

Network Coding of Rateless Video in Streaming Overlays

Nikolaos Thomos, *Member, IEEE*, and Pascal Frossard, *Senior Member, IEEE*

Abstract—We present a system for collaborative video streaming in wired overlay networks. We propose a scheme that builds on both rateless codes and network coding in order to improve the system throughput and the video quality at clients. Our hybrid coding algorithm permits to efficiently exploit the available source and path diversity without the need for expensive routing nor scheduling algorithms. We consider specifically an architecture where multiple streaming servers simultaneously deliver video information to a set of clients. The servers apply Raptor coding on the video packets for error resiliency, and the overlay nodes selectively combine the Raptor coded video packets in order to increase the packet diversity in the system. We analyze the performance of selective network coding and describe its application to practical video streaming systems. We further compute an effective source and channel rate allocation in our collaborative streaming system. We estimate the expected symbol diversity at clients with respect to the coding choices. Then we cast a *minmax* quality optimization problem that is solved by a low-cost bisection based method. The experimental evaluation demonstrates that our system typically outperforms Raptor video streaming systems that do not use network coding as well as systems that perform decoding and encoding in the network nodes. Finally, our solution has a low complexity and only requires small buffers in the network coding nodes, which are certainly two important advantages toward deployment in practical streaming systems.

Index Terms—Network coding, optimization, overlay networks, p2p streaming, Raptor codes, rate allocation.

I. INTRODUCTION

THE LAST DECADE has witnessed the rapid development of novel overlay network architectures such as peer-to-peer or wireless mesh networks that offer source and path diversity to data delivery applications. Network diversity permits to augment the quality of service for media streaming applications, with increased throughput and improved resilience to failures. Streaming applications can typically benefit from the network diversity by designing multi-path or multi-source streaming strategies. Such algorithms, however, generally rely on appropriate source coding, routing mechanisms or coordination of the network peers in order to prevent packet duplicates that cause waste of resources. The need for coordination

can, however, be avoided if the overlay network nodes are able to perform simple processing or coding operations on the packet streams. Network coding [1] can prove to be particularly useful in overlay streaming applications, where it could help to increase the network throughput and data persistence.

In this paper, we propose a network coding algorithm for video streaming in lossy overlay networks with low dynamics. We consider a framework where a compressed video sequence is sent from possibly multiple servers to multiple clients via several network peers which are able to perform network coding operations by combination of packets. This framework is illustrated in Fig. 1. The overlay nodes are organized in a mesh network, which is generally more appropriate than tree-based overlays for streaming applications with stringent timing constraints. Efficient solutions for streaming on overlay networks have to prevent important packet duplication for effective resources usage. Network coding techniques bring interesting benefits in such a context [2], since they permit to take advantage of the network diversity. Hence they lead to increased goodput without the need for strict coordination between peers. The video packets are simply combined together in the overlay nodes and then forwarded to the children nodes toward the clients. Compared to successive decoding and re-coding operations in the network nodes [3], network coding typically leads to lower end-to-end delay and involves smaller computational complexity in the network nodes.

We combine network coding with rateless coding at the source. As shown initially in [4], this novel hybrid coding strategy permits to exploit both the source and path diversity for improved quality at the clients. In particular, we first encode the source packets at the servers with Raptor codes [5] that offer interesting rateless properties and low complexity. When several servers are used, the encoding process forms disjoint sets of packets at each server, in order to provide additional robustness to failures and losses. After Raptor encoding, the packets are sent toward the clients via overlay nodes. The overlay nodes implement a new network coding algorithm by combining Raptor symbols. The coding strategy depends on the set of received packets and the output bandwidth. The selective combination of packets in a node permits to increase the symbol diversity in the overlay network that proves to be particularly efficient to combat packet erasures. In addition, selective network coding¹ in the nodes only necessitates the

Manuscript received October 13, 2009; revised February 9, 2010 and June 11, 2010; accepted June 16, 2010. Date of publication October 18, 2010; date of current version January 22, 2011. This work was supported by the Swiss National Science Foundation, under Grant PZ00P2-121906. This paper was recommended by Associate Editor T. Nguyen.

The authors are with the Signal Processing Laboratory, Swiss Federal Institute of Technology, Lausanne 1015, Switzerland (e-mail: nikolaos.thomos@epfl.ch; pascal.frossard@epfl.ch).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2010.2087830

¹Throughout the paper “selective network coding” and “Raptor network coding” are used interchangeably.

use of small buffers and hence induces small latency. This is a clear advantage compared to rateless coding schemes that decode packets in the overlay peers (e.g., [6]). We then show how our novel collaborative method can be implemented in practical video streaming solutions. We further address the source and parity rate allocation problem and determine efficient coding strategies under given network statistics. The optimization problem is formulated as a *minmax* problem, and the presented algorithm seeks for the optimal tradeoff between source distortion and resiliency to loss such that the goodput is maximized for the less reliable client. We evaluate the performance of our system in various random network topologies, and we show that it outperforms Raptor coding systems that do not utilize peers for network coding. Our system also leads to low latency and small computational complexity in the network nodes, which certainly represent important characteristics in practical streaming systems.

This paper is arranged as follows. In Section II, we give an overview of Raptor codes and describe the selective network coding algorithm applied to Raptor streams. In Section III, we present the application of the novel network coding algorithm to compressed video sequences. We propose a source and channel rate allocation algorithm that minimizes the maximal distortion at the clients. Simulation results are proposed in IV for a variety of scenarios that demonstrate the benefits of the proposed solution. The related work is discussed in Section V. Finally, conclusions are drawn in Section VI.

II. NETWORK CODING OF RAPTOR STREAMS

A. Overview of Raptor Codes

Our collaborative streaming system proposes to use Raptor codes [5] at sources and network coding in overlay nodes in order to combat packet erasures and augment the network throughput. Maximum distance separable codes such as Reed-Solomon codes are often used in packet erasure networks, but their application is generally limited to small coding blocks due to their decoding complexity that typically grows quadratically with the block size. On the other hand, the family of Fountain codes and specifically Raptor codes has received a lot of interest recently to overcome these limitations, since they provide advanced performance with reduced complexity. In addition, they have an implicit coding structure that permits to generate encoded packets on the fly in a flexible way. They typically combine packets to generate encoded symbols, in a similar way to the randomized network coding systems [7] that perform simple linear operations with packets. We present below an overview of Raptor codes, and their application in our network coding framework.

Raptor codes [5] are based on Luby transform (LT) codes [8] and designed to achieve linear encoding-decoding time complexity. They typically concatenate weakened LT codes with outer codes (pre-coding step). In the LT coding stage, the source symbols are XOR-ed for generating the encoded symbols, and the implicit structure of the corresponding Tanner graph is passed to the decoder as a decoding key along with each of the encoded symbols. This key, also called the encoded symbol ID (ESI) represents the seed of a pseudo-

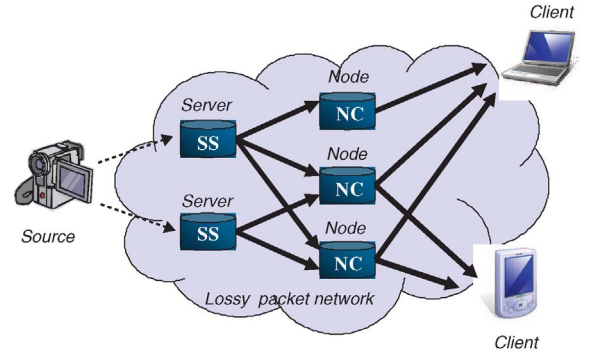


Fig. 1. Illustration of a system for video streaming on overlay networks. Multiple streaming servers (SS) send information to video clients on a lossy packet network via network coding (NC) peers. Each network segment is typically characterized by an available bandwidth and a packet loss probability.

random generator that determines both the degree of the encoded symbols as well as the source symbols to be combined. Raptor codes perform close to perfect codes as they are able to recover the source symbols from any set of encoded symbols that is slightly larger than the set of encoded symbols. Hence, they have a very small overhead that further tends to zero when the number of encoded symbols in a block increases. This rateless property is particularly attractive for streaming applications with strict timing constraints. Raptor codes have been proposed for peer-to-peer multimedia delivery [6] and wireless broadcast systems [9]. The Raptor codes have also been recently adopted in the 3GPP standard [10].

We describe now in more detail the 3GPP implementation that will be used in this paper. In this implementation, the pre-coding step consists of regular LDPC codes and high density Half codes. This encoding stage enables low complexity encoding and decoding, as it permits the use of LT codes with sparse parity check matrices. In more detail, the encoding procedure grows as follows. Let $\mathbf{C}$ denote a source vector of K input (source) symbols with $\mathbf{C} = [C_1^T C_2^T \dots C_K^T]$. If we represent as $\mathbf{C}_P$ the redundant pre-coded symbols, the coding constraints among source and redundant symbols are given by

$$\mathbf{C}_P = \mathbf{G}_P \cdot \mathbf{C}$$

where $\mathbf{G}_P$ is the generator matrix of the pre-coder. The LT encoder then takes as input the pre-coding symbols $\mathbf{F}_{1:L} = [\mathbf{C}^T \mathbf{C}_P^T]^T$ and generates the Raptor symbols, where $L = K + S + H$ is the number of pre-coded symbols, and S and H are, respectively, the number of LDPC and Half codes constraints. If $\mathbf{G}_{LT}$ is the generator matrix of non-systematic LT encoder, then it holds that

$$\mathbf{E}_{1:N} = [\mathbf{E}_1^T \mathbf{E}_2^T \dots \mathbf{E}_N^T]^T = \mathbf{G}_{LT}(1, 2, \dots, N) \cdot \mathbf{F}$$

where N is the number of Raptor encoded symbols. The generator matrix of the LT coder $\mathbf{G}_{LT}$ is defined as

$$\mathbf{G}_{LT} = [\Gamma_1^T \Gamma_2^T \dots \Gamma_n^T]$$

where Γ_i^T with $i = 1, \dots, n$ is a row vector with ones at the positions corresponding to the indices of the pre-coded symbols that are combined in the generation of encoded symbols.

Therefore, each encoded symbol E_i , with $i = 1, \dots, n$, can be written as

$$E_i = \sum_{j \in \Psi_i} \bigoplus F_j$$

where Ψ_i is the set of pre-coding symbol indices that are combined and $\sum \bigoplus$ is the XOR operator. The Ψ_i is determined by a pseudo-random generator whose seed is controlled by the ESI information.

On the receiver side, Raptor decoding first constructs the generator matrix $\mathbf{A}$ with the encoding constraints given by the ESIs of the received symbols. The first rows of $\mathbf{A}$ contain the pre-coding conditions given by the generator matrices $\mathbf{G}_{LDPC}$ and $\mathbf{G}_{Half}$ for both the LDPC and the Half encoders, respectively. The last rows of the matrix $\mathbf{A}$ represent the operations of the LT encoder given by $\mathbf{G}_{LT}$. The matrix $\mathbf{A}$ can be written as

$$\mathbf{A} = \begin{bmatrix} \mathbf{G}_{LDPC} & I_S & O_{S \times H} \\ & \mathbf{G}_{Half} & I_H \\ & & \mathbf{G}_{LT} \end{bmatrix}$$

where I_S and I_H are square unit matrices of size S and H , respectively, and $O_{S \times H}$ is a zero matrix of size $S \times H$. It is interesting to note that the matrices $\mathbf{G}_{LDPC}$ and $\mathbf{G}_{Half}$ are not transmitted to the decoder as they can be constructed based only on the number K of input symbols. The matrix $\mathbf{G}_{LT}$ can be constructed based on the ESI information.

Whenever symbols have been erased during transmission, the decoder modifies the original generator matrix and forms a new matrix $\mathbf{G}'_{LT}$ by pruning the rows in $\mathbf{G}_{LT}$ that correspond to the symbols that have been lost. This results in a modified decoder matrix $\mathbf{A}_d$ that is given by

$$\mathbf{A}_d = \begin{bmatrix} \mathbf{G}_{LDPC} & I_S & O_{S \times H} \\ & \mathbf{G}_{Half} & I_H \\ & & \mathbf{G}'_{LT} \end{bmatrix}.$$

If $\mathbf{A}_d$ has full rank, then the probability for decoding the original source symbols stays high even in the case of symbol erasures. The decoder has to solve the following linear system of equations:

$$\mathbf{A}_d \cdot \mathbf{C} = \mathbf{D}'.$$

$\mathbf{D}'$ is a vector composed of both a zero vector of length $S+H$ and the vector of received symbols. The above linear system is typically solved by Gauss-Jordan elimination or by fast belief propagation algorithms for some specific codes.

B. Selective Network Coding

We propose to implement network coding in the nodes of the overlay network in order to augment the performance of the streaming system. The peers selectively perform combinations of packets that have been encoded with non-systematic Raptor codes [10] by the servers in order to compensate for packet loss in the network (see Fig. 2).

We model the overlay network as a directed acyclic graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$, where the set of vertices $\mathcal{N}$ corresponds to

the nodes n_i in the overlay network, and the set of edges $\mathcal{E}$ represents the links between the nodes. Each link ϵ_{ij} between the nodes n_i and n_j is characterized by a loss probability π_{ij} and an available bandwidth b_{ij} , given in packets per time unit. We call *parents* and *children* of the node n_i the peers that transmit information to, or, respectively, receive packets from n_i . We denote by $\mathcal{P}_i$ the set of parents nodes for node n_i , and by $\mathcal{C}_i$ the set of its children nodes. We further denote by t_i^j the number of packets sent by the node n_i to its children n_j in each time unit. Similarly, r_i^h is the number of packets received by the node n_i from its parent n_h in each time unit. The network constraints are finally given as

$$t_i^j \leq b_{ij}$$

and

$$r_i^j = (1 - \pi_{ij}) t_i^j.$$

Network coding is applied selectively in overlay nodes, and new packets are generated only if packets have been lost, as long as the outgoing links can accommodate new packets. In more detail, a network node n_i gathers r_i encoded symbols from its parents nodes, where $r_i = \sum_{j|n_j \in \mathcal{P}_i} r_i^j$ represents the sum of the packets received from all the parent nodes. If the overall outgoing bandwidth b_i with $b_i = \sum_{j|n_j \in \mathcal{C}_i} b_{ij}$ is smaller than the number of received packets r_i , then the node n_i forwards to its children the received packets in the order of their arrival and distribute them randomly on the outgoing paths. It further drops the packets that cannot be forwarded due to bandwidth limitations. If, on the other hand, the outgoing bandwidth b_i is larger than the number of received packets ($b_i > r_i$), the node implements selective network coding in order to fully use the outgoing links. Selective network coding combines incoming packets and forwards the resulting packets to the children nodes such that the total number of transmitted packets $t_i = \sum_{j|n_j \in \mathcal{C}_i} t_i^j$ is equal to the outgoing bandwidth, i.e., $t_i = b_i$.

The coding strategy is a crucial part of the collaborative streaming algorithm. It is important for the efficiency of the overall system that the symbol diversity stays as high as possible, such that the probability of decoding failure is low on the client side. The selective network coding has therefore to choose properly the symbols to be combined together, in order to maximize the symbol diversity. The network coding node first forms a modified generator matrix $\mathbf{A}_{nc}$ based on the ESI information that is present in the received packets. This matrix typically contains some rows of the initial generator matrix $\mathbf{A}$ that correspond to the original symbols that have been correctly received. It also contains rows that describe the network coding operations in the parent nodes. Network coded packets are generated by the combination of a random pair of received symbols that respect the following conditions.

- 1) *Rule 1*: the selected symbols correspond to some rows of $\mathbf{A}_{nc}$ that are orthogonal to all the rows in the original matrix $\mathbf{A}$ that correspond to lost symbols.
- 2) *Rule 2*: the selected symbols correspond to some rows of $\mathbf{A}_{nc}$ that are orthogonal to each other.

These rules permit to maintain a high symbol diversity. They further permit to prevent decoding problems in the Gauss-Jordan elimination process at the receiver. Some bad symbol

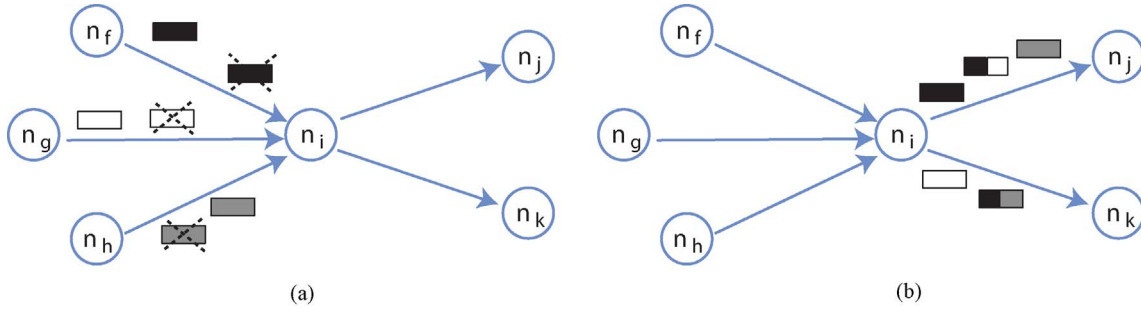


Fig. 2. Selective network coding in a network peer. (a) Some of the packets arrive at node n_i while other are erased before reaching node n_i . (b) Successfully received packets are forwarded by node n_i to the children peers, along with network coded packets that replace the lost packets and permit to fully use the available output bandwidth.

combinations could indeed mislead the decoder, where the matrix $\mathbf{A}_{nc}$ stays full rank, but the solution differs from the correct one. In particular, the first condition ensures that the new symbols are independent from the lost symbols. The second condition prevents the combination of non-orthogonal symbols that can result in erroneous codewords. Once a symbol is used in network coding, it is marked in order to force the network coding node to use other symbols for encoding. Once the network coding node cannot find valid pairs of symbols for encoding, all the symbols markers are set to zero. All the symbols can be used again for encoding. The encoding proceeds until the outgoing bandwidth can be fully utilized. In practice, the constraint imposed by *Rule1* is often relaxed, as it is difficult to track the missing symbols in distributed settings. Note that in our system, the nodes only combine pairs of packets, in order to maintain the sparsity of the modified generator matrix $\mathbf{A}_{nc}$ even in multi-hop networks. It is possible to combine more than two packets in the network coding process, but it decreases the probability to find symbols that satisfy the two conditions above.

Obviously, the re-encoding strategy does not increase the rank of the matrix $\mathbf{A}$, since the symbols generated by network coding are combinations of received symbols. The redundancy therefore augments with the number of coding stages. The performance of the coding system decreases as the number of network coding nodes augments, since some information conveyed by symbols generated by symbol combination is also available in other encoded symbols. Symbol erasures and re-encoding render the generating matrix $\mathbf{A}$ less sparse, and network coding can only be applied a limited number of times. This phenomenon could, however, be mitigated by taking advantage of the path diversity in the overlay network. In particular, the rank of $\mathbf{A}$ decreases more gracefully when multi-path transmission is combined with efficient packet splitting policy. The peers in our system therefore implement a routing policy that splits the packets to be transmitted to children nodes into different subsets, which are sent on different paths. In general, the packets that have been combined together and the resulting network coded packet are sent on different outgoing links. If the network topology does not permit it, the encoded packet is sent on the same outgoing channel as one of the two packets used in the combination. This simple splitting policy tries to prevent a rapid increase of redundancy in children peers.

Finally, it should be noted that the choice of a non-systematic version of the Raptor codes is vital in the proposed scheme. It provides high symbol diversity that is crucial in network coding. Even if they directly provide some original symbols among the received symbols, systematic Raptor codes are not appropriate in our network coding scheme since they lead to increased probability of receiving duplicate symbols. Systems that use systematic codes are thus more fragile to erasures and the decoding probability drops after a few network coding stages as the symbol diversity decreases extremely fast.

C. Performance Analysis

We now analyze the performance of the proposed network coding algorithm in a few sample scenarios in regular topologies. For the clarity of the analysis, we consider here that the network nodes are grouped into *coding stages* depending on the hop-distance to the server nodes. We consider here regular topologies with three peers per coding stage, which are connected to all the peers in the neighboring coding stages. We assume that each link has a capacity of 300 kb/s, and that the packet loss probability is the same on each segment. Two servers encode 158 packets of 512 bytes from 140 source packets with different ESIs. We compare the performance of the network coding algorithm to an algorithm that performs Raptor decoding and re-coding at each of the network peers. Whenever Raptor decoding is not successful in a peer, a random packet replication is performed so that the available bandwidth is fully used in both schemes.

We look at the evolution of the performance of the Raptor network coding algorithm with respect to the number of coding stages between the encoders and the decoder for different packet loss probabilities, as illustrated in Fig. 3(a). We first observe that the performance of both schemes decreases with the number of coding stages, as expected. The decoding/recoding strategy performs better for low to medium loss rates (up to 10% in our sample scenario). In this case, it is indeed better to try to completely recover from losses at each coding stage and then to generate new Raptor symbols. However, when the loss rate gets high (e.g., 15%), Raptor decoding in the nodes cannot be performed correctly, and network coding becomes more beneficial as it permits to maintain a higher packet diversity. These observations are confirmed in Fig. 3(b). It shows that the probability of decoding failure in the network coding algorithm decreases faster when the

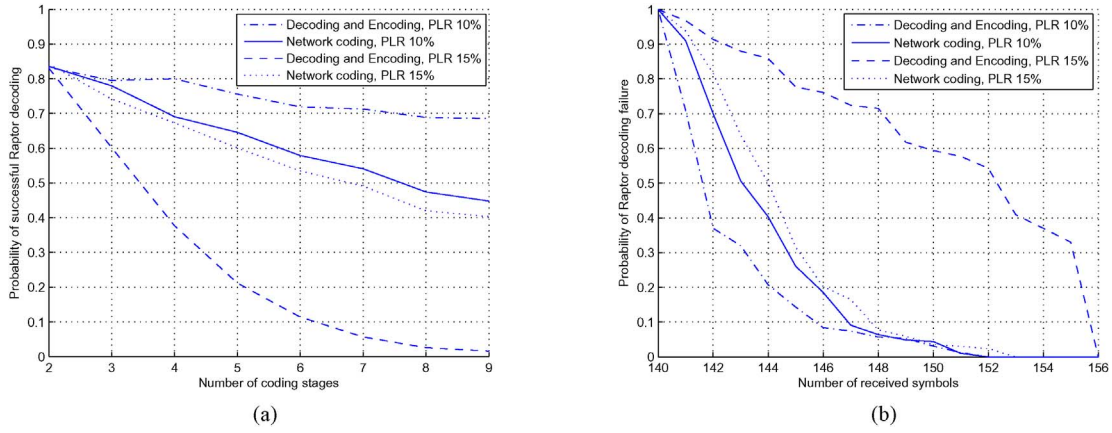


Fig. 3. Probability of decoding versus (a) number of coding stages in a regular topology with three nodes per stage, and (b) number of symbols in a regular topology with five coding stages and three nodes per stage. The schemes under comparison are the proposed selective network coding (NC) and a scheme where each node performs Raptor decoding and encoding successively.

number of received packets increases. This is mostly due to the high number of packet duplicates in the decoding/recoding strategy when the loss rate is high. Note that at high loss rates, the decoding/recoding strategy tends to behave as poorly as a random packet replication strategy.

Decoding and recoding is probably not a viable option in large networks, since it introduces important complexity penalties and large delays. The computational complexity requirements are also more important than in the network coding scheme. From [11], the number of XORs operations in a (N, K) Raptor code is given by

$$C_{enc} = K \cdot \left(10 + 4.5 \cdot \frac{K}{N} \right).$$

At the same time, Raptor decoding requires C_{dec} XORs operations, with

$$C_{dec} = 10 + 4.5 \cdot \frac{\min(N - K, K)}{K}.$$

In every node, decoding and re-coding necessitates a total number of XOR operations that can be written as

$$\begin{aligned} C_{total} &= N \cdot C_{enc} + K \cdot C_{dec} \\ &= 10 \cdot K \cdot (N + 1) + 4.5 \cdot (K^2 + \min(N - K, K)). \end{aligned}$$

For large values of N and K the C_{total} is approximated by

$$C_{total} = 10 \cdot K \cdot N + 4.5 \cdot K^2.$$

On the other hand, nodes perform packet combinations only in case of loss in the network coding scheme. Therefore, the number of XORs operations is directly driven by the packet loss rate p . It can be written as

$$C_{NC} = p \cdot N$$

and tends to zero when the loss probability gets small. In practice, C_{NC} is slightly larger than $p \cdot N$, since the second network coding condition has to be satisfied. This might necessitate a few additional combinations or tests. In any case, the complexity of the network coding solution is dramatically smaller than the complexity of decoding and recoding strategies, even if we assume that Raptor decoding is performed

only in case of packet loss. In the case of an independent loss process, we have

$$C_{total} = (1 - (1 - p)^K) (10 \cdot K \cdot N + 4.5 \cdot K^2).$$

We finally investigate the latency issues related to the selective network coding solution. We simulate a regular and homogeneous network topology with links of 400 kb/s and a packet loss rate of 5%, and we send 270 packets of 512 bytes each from two servers. We compute the time required for the delivery of a number of packets sufficient to enable the decoding of all source packets, in the case where each coding stage has three network coding nodes. We compare the proposed scheme to a solution where nodes simply replicate randomly chosen packets in case of loss. We also provide comparisons with a randomized network coding solution (RNC) [12] where all the packets buffered at a node are combined before forwarding to next hop nodes. Note that the header in this case needs to be adapted in order to keep track of the coding operations. Typically, we need a header of 270 bytes for 270 packets, if network coding is performed in a Galois field $GF(2^8)$. We omit the comparisons to decoding and recoding strategies in the nodes, where the delay becomes rapidly very large as each node has to perform full decoding prior to re-coding.

Fig. 4 illustrates the delivery time as a function of the number of coding stages, when the buffer size in the node is set to 32 packets. We can see that selective network coding performs closely to the packet replication solution. The randomized scheme experiences larger latencies as each node forwards packets when its buffer is full. It is interesting to note that the selective network coding scheme does not present any significant latency increase as the network size grows. Recall that network coded packets are generated only when loss happens. Since the packet loss rate is small and the buffer size sufficiently large, the probability that the network coding peer is able to find orthogonal packets in the buffer is kept high. The gain of the proposed Raptor network coding systems over randomized network coding scheme is partly due to the lower overhead of the proposed scheme, which requires an overhead

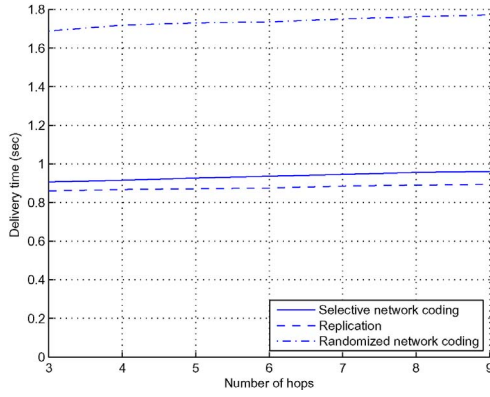


Fig. 4. Delay necessary for receiving enough packets to decode 270 source packets. The compared schemes are: the selective network coding, the packet replication where intermediate nodes follow store and forward approach and randomized network coding [12]. The delay is given with respect to the number of coding stages (three nodes per coding stage, buffer size of 32 packets).

of two bytes that increases only when packets are combined. Smaller overhead typically leaves more bandwidth for useful information and thus increases the decoding performance. Finally, it can be pointed out that the buffer size in the peers does not influence the delivery time as long as it is bigger than a few packets (typically 16 packets in our simulations); large buffers only offer a larger set of solutions for packet combinations.

III. NETWORK CODING OF RAPTOR VIDEO

A. Overview

Network coding offers a promising alternative for media streaming systems, as it is able to improve the error robustness and to increase the network throughput. The selective network coding solution proposed above is particularly interesting in this context, as it is highly adaptive, low complexity and induces only very small delays.

The extension of network coding to video streaming applications, however, requires a proper analysis of the source and channel coding rates, so that the overall performance is maximized. Under timing, complexity and bandwidth constraints, a joint source and channel rate allocation strategy is necessary in order to find the proper tradeoff between distortion due to compression of the video sequence and distortion due to channel loss. If the coding rate or the number of source symbols increases, the source distortion decreases. The redundancy created by the Raptor encoders or the network nodes generally tends to reduce the loss probability, or equivalently to increase the decoding probability for the video clients. When the overall network resources are constrained, one typically has to find the right tradeoff between the source rate and the redundancy in the network, so that the end-to-end quality is maximized.

We first compute the decoding failure probability, as a function of the number of source symbols when the nodes implement selective network coding. Then we propose an optimization algorithm that determines the coding rate for the Raptor encoders for a given network topology, such that

the distortion is minimized for the client that experiences the lowest decoding probability. This maximizes the probability that all clients are able to decode the video stream. We finally describe the video streaming system built on selective network coding of Raptor encoded video.

B. Decoding Failure Probability

The decoding failure probability p_R at a client denotes the fact that the decoder cannot recover the source symbols when Raptor decoding fails. This probability depends on the network topology and the number of source symbols K used by the Raptor encoders. We propose to estimate the decoding failure probability based on the redundancy of the modified generator matrix built by the clients. Each client m forms an equation system with the received symbols and typically solves it by Gauss-Jordan elimination. The decoding probability is directly linked to the rank of the matrix. The redundancy of the system is, however, increased by the successive network coding operations in the network nodes that can generate symbol replicates. Therefore, the rank of the system at decoder might not increase as fast as the number of packets that are received. The linear system of equations at the client might therefore not be full rank anymore and thus not solvable. Based on the probability of unsuccessful decoding developed in [13], we write the probability of unsuccessful decoding at client m in the case of non-systematic 3GPP Raptor codes [10], as

$$p_R(r_m, K) = \begin{cases} 1 & r_m < K \\ 0.85 \cdot 0.567^{r_m - K - \delta_m} & r_m \geq K \end{cases} \quad (1)$$

where the penalty term δ_m has been added to capture the effect of network coding. This penalty term represents the number of packets that are purely redundant and thus useless at decoder.

We estimate an upper-bound on the penalty term δ_m , which eventually leads to an upper-bound on the unsuccessful decoding probability. We consider each node independently. We estimate the number of redundant packets transmitted by a network coding node, which finally reach the client m . Recall that, per construction of the proposed selective coding algorithm, the number of transmitted packets in a node is equal to the outgoing bandwidth, i.e., $t_i = b_i$. Under the assumption that network peers can always find orthogonal symbols in their buffer to perform selective network coding, the number c_i of network coded packets generated at node n_i can be computed as

$$c_i = \begin{cases} 0, & \text{if } t_i \leq r_i \\ t_i - r_i, & \text{if } t_i > r_i \end{cases} \quad (2)$$

It is equal to the number of packets that are necessary to compensate for the difference between the outgoing bandwidth b_i and the incoming rate r_i at node n_i . We further compute the probability p_i^m for a packet sent by the node n_i to reach the client node m , where both nodes are connected by a set of paths $\mathcal{W}_i^m$. This probability is computed by considering all the paths $w \in \mathcal{W}$, as

$$p_i^m = \sum_{w \in \mathcal{W}_i^m} \left\{ \prod_{\epsilon_{jk} \in w} \rho_{jk} \cdot (1 - \pi_{jk}) \right\}$$

where ρ_{jk} is the percentage of the packets sent by node n_j that are forwarded toward the children node n_k on the network link ϵ_{jk} with a packet loss probability π_{jk} . The percentage of packets sent over the link ϵ_{jk} is driven by the respective bandwidth on this link. ρ_{jk} is equal to

$$\rho_{jk} = \begin{cases} \frac{t_j^k}{r_j}, & \text{if } t_j < r_j \\ \frac{t_j^k}{t_j}, & \text{otherwise.} \end{cases} \quad (3)$$

We can then estimate the expected number of packet triples τ_m received by each client m , where a packet triple represents a network coded packet along with the two packets used in the generation of this packet. When a triple is present at the decoder, one of the packets is completely redundant, and it does not decrease the probability of decoding failure. In this case, the rank of the system does not increase proportionally to the number of received packets, and the unsuccessful decoding probability of (1) is penalized by incrementing δ_m . The expected number of packet triples received by the client m that includes a network packet generated at node n_i is given by

$$\tau_i^m = \alpha_i c_i (p_i^m)^3. \quad (4)$$

The parameter α_i captures the influence of the participation of packets to multiple network coding operations in a node. Our selective network coding policy indeed authorizes multiple packet combinations if the outgoing bandwidth at node n_i is very large compared to the incoming rate, i.e., if $2c_i > r_i$. In this case, the redundancy at decoder increases faster. This acceleration is, however, upper-bounded by the number of combinations for each packet. We, therefore, have

$$\alpha_i = \begin{cases} \frac{2 \cdot (t_i - r_i)}{r_i}, & \text{if } 2 \cdot (t_i - r_i) \geq r_i \\ 1, & \text{otherwise.} \end{cases} \quad (5)$$

Finally, we can estimate an upper-bound on the penalty term δ_m . When a set of servers $\mathcal{S}$ is available for stream delivery, the penalty term for a client $m \in \mathcal{M}$ is given by

$$\delta_m = \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{W}_s^m} \sum_{i \in w} \tau_i^m.$$

This clearly represents an upper-bound on the packet redundancy at client m , as we have considered all the network coding nodes independently. The actual redundancy can only be smaller than the value captured by δ_m .

C. Source Rate Selection

When the network nodes process packets distributedly by selective network coding as described in Section II-B, the optimization of the decoding quality becomes equivalent to the proper selection of the source rate. The source rate or equivalently the number of source symbols per timeframe is determined such that the distortion is minimized for the client that experiences the lowest decoding probability. This maximizes the probability that all clients are able to decode the video stream.

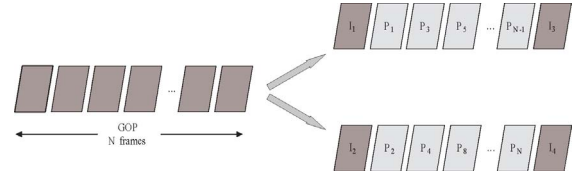


Fig. 5. GOP structure of the employed MSVC encoder [14] which splits the H.264/AVC encoded video sequence into two independently decodable video streams (descriptions): one with the odd frames and the other with the even.

Formally, the problem of optimal allocation of the source and Raptor coding rate in the streaming servers is cast as a *minmax* problem that minimizes the maximal distortion among all clients $m \in \mathcal{M}$. Let us denote by D_s and D_c the source and channel distortion, respectively. The source distortion represents the distortion due to source encoding only. Therefore, the source distortion is directly driven by the number of source symbols K in the Raptor code. The channel distortion describes the degradation due to packet loss and happens when the Raptor decoding fails. It includes the effect of error concealment when available and represents the distortion of the video displayed at the decoder. Finally, we denote by $p_R(r_m, K)$ the probability of failure of the Raptor decoding process when the clients m receives r_m symbols. The optimization problem can finally be written as follows:

Optimization problem: **RA**

$$K^* = \arg \min_K \left\{ \max_{m \in \mathcal{M}} \left\{ (1 - p_R(r_m, K)) \cdot D_s(K) + p_R(r_m, K) \cdot D_c(K) \right\} \right\} \quad (6)$$

subject to $t_i^j \leq b_{ij}$.

The condition expresses the link capacity constraints, as the number of transmitted packets on a given network segment cannot be larger than the available bandwidth on this segment. Other constraints are given by the construction of the selective network coding algorithm. In particular, the network coding algorithm transmits new packets when loss happens or when the outgoing bandwidth is larger than the incoming, as explained in Section II-B.

We compute for each client its $p_R(r_m, K)$ and then we apply a bisection search algorithm in order to solve the RA optimization problem. The source distortion $D_s(K)$ decreases when K increases. On the other hand, the decoding failure probability $p_R(r_m, K)$ also increases when K gets larger. The optimization problem is therefore convex with K . The proof of convexity is presented in [17]. Initially, the lower bound for the source rate is set to zero, which means that only channel symbols are transmitted. The upper bound is equal to the receiving rate of each client. Bisection search then converges very fast to the optimal source rate per client K_m . The overall optimal source rate is calculated as $K^* = \min_{m \in \mathcal{M}} K_m$.

D. Video Streaming System

We consider a system where the video sequence is possibly available at multiple servers. These may offer different subsets of Raptor packets. Since we use a non-systematic Raptor code, the efficiency of error concealment at the decoder is

very limited in case of decoding failures. We, therefore, propose to improve the robustness to errors with a simple multiple description coding (MDC) scheme [14] that provides a sufficient high number of source packets. The video sequence is initially split into two independent encoding threads, as illustrated in Fig. 5. The odd P frames use the first I frame as reference frame, while the even P frames refer to the second I frame. The MDC coder enhances system robustness because the simultaneous loss of both descriptions is quite improbable. Whenever the decoder fails to retrieve the source information from one description, the other description is used for error concealment with simple frame replication. Note, however, that our scheme is independent on the source coding scheme and it can be used with any video coder with advanced concealment properties like the scalable extension of the H.264/AVC standard.

We then segment each subsequence into independent group of pictures (GOP). For transmission of CIF sequences coded at 30 frames/s, we typically select a GOP size of 30 frames for each description. It represents a good compromise between latency constraints and number of source symbols, which has to be large for Raptor and network coding efficiency. Each GOP is then compressed with a source rate corresponding to the optimal value of K^* symbols. The source symbols typically correspond to fixed-length chunks of the compressed video stream. The size of these chunks is a tradeoff between coding efficiency that increases with the number of symbols, and the overhead due to the header that is appended to each symbol by the transmission protocols.

The source symbols are then encoded with a non-systematic Raptor coder. The total number of Raptor symbols is driven by the transmission bandwidth availability. Each symbol is then fed into a transmission packet, which is augmented with the RTP/UDP/IP protocol headers for transmission. As proposed in the 3GPP implementation of the Raptor codes [10], each packet has a size of 512 bytes. We also append ESI information to each packet as a header of 2 bytes that convey the information about the Raptor encoding structure. The Raptor symbols are sent in packets to the overlay nodes, which implement selective network coding as described in Section II-B. The video clients finally decode the network coded symbols and eventually the Raptor symbols to re-assemble the video streams. The video streams are finally decoded. Error concealment can finally be activated to mask the effect of losses or decoding failures.

IV. SIMULATION RESULTS

A. Simulation Setup

We analyze in this section the performance of the proposed streaming system in various scenarios. Here, we focus only on wired streaming scenarios. Extension to wireless case is possible, however, parameters such as interference and others should be taken into account. The video sequences are encoded with a multiple state video coding (MSVC [14]) variant of the H.264 AVC encoding (i.e., JM 12.2 [15]), which divides the original video sequence into two independently decodable sub-streams of odd and, respectively, even frames. The overall

GOP size is set to 60 frames, which corresponds to GOPs of 30 frames in each of the descriptions. The frame rate is equal to 30 frames/s.

The Raptor encoder is non-systematic version of 3GPP Raptor codes [10], with a Raptor symbol size of 512 bytes. Each of the Raptor symbols forms a network packet. This symbol size is a good compromise between video encoding performance and Raptor decoding performance. The employment of larger symbols imposes shorter code-blocks, which reduces the performance of Raptor codes as the decoding performance typically improves with the code-block size. A smaller symbol size leads to lower source coding rate, which reduces the encoding performance of the video encoder. Finally, we do not permit partial decoding of descriptions, in order to avoid error propagation. A description is either decoded fully, or not decoded at all. In this case, the corresponding pictures are reconstructed by replicating the images that have been correctly decoded in the other description.

Simulations are performed on several network topologies. Regular overlay network topologies consist in several coding stages, which group the nodes that have the same hop distance to the server. Although, regular networks are not very good models of real peer-to-peer networks, they can give useful insights about the efficiency the streaming algorithms. The peer nodes in regular networks have constant node degree, so that all peers in the same coding stages have the same number of ancestors, respectively, descendants. All network paths connecting servers and clients have an identical hop distance. An overlay network with T sources, P coding stages with Q_p nodes in the p th coding stage, and R receivers is denoted as a $T \rightarrow Q_1 \rightarrow \dots \rightarrow Q_P \rightarrow R$ network topology. In addition to regular topologies, we also build irregular network overlays for our simulations. The irregular graphs are generated from regular ones, where some links are randomly pruned or shifted. A link is shifted if the destination is changed to a node in one of the subsequent coding stage. Shifting does not affect the out-degree of a peer. After shifting and pruning operations we examine whether the resulting network graph is directed acyclic. Only directed acyclic graphs are allowed as network coding faces problems in network with cycles. The links are pruned and shifted with a uniform probability distribution. We, however, constrain all the peers to have at least two incoming and two outgoing links, as path diversity is critical for illustrating the performance of network coding. Network coding obviously does not perform nicely if there is no path diversity in the network.

We simulate packet loss on the network links with a Gilbert-Elliott model, which is a two-state Markov chain, where the “good” and “bad” state represent, respectively, the correct reception, or the loss of a packet. The transition probabilities between the two states drive the packet loss ratio and the average error burst length. In our model, the average burst length is equal to nine packets. We compare the performance of the proposed system with a strategy that decodes and re-encodes packets in the network nodes and with end-to-end error protection. All results reported are averages of 150 simulations.

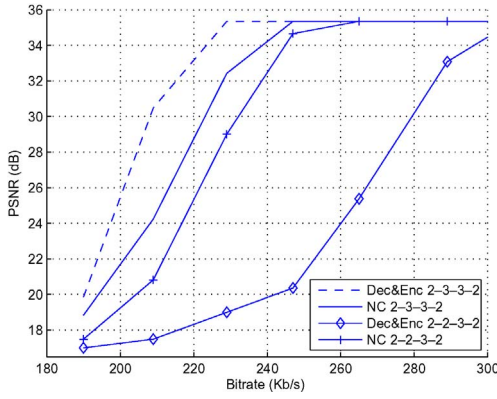


Fig. 6. PSNR comparison of the proposed selective network coding scheme (NC) and a scheme that follows Raptor decoding and encoding approach (Dec&Enc) for transmission of the *Foreman* CIF sequence coded at 512 kb/s with respect to average link capacity for 2-3-3-2 and 2-2-3-2 overlay network topologies.

B. Baseline Distributed Streaming

We first study the performance of the proposed streaming system in a totally distributed scenario, where the rate allocation is chosen without knowledge about the network topology. We compare the selective network coding algorithm with a strategy that performs Raptor decoding and re-coding in the network nodes. This baseline scheme replicates randomly the received packets, whenever Raptor decoding fails or when the number of encoded packets is not sufficient to completely fill in the outgoing bandwidth. When the outgoing bandwidth is insufficient, both schemes randomly select a subset of packets to be transmitted. It is worth noting that both schemes do not assume any reconciliation among network nodes.

We first analyze the behavior of the network coding strategy as a function of the average link bandwidth, $\bar{b}$. The average packet loss ratio is set to 5%, while the link bandwidth is randomly selected in $[\bar{b} - 40, \bar{b} + 40]$ kb/s. The *Foreman* CIF sequence is encoded at 512 kb/s. We compare in Fig. 6 the performance of the selective network coding and the baseline schemes for two regular topologies, respectively, of the form 2-3-3-2 and 2-2-3-2. It can be seen that the decoding and encoding approach performs better than the Raptor network coding approach for the 2-3-3-2 topology, which provides increased path diversity. The performance gap is quite significant for low average link bandwidth. The performance difference then decreases sharply as the average link capacity increases, and both methods perform similarly when the bandwidth is high. The improved performance of the decoding and encoding approach is due to the replication policy implemented in the intermediate nodes. It ensures successful decoding at the second hop nodes and at the receivers. On the contrary, the Raptor network coding scheme cannot maintain a sufficiently high packet diversity for low bandwidth links. It hardly succeeds in exploiting path diversity to compensate for the decay of the packet diversity, which reduces the probability that the end-nodes receive a set of packets that corresponds to full rank generator matrix.

The influence of the link capacity is even more apparent when we evaluate the coding scheme in the 2-2-3-2

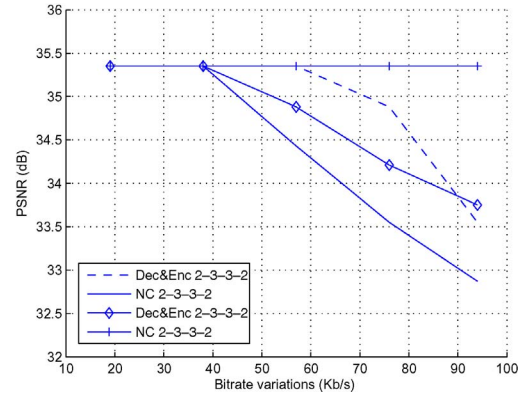


Fig. 7. PSNR comparison of the proposed selective network coding scheme (NC) and a scheme that follows Raptor decoding and encoding approach (Dec&Enc) for transmission of the *Foreman* CIF sequence coded at 512 kb/s with respect to the variation at the average link capacity for 2-3-3-2 and 2-2-3-2 overlay network topologies.

topology, as shown in Fig. 6. Since the first hop has only two nodes, the packet replication policy cannot assist decoding and encoding scheme because the set of packets that arrive at the second hop does not correspond to a linear equation system with full rank. For this topology, the network coding system even performs best, since it becomes less sensitive to the above issues. However, it should be noted that the system needs slightly more bandwidth to guarantee error free video decoding, when compared to the 2-3-3-2 topology.

We also compare the performance of both schemes when the link bandwidth is heterogeneous. The average link capacity $\bar{b}$ is set to 250 and 300 kb/s for the 2-3-3-2 and the 2-2-3-2 topologies, respectively. The actual link bandwidths are then randomly selected as $\bar{b} + \sigma$, with $\sigma \in [20, 100]$ kb/s. The results in Fig. 7 confirm the previous analysis, as the decoding and re-encoding strategy performs best for the regular topology in presence of large bandwidth variations. For the irregular topology, the performance are again better for the network coding algorithm, as the baseline scheme fails to transmit enough different packets due to the reduced number of peers.

While the baseline decoding/recoding solution generally performs well for distributed streaming, its computational complexity and latency are serious limitations for its deployment in practical systems. On the other hand, the selective network coding strategy generally performs close to the baseline solution in network with high path diversity, and performs better when the diversity reduces. It has low complexity and requires only small buffers in the network nodes. It provides a viable solution for low cost networks and we focus on this algorithm in the rest of this section.

C. Selective NC with Optimized Rate Allocation

We have shown above that the selective network coding strategy represents an interesting solution for distributed streaming. We evaluate now the benefits of optimized rate allocation in order to take full advantage of the proposed algorithm. We first analyze the solution provided by the rate allocation algorithm in Section III-C, and then we compare the performance of the optimized algorithm with a scheme that

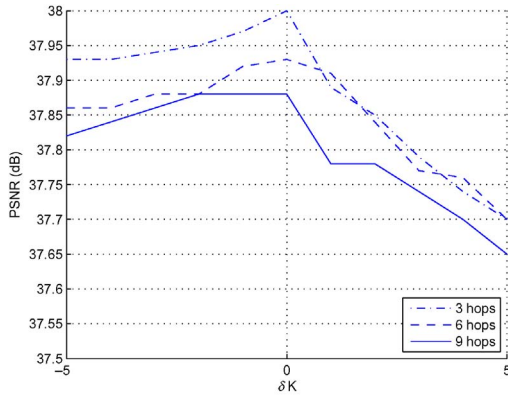


Fig. 8. Analysis of the optimal rate allocation strategy. Transmission of *Foreman* CIF sequence with the selective network coding scheme was considered for regular network topologies. δK is the searching area around the solution determined by the optimization algorithm.

optimizes FEC protection end-to-end. In this latter scheme, the peer nodes only forward packets. They replicate packets to fill the outgoing links and provide additional robustness. The optimal source rate K in this case is found by solving the optimization problem of (6) with exhaustive search, where we assume that the servers are aware of the channel statistics and that the peer nodes do not perform network coding. In order to solve the rate allocation problem, we consider a source rate-distortion model of the form $D_s = x \cdot r^y$, where r is the encoding rate or the number of source symbols. The parameters x and y are determined by training the model for a large number of points acquired from different encodings. We adopt this model in our system because of its simplicity and accuracy in modeling the average distortion of compressed video sequence [16].

We evaluate the solution obtained by the rate allocation algorithm in Section III-C. In particular, we examine the quality of the received video for different values of K selected around the K^* value computed by the rate allocation algorithm. We consider the transmission of the *Foreman* CIF sequence for regular networks with three nodes per coding stage ($3 - 3 - \dots - 3$) and different number of coding stages. The loss link rate and the link capacity have been, respectively, set to 5% and 400 kb/s. The PSNR evaluation is presented in Fig. 8 for different source rate values. The horizontal axis denotes the difference δK (in packets) between the selected source rate allocation and the one computed by the proposed optimization algorithm. We can see that the optimization algorithm is able to find the optimal rate allocation in all the topologies under consideration, which confirms the validity of the proposed algorithm. Note that comparisons with the decoding and encoding scheme are not reported here since this scheme cannot be deployed in real systems due to the high latency imposed by decoding operations in each node. Finally, this analysis does not provide a proof of optimality, but rather shows that the rate allocation algorithm provides good performance despite the conservative assumptions necessary for solving the RA problem.

We now compare the selective network coding scheme with optimized rate allocation to optimal end-to-end error

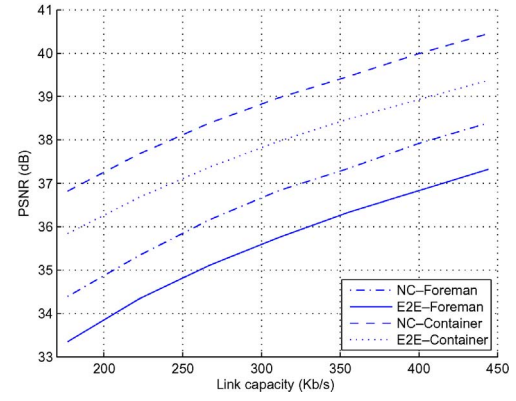


Fig. 9. PSNR performance evaluation for the proposed selective network coding (NC) and an end-to-end Raptor coding system (E2E), for regular topologies with six coding stages and three nodes per stage. The PSNR is measured as a function of the link bandwidth $\bar{b}$ ($\pi = 5\%$).

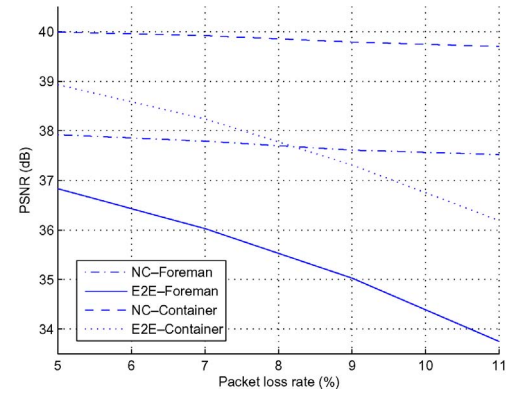


Fig. 10. PSNR performance evaluation for the proposed selective network coding (NC) and an end-to-end Raptor coding system (E2E), for regular topologies with six coding stages and three nodes per stage. The PSNR is measured as a function of the packet loss rate π ($\bar{b} = 400$ kb/s).

protection with packet replication in the nodes. We first study the performance of both schemes as a function of the link bandwidth $\bar{b}$. The packet loss ratio is set to 5% while the link bandwidth $\bar{b}$ varies between 170 kb/s and 450 kb/s. We consider a regular topology with six encoding stages between the servers and the decoders and three nodes per coding stage. We examine the performance of both schemes for the transmission of two CIF sequences *Foreman* and *Container*. Note that these very different sequences lead to very different performances of the error concealment scheme. Typically, MSVC decoding and error concealment performs relatively poorly for the high motion *Foreman*, while it gives excellent results for *Container* that has quite a static background. The streaming results for *Foreman* and *Container* sequences are presented in Fig. 9. We can see that the selective network coding algorithm performs better than the end-to-end solution, by roughly 1 dB. We also observe that the performance gap remains unaltered as the link capacity increases. This is due to fact that the schemes are not very sensitive to bandwidth variations, but rather to the overall packet loss rate.

We also investigate the influence of the packet loss rate for the same network topology. We vary the link loss rate between 5% and 11%, while the link capacity is fixed to 400

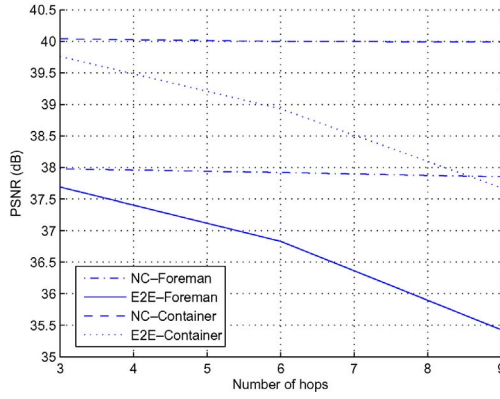


Fig. 11. PSNR performance evaluation for the proposed selective network coding (NC) and an end-to-end Raptor coding system (E2E) for link capacity of $\bar{b} = 400$ kb/s and loss rate of $\pi = 5\%$. The quality is measured for different path lengths.

kb/s. Fig. 10 presents the results for streaming of *Foreman* and *Container* sequences. We can see that network coding performs better by more than 1 dB when the loss rate is low. When channel conditions deteriorate the performance of the network coding schemes degrades smoothly, while the end-to-end scheme collapses. For 11% packet loss ratio the performance of the proposed scheme for *Foreman* is close to 3.5 dB, while for *Container* the performance gaps grows up to 4 dB. This additional gain for the *Container* is due to the better concealment performance of the MSVC scheme. This impressive performance difference shows the resiliency of the network coding schemes for channels with heavy losses; in these conditions the replication policy cannot assist the end-to-end scheme as the optimized source rate is selected very low.

Finally, we have assumed that the network statistics are gathered by each peer node and periodically sent back to the servers, which perform the rate allocation. Since, we focus on wired overlay networks with low dynamics, the servers are always able to acquire the latest network statistics and update the optimal rate allocation policy in due time. In the rare cases where the statistics are not timely or not accurate the performance of the proposed system stays consistent due to the resiliency of Raptor and network coding. The transmitted streams can still be decoded with high probability.

D. Influence of the Network Topology

We study now the influence of the network topology on the performance of the network coding algorithm. We compare both streaming algorithms with respect to the hop distance between the encoder and the decoder for regular topologies of three nodes per coding stage. The link capacity is set to 400 kb/s and the packet loss ratio to 5%. As above, we present results for streaming *Foreman* and *Container* CIF sequences in Fig. 11. The results show that for small size topologies the end-to-end scheme performs close to the Raptor network coding. However, as the number of hops increases, the advantages of the network coding become more apparent and the performance difference widens. In general, the performance degrades when the path length increases, as the diversity of symbols becomes smaller in the last coding stages. Still, the

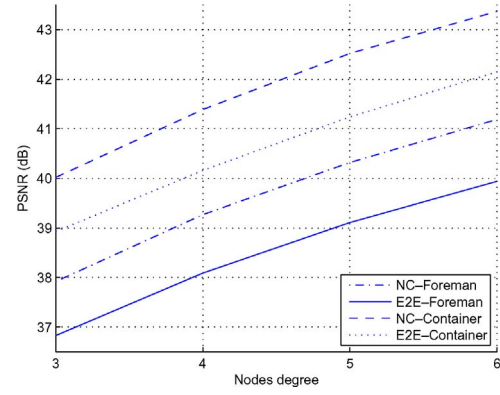


Fig. 12. PSNR performance evaluation for the proposed selective network coding (NC) and an end-to-end Raptor coding system (E2E) for link capacity of $\bar{b} = 400$ kb/s and loss rate of $\pi = 5\%$. The quality is measured for different node degrees.

performance of the network coding scheme remains high, while the end-to-end scheme performs worse. This difference is attributed to the high loss rate that reduces significantly the optimized source rate for the end-to-end scheme. At the same time, the optimized source rate for the Raptor network coding is not significantly affected by the loss rate value due to its ability to properly exploit path diversity.

We also examine the performance of both schemes considering regular network topologies with six coding stages. The number of nodes per coding stage is in the range [3, 6]. Again the transmission of *Foreman* and *Container* CIF sequences is considered and the results are presented in Fig. 12. It can be seen that the network coding outperforms the end-to-end scheme by approximately 1 dB. The performance generally improves with the node degree as the path diversity becomes more important. For high nodes degree, the sequence *Container* benefits from the increased number of nodes per coding stages and the performance gain reaches 1.5 dB due to improved error concealment compared to the *Foreman* sequence. The advanced performance of the proposed streaming solution is due to the efficiency of both the network coding scheme and the channel rate allocation algorithm that permit the use of higher code rate in the Raptor encoder. Furthermore, as the number of nodes per hop increases, the symbol diversity in the system becomes also higher, which is beneficial for our network coding algorithm.

Finally, we study the effect of the network regularity on the streaming performance. We consider transmission of *Foreman* CIF sequence over irregular networks that are generated from regular topologies by link pruning and shifting. We run simulations over different random topologies and provide average performance results for different degrees of irregularity in the topology. We avoid pruning the links that connect the last coding stages to the clients, since this would mislead the system toward very low source rate values. We consider network topologies with six coding stages and three nodes per coding stage, where we set the link pruning and shifting probabilities to 10% for the irregular topologies. We show in Fig. 13(a) and (b) the PSNR performance as a function of the link bandwidth $\bar{b}$ (with $\pi = 5\%$) and the loss probability π (with $\bar{b} = 400$ kb/s). It can be seen that the network coding

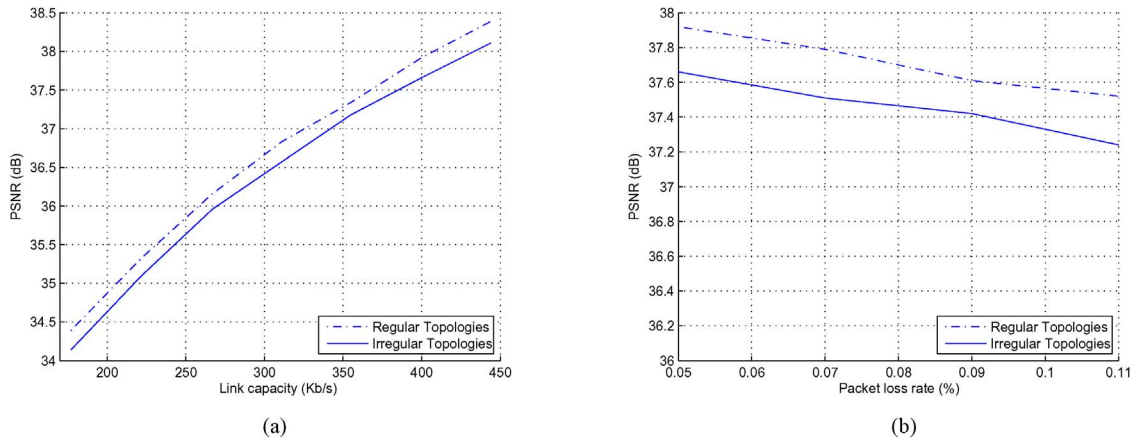


Fig. 13. PSNR performance of the proposed selective network coding scheme for transmission of the *Foreman* CIF video sequence on regular and irregular topologies where link shifting and pruning probabilities are set to 10%. The topologies have six coding stages and three nodes per coding stage. Performance are given as a function of (a) link capacity $\bar{b}$ ($\pi = 5\%$) and (b) loss probability π ($\bar{b} = 400$ kb/s).

scheme is resilient to the irregularity of the network topology, and that the performance only degrades by 0.2 dB compared to regular topologies at high loss rates. The proposed method is resilient to network variations as the scheme takes advantage of both path and symbol diversity.

V. RELATED WORK

Due to the advent of novel network architectures with path and source diversity, several works have addressed the problem of exploiting the network diversity for increased streaming performance. For example, a system for data overlay networks called CoolStreaming [18] has recently been designed. The system exploits path diversity and systems' scalability. Specifically, the peer nodes periodically inform a set of neighboring nodes for their data availability and retrieve missing data from them. Each node maintains two lists: one with the neighboring nodes and another with the available data that neighboring nodes possess. However, the periodic data exchange increases significantly the delay in the system. Avalanche [19] has been developed for large scale content distribution on peer-to-peer networks. It envisions the deployment of huge overlay networks that allow fast downloading. This collaborative system architecture significantly improves file download times over BitTorrent [20].

In parallel, network coding [1] has received a lot of attention from the research community in the past few years as it permits to effectively exploit the network diversity for increased throughput or robustness to losses. It is a promising method to improve network throughput utilization and better approach max-flow min-cut bound. In the network coding systems, the nodes do not only forward packets, but perform operations with the received packets for exploiting network diversity and improving data persistence. The network coding schemes in the literature can be classified into two major categories, namely the systems based on algebraic and channel coding. An algebraic perspective on network coding has first been provided in [21], which shows that maximal robustness to non-ergodic link failures can be achieved. Linear network codes [22] have been considered for multi-hop networks,

where they achieve the max-flow capacity bound with controllable complexity in the network nodes. Recently, the benefits of network coding have been thoroughly examined in wireless sensor networks with dynamically changing topologies [23]. Alternatively, channel codes have also been proposed for network coding. For example, a network coding system based on LDPC codes has been proposed in [24] for transmission over wireless relay networks. Joint network-channel coding has been presented for transmission over relay channels [25], based on Turbo codes.

A few works have addressed lately the application of network coding principles to streaming applications. Reed-Solomon codes or LT codes have been implemented in network-embedded FEC nodes [3] and in network peers [26], respectively, in order to enhance the robustness to transmission errors. In both cases, the packets are decoded and re-encoded in the network nodes, before transmission toward the streaming clients. Both schemes show that the network throughput can be significantly improved with network coding. However, decoding and re-coding operations in network nodes augment the latency of the streaming system. To reduce the end-to-end delays of [26], in [27] the relay and encode approach is proposed. The peers forward the received packets to their descendant peers upon their reception. Packet duplicates are eliminated since each packet is forwarded once. As soon as a peer is able to decode the LT encoded packets, the peer first performs LT encoding and then it generates a fresh set of packets. Practical network coding for streaming applications [28] generally relies on randomized linear network coding [12] that permits the deployment of distributed solutions with low complexity. In addition, streams are generally split into generations of symbols, in order to deal with the timing issues in streaming scenarios [28]. In [28], prioritized network coding is achieved by employing a MD-FEC scheme [29]. Motivated by this system, several unequal error protection schemes based on RNC have been presented [30]–[32].

Network coding has also been proposed for multicasting in overlay or peer-to-peer networks [33]–[36], where it takes advantage of path diversity. The benefit of network coding with peer-to-peer networks has been evaluated in [37], which shows

that network coding is very useful in peer-to-peer networks, since it provides granularity, resiliency to network dynamics, and leads to better bandwidth exploitation. A new streaming algorithm called R^2 [38] incorporates random network coding along with a random pushing algorithm to make feasible live peer-to-peer streaming and smoothening latency problems. Recently, a network coding system that benefits from both network coding and Raptor codes was presented in [39]. This scheme is very robust to erroneous channel estimations, especially for high loss rates and limited network diversity.

A method similar in many aspects to [39] was presented in [40] where network coding was used for video streaming over wireless mesh networks. It takes into account the significance of each video packet and intelligently selects network codes for combining packets that can be decoded by several peers. To select the most important packet, the scheme ignores the packets that have been already transmitted. This scheme employs the rate-distortion optimization framework proposed in RaDiO [41] and incorporates it into the design of the network coding algorithm such that the expected distortion is minimized. In [42], a network coding system was used for WLAN-like access point or WiMAX-like broadcast stations. This method employs an optimized scheduling algorithm based on the Markov decision process to maximize the multimedia transmission in both broadcast and unicast settings. A practical wireless multiparty video conferencing with network coding has been presented in [43]. A special protocol is devised for achieving real-time video scheduling of network coded video streams. LT codes [8] have been proposed for peer-to-peer multimedia delivery [6] eliminating the need for coordination among peer nodes and packet scheduling. This approach ensures very low, but not negligible probability of multiple reception of the same packet. Although the scheme is efficient, it has high computational complexity as it performs LT decoding/encoding in every peer node. This prohibits its deployment into real time systems with strict timing constraints and limited computational resources.

In this paper, we have proposed to benefit from the advantage of both the Raptor codes and the selective network coding. The Raptor codes provide a low complexity solution for rateless coding that augments the symbol diversity. Selective network coding efficiently exploits the path diversity with low complexity and small buffering requirements. It provides an effective and practical solution for distributed streaming that is robust to packet loss and network irregularity.

VI. CONCLUSION

We proposed in this paper a video network coding algorithm based on Raptor codes at the server and selective packet combinations in the network nodes for transmission over wired overlay networks. Raptor encoding permits to achieve a high symbol diversity in the network, and selective network coding effectively exploits path diversity in the network. The proposed solution has a low complexity and low delay requirements compared to state-of-the-art methods. These advantages come from the utilization of Raptor codes that permits the usage of short buffers. Furthermore, the selective property of the

proposed scheme allows on the fly combinations of the packets without collecting a full rank system. The proposed scheme also requires significantly lower overhead compared to randomized network coding method.

We also proposed an optimized rate allocation solution, such that the system minimizes the average distortion at the least reliable client. The presented rate allocation algorithm finds the optimal rate allocation by estimating the expected symbol diversity at each client for different coding strategies. Compared to optimized end-to-end FEC protection, the selective network coding scheme shows improved quality performance and increased robustness to packet loss and network irregularity. Overall, the low complexity nature and the effective behavior of the proposed algorithm in regular as well as in irregular network topologies, position it as an interesting solution for distributed streaming in practical overlay network scenarios.

REFERENCES

- [1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.
- [2] Y. Zhu, B. Li, and J. Guo, "Multicast with network coding in application-layer overlay networks," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 1, pp. 107–120, Jan. 2004.
- [3] M. Wu, S. Karande, and H. Radha, "Network embedded FEC for optimum throughput of multicast packet video," *EURASIP J. Appl. Signal Process.*, vol. 20, no. 8, pp. 728–742, Sep. 2005.
- [4] N. Thomos and P. Frossard, "Collaborative video streaming with Raptor network coding," in *Proc. ICME*, Jun. 2008, pp. 497–500.
- [5] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2551–2567, Jun. 2006.
- [6] C. Wu and B. Li, "rStream: Resilient and optimal peer-to-peer streaming with rateless codes," *IEEE Trans. Parallel Distributed Syst.*, vol. 19, no. 1, pp. 77–92, Jan. 2008.
- [7] P. Chou and Y. Wu, "Network coding for the internet and wireless networks," *IEEE Signal Process. Mag.*, vol. 24, no. 5, pp. 77–85, Sep. 2007.
- [8] M. Luby, "LT codes," in *Proc. 43rd Annu. IEEE Symp. FOCS*, Nov. 2002, pp. 271–280.
- [9] M. Luby, M. Watson, T. Gasiba, T. Stockhammer, and W. Xu, "Raptor codes for reliable download delivery in wireless broadcast systems," in *Proc. 3rd IEEE CCNC*, vol. 1, Jan. 2006, pp. 192–197.
- [10] 3GPP TS 26.346 V7.1.0, Technical Specification Group Services and System Aspects, Multimedia Broadcast/Multicast Service, Protocols and Codescs, Jun. 2005.
- [11] *Why Digital Fountain's Raptor Code Is Better Than Reed Solomon Erasure Codes For Streaming Applications*. Digital Fountain, 2005.
- [12] T. Ho, M. Medard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," in *Proc. 41st Allerton Annu. Conf. Commun. Control Comput.*, Oct 2003.
- [13] M. Luby, T. Gasiba, T. Stockhammer, and M. Watson, "Reliable multimedia download delivery in cellular broadcast networks," *IEEE Trans. Broadcasting*, vol. 53, no. 1, pp. 235–246, Mar. 2007.
- [14] J. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," in *Proc. Visual Commun. Image Process.*, Jan. 2001, pp. 392–409.
- [15] *JVT Reference Software Version 12.2* [Online]. Available: <http://bs.hhi.de/suehring/tml>
- [16] P. Frossard and O. Verscheure, "Joint source/FEC rate selection for quality-optimal MPEG-2 video delivery," *IEEE Trans. Image Process.*, vol. 10, no. 12, pp. 1815–1825, Dec. 2001.
- [17] N. Thomos and P. Frossard, "Network coding of rateless video in streaming overlays," EPFL, Lausanne, Switzerland, EPFL Tech. Rep. LTS-REPORT-2009-011, 2009.
- [18] X. Zhang, J. Liu, B. Li, and Y. S. P. Yum, "CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming," in *Proc. IEEE INFOCOM*, vol. 3, Mar. 2005, pp. 2102–2111.
- [19] C. Gkantsidis and P. R. Rodriguez, "Network coding for large scale content distribution," in *Proc. IEEE INFOCOM*, vol. 4, Mar. 2005, pp. 2235–2245.

- [20] B. Cohen, "Incentives build robustness in BitTorrent," in *Proc. P2P Econ. Workshop*, 2003, pp. 251–260.
- [21] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. Netw.*, vol. 11, no. 5, pp. 782–795, Oct. 2003.
- [22] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inform. Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.
- [23] C. Fragouli, J. Widmer, and J. Y. L. Boudec, "A network coding approach to energy efficient broadcasting: From theory to practice," in *Proc. IEEE INFOCOM*, Apr. 2006, pp. 1–11.
- [24] X. Bao and J. Li, "A unified channel-network coding treatment for wireless ad-hoc networks," in *Proc. IEEE ISIT*, Jul. 2006, pp. 202–206.
- [25] C. Hausl and P. Durpaz, "Joint network-channel coding for the multiple-access relay channel," in *Proc. IWWAN*, Jun. 2006, pp. 817–822.
- [26] C. Wu and B. Li, "Optimal peer selection for minimum-delay peer-to-peer streaming with rateless codes," in *Proc. ACM P2PMMS'05*, Nov. 2005, pp. 69–78.
- [27] M. Grangotto, R. Gaeta, and M. Sereno, "Rateless codes network coding for simple and efficient P2P video streaming," in *Proc. ICME*, Jun. 2009, pp. 1500–1503.
- [28] P. Chou and Y. Wu, "Network coding for the internet and wireless networks," *IEEE Signal Process. Mag.*, vol. 24, no. 5, pp. 77–85, Sep. 2007.
- [29] R. Puri, K. Ramchandran, and K. W. L. V. Bharghavan, "Forward error correction (FEC) codes based multiple description coding for internet video streaming and multicast," *Signal Process. Image Commun.*, vol. 16, no. 8, pp. 745–762, May 2001.
- [30] N. Thomos, J. Chakareski, and P. Frossard, "Randomized network coding for UEP video delivery in overlay networks," in *Proc. ICME*, Jun. 2009, pp. 730–733.
- [31] Y. Lin, L. Yunfeng, B. Liang, and B. Li, "Priority random linear codes in distributed storage systems," *IEEE Trans. Parallel Distributed Syst.*, vol. 20, no. 11, pp. 1653–1667, Nov. 2009.
- [32] H. Wang, S. Xiao, and C. C. J. Kuo, "Robust and flexible wireless video multicast with network coding," in *Proc. IEEE GLOBECOM*, Nov. 2007, pp. 2129–2133.
- [33] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang, "LION: Layered overlay multicast with network coding," *IEEE Trans. Multimedia*, vol. 8, no. 5, pp. 1021–1032, Oct. 2006.
- [34] Y. Liu, Y. Peng, W. Dou, and B. Guo, "Network coding for peer-to-peer live media streaming," in *Proc. 5th Int. Conf. GCC*, Oct. 2006, pp. 149–155.
- [35] C. Gkantsidis, J. Miller, and P. Rodriguez, "Comprehensive view of a live network coding P2P system," in *Proc. ACM SIGCOMM/USENIX IMC*, Oct. 2006, pp. 177–187.
- [36] K. Jain, L. Lovász, and P. A. Chou, "Building scalable and robust peer-to-peer overlay networks for broadcasting using network coding," *J. Distributed Comput.*, vol. 19, no. 4, pp. 301–311, Dec. 2006.
- [37] M. Wang and B. Li, "Lava: A reality check of network coding in peer-to-peer live streaming," in *Proc. IEEE INFOCOM*, May 2007, pp. 1082–1090.
- [38] M. Wang and B. Li, " R^2 : Random rush with random network coding in live peer-to-peer streaming," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 9, pp. 1655–1666, Dec. 2007.
- [39] N. Thomos and P. Frossard, "Raptor network video coding," in *Proc. 1st ACM Int. Workshop Mobile Video (in Conjunction with ACM Multimedia 2007)*, Sep. 2007, pp. 19–24.
- [40] H. Seferoglu and A. Markopoulou, "Video-aware opportunistic network coding over wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 5, pp. 713–728, Jun. 2009.
- [41] P. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 390–404, Apr. 2006.
- [42] D. Nguyen, T. Nguyen, and X. Yang, "Multimedia wireless transmission with network coding," in *Proc. PV*, Nov. 2007, pp. 326–335.
- [43] H. Wang, R. Y. Chang, and C.-C. J. Kuo, "Wireless multi-party video conferencing with network coding," in *Proc. ICME*, Jun.–Jul. 2009, pp. 1492–1495.



Nikolaos Thomos (S'02–M'06) received the Diploma and the Ph.D. degree from the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2000 and 2005, respectively.

Currently, he is a Post-Doctoral Researcher with the Signal Processing Laboratory, Swiss Federal Institute of Technology, Lausanne, Switzerland. Previously, he was a Post-Doctoral and Post-Graduate Researcher with the Informatics and Telematics Institute/Center for Research and Technology Hellas, Thessaloniki.

His current research interests include network coding, multimedia communications, joint source and channel coding, and distributed source coding.

Dr. Thomos received the Swiss NSF Researcher Award in 2008. He is a member of the Technical Chamber of Greece.



Pascal Frossard (S'96–M'01–SM'04) received the M.S. and Ph.D. degrees, both in electrical engineering, from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1997 and 2000, respectively.

From 2001 to 2003, he was a Member of the Research Staff with the IBM T. J. Watson Research Center, Yorktown Heights, NY, where he worked on media coding and streaming technologies. Since 2003, he has been a Professor with EPFL, where he heads the Signal Processing Laboratory. His current

research interests include image representation and coding, visual information analysis, distributed image processing and communications, and media streaming systems.

Dr. Frossard has been the General Chair of IEEE ICME 2002 and Packet Video 2007. He has been the Technical Program Chair of EUSIPCO 2008, and a member of the organizing or technical program committees of numerous conferences. He has been an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA from 2004 to 2010, the IEEE TRANSACTIONS ON IMAGE PROCESSING since 2010, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY since 2006. He has been an elected member of the IEEE Image and Multidimensional Signal Processing Technical Committee since 2007, the IEEE Visual Signal Processing and Communications Technical Committee since 2006, and the IEEE Multimedia Systems and Applications Technical Committee since 2005. He served as a Vice-Chair of the IEEE Multimedia Communications Technical Committee from 2004 to 2006, and as a member of the IEEE Multimedia Signal Processing Technical Committee from 2004 to 2007. He received the Swiss NSF Professorship Award in 2003, the IBM Faculty Award in 2005, and the IBM Exploratory Stream Analytics Innovation Award in 2008.