0$ for all $i \in \{1, 2, \dots, L\}$, then there are no corrupted packets in the network and (2) reduces to (1).

According to the above RLNCC model, the concepts of erasure, errors, and jamming can also be better clarified for NC. In particular: i) erasures happen when the rank of AX is smaller than the rank of X , and ii) errors and jamming happen when $Z_i \neq 0$ for some $i \in \{1, 2, \dots, L\}$ [14].

3 Error-Correcting Codes in Projective Spaces

In this section, we summarize the main contributions on the analysis and design of error-correcting codes, introduce the concept of code design in projective spaces pioneering conceived in [14], and describe the latest developments in this research area.

The first landmark approaches to the design of error control codes for network-coded systems were presented in [16]–[18]. In those papers, the authors introduced the concept of *network error correction*, whose main idea is to design the network code so that it can be used for error correction. The underlying idea is to exploit the network code for protecting the messages transmitted through the network from distributed errors occurring over the individual links, which are not assumed to be error-free. Network error correction generalizes the usual link-to-link error correction methods adopted in conventional networks. Broadly speaking, the method introduced in [16]–[18] considers the design of a network code as part of an error control problem. Moving from the original idea of network error correction introduced in [16], many subsequent papers investigated that problem with the main aim of computing fundamental performance bounds,

and propose code constructions and efficient decoding algorithms for network error-correcting codes. Notable examples along this line are [13], [19]–[21], and references therein.

A radical shift in paradigm on the design of error-correcting codes for random NC was introduced by Koetter and Kschischang in [14], who conceived the principle of coding for operator channels. This clever idea has originated an active field of research that is also known as error-correcting codes design in projective spaces (see, *e.g.*, [22]). The main idea behind [14] resides in recognizing that the natural transmission model of random NC consists of inputs and outputs that are subspaces of a given vector space. The interesting feature and main difference of the method introduced in [14], with respect to previous approaches available in the literature, is to be oblivious to both the network topology and the particular network code. In other words, the method introduced in [14] seeks to design an outer code that can be applied end-to-end without requiring any modifications on (or even the knowledge of) the underlying network code. The basic idea is to encode the information in the choice, at the transmitter, of a vector space (rather than a vector), and to design, at the receiver, a suitable algorithm to reconstruct the subspace sent by the transmitter in the presence of different kinds of errors.

3.1 How It Works

The theoretic motivation behind the design of error-correcting codes in projective spaces can be captured by using the RLNCC model introduced in Section 2. Let us consider, for ease of illustration, the channel model in (1). Since in random NC the matrix A of the overall linear transformation applied by the network is unknown, it follows that, even in the absence of errors, the only property of the transmitted packets that is kept invariant after propagation through the RLNCC model in (1) is the product AX , which is the row space of X . In other words, from the point of view of the destination node, any of the possible generating sets for the space AX are equivalent. As a consequence, the conventional link-to-link code design, which foresees the transmission of the information via a suitable design of X , needs to be modified and generalized to convey the information via the vector space spanned by the row space of X .

Mathematically speaking, the transmission (and so the encoding via sub-space selection) can be stated as follows. Let $\mathcal{P}_q(n)$ be the projective space of order¹ n over the finite field \mathbb{F}_q , which is defined as the set of all subspaces of the vector space \mathbb{F}_q^n . Let us consider a subspace code $\Omega \subseteq \mathcal{P}_q(n)$, which is a non-empty set of subspaces of $\mathcal{P}_q(n)$, with maximum dimension n . According to [14], the source node selects a subspace $V \in \Omega$ and transmits it over the RLNCC model as the matrix X in (1) and (2), and such that $V = \langle X \rangle$, with $\langle \cdot \rangle$ denoting the row space of X . The destination node receives Y and compute $U = \langle Y \rangle$, from which the transmitted subspace V can be inferred by using the minimum distance decoder

¹ According to Section 2, n is the number of packets injected, for each transmission, in the network by the source node.

as follows:

$$\hat{V} = \arg \min_{V \in \Omega} \{d_S(V, U)\} \quad (3)$$

where $d_S(\cdot, \cdot)$ is the subspace distance defined as:

$$d_S(V, U) = \dim(V) + \dim(U) - 2 \dim(V \cap U) \quad (4)$$

and $\dim(\cdot)$ denotes the dimension of a vector space.

In particular, the minimum distance decoder in (3) guarantees perfect decoding capabilities, *i.e.*, $\hat{V} = V$, if $d_S(V, U) < d_S(\Omega)/2$, where $d_S(\Omega)$ is the minimum subspace distance of the subspace code Ω , *i.e.*:

$$d_S(\Omega) = \min_{\substack{V_1, V_2 \in \Omega \\ V_1 \neq V_2}} \{d_S(V_1, V_2)\} \quad (5)$$

with V_1 and V_2 being arbitrary subspaces in Ω .

3.2 Recent Developments

Besides introducing the principle of error-correcting codes in projective spaces in [14], the authors also introduced a Reed–Solomon–like construction and described a Sudan–style minimum–distance decoding algorithm for the new family of subspace codes. Furthermore, the class of constant–dimension codes was introduced and investigated. Soon after [14], several contributions appeared in the literature with the goal of generalizing and improving the original idea. Relevant results in this research area are [23]–[31]. In [23], the authors study optimal constant–dimension codes for their application to NC, and show that Steiner structures are optimal constant–dimension codes. Two Johnson–type bounds are also computed. In [24], several new codes and bounds for the subspace metric introduced in [14] are derived. In [25], a large class of constant–dimension subspace codes is investigated. It is shown that codes in that class can be easily constructed from rank–metric codes, while preserving their distance properties. Moreover, it is shown that minimum distance decoding of such subspace codes can be reformulated as a generalized decoding problem for rank–metric codes where partial information about the error is available. Furthermore, for the important family of maximum rank–distance codes known as Gabidulin codes, an efficient decoding algorithm is proposed. In [26], the authors construct many new constant–dimension codes with a larger number of codewords than previously known codes. In [28], the authors study bounds and code constructions for the family of codes in [14] targeting the correction of insertions/deletions. In [29], the authors analyze the geometrical properties of rank–metric codes. They derive upper and lower bounds on the minimum cardinality of a code with a given rank covering radius and show that the proposed geometrical properties and bounds can be significant to the design, decoding, and performance analysis of rank–metric codes. In [22], a novel multilevel coding approach to construct codes in the projective space is presented. The method makes usage of four tools: an

appropriate constant-weight code, the reduced row echelon form of a linear subspace, the Ferrers diagram related to this echelon form, and rank-metric codes related to the Ferrers diagram. The authors show that the codes proposed in [14] are a special case of the proposed family of codes. In [30], the error correction problem in both coherent and non-coherent NC is considered under an adversarial model. In particular, as far as non-coherent NC is concerned, the authors introduce a different metric with respect to [14], and prove that it yields a measure of code performance that is more precise, when a non-constant-dimension code is used, than [14]. The new metric is called injection metric. In [27], the authors introduce a Gilbert-Varshamov bound for the codes constructed in [30] according to the definition of injection metric. Moreover, the construction framework in [22] is exploited to obtain new non-constant-dimension codes, which are shown to contain a large number of codewords than comparable codes designed for the subspace metric. Finally, in [31] the authors address the very important problem of understanding if the codes introduced in [14] are feasible and suitable for hardware implementations. They show that the construction of these codes over small fields and limited error-correcting capabilities is not only feasible, but the resulting codes can achieve a high throughput.

4 Joint Network-Channel Decoding

In this section, we provide an overview of a slight different approach to improve the reliability of network-coded wireless architectures. The main idea consists in taking a cross-layer approach and leveraging technologies from the physical and network layers to combat the dominant impairment for an error-free delivery of information over wireless networks, *i.e.*, the channel fading. Of particular interest in the research community is the joint treatment of network and channel coding for improving the end-to-end performance and exploiting in an optimal way the spatial diversity and redundancy in both codes. This research field is motivated by recent results, which have clearly highlighted the fragility of a disjoint design of network and channel coding, as well as network and source coding [32]. By studying some canonical networks, it has been shown that source-channel separation may still hold for some networks, but source-network separation and channel-network separation usually break. Hence, although source coding and channel coding may still be treated separately in some network scenarios, separating routing (or more in general NC) from source or channel coding could fail to bring the desired end-to-end optimality. Furthermore, an end-to-end code design is advocated in [32] as well, where compression, channel coding, and NC/routing might not be separable functions in generic networks. Finally, in [33] it has also been shown that, even though for some networks the separation condition does not break optimality, a separate design of, *e.g.*, source and network codes may yield a higher cost (*e.g.*, may require more power or bandwidth) than their joint optimization.

The application of a joint treatment of network and channel coding finds an important application in lossy networks. As opposite to wireline networks, where it is usually considered that the lower layers deliver error-free or erasure-based links with the help of channel coding, in lossy networks the links are assumed to be error-prone. The principle of a joint design of network and channel codes resides in the exploitation of the redundancy of the network code to help the channel code for a better error protection. In other words, instead of guaranteeing the error-free transmission for each point-to-point link, one is only interested in guaranteeing error-free decoding at the destination nodes. These latter nodes have to decode the data using the input from all incoming links. If they have more than one incoming link, error-free decoding can be possible even if error-free decoding of the point-to-point links is not possible. So, joint network-channel decoding is useful, if the network code contains redundancy. The first practical application of this concept to relay networks is due to Hausl *et al.*, who conceived iterative network and channel decoding methods for the two-way and the multiple-access relay channels in [15] and [34], respectively. The results in these latter papers have evidenced that some performance improvements can be obtained by jointly decoding network and channel codes.

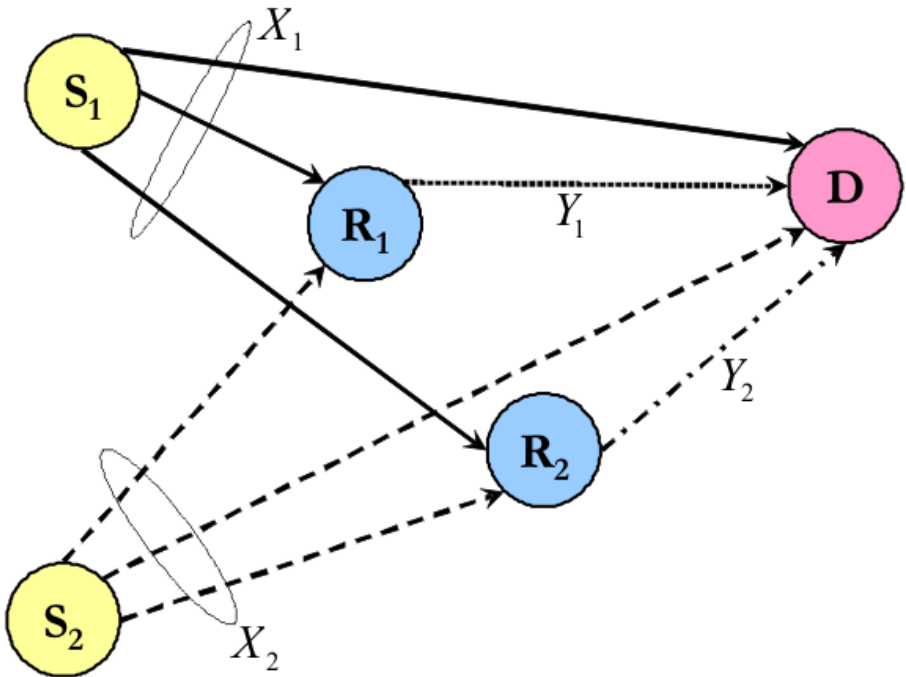


Fig. 2. Example of a two-source and two-relay network topology. Lines with different styles represent the transmission over orthogonal channels to avoid mutual interference.

4.1 Understanding Joint Network–Channel Decoding

Let us provide a simple example for understanding the rationale behind the joint treatment of network and channel coding. Let us consider, *e.g.*, a simple two–source and two–relay topology as shown in Fig. 2. We emphasize here that the conclusions drawn for the simple scheme in Fig. 2 can be extended to the general network topology described in Section 2. However, due to space constraints, this generalization is omitted in this paper. In Fig. 2, two sources S_1 and S_2 transmit two independent packets, of M symbols each, X_1 and X_2 to a common destination node D with the help of two relay nodes R_1 and R_2 , respectively. Let us assume, for the sake of illustration, but without loss of generality, that all links are lossless, and that all communications take place over orthogonal channels such that mutual interferences can be neglected. Accordingly, the destination node will receive four packets from which it wants to infer the information messages, U_1 and U_2 , emitted by the sources.

Let us assume that both packets X_1 and X_2 are channel–coded versions of the information messages U_1 and U_2 (of \tilde{M} symbols each) actually produced by the sources S_1 and S_2 , respectively. We assume general non–binary channel encoding over the field \mathbb{F}_q , as follows:

$$\begin{cases} X_1 = U_1 G_1 \\ X_2 = U_2 G_2 \end{cases} \tag{6}$$

where G_1 and G_2 are the $\tilde{M} \times M$ generator matrices describing the encoding at the source node S_1 and S_2 , respectively.

By assuming error–free links, the relays R_1 and R_2 will ideally be able to retrieve² and to re–encode the messages U_1 and U_2 by performing network and channel coding. For illustrative purposes, let us consider the channel– and network–coded packets emitted by the relays as follows:

$$\begin{cases} Y_1 = a_{11} U_1 G_{11} + a_{12} U_2 G_{12} \\ Y_2 = a_{21} U_1 G_{21} + a_{22} U_2 G_{22} \end{cases} \tag{7}$$

where $\{a_{ij}\}_{i,j=1}^2$ are the coefficients of the network code and $\{G_{ij}\}_{i,j=1}^2$ the generator matrices for error protection.

Accordingly to (6) and (7), the destination node D receives four packets X_1 , X_2 , Y_1 , and Y_2 from which it tries to retrieve the transmitted messages U_1 and U_2 . By using a matrix notation, the following end–to–end equations can be obtained:

$$\begin{bmatrix} X_1 \\ X_2 \\ Y_1 \\ Y_2 \end{bmatrix} = \underbrace{\begin{bmatrix} G_1 & 0 \\ 0 & G_2 \\ a_{11} G_{11} & a_{12} G_{12} \\ a_{21} G_{21} & a_{22} G_{22} \end{bmatrix}}_{G_{\text{joint}}} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \tag{8}$$

² The relays must know G_1 and G_2 and the related parity check matrices for decoding the individual links.

where G_{joint} can be seen as the generator matrix of a joint network–channel code that includes both network and channel coding operations.

The result in (8) clearly highlights that network and channel codes can be regarded, from an end–to–end point of view, as a single integrated code with generator matrix G_{joint} . Accordingly, the messages U_1 and U_2 could be decoded at the destination node by representing the integrated code in (8) by a factor graph and using some iterative decoding methods. This way, network and channel decoding can be performed jointly, by enabling the exchanging of information within and across the received packets to better exploit the redundancy of the network code and achieve some performance improvements. On the contrary, separate network and channel decoding foresee, in general, to exchange the information only within individual packets. A specific iterative decoding algorithm introduced in [35] for the simple network shown in Fig. 2 has clearly evidenced the potential gain of a joint decoding approach.

4.2 Recent Results

Moving from the basic idea in [15] and [34], various studies about the performance improvement of joint network and channel decoding are available in the literature. Most of these studies have the main objective to analyze the effectiveness of such a joint decoding design for the robust and reliable operation of network–coded wireless architectures over lossy networks and to overcome some initial assumptions retained in, *e.g.*, [15]. For example, in [15] ideal error–correcting codes are assumed for the source–to–relay channels, which results in having error–free communication over these links, as well as in introducing a diversity loss since the local channel code blocks the whole frame if just a single bit is erroneous (see, *e.g.*, [36], [37]). Some examples of recent research results addressing the exploitation and the benefits of a joint network–channel code design and decoding can be found in [38]–[48].

5 Concluding Remarks

In this paper, we have provided an overview of two important research fields for the robust design of network–coded wireless architectures over lossy networks: error–correcting code design in projective spaces and joint network–channel iterative decoding. We have clearly shown that both research fields are receiving an upsurge of interest in the research community. However, research in both fields is still at its infancy, and fundamental and open issues need to be still addressed for their practical and effective application to distributed wireless networks.

Acknowledgment

This work is supported by the research projects “New Communications with Network Coding (N2C)” funded by DIGITEO, and “JNCD4CoopNets” funded by CNRS - GDR 720 ISIS.

References

1. Ho, T., Koetter, R., Medard, M., Karger, D.R., Effros, M.: The benefits of coding over routing in a randomized setting. In: *IEEE Int. Symposium Inform. Theory*, p. 442 (June/July 2003)
2. Ahlswede, R., Cai, N., Li, S.-Y.R., Yeung, R.W.: Network information flow. *IEEE Trans. Inform. Theory* 46(4), 1204–1216 (2000)
3. Li, S.-Y.R., Yeung, R.W., Cai, N.: Linear network coding. *IEEE Trans. Inform. Theory* 49(2), 371–381 (2003)
4. Koetter, R., Medard, M.: An algebraic approach to network coding. *IEEE/ACM Trans. Networking* 11(5), 782–795 (2003)
5. Chou, P.A., Wu, Y., Jain, K.: Practical network coding. In: *Allerton Conf. Commun. Control, Computing* (October 2003)
6. Ho, T., Koetter, R., Medard, M., Karger, D.R., Effros, M., Shi, J., Leong, B.: A random linear network coding approach to multicast. *IEEE Trans. Inform. Theory* 52(10), 4413–4430 (2006)
7. Katti, S., Gollakota, S., Katabi, D.: Embracing wireless interference: Analog network coding. In: *ACM SIGCOMM*, pp. 397–408 (August 2007)
8. Katti, S., Katabi, D.: MIXIT: The network meets the wireless channel. In: *ACM HotNets* (November 2007)
9. Katti, S., Katabi, D., Balakrishnan, H., Medard, M.: Symbol-level network coding for wireless mesh networks. In: *ACM SIGCOMM*, pp. 401–412 (October 2008)
10. Katti, S., Rahul, H., Hu, W., Katabi, D., Medard, M., Crowcroft, J.: XORs in the air: Practical wireless network coding. *IEEE/ACM Trans. Networking* 16(3), 497–510 (2008)
11. Fragouli, C., Soljanin, E.: *Network Coding Fundamentals* 2(1) (2007)
12. Fragouli, C., Soljanin, E.: *Network Coding Applications* 2(2) (2007)
13. Zhang, Z.: Linear network error correction codes in packet networks. *IEEE Trans. Inform. Theory* 54(1), 209–218 (2008)
14. Koetter, R., Kschischang, F.R.: Coding for errors and erasures in random network coding. *IEEE Trans. Inform. Theory* 54(8), 3579–3591 (2008)
15. Hausl, C., Hagenauer, J.: Iterative network and channel decoding for the two-way relay channel. In: *IEEE Int. Commun. Conf.*, pp. 1568–1573 (June 2006)
16. Cai, N., Yeung, R.W.: Network coding and error correction. In: *IEEE Inform. Theory Workshop*, pp. 119–122 (October 2002)
17. Yeung, R.W., Cai, N.: Network error correction, part I: Basic concepts and upper bounds. *Commun. Information Systems* 6(1), 19–36 (2006)
18. Cai, N., Yeung, R.W.: Network error correction, part II: Lower bounds. *Commun. Information Systems* 6(1), 37–54 (2006)
19. Matsumoto, R.: Construction algorithm for network error-correcting codes attaining the singleton bound. *IEICE Trans. Fundamentals* E90–A(9), 1–7 (2007)
20. Jaggi, S., Langberg, M., Katti, S., Ho, T., Katabi, D., Medard, M., Effros, M.: Resilient network coding in the presence of byzantine adversaries. *IEEE Trans. Inform. Theory* 54(6), 2596–2603 (2008)
21. Balli, H., Yan, X., Zhang, Z.: On randomized linear network codes and their error correction capabilities. *IEEE Trans. Inform. Theory* 55(7), 3148–3160 (2009)
22. Etzion, T., Silberstein, N.: Error-correcting codes in projective spaces via rank-metric codes and Ferrers diagrams. *IEEE Trans. Inform. Theory* 55(7), 2909–2919 (2009)

23. Xia, S.-T., Fu, F.-W.: Johnson type bounds on constant dimension codes. *Designs, Codes, and Cryptography* 50(2), 163–172 (2008)
24. Gabidulin, E.M., Bossert, M.: Codes for network coding. In: *IEEE Int. Symposium Inform. Theory*, pp. 867–870 (July 2008)
25. Silva, D., Kschischang, F.R., Koetter, R.: A rank–metric approach to error control in random network coding. *IEEE Trans. Inform. Theory* 54(9), 3951–3967 (2008)
26. Kohnert, A., Kurz, S.: Construction of large constant dimension codes with a prescribed minimum distance. *LNCS*, pp. 31–42. Springer, Heidelberg (December 2008)
27. Khaleghi, A., Kschischang, F.R.: Projective space codes for the injection metric, arXiv.org (April 2009), http://arxiv.org/PS_cache/arxiv/pdf/0904/0904.0813v2.pdf
28. Ahlswede, R., Aydinian, H.: On error control codes for radom network coding. In: *IEEE Int. Workshop Network Coding Theory and Applications*, pp. 68–73 (June 2009)
29. Gadouleau, M., Yan, Z.: Bounds on covering codes with the rank metric, arXiv.org (June 2009), http://arxiv.org/PS_cache/arxiv/pdf/0809/0809.2968v2.pdf
30. Silva, D., Kschischang, F.R.: On metrics for error correction in network coding, arXiv.org (August 2009), http://arxiv.org/PS_cache/arxiv/pdf/0805/0805.3824v4.pdf
31. Chen, N., Gadouleau, M., Yan, Z.: Rank metric decoder architectures for noncoherent error control in random network coding. In: *IEEE Workshop Sig. Process. Systems* (October 2009)
32. Effros, M., Medard, M., Ho, T., Ray, S., Karger, D., Koetter, R.: Linear network codes: A unified framework for source, channel and network coding. In: *DIMACS Workshop Network Information Theory* (March 2003)
33. Lee, A., Medard, M., Haigh, K., Gowan, S., Rubel, P.: Minimum–cost sub–graphs for joint distributed source and network coding. In: *IEEE Workshop Network Coding, Theory and Applications* (January 2007)
34. Hausl, C., Dupraz, P.: Joint network–channel coding for the multiple–access relay channel. In: *IEEE Commun. Society on Sensor and Ad Hoc Commun. and Networks*, pp. 817–822 (September 2006)
35. Guo, Z., Huang, J., Wang, B., Cui, J.-H., Zhou, S., Willett, P.: A practical joint network–channel coding scheme for reliable communication in wireless networks. In: *ACM Int. Symposium on Mobile Ad Hoc Networking and Computing*, pp. 279–288 (May 2009)
36. Xiao, L., Fuja, T.E., Kliever, J., Costello, D.: A network coding approach to cooperative diversity. *IEEE Trans. Inform. Theory* 53(10), 3714–3722 (2007)
37. Al-Habian, G., Ghayeb, A., Hasna, M.: Controlling error propagation in network–coded cooperative wireless networks. In: *IEEE Int. Commun. Conf.*, pp. 1–6 (June 2009)
38. Bao, X., Li, J.: A unified channel–network coding treatment for user cooperation in wireless ad–hoc networks. In: *IEEE Int. Symposium Inform. Theory*, pp. 202–206 (July 2006)
39. Gowaikar, R., Dana, A.F., Hassibi, B., Effros, M.: A practical scheme for wireless network operation. *IEEE Trans. Commun.* 55(3), 463–476 (2007)
40. Zhang, S., Zhu, Y., Liew, S.C., Letaief, K.B.: Joint design of network coding and channel decoding for wireless networks. In: *IEEE Wireless Commun. Conf.*, pp. 779–784 (March 2007)

41. Nguyen, H.T., Nguyen, H.H., Le-Ngoc, T.: A joint network–channel coding scheme for relay–based communications. In: IEEE Canadian Conf. Electrical and Computer Engineering, pp. 904–907 (April 2007)
42. Yang, S., Koetter, R.: Network coding over a noisy relay: A belief propagation approach. In: IEEE Int. Symposium Inform. Theory, pp. 801–804 (June 2007)
43. Kliewer, J., Dikaliotis, T., Ho, T.: On the performance of joint and separate channel and network coding in wireless fading networks. In: IEEE Workshop Inform. Theory for Wireless Networks, pp. 1–5 (July 2007)
44. Thobaben, R.: Joint network/channel coding for multi–user hybrid–ARQ. In: Int. ITG Conf. Source and Channel Coding, pp. 1–6 (January 2008)
45. Xu, X., Flanagan, M.F., Goertz, N.: A shared–relay cooperative diversity scheme based on joint channel and network coding in the multiple access channel. In: IEEE Int. Symposium Turbo Codes and Related Topics, pp. 243–248 (September 2008)
46. Bing, D., Jun, Z.: Design and optimization of joint network–channel LDPC code for wireless cooperative communications. In: IEEE Singapore Int. Conf. Commun. Systems, pp. 1625–1629 (November 2008)
47. Li, Q., Ting, S.H., Ho, C.K.: Joint network and channel coding for wireless networks. In: IEEE Conf. Sensor, Mesh and Ad Hoc Communications and Networks, pp. 1–6 (June 2009)
48. Li, Y., Song, G., Wang, L.: Design of joint network–low density parity check codes based on the EXIT charts. *IEEE Commun. Lett.* 13(8), 600–602 (2009)