A Framework for Joint Multiple Description Coding and Network Coding over Wireless Ad-hoc Networks

Irina Delia Nemoianu, Claudio Greco, Marco Cagnazzo, Béatrice Pesquet-Popescu
TELECOM-ParisTech, TSI department, 46 rue Barrault, 75634 Paris Cedex 13, FRANCE
{nemoianu,greco,cagnazzo,pesquet}@telecom-paristech.fr

***Abstract*—Network coding (NC) can achieve the maximum information flow in the network by allowing nodes to combine received packets before retransmission. Several works have shown NC to be beneficial in mobile ad-hoc networks, but the delay introduced by buffered decoding raises a problem in real-time streaming applications. Here, we propose to use NC jointly with multiple description coding (MDC) to allow instant decoding of the received packets. The optimal encoding coefficients are chosen via distributed optimisation of the expected video quality. Nodes receive up-to-date information about the network topology through a recently proposed protocol, originally designed for real-time streaming of MDC video. Results show that, due to the limitations imposed by instant decoding to the coding window size, our approach consistently outperforms the popular technique of random linear network coding.**

***Index Terms*—Network coding, multiple descriptions, video coding, mobile ad-hoc networks.**

I. Introduction

Network Coding (NC) [1–4] has recently received a lot of interest from the research community for its potential to become a very efficient alternative to classical routing. With this technique, messages sent over a node's output link are not just copies, but combinations of messages received on the node's input links.

Ahlswede *et al.* [1] have shown that, by allowing coding at the intermediate nodes, information can be multicast at a rate approaching the smallest minimum cut between the source and any of the receivers, which is not possible in traditional routing. Moreover, Li *et al.* [2] have shown that, in *directed acyclic graph* (DAG) networks, the multicast capacity can be achieved by linear coding as long as an alphabet with a large enough size is used. For general networks, similar results can be obtained by randomly choosing the coding coefficients from a finite field [3]. *Random Linear Network Coding* (RLNC) is a distributed random NC scheme introduced by Chou *et al.* [4], where the data stream is divided into generations, with each generation grouped into G data packets. The source generates G new data packets by prepending the G-dimensional i-th unit vector to the corresponding i-th data packet. In this way, when intermediate nodes generate a new packet by linearly combining their input packets using random coefficients, the G-dimensional overhead will correspond to the global encoding vector. Intermediate nodes store the received packets into buffers sorted by generation number and the packet to be sent is generated as a random combination of all the packets in the current generation. The original packets are obtained at the receivers by performing Gaussian elimination as soon as G independent coding vectors have been received.

One class of networks where NC can prove itself a viable tool is that of *mobile ad-hoc networks* (MANETs). A MANET is a dynamic network of mobile devices connected by wireless links, self-organised in a mesh topology [5]. Thanks to their properties of flexibility, ease of deployment and robustness, they are fundamental in design of video streaming applications in environments without preexisting infrastructure, such as military or disaster-relief applications. There exist several NC techniques specifically designed for wireless networks, *e.g.*, the COPE protocol [6], that exploits the broadcast medium by allowing nodes to perform *opportunistic listening* and *opportunistic NC*, *i.e.*, the mobile nodes listen to all transmissions in their neighbourhood, store all the packets even though they are not intended for them, and use these packets to construct coded packets or to decode received packets.

Some of the approaches proposed for video delivery [7, 8], based on RLNC, imply dividing the stream into layers of priority and providing *unequal error protection* (UEP) for the different layers. Because such methods need to ensure that all end-users receive at least the base layer, all received packets must be stored in a buffer, until a sufficient number of independent coding vector are received, which introduces in the decoding process a delay unacceptable in real-time streaming applications. In order to avoid this delay, we employ NC jointly with *multiple description coding*. Multiple description coding (MDC) is a paradigm based on splitting a media content into d sub-streams, referred to as *descriptions*. Any description can be independently decoded for representing the content, but the quality improves with the number of descriptions. The quality can be expected to be roughly proportional to the bitrate sustained by the receiver. MDC is considered a valuable tool to cope with packet losses in MANETs [9].

Joint use of MDC and RLNC has been pointed out as a viable technique for video multicast with delay constraints. Iwaza *et al.* [10] recently proposed a coding scheme for multiple description network-coded data. Our work integrates this approach by focusing on the optimisation of the overlay

network constructed to deliver the streams.

There exist works [11] proposing the construction of an overlay that is optimal with respect to the quality of service provided to the users. However, these works employ a centralised method, wherein the server must have complete information about the number of descriptions obtained by each user. Since this cannot always be applied in the case of multi-hop wireless networks with node mobility, we propose a novel approach that aims to achieve similar results by means of local optimisation only. The main challenge with this approach is to define an objective function that each node can compute locally, *i.e.*, using only information available within its neighbourhood. This is nonetheless coherent with the global performance of the network, keeping into account the particular rate-distortion properties of multiple description coded video. Also, we have to guarantee both that the nodes have up-to-date information about the overlay topology and that the protocol overhead is minimised.

The rest of this work is organised as follows. In Sec. II, we define our model and derive a local optimisation problem that maximises the video quality perceived by the users. We integrate this framework in a recently proposed overlay construction protocol in order to exploit its capability of maintaining the overlay in an efficient fashion. Then, in Sec. III we provide an experimental validation of the proposed approach and compare its performance both with the RLNC technique and with the theoretical bound. Finally, in Sec. IV, we draw conclusions and outline future work.

II. Proposed approach

In this section, we model the diffusion of a content, encoded in d descriptions, on a MANET, from a single sender to a multitude of receivers. We assume that each node of the network will contribute to the diffusion, but it will be able to sustain an up-link bitrate sufficient for only one description.

We start our discussion from the model proposed by Chou *et al.* [4], where $G = (V, E)$ is a directed acyclic graph having unit capacity and representing the *communication network*. Let $s \in V$ be the sender, or *source node* and $T \subset V$ the set of receivers, or *peers*. In our scenario, without loss of generality we assume that the set of receivers includes any node except the source. At each sending opportunity, the source has to broadcast a video frame x, encoded in packets $x_0, x_1, \ldots, x_{d-1}$ (one per description), to each peer in the network. In the standard model, each transmitted symbol is associated to the edge (or *channel*) it is carried over. Let us define $y(e_{ij})$ the symbol carried over channel e_{ij}. Since we want to model a broadcast network, we need to introduce a further constraint. Namely, we want that each node n_i transmits the same symbol, that we denote y_i, over all its outgoing channels. Once all symbols y_i are assigned, it is possible to revert to the standard model by imposing $y(e_{ij}) = y_i,\ \forall e_{ij} \in E$.

Imposing that for all nodes the same symbol is sent on all unitary capacity channels raises a problem: the source would be constrained to send packets from just one out of d descriptions. To avoid this problem, we can model the video source as a set S of d virtual sources, each having a copy of the outgoing channels of s and emitting exactly one description of x over all its channels, thus maintaining uniformity in the unit capacity of channels: $y_i = x_i,\ \forall i \in \{0, 1, \ldots, d-1\}$. Set V, by our definition, does not contain the "original" source and contains instead the set S. In this model, there are thus $N = |V|$ nodes, d virtual sources and $N - d$ peers.

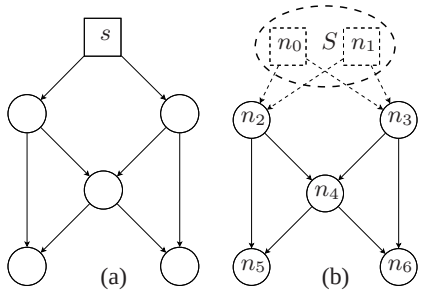


Figure 1. (a) Butterfly wireless network. (b) Equivalent network with virtual sources and node labelling.

Symbols emitted by a node $n_i \in T$ must be linear combinations of the symbols carried over the channels entering n_i. Let us define $\mathcal{I}_i$ the set of nodes n_j such that a channel exists from n_j to n_i. The symbol emitted by n_i will be in the form:

$$y_i = \sum_{n_j \in \mathcal{I}_i} m_i(n_j) y_j, \qquad \forall n_i \in T. \tag{1}$$

The *local encoding vector* $\mathbf{m}(n_i) = [m_i(n_j)]_{n_j \in \mathcal{I}_i}$ represents the encoding function of node n_i along all its outgoing channels $e_{ik} \in E$. The source does not have any entering channels, but this does not pose a problem, as its emitted symbol can be assigned beforehand.

Let us define the height of a node $h(n_i)$ as the length of the *longest* finite path in G from any node in S to node n_i; this induces a partial order on set V, well defined, as G is acyclic. Let us label the nodes with indices $0, 1, \ldots, N-1$ such that $i < j \implies h(n_i) \leq h(n_j)$ (see Fig. 1). It follows that the nodes in S are labelled $n_0, n_1, \ldots, n_{d-1}$.

This order is consistent with the propagation of packets outgoing from the source. If we define an *encoding matrix* M as follows:

$$M_{ij} = \begin{cases} m_i(n_j) & \text{if } e_{ji} \in E, \\ 0 & \text{otherwise,} \end{cases}$$

we can rewrite equation (1) as:

$$y_i = \sum_{j<i} M_{ij} y_j \qquad \forall i \in \{d, d+1, \ldots, N-1\},$$

while for the virtual sources we impose $y_i = x_i,\ \forall i \in \{0, 1, \ldots, d-1\}$.

We are interested in the number of packets a node is able to decode. Let us define $\mathbf{x}$ as a column vector that has for components the packets encoding x, *i.e.*, $\mathbf{x} = [x_i],\ \forall i \in \{0, 1, \ldots, d-1\}$. Since any emitted symbol y_i is a linear combination of packets from all the descriptions, there exists a row

weight vector $\mathbf{w}_i$ with d components such that $y_i = \mathbf{w}_i \cdot \mathbf{x}$. The vector $\mathbf{w}_i$ for nodes in S has only one non-zero component, corresponding to the description emitted, while for nodes in T it can be inferred from matrix M:

$$\mathbf{w}_i = \sum_{j<i} M_{ij}\mathbf{w}_j \qquad \forall i \in \{d, d+1, \ldots, N-1\}.$$

By collecting symbols and weight vectors on its entering channels, a node n_i is able to construct a system of linear equations $W_i\mathbf{x} = \mathbf{y}_i$, where W_i is a *weight matrix* obtained by horizontal concatenation of vectors $\mathbf{w}_j$, and $\mathbf{y}_i$ is a column vector with components equal to y_j, for all j such that $e_{ji} \in E$.

A node n_i is able to perform *central decoding*, *i.e.*, to decode all d descriptions, if and only if $\mathbf{rank}(W_i) = d$. However, the rank is not a reliable tool to estimate the number of descriptions used in *side decoding*, *i.e.*, decoding only a subset of the d descriptions. For instance, a node n_i having $\mathbf{rank}(W_i) = 1$ could be receiving an equation in the form $y_i = w_{i,0}x_0$, which is trivial and allows the decoding of x_0. But it could also be receiving $y_i = w_{i,0}x_0 + w_{i,1}x_1$, which is impossible to solve without further information.

Let us assume we have an operator $\mathbf{dec}(W_i)$ able to infer how many descriptions a node n_i will be able to decode, given its weight matrix W_i. This operator can be easily implemented in practice by counting the number of trivial equations. We also define a *value* operator $\varphi(\cdot)$ that associates a quality metric to a number of decoded descriptions $\mathbf{dec}(W_i)$. The choice of the quality metric will depend of course on the requirements of the application. If we use, for instance, the expected PSNR, $\varphi(\cdot)$ should reflect the fact that the PSNR gap between a node not receiving any description and one receiving just one is bigger than the gap between a node receiving one description and one receiving two, *i.e.*, $\varphi(1) - \varphi(0) \geq \varphi(2) - \varphi(1)$.

Given this model, our approach is straightforward. At each sending opportunity, each node n_i inspects the state of the buffers of its neighbours, then it chooses an optimal weight vector w_i^* as:

$$w_i^* = \arg\max_{w \in \mathcal{W}_i} \left\{ J(w) = \sum_{j \in \mathcal{N}_i} \varphi\left(\mathbf{dec}(W_j)\right) \right\}, \qquad (2)$$

where $\mathcal{W}_i$ is the set of weighting vectors available to n_i, and $\mathcal{N}_i$ is the set of indices of the neighbours of n_i. Notice that $\mathcal{W}_i$ is restricted by the symbols actually received by n_i: a node can only choose to send one of the packets it received or a combination thereof. The optimisation of the emitted symbol is independent from the decoding capability, *i.e.*, even if a node is unable to decode any description (*e.g.*, if it is receiving just a combination $w_0x_0 + w_1x_1$), it could choose to forward a combination it received if that would benefit its neighbours.

In Fig. 2 we present an example of optimisation performed by a node n_1. Here, $J([1,0]) = \varphi(2) + \varphi(2) + \varphi(1)$, as only the rank of W_3 is affected by the combination, $J([0,1]) = \varphi(2) + \varphi(1) + \varphi(2)$ as only the rank of W_4 is affected, and $J([1,1]) = \varphi(2) + \varphi(2) + \varphi(2)$ as both the rank of W_3 and W_4 are affected. Since $\varphi(2) > \varphi(1)$, the optimal choice is $w_1^* = [1,1]$.

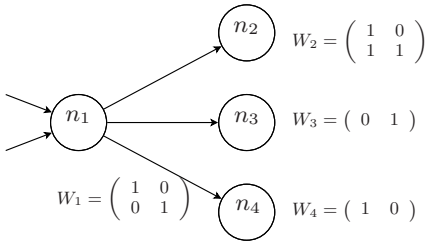


Figure 2. Example of optimisation of the weight vector of a node n_1.

There are two main challenges that need to be dealt with in order to use this approach: firstly, a mobile ad-hoc network is hardly a DAG; secondly, the nodes need to inspect the buffer state of their neighbours in order to solve the optimisation problem (2). We therefore apply the algorithm not directly on the ad-hoc network, but rather on an *overlay network* built on top of it. We generate a DAG overlay using the ABCD protocol [12], a cross-layer protocol for content delivery over MANETs. ABCD forms an overlay consisting in the superposition of d different diffusion trees, one per each description, and is therefore acyclic. Also, the control messages sent in order to build and maintain the overlay can be used to propagate information about the local buffer W_i from node n_i to its neighbours. The ABCD protocol also ensures that the nodes have an up-to-date view of the topology, even in presence of node mobility and churn, and minimises the protocol overhead.

III. Experimental Results

In the following, we present the results of the proposed technique and compare them with the result achievable via the RLNC scheme proposed by Chou *et al.* [4].

To generate the DAG, we randomly construct a MANET consisting of 100 nodes in a $100 \times 100\,\mathrm{m}^2$ playground. The nodes have a nominal transmission range of $25\,\mathrm{m}$, then the ABCD protocol [12] is run in order to form a directed acyclic overlay. When the source starts broadcasting the stream, the proposed coding strategy is applied at each node. Observing which packets have been decoded, we compute average Y-PSNR observed by the various users.

In order to generate the video content we encoded the "Foreman" video sequence (CIF, 30 fps, 288 frames) using a recently proposed MD coder [13]. The stream is encoded in two descriptions, $d = 2$, balanced in terms of rate-distortion properties.

The test is repeated 100 times, with different initial position of the nodes, in order to take into account the variability of the network topology. Several tests have been performed with other video contents, with similar results. By lack of space, we only present in Fig. 3 the results for the average PSNR obtained with our method and RLNC. A theoretical bound, obtained by exhaustive exploration of the solutions, is also reported for reference.

For the RLNC implementation we assumed that the coding window cannot be set along the time axis, *i.e.*, we do not mix

packets with different due-dates, in order to avoid decoding delay, crucial in real-time applications. Therefore, combination of packets can only occur along the descriptions axis, *i.e.*, we mix packets from different descriptions, but having the same due-date. This implies that the length of the coding window equals the number d of descriptions, which is 2 in our test. The coding coefficients are chosen randomly in the Galois Field of size 256 (*i.e.*, $\mathrm{GF}(256)$), which has been shown [4] to give a low probability of building duplicate packets. In our RLNC scheme each node uses a forwarding factor of $75\,\%$.

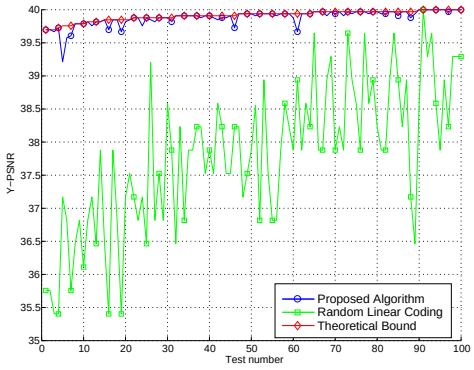


Figure 3. Comparison between the reference and the proposed technique, for video sequence "foreman", CIF, 30 fps, 1.8 Mbps.

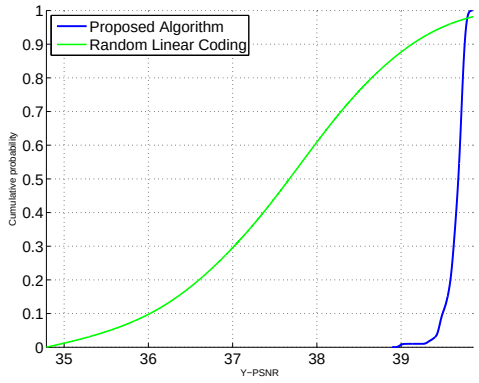


Figure 4. Comparison of PSNR CDFs for the same conditions as Fig. 3.

We notice that the proposed technique performs on average about 2 dB better than RLNC. Moreover in Fig. 4, we report the PSNR cumulative distribution functions for the two techniques. We observe that while in the reference technique the distribution of PSNR is very widespread, using our technique almost all nodes achieve very close, and very high qualities. We ascribe this result to the fact that the efficiency of RLNC is considerably affected by the length of the coding window (*i.e.*, the size of the generation).

To achieve these qualities, using the reference technique all nodes collectively injected into the network $4.6 \cdot 10^3$ packets per second on the average, where as with our technique, only $1.2 \cdot 10^3$ packets are sent. We also notice that from one experiment to another, these values do not vary much.

We conclude that in this particular scenario, with stringent constraints on delay, an optimisation technique outperforms a technique based on random coefficients, providing a better video quality to the end users.

IV. Conclusions

In this work we presented a novel technique of video diffusion over ad-hoc networks based on the joint use of network coding and multiple descriptions coding.

First, we formulated the problem of broadcasting a video stream encoded in multiple descriptions on an ad-hoc network in terms of finding an optimal set of combination coefficients. Then, we introduced an objective function that takes into account the effect that decoding a given number of description has on the total distortion. This framework has been integrated with a recently proposed cross-layer protocol that provides both an acyclic overlay network and knowledge of the neighbours' state. Finally, we compared the performance of our technique with the well-known random linear coding technique. We observe that the limitations of the generation size to the number of descriptions, imposed by the delay constraints, severely affect the performance of the reference technique, which as a result is consistently outperformed by the proposed approach. The result we obtained suggests that further research in this direction could be promising, in particular in the direction of a joint design of the overlay management protocol and the optimal choice of network coding coefficients.

References

[1] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inf. The.*, vol. 46, no. 4, pp. 1204–1216, 2000.

[2] S.-Y. Li, R. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inf. The.*, vol. 49, no. 2, pp. 371–381, 2003.

[3] T. Ho, M. Médard, J. Shi, M. Effros, and D. Karger, "On randomized network coding," in *IEEE Intern. Symp. Inform. Theory*, vol. 41, no. 1, 2003, pp. 11–20.

[4] P. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Allerton Conf. on Commun. Control and Comput.*, 2003.

[5] M. Frodigh, P. Johansson, and P. Larsson, "Wireless ad-hoc networking: the art of networking without a network," *Ericsson Review*, vol. 4, pp. 248–263, 2000.

[6] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: practical wireless network coding," *ACM SigComm Computer Comm. Rev.*, vol. 36, pp. 243–254, August 2006.

[7] N. Thomos, J. Chakareski, and P. Frossard, "Prioritized distributed video delivery with randomized network coding," *IEEE Trans. Multimedia*, vol. 13, no. 99, pp. 776–787, 2011.

[8] D. Vukobratović and and V. Stanković, "Unequal error protection random linear coding for multimedia communications," in *IEEE Worksh. Multim. Sign. Proc.*, Saint-Malo, France, Oct. 2010, pp. 280–285.

[9] Y. Wang, A. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proc. IEEE*, vol. 93, no. 1, pp. 57–70, 2005.

[10] L. Iwaza, M. Kieffer, L. Liberti, and K. Al-Agha, "Joint decoding of multiple-description network-coded data," in *IEEE Intern. Symp. on Network Cod.*, Hangzhou, China, Nov. 2011.

[11] A. Ramasubramonian and J. Woods, "Multiple description coding and practical network coding for video multicast," *IEEE Signal Proc. Lett.*, vol. 17, no. 3, pp. 265–268, March 2010.

[12] C. Greco and M. Cagnazzo, "A cross-layer protocol for cooperative content delivery over mobile ad-hoc networks," *Inderscience Intern. J. of Comm. Networks and Distrib. Syst.*, vol. 7, no. 1–2, pp. 49–63, 2011.

[13] C. Greco, G. Petrazzuoli, M. Cagnazzo, and B. Pesquet-Popescu, "An mdc-based video streaming architecture for mobile networks," in *IEEE Worksh. Multim. Sign. Proc.*, Hangzhou, China, October 2011.