Distributed Multiple Description Video Transmission via Noncooperative Games with Opportunistic Players

S. Milani, *Member, IEEE,* and G. Calvagno, *Member, IEEE,*

Abstract—**Recent works have shown how Multiple Description Coding (MDC) proves to be an effective solution for the multimedia streaming over Peer-to-Peer (P2P) and Content Delivery Networks (CDN). However, the presence of losses and congestions throughout the network affects the visual quality of the reconstructed sequence at the end terminal. These inconveniences can be mitigated by specifying different levels of Quality-of-Service, but an optimal packet classification is hard to obtain since P2P and CDN protocols operate at higher protocol layers (ignoring network conditions of the lowest stages), the network can be quite distributed, and few information can be available regarding other network segments involved in the transmission. The peculiarities of the transmission scenario requires a distributed and robust packet classification strategy that grants both intra-stream and inter-stream diversities among the loss patterns for the different streams. The classification approach presented here is modelled via a noncooperative game where the different uploading nodes are players/descriptions competing for the allocation of the available network resources. Within this modelization, each player may switch from a selfish strategy to a more cooperative strategy according to its convenience (opportunistic players). Experimental results shows that the proposed solution proves to be quite effective under different network scenarios.**

Index Terms—**Multiple Description Coding, Game Theory, opportunistic players, CDN networks, Peer-to-Peer network.**

I. INTRODUCTION

The distributed delivery of data (e.g. in Peer-to-Peer or in Content Delivery networks) permits an effective and flexible delivery of various multimedia data [1], [2] via the interconnection and the cooperation of the different nodes (peers) in the network [3]. As a matter of fact, research activity has been dedicated worldwide to the development of novel video coding architectures that fit the characteristics of distributed systems. Together with scalable video coding [4], Multiple Description (MD) architectures [5] prove to be an effective solution for distributed video transmission since they permit solving several problems concerning the varying topologies of Peer-to-Peer (P2P) or Content Delivery Networks (CDN) networks.

In MDC, the original video source is characterized *with several chunks of data ("descriptions") so that the source can be approximated from any subset of the chunks* [6]. This approximation is possible since each description is correlated with the others, and therefore, it can be estimated by the available data whenever it is missing. Note also that each MDC substream (associated to one description) can be independently decoded with respect to the others, and the perceptive quality of the signal reconstructed at the end terminal only depends on the number of correctly-received substreams (since each one is equally-significant). In the case of MDC video sequences, the reconstructed sequence could present different temporal and/or spatial resolutions, as well as different visual quality levels, according to the number of available descriptions and the adopted MDC scheme [6]. This coding strategy proves to be extremely helpful in granting a minimum level of Quality-of-Service (QoS) to the end user since the fruition of the transmitted multimedia content is still possible even with high churning rates and packet losses.

Experimental results have shown that the effectiveness of video transmission depends on how packets are routed and transmitted [1], [7]. Within P2P and CDN strategies, several approaches have been proposed to find the optimal topology for the distribution trees of each description. The solution in [8] mixes push and pull strategies according to delay requirements. In [9] Wu *et al.* propose a delivery algorithm to maximize the number of descriptions received by the terminal nodes. Other solutions rely on an effective building of the delivery trees [10]. It is worth noticing that these approaches imply the election of a controller node that manages the P2P network and finds out the optimal configuration.

Unfortunately, a centralized strategy proves to be ineffective to grant some control over the QoS level of the different streams. In fact, the central node needs to be updated timely about network conditions and to propagate quickly the required configuration changes to the different peers involved in the data delivery. This task proves to be quite hard to fulfill since the network could be quite heterogeneous and complex, and as a matter of fact, distributed QoS control algorithms are needed [11]. To this purpose, Game Theory (GT) may provide some help in modelling a distributed algorithm to manage the QoS problem over a distributed network.

Game Theory has recently proved to be an interesting theoretical framework to analyze and optimize resource allocation problems in digital communication scenarios [12]. As an example, bandwidth contentions between different users or terminals in a shared wireless environment can be modelled as a "game" where each device accessing the network is a player competing with the others for the access to the channel [13].

S. Milani and G. Calvagno are with the Dept. of Information Engineering, University of Padova, via Gradenigo 6/B, 35131 Padova, Italy. Phone: +39 049 827 7641, Fax: +39 049 827 7699, e-mail: {simone.milani,calvagno}@dei.unipd.it.
This work was partially supported by the PRIN 2008 project prot. 2008C59JNA founded by the Italian Ministry of University and Research (MIUR).

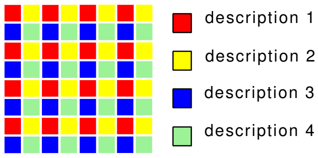


Fig. 1. MD polyphase subsampling.

However, recent works have also shown that the game theory analysis can also be employed for other purposes, such as wireless channels modelling [14] and distributed optimization [15]. The approach proposed in this paper can be included within the last field since it deals with packet classification in a distributed network scenario [11].

In this paper, we apply GT principles on a cross-layer distributed classification algorithm that runs independently on each uploading peer/node of a distributed content delivery network. Packets are scheduled into multiple QoS classes via a non-cooperative game that aims at minimizing the channel distortion affecting each stream and the final reconstructed sequence. A first work using GT to classify packets has been published in [11], where a noncooperative game between two players/descriptions is employed and the behavior of each player is fixed. The proposed approach proved to be an effective solution with respect to other independent scheduling approaches both in case the network congestion is affecting a common segment (i.e., peers/nodes are competing among each others) and in case the congested segment is related to one description only.

In the current work we adopt a noncooperative game with 4 players, where each player can opportunistically vary its behavior from selfish to cooperative (and viceversa) according to its convenience [16]. The introduction of opportunistic players improves the performance of the previous GT approach since it allows each player to take advantage of the QoS level provided to other players.

In the following, Section II presents the adopted MDC scheme, while Section III describes the problem of packet scheduling for MDC coded video. Section IV presents some basic notions about Game Theory and how it can be applied to packet classification. Section V illustrates the proposed strategy which, as simulation results in Section VI show, provides better results with respect to previous works. Conclusions are drawn in Section VII.

II. Multiple Description Video Coding over Peer-to-peer and Content Delivery Networks

During the last years, several MD video coding schemes have been proposed in literature with different efficiencies and computational complexities. In this work we focus on the packet classification for MDC schemes, and therefore, we adopt a quite simple and general MDC approach based on the spatial subsampling of phase-shifted frames from the original video sequence. Despite it is possible to design more efficient MDC coding architectures [17], this simple polyphase scheme allows us to generalize the obtained results to other approaches.

Assuming that each input frame is made of pixels $F(n,m)$ (being (n,m) the vertical and horizontal coordinates), the MD generation unit partitions the input samples into four subsequences $F(2\,i, 2\,j)$, $F(2\,i{+}1, 2\,j)$, $F(2\,i{+}1, 2\,j)$, and $F(2\,i{+}1, 2\ j+1)$, with $i = 0, \ldots, \lfloor h/2 \rfloor$ and $j = 0, \ldots, \lfloor w/2 \rfloor$ where h and w are the number of pixel rows and columns of the input video sequence (see Fig. 1). As a result, the input video sequence is split into four subsequences with halved spatial resolution and different phases. Each subsequence is then independently coded by an H.264/AVC codec [18] and uploaded in the network in separate repositories (peers or servers). Whenever an end user requires the downloading of the sequence and a delivery network has been defined, each storage terminal starts transmitting the video packets to the destination terminal. At the receiver, packets are sent to four independent H.264/AVC decoders that share a common MD error concealment unit. In case all descriptions are correctly received, the coded information can be reconstructed without quality loss. In case some parts of one description are missing, the MD concealment unit can estimate the lost information thanks to the intrinsic spatial correlation existing among adjacent pixels. In our implementation, the missing pixels are approximated via a simple bilinear interpolation involving the available neighbors. Note that the quality of the reconstructed subsequence increases as the number of available correlated pixels augments.

In case all the descriptions have been lost, the missing data are copied from the previously-decoded frame like in most of the Single Description (SD) coding schemes.

The performance of the designed MD scheme can be improved by effective algorithms that classify packets varying their loss probabilities [7]. Several multi-source video streaming strategies have been proposed in literature. However, when dealing with P2P networks, several aspects of the underlying network remain unknown. P2P and CDN protocols usually operate at the transport (or higher) layers of the protocol stacks, and therefore, buffer fullness, the presence of concurrent streams, channel and network conditions related to the lowest protocol layers are not known. Similarly, packet classification strategies operate at the same layers of the P2P or CDN applications, but their performance is an effect of what happens at the lowest stages. From these premises, optimization strategies must operate in a distributed way and process limited and partial network statistics. The following section presents some possible strategies.

III. MDC packet scheduling in QoS-aware networks

The limited upload bandwidth, the changing network topology, the presence of wireless links, and the significant amount of coded data associated to the transmission of multimedia signals increase the congestion and loss probabilities affecting the network. As a matter of fact, it is possible to mitigate the channel distortion produced on the received video signal by ruling the network access according to different QoS classes. Several QoS-aware transmission and network access protocols have been defined ([19], [20], [21]) in order to differentiate the loss probability of packets according to their significance and protect the network from congestions.

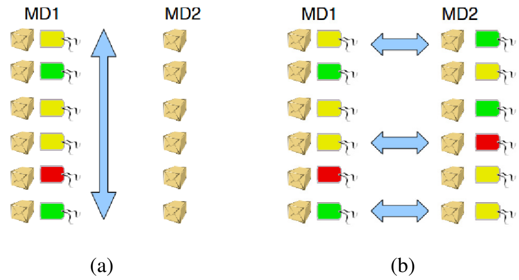


Fig. 2. Different types of diversity provided via DiffServ labelling (considering two descriptions). a) intra stream diversity b) inter stream diversity.

The performance of these solutions on the quality of the reconstructed video sequence strongly depends on how packets are assigned to the different classes. The Two-Rates-Three-Colors-Markers (TRTCM) approach [21] distributes packets among three classes according to the negotiated average bit rates. A more effective solution has been proposed in [22], where packets are assigned to different classes with different loss probabilities according to their significance in the decoding process. The effectiveness of the classification approach rely on creating a certain *intra-stream* diversity in the loss pattern such that crucial data are characterized by a lower loss probability with respect to the least important data (see Fig. 2(a)). Other solutions address the problem of packet scheduling in relation with MDC video transmission and aim at granting a certain degree of diversity among the loss patterns affecting the different descriptions (*inter-stream* diversity) [7], [23]. Figure 2 reports a schematic representation of both types of diversity. Arrows denote the dimension along which diversity must be granted. In case the congestion affects all the descriptions, the effectiveness of the MDC error concealment algorithm relies on the fact that at least one description is available at the decoder [24], and therefore, the probability that all the descriptions are lost has to be minimized by differentiating their relative loss probability, e.g., by assigning different QoS classes to different descriptions [23]. On the other hand, in case the congestion affects a single description, intra-stream diversity like in the single description case needs to be granted since the other descriptions are received correctly in any case. As a matter of fact, the classification algorithm has to take into account both requirements since a wrong packet classification may lead to uncontrolled bandwidth contentions which significantly degrade the quality of the reconstructed sequence.

In case all the descriptions are transmitted by a common server, an optimal classification can be easily performed. However, in case descriptions are stored into multiple nodes (like in the P2P and CDN cases), optimal packet scheduling can be achieved by the service provider (or an elected controller node) who should evaluate all the channel conditions throughout the network, compute the optimal classification, and send it to the different uploading peers/nodes. Such a centralized strategy is significantly time-demanding and needs a continuous update of the information related to transmission statistics for the different network paths.

A faster solution can be obtained via a distributed classification performed independently by the single uploading peers/nodes. In this case, each source node is neither aware of the classification choices of the other source nodes nor aware of where the congestions are located. As a matter of fact, the classification choices of each node could compete with the choices of the other peers for the fruition of network resources leading to sub-optimal configurations.

It is possible to mitigate the inefficiencies of a distributed classification using a GT-based packet classification strategy that provides a satisfying degree of intra-stream and inter-stream diversities in the loss patterns affecting the sequence reconstructed at the decoder.

IV. A Game Theory Classification approach

In case packet classification is performed independently by uploading nodes that can not communicate, each source node that transmits the stored description monitors the network state via transmission control protocols and classifies the video packets according to the characteristics of the coded video signal and to the loss probability. Since each peer/uploader is unaware of the choices and the channel state of the other peers/nodes, it is highly probable that each uploading node turns out to be in competition with the others in order to provide an adequate intra-stream diversity to its own packets. As a matter of fact, it is possible to analyze the classification problem via the theoretical framework of game theory, which permits taking into account the behavior of the other nodes and granting a certain level of inter-stream diversity in the optimization.

A. Basic game theory notions and notation

A competitive behavior among a set of individuals can be represented by a "strategic game", i.e., a *model of interacting decision-makers* [25] that evaluates how each individual is affected by the actions of the others. In analytical terms, the game is identified by a set of n decision-makers (commonly-named "players"), a set $\mathcal{N}_c$ of strategies available for each individual, and a set of preferences for each player. The preferences (or utility) of one player can be parameterized by a pay-off (or cost) function $f_d(\cdot)$, $d = 1, \dots, n$, which assumes its values among real numbers $\mathbb{R}$ and characterize the satisfaction (or dissatisfaction) level p of the player. Since the decisions of each player are affected by the strategies of the other players, the pay-off function depends on the decisions of all the players, i.e., for the d-th player it is possible to define

$$\begin{aligned} f_d &: \mathcal{N}_c^n \rightarrow \mathbb{R} \\ f_d &: \mathbf{c} = \left[c^1, \dots, c^n\right] \mapsto p \end{aligned} \tag{1}$$

where c^d is the strategy chosen by the d-th player, $d = 1, \dots, n$. Assuming that players are rational and selfish, their main aim is maximizing (or minimizing) their pay-off (or cost) functions f_d. However, a blind choice that does not take into consideration the behaviors of the other players may lead to inefficiencies whenever the game presents conflicting configurations. Assuming that players can not cooperate (*non-cooperative game*), a conflict appears whenever a player needs to decrease the utilities of the other players in order to increase his/her own. As a matter of fact, the rationality of the decision-makers and the correctness of their beliefs about the actions of the other players[1] lead towards some points of equilibrium, where every player has no reasons to change his/her strategy as his/her utility can not improve. These configurations are called *Nash equilibria* [25], and in analytical terms, an array of strategies $\mathbf{c}^*$ is a Nash equilibrium if

$$f_d\left(\left[\mathbf{c}_i^{-d,*}, c^{d,*}\right]\right) \leq f_d\left(\left[\mathbf{c}^{-d,*}, c^{d,*}\right]\right) \quad \forall d = 1,\dots,n \quad \text{and} \quad \forall c^d \in \mathcal{N}_c, \tag{2}$$

where $\mathbf{c}^{-d}$ is the array of strategies of all the players made exception for the d-th and $f_d(\cdot)$ is a cost function. In this case, no other choice allows player d to reduce $f_d(\cdot)$, therefore, the strategy $c^{d,*}$ is the best he/she can adopt. Note that this optimality must be verified for all the players at the same time making $\mathbf{c}^*$ a stable configuration. In the following, we will show how it is possible model the distributed MDC packet classification by a noncooperative strategic game.

B. Game Theory applied to MDC packet classification

Given an MDC video sequence coded into n-descriptions, which are stored into n separate servers or uploading peers, each source node behaves like a player in an n-players non-cooperative game, where the d-th player (description MDd) can choose its strategy (QoS class) $c_i^{\text{MD}d}$ for the i-th packet among a set $\mathcal{N}_c$ of N_c possible choices. The non-cooperative nature of the game depends on the fact that terminals can not communicate among each other. Let $\mathbf{c}_i = \left[c_i^{\text{MD}1}, \dots, c_i^{\text{MD}n}\right]$ be the array that groups the chosen strategies for the n descriptions.

Each class k can be modelled via a queue with maximum length[2] B_k^M and with an emptying server modelled by a Poisson process with parameter λ_k. The parameter λ_k varies according to the congestion level of the network and the transmission priority associated to the k-th class, which changes the probability of accessing the medium and, as a consequence, the waiting time. Each class is also associated to a transmission channel with loss probability $P_{chan,k}$, which depends on the parameter setting (e.g., the adopted channel code, the transmission power, etc...) for the k-th QoS class, and the network state. Since packets are transmitted by each source at a constant packet rate, it is possible to model each QoS class with a G/M/1/B_k^M queue model [26], where the generic arrival rate depends on the classification choices of the algorithm.

When considering the best classification for the packets of description MDd, the loss percentages $P_{loss,k}$, $k = 0, 1, \dots, N_c - 1$, are affected by the channel parameters related to the k-th class, by the parameter λ_k, and by the number of packets B_k buffered in the k-th queue. Note that the parameter B_k depends on the chosen classes for all the descriptions, i.e., on the arrays $\mathbf{c}_t$ with $t \leq i$. More precisely, assuming that $B_k < B_k^M$, the buffer level can be written as

$$B_k = \sum_{t \leq i} \sum_{d=1}^{n} \mathcal{I}\left(c_t^{\text{MD}d} = k\right) - R_{p,k}(i), \tag{3}$$

where $\mathcal{I}(\cdot)$ is the indicator function and $R_{p,k}(i)$ is the number of transmitted packets in the k-th class at the instant i. As a consequence, the loss probability $P_{loss,k}$ for the d-th description can be written as

$$P_{loss,k}(\mathbf{c}_i) = \begin{cases} P_{i,k}^{late}(B_k) + \left(1 - P_{i,k}^{late}(B_k)\right) P_{chan,k} & \text{if } B_k < B_k^M \\ 1 & \text{if } B_k = B_k^M \end{cases} \tag{4}$$

where $P_{i,k}^{late}(B_k)$ is the probability for the class k that the i-th packet can not be delivered in time (i.e., later than a certain time limit), and $P_{chan,k}$ characterizes the packet loss probability related to the network congestion level. Note that $P_{chan,k}$ can vary for the different uploading nodes since network states can differ throughout the network. Unfortunately, the lack of messaging between source terminals makes each node unaware of the transmission conditions for the other terminals, and therefore, it is not possible to obtain a global optimum for the classification process. As a matter of fact, $P_{chan,k}$ corresponds to the loss probability measured by the node itself via some transmission control protocol (in our case, the Real Time Control Protocol - RTCP) and may differ from the actual values throughout the network.

The probability $P_{i,k}^{late}(B_k)$ can be expressed as

$$P_{i,k}^{late}(B_k) = \mathrm{P}\left[\sum_{m=0}^{B_k - 1} \Delta T_m > T_L\right] \tag{5}$$

where ΔT_m is the waiting time for the m-th buffered packet after the $(m-1)$-th packet has been served, and the threshold T_L denotes the maximum time limit after which the packet is discarded since it has become obsolete (in our case we have set T_L to 0.2 s). Assuming that the waiting time ΔT_m is an exponential variable, the overall waiting time for the m-th buffered packet is an Erlang variable, and the resulting probability of losing the m-th packet with $B_k < B_k^M$ is

$$P_{loss,k}(\mathbf{c}_i) = P_{chan,k} + (1 - P_{chan,k}) \ \frac{\gamma(B_k, T_L/\lambda_k)}{\Gamma(B_k)} \tag{6}$$

where $\gamma(\cdot,\cdot)$ and $\Gamma(\cdot)$ are the lower incomplete and the standard Gamma functions, respectively.

In order to evaluate the distortion produced by the class assignment $\mathbf{c}_i$, the model in [27] allows us to parameterize the channel distortion $\delta\text{PSNR}_i^{\text{MD}d}$ associated to the loss of the i-th packet in description MDd as

$$\delta\text{PSNR}_i^{\text{MD}d} = \frac{\text{PSNR} - \text{PSNR}_{l,i}^{\text{MD}d}}{\text{PSNR}} = h_{l,0}^{\text{MD}d} + h_{l,1}^{\text{MD}d} \rho_i^{\text{MD}d}. \tag{7}$$

The parameter PSNR in eq. (7) identifies the Peak Signal-to-Noise Ratio value of the reconstructed frame in case all the packets are correctly received, $\rho_i^{\text{MD}d}$ is the percentage of null transform coefficients for the i-th packet of description MDd,

[1] In this case, the term *correctness* refers to the fact that the beliefs of two players about the actions of a third player are the same [25].

[2] The parameter B_k^M is expressed in terms of maximum number of packets.

TABLE I
NORMAL FORM OF THE CLASSIFICATION GAME.

	$\mathbf{c_i^{MD1} = 0}$	$\mathbf{c_i^{MD1} = 1}$	$\mathbf{c_i^{MD1} = 2}$
$\mathbf{c_i^{MD2} = 0}$	$f_1(0,0)$ $f_2(0,0)$	$f_1(1,0)$ $f_2(1,0)$	$f_1(2,0)$ $f_2(2,0)$
$\mathbf{c_i^{MD2} = 1}$	$f_1(0,1)$ $f_2(0,1)$	$f_1(1,1)$ $f_2(1,1)$	$f_1(2,1)$ $f_2(2,1)$
$\mathbf{c_i^{MD2} = 2}$	$f_1(0,2)$ $f_2(0,2)$	$f_1(1,2)$ $f_2(1,2)$	$f_1(2,2)$ $f_2(2,2)$

$h_{l,j}^{\text{MD}d}$ $(j = 0, 1)$ are the parameters of the distortion model, and $\text{PSNR}_{l,i}^{\text{MD}d}$ is the PSNR value of the reconstructed frame when the i-th packet is lost.

From the distortion model in eq. (7), it is possible to define the cost function for the d-th class as the expected relative PSNR loss

$$\begin{aligned} &f_d : \mathcal{N}_c^n \to \mathcal{R} \\ &f_d(\mathbf{c}_i) = (i-1)\text{E}[f_d]_{i-1} + P_{loss,c_i^{\text{MD}d}}(\mathbf{c}_i)\ \delta\text{PSNR}_i^{\text{MD}d} \end{aligned} \tag{8}$$

where $\text{E}[f_d]_{i-1}$ is the average relative PSNR loss for the previous packets of the d-th description until the $(i-1)$-th instant, i.e.,

$$\text{E}[f_d]_{i-1} = \frac{1}{i-1} \sum_{t<i} f_d(\mathbf{c}_t) \quad \text{for } i > 1. \tag{9}$$

Given the classification $\mathbf{c}_i$, the loss probabilities $P_{loss,c_i^{\text{MD}d}}$, and the percentages $\rho_i^{\text{MD}d}$ for the i-th packet of each description, the adopted classification strategy $\mathbf{c}_i^*$ identifies a Nash equilibrium [13] for the game expressed by equation (8) whenever the condition in eq. (2) is satisfied. Substituting eq. (8) into equation (2), an equilibrium is found if

$$\begin{aligned} &P_{loss,c_i^{\text{MD}d}}(\mathbf{c}_i^*) < P_{loss,k}\left(\left[c_i^{-\text{MD}k,*}, k\right]\right) \\ &\forall d = 1, \ldots, n \qquad \text{and} \qquad \forall k \in \mathcal{N}_c. \end{aligned} \tag{10}$$

Among all the Nash equilibria, the classification strategy identifies the configuration minimizing the average distortion

$$\delta\text{PSNR} = \frac{1}{n} \sum_{d=1}^{n} \text{E}[f_d]_i. \tag{11}$$

According to this strategy, the algorithm avoids that one description jeopardizes all the available bandwidth in case of congestion on a shared link, and at the same time, it creates a certain differentiation between the packets belonging to the same description improving the performance in case of congestions affecting a single description.

In our simulations, we considered an MDC coded video sequence made of 4 descriptions whose packets can be assigned to three different priority classes $\mathcal{N}_c = \{\text{green}, \text{yellow}, \text{red}\}$ [19], [21]. For the sake of conciseness, we will refer to them with the indexes $0, 1, 2$ respectively. Table I reports the normal form of the non-cooperative game associated to the classification task.

Simulations results will show how the proposed classification approach proves to be extremely effective with different congestion settings.

V. CLASSIFICATION USING NONCOOPERATIVE GAMES WITH OPPORTUNISTIC PLAYERS

Experimental results have shown how it is possible to improve the performance of classification by allowing each player to adopt an opportunistic behavior. Since the different descriptions are correlated, it is possible that the utility of player d provides some advantages to players $d' \neq d$. Lost data can be easily approximated from other descriptions, and therefore, a limited distortion in the correlated video streams permits estimating the missing information with a satisfying accuracy. An additional advantage is obtained by considering the configurations $\mathbf{c}_i$ between different consecutive packets (which can be related to one frame or one GOP). Therefore, it is possible to average distortion values for more than one packet considering all the packet data within the set W_p. As a matter of fact, the cost function of eq. (8) for player/description d can be modified as follows:

$$f_d(\mathbf{c}_i) = (i-1)\text{E}[f_d]_{i-1} + D_i^d(\mathbf{c}_i) \tag{12}$$

where the distortion function D_i^d related to packet $i \in W_p$ is computed as

$$\begin{aligned} D_i^d(\mathbf{c}_i) = \sum_{j \in W_p} \Big\{ &(1-\alpha)\ \delta\text{PSNR}_j^{\text{MD}d}\ P_{loss,c_j^{\text{MD}d}}(\mathbf{c}_j) \\ &+ \alpha \prod_{d'=1}^{n} \delta\text{PSNR}_j^{\text{MD}d'}\ P_{loss,c_j^{\text{MD}d'}}(\mathbf{c}_j) \Big\}. \end{aligned} \tag{13}$$

The first term of equation (13), i.e.,

$$\delta\text{PSNR}_j^{\text{MD}d}\ P_{loss,c_j^{\text{MD}d}}(\mathbf{c}_j) \tag{14}$$

focus on intra stream diversity, while the second term

$$\prod_{d'=1}^{n} \delta\text{PSNR}_j^{\text{MD}d'}\ P_{loss,c_j^{\text{MD}d'}}(\mathbf{c}_j) \tag{15}$$

accounts for the influence of inter stream diversity on the final distortion.

Although in theory $\mathbf{c}_i$ can change within W_p for every packet, experimental result has proved that keeping $\mathbf{c}_i$ constant for all the packets in W_p leads to better performances. This behavior has to be related to the fact that it is possible to consider the classification problem as a sequential game where the trigger strategy $\mathbf{c}_i = const.$ within W_p leads to more effective configurations. Changing $\mathbf{c}_i$ within W_p, could imply a temporary lower distortion for some players. Unfortunately, the following reaction of other players would decrease significantly the performance (credibility of the threats).

The proportional weight α $(0 \leq \alpha \leq 1$, which is here called the opportunism ratio), controls how much both types of diversity should influence the final distortion and is related with the characteristics of the coded sequence.

For sequences with high spatial correlation and low temporal correlation, lost descriptions can be easily estimated from the available ones, and therefore, providing a high degree of diversities among the different MD streams (inter stream diversity) increases the robustness of the transmission. On the other hand, the visual quality of the reconstructed sequences with low spatial and high temporal correlations is poorly affected by inter stream diversity and proves to increase whenever inter stream diversity is preserved. As a matter of fact, the parameter α has to be tuned according to the spatial correlation of the coded sequence. In our implementation, we set α as

$$\alpha = \begin{cases} 0 & \text{if } R_S > 20 \\ 1 & \text{if } R_S < 4 \\ 1 - \dfrac{R_S - 4}{16} & \text{otherwise,} \end{cases} \tag{16}$$

with

$$R_S = \frac{|S_x| + |S_y|}{|\Delta_t|} \tag{17}$$

where S_x, S_y are the average vertical and horizontal Sobel operators, and Δ_t is the average temporal gradient. The ratio R_S compares the spatial and temporal correlation (measured via the Sobel operators and the temporal gradients, respectively). This is one of the main improvements with respect to the work in [11], where the classification approach did not take into consideration the possibility of varying the distortion metric according to the characteristics of the coded sequence. Moreover, the employment of mode descriptions enhances the flexibility of the original scheme, as the results in Section VI will show.

Equation (13), together with the parameter adaptation of eq. (16), provides an effective metric that can be used to minimize the channel distortion on the final reconstructed sequence at the decoder. The proposed classification can be adapted to different QoS-aware networks provided that different class of service are defined.

As a matter of fact, the adaptation of the set of strategies for each player to the specified set of QoS classes permits its use in other protocols, e.g., RSVP or MPLS (that is currently supported by many existing network infrastructure and software, like Cisco IOS on Cisco routers). The priority or QoS class can be specified in the DS field of packets (RFC 2474).

In the following section, we will evaluate how the proposed strategy compares with some existing packet classification algorithms.

VI. Simulation results

The proposed classification approach can be applied to several types of networks that differentiate the priority and the handling strategy for the packets. Despite P2P and CDN protocols operate at the highest layers of the protocol stacks, congestions, delays, and packet scheduling take place at the lowest layers of the protocol stacks. Therefore, in our tests we simulated queues and reduction of the transmission capacity for the links at the lowest layers evaluating the effects at the highest layers. In this work we adopted the NS2 network simulator generating different random network topologies that can be underlying a given overlay tree built by a P2P protocol or distributing multimedia packets in a CDN network. This methodology permits both the reproducibility of the tests and the evaluation of the classification on a complex network.

Nowadays, many protocols [19], [20] define several QoS classes which are associated with different transmission priorities, retry limits, and buffer sizes. In this paper we will consider the special case of DiffServ networks [19], but the approach can be easily extended to other scenarios as well.

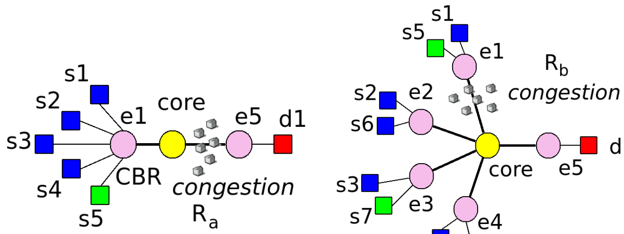
The experimental settings have been simulated using the NS2 simulator and considering a DiffServ-enabled network [19] which classifies the injected packets using three different labels (green, yellow, and red). Each label characterizes the relevance of the packet and the handling strategy. The values of the parameters λ_k, $k = 0, \ldots, N_c - 1$, were estimated from a training set of packet transmissions, while $P_{chan,k}$ are independently evaluated by each node via the RTCP protocol.

In our testing, we simulated the transmission of CIF video sequences coded into 4 descriptions via a spatial polyphase-subsampling. Each sequence was coded with fixed QP into GOPs of 15 frames with structure IP...P. The CABAC entropy coder was enabled as well. Tests were done for different values of the QP parameter. For the sake of conciseness, in the paper we report experimental values for $QP = 25$ and $QP = 30$ since they proved to be representative to evaluate the performance of the algorithm. Other experimental data can be found at the web-site [28].

The algorithm was tested on two different types of congestion, where inter-stream and intra-stream diversities are respectively required in the loss patterns. More precisely, we adopted two different network settings designed in such a way that congestions affect data streams from different peers/servers (see Fig. 3(a) and 3(b)).

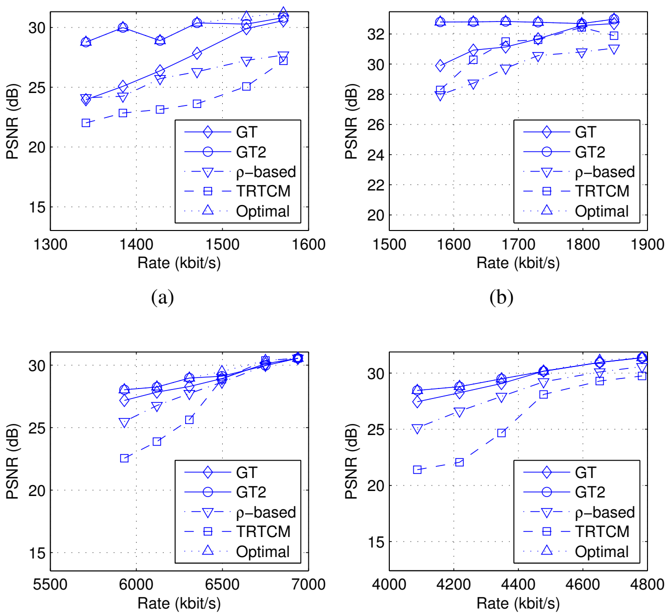
In a first set of simulations, we adopted the network topology shown in Fig. 3(a). The node $d1$ is downloading the MDC coded video sequence from the source nodes $s1$, $s2$, $s3$, and $s4$, while the node $s5$ is streaming a Constant Bit Rate (CBR) traffic towards the node $e5$. The node si, with $i = 1, \ldots, 4$, stores the i-th description of the MDC video sequence while node $s5$ has the only purpose of adding extra packets in the network. The access to the core network is ruled by the DiffServ-enabled node $e1$ [19] that process the packets according to their labels and the buffer levels. The congestion is varied reducing the available bandwidth R_a of the link from the the core network to node $e5$. As a matter of fact, all the descriptions are contending the same link, and therefore, congestion affects all the uploading peers/nodes. The performance of the GT-based approach (labelled GT2 in the reported graphs) is compared with the performances of the classification strategy in [11] adapted to 4 MDC descriptions (labelled GT), the Algorithm 1 proposed in [7] (labelled ρ-based), and with the standard technique TRTCM [21]. In the graphs we also added a fifth curve showing the performance of an optimal classification (labelled Optimal) that assumes to know all the channel conditions for all the uploading peers/nodes.

The simulation results reported in Fig. 4 show that the GT2 approach improves the PSNR value of 4 dB on average for the sequence `foreman` (see Fig. 4(a)) with respect to the TRTCM approach. Note also that the GT approach in [11] does not



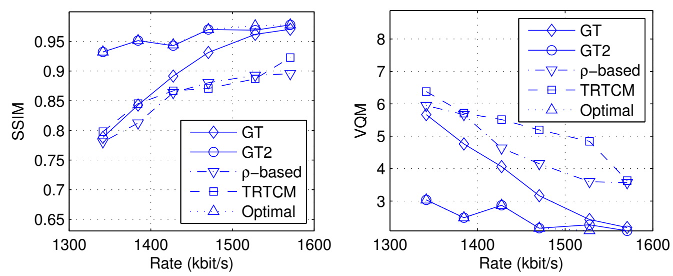
(a) (b)

Fig. 3. Network settings. a) Setting 1: congestion on a shared link; b) Setting 2: congestion on a non-shared link.



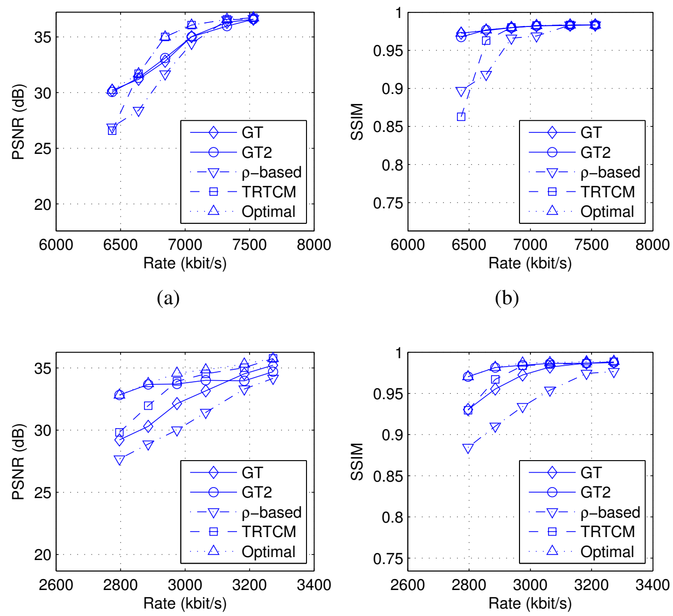
(a) (b) (c) (d)

Fig. 4. Simulation results for different sequences coded with $QP = 30$ on network 1. The graphs report the value of average PSNR as function of the transmission rate R_a. a) `foreman`, b) `crew`, c) `mobile` d) `flower`.



(a) (b)

Fig. 5. Simulation results for the sequence `foreman` coded with $QP = 30$ on network 1. The graphs report the value of average SSIM and VQM as function of the transmission rate R_a. a) SSIM, b) VQM.



(a) (b) (c) (d)

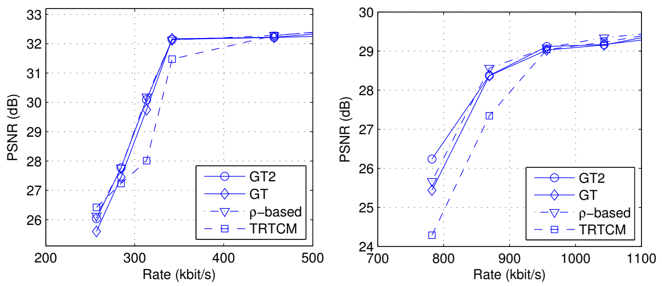
Fig. 6. Simulation results for different sequences coded with $QP = 25$ and transmitted on network 1. The graphs report the value of PSNR as function of the transmission rate R_a. a) PSNR and b) SSIM for `flower`; c) PSNR and d) SSIM for `foreman`.

perform quite effectively on 4-descriptions MDC streams. A quality improvement can be noticed for other sequences as well (see Fig. 4(b) and Fig. 4(c)). However, in our tests we also measured the quality improvement using different quality metrics including the Structured Similarity Index Measure (SSIM) [29] and the Video Quality Metric (VQM) [30].[3] This evaluation aimed at discriminating false quality improvements which are highlighted by a single metric but not confirmed by the others. As an example, we report here the average values for SSIM (Fig. 5(a)) and VQM (Fig. 5(b)) for the sequence `flower`. The displayed data confirm that the GT2 algorithm provides the best quality at the receiver.

The GT2 algorithm also proves to be effective with MDC sequences coded at different quality levels. Figure 6 reports the average PSNR and SSIM values for the sequences `foreman` and `flower` coded with $QP = 25$. For the sequence `flower`, the proposed strategy improves the average PSNR value of approximately 3.3 dB at high congestion levels with respect to the TRTCM approach (Fig. 6(a)). However, for high R_a values TRTCM seems to provide a better visual quality since the average PSNR values are higher. This fact is not confirmed by the other quality metrics (Fig. 6(b)) which shows that at low congestions (high R_a values) the visual quality provided by TRTCM and GT2 is the same. As for the sequence `foreman`, the performance proves to be slightly worse than TRTCM at high rates (low congestion), but the PSNR values allow to assert that this difference is negligible (Fig. 6(c)). On the contrary, at high congestions the GT2 algorithm significantly improves the quality of the reconstructed sequence up to 3 dB.

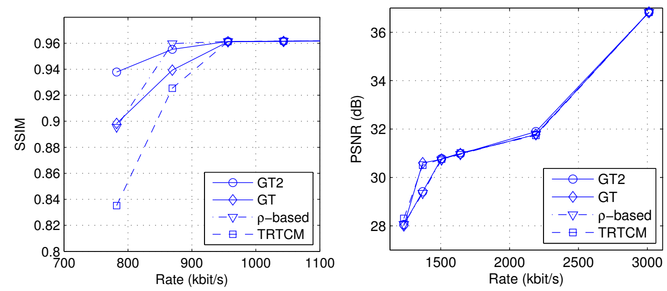
In a second set of simulations, the network topology has been slightly modified, as Fig. 3(b) shows. In this case, the congestion is simulated reducing the available bandwidth R_b of the link from node $e1$ to core network, while descriptions

[3]Note that for most of the reconstructed sequences the visual quality increases as the VQM value decreases and the SSIM index increases.



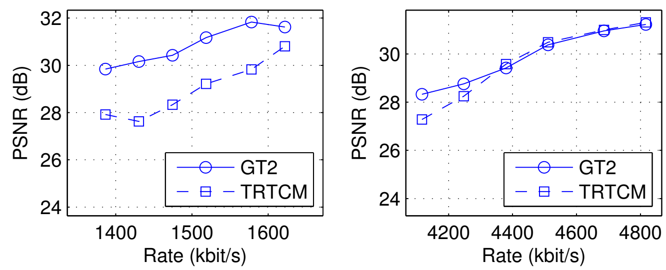
(a) (b)

Fig. 7. Simulation results for different test sequences coded with $QP = 30$ and transmitted on network 2. The graphs report the average values of PSNR (dB) as function of the transmission rate available R_b. a) `foreman`, b) `flower`.



(a) (b)

Fig. 8. Simulation results for the sequence `flower` on network 2. The graphs report the average values of SSIM and PSNR as functions of the transmission rate available R_b. a) average SSIM for $QP = 30$, b) average PSNR for $QP = 25$.



(a) (b)

Fig. 9. Simulation results for unbalanced descriptions. Descriptions MD1, MD2, MD3, and MD4 are coded with QP values 28, 29, 31, and 32, respectively, and transmitted on network 1. The graphs report the value of average PSNR as function of the transmission rates R_a. a) `foreman`, b) `flower`.

MDi, $i = 2, 3, 4$, are transmitted to core network with no congestions. Nodes ek, $k = 1, \ldots, 5$, are DiffServ-enabled, while nodes sk, with $k = 5, \ldots, 8$, have the only purpose of adding extra CBR packets in the network.

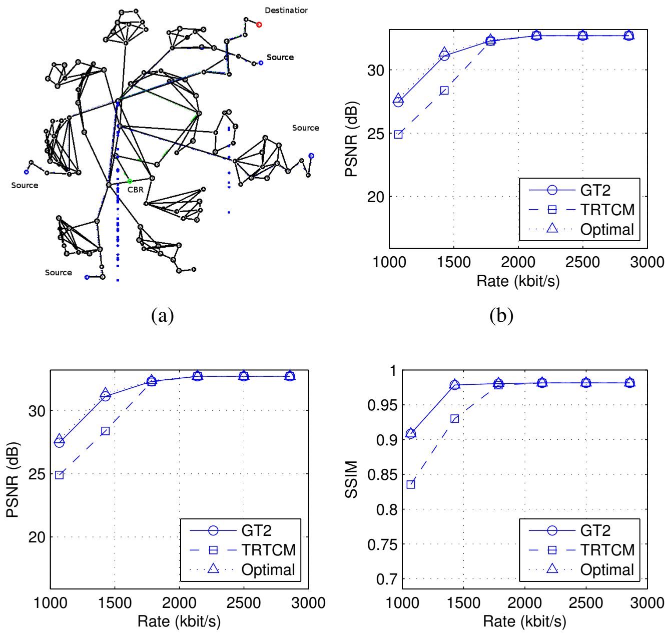
Fig. 7 reports the average PSNR values vs. the rate R_b for the GT2, the GT, the ρ-based, and the TRTCM algorithms. For the sequence `foreman` coded with $QP = 30$, the GT2 approach increases the PSNR value of 0.5 dB on average with respect to the TRTCM approach (see Fig. 7(a)). This improvement is possible since the GT2 approach is able to preserve the most important packets in the decoding process. The reported results also show that the ρ-based approach performs as well as the GT2 algorithm since also in this case the QoS class for each packet is changed according to the distortion introduced by its loss. As for the sequence `flower` (see Fig. 7(b)), the average PSNR value for the GT2 and the ρ-based approaches improves of 2 dB for $R_b = 800$ kbit/s and tends to 0 dB for higher rates. This behavior is confirmed by the VQM and SSIM quality metrics (see Fig. 8(a)) since no significant differences can be noticed among the curves in Fig. 8(a) for $R_b \geq 1$ Mbit/s. Comparing approaches GT2 and GT, it is possible to notice that sometimes the performance of GT2 is slightly lower. A major motivation of this is to be found in the opportunistic behavior of players in GT2 that are more willing to help other nodes in order to gain some advantages. Note that in case the congestion is affecting a single node, opportunism is useless (but this can not be known by the nodes). As a matter of fact, the performance of GT is higher. It is also worth of observation that this difference is minimal (< 0.1 dB in most cases), and therefore, the performance of both algorithms is approximately the same for network 2. The improvements obtained by GT2 on network 1 justifies its adoption. Whenever the sequences are coded at a higher bit rate ($QP = 25$), it is possible to verify that the performance of the different algorithms does not differ too much. However, in most of the cases the performance improvement of GT2 in network setting 2 is lower with respect to that in network setting 1.

In the end, we evaluated the performance when the MDC scheme produces unbalanced descriptions coding each sub-sampled sequence with different QPs. Experimental results compare the TRTCM algorithm with the GT2 approach (for the sake of conciseness) on network setting 1 and show an improvement up to 2 and 1.5 dB for the sequences `foreman` and `flower`, respectively (see Fig. 9). No significant differences have been found on network setting 2.

The third network setting considered a random network of nodes (generated in the NS2 simulator with the software GT-ITM), where the source and the destinations nodes are attached randomly at each network realization. An example is reported in Fig. 10(a). In this way it is possible to simulate a more realistic condition where both network setting 1 and 2 are combined together multiple times. The bandwidth capacities of the different links are assigned randomly following a normal distribution with different averages and standard deviation equal to 100 kbit/s. Note that congestions may affect a variable number of descriptions since paths could be completely independent or could share one or more links. The displayed results show how the performances verified on network 1 and 2 are combined and prove the effectiveness of GT2 approach once again.

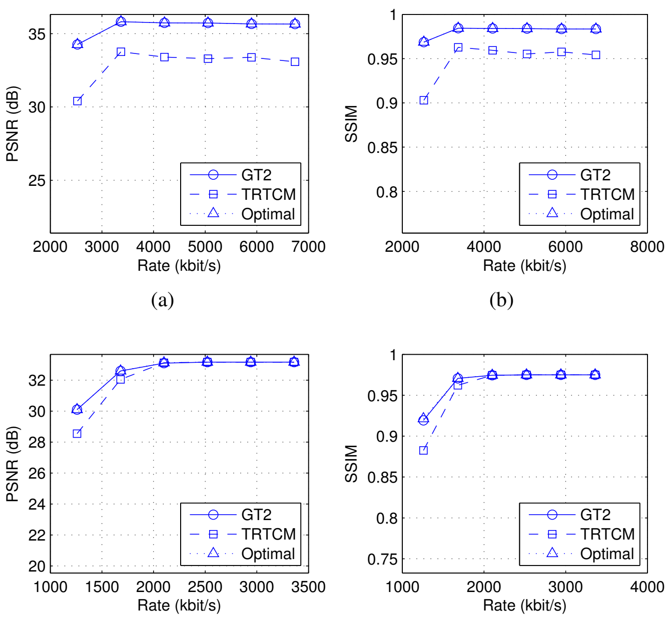
Figures 10 b,c,d report the value of PSNR, VQM, SSIM obtained for the sequence `foreman` with $QP = 30$. It is possible to appreciate that the performance of the GT2 strategy is close to the optimal one for all the considered metrics.

Figure 11 reports the value of PSNR and SSIM obtained for the sequence `crew` with $QP = 25$ and 30. In this case, the gain provided by the GT2 approach is higher since the stronger



(a) (b) (c) (d)

Fig. 10. Average simulation results on GT-ITM random networks (10 realizations) for sequence `foreman` with $QP = 30$. The graphs report the value of PSNR, VQM, and SSIM as a function of the average transmission rate of links (Gaussian distribution). a) network realization 0; b) PSNR; c) VQM ; d) SSIM.



(a) (b) (c) (d)

Fig. 11. Average simulation results on GT-ITM random networks (10 realizations) for sequence `crew` with $QP = 25, 30$. The graphs report the value of PSNR and SSIM as a function of the average transmission rate of links (Gaussian distribution). a) PSNR $QP = 25$; b) SSIM $QP = 25$; c) PSNR $QP = 30$; d) SSIM $QP = 30$.

spatial correlation between the different streams is exploited better by the GT2 optimization. This fact is more evident at low QPs (see Fig. 11(a) and 11(b)), where the performance of TRTCM is always lower with respect to that of GT2 since GT2 is more effective in preserving inter-stream diversity.

TABLE II
COMPUTATIONAL TIME FOR DIFFERENT CLASSIFICATION

Algorithm	**Execution Time (ms)**	**Relative Time Sav. w.r.t. TRTCM**
Fixed	7.77	+0 %
GT	12.80	+64.74 %
GT2	12.82	+64.99 %
TRTCM	–	–
ρ-based	7.89	1.54 %
Optimal	12.92	66.28 %

A. *Computational complexity of the approach*

The proposed classification approach requires an approximate knowledge on the transmission conditions (inferred from RTCP statistics) and of the statistics of data to be transmitted. The transmitted sequence can be described by a set of distortion values which can be computed during the coding operations with negligible additional computation with respect to the encoding complexity. As a consequence, the classification strategy has to process a limited amount of the data and combine them in a 4-players game to find its equilibria. The algorithm runs in 0.22 ms per frame on a 1.7 GHz processor with 1 GB RAM, and therefore, it proves to be suitable for low complexity devices. Table II reports the computational time required by different packet classification approaches (for a whole sequence). It is possible to notice that the required time is quite low for all the approaches despite the `fixed`, ρ-based, and TRTCM strategies requires the minimum amount of calculation. No values are displayed for the TRTCM strategy since the classification is performed according to the occupancy levels of the queues on the network (no content-dependent packet classification is applied). Note also that the complexity of the optimal approach is not far from that of GT and GT2 approaches but requires the knowledge of transmission conditions for all the peers (which is not feasible in a real scenario).

VII. CONCLUSION

The paper presents a classification approach for an Multiple Description Coded video sequence which is based on a non-cooperative game. The strategy identifies Nash equilibria and chooses the configuration that minimizes the estimated distortion in the sequence reconstructed at the decoder. The approach grants an adequate level of intra-stream and inter-stream diversities improving the PSNR values both in case the congestion affects a network link shared by the different descriptions and in case the congestion only affects a single description. Future work will be focused both on extending this approach to more efficient MDC video schemes and on testing the algorithm using more complex and heterogeneous network topologies.

REFERENCES

[1] J. Rckert, O. Abboud, T. Zinner, R. Steinmetz, and D. Hausheer, “Quality adaptation in p2p video streaming based on objective qoe metrics,” in *NETWORKING 2012*, ser. LNCS. Springer, 2012, vol. 7290, pp. 1–14.

[2] S. Milani and G. Calvagno, “Reliable 3d video p2p transmission enabling synthesis of virtual views,” May 2 – 4, 2012, pp. 1–4.

[3] O. Abboud, K. Pussep, A. Kovacevic, K. Mohr, S. Kaune, and R. Steinmetz, “Enabling resilient p2p video streaming: survey and analysis,” *Multimedia Systems*, vol. 17, no. 3, pp. 177–197, Jun. 2011.

[4] N. Ramzan, E. Quacchio, T. Zgaljic, S. Asioli, L. Celetto, E. Izquierdo, and F. Rovati, "Peer-to-peer streaming of scalable video in future internet applications," *IEEE Comm. Mag.*, vol. 49, no. 1, pp. 128–135, Feb. 2011.

[5] E. Akyol, A. M. Tekalp, and M. R. Civanlar, "A flexible multiple description coding framework for adaptive peer-to-peer video streaming," *IEEE J. Select. Topics Signal Processing*, vol. 1, no. 2, pp. 231 – 245, Aug. 2007.

[6] V. K. Goyal, "Multiple Description Coding: Compression Meets The Network," *IEEE Signal Process. Mag.*, vol. 8, no. 5, pp. 74–93, Sep. 2001.

[7] S. Milani, G. Calvagno, R. Bernardini, and R. Rinaldo, "A Low-Complexity Packet Classification Algorithm for Multiple Description Video Streaming over IEEE802.11e Networks," in *Proc. of the 2008 IEEE International Conference on Image Processing (ICIP 2008)*, San Diego, CA, USA, Oct.12 – 15, 2008.

[8] A. Russo and R. L. Cigno, "Delay-aware push/pull protocols for live video streaming in p2p systems," in *Proc. of ICC 2010*, Cape Town, South Africa, May 23-27, 2010, pp. 1–5.

[9] X. Wu, B. Ma, and N. Sarshar, "Rainbow network problems and multiple description coding," *IEEE Trans. Inf. Theory*, vol. 54, no. 10, pp. 4565 – 4574, Oct. 2008.

[10] S. Zezza, E. Magli, G. Olmo, and M. Grangetto, "SEACAST: a protocol for peer-to-peer video streaming supporting multiple description coding," in *Proc. of ICME 2009*, New York, USA, Jun. 28 – Jul. 3, 2009.

[11] S. Milani and G. Calvagno, "A Game Theory Based Classification for Distributed Downloading of Multiple Description Coded Videos," in *Proc. of the 2009 IEEE International Conference on Image Processing (ICIP 2009)*, Cairo, Egypt, Nov. 24 – 28, 2009, pp. 3077 – 3080.

[12] E. Jorswieck, E. Larsson, M. Luise, and H. Poor, "Game theory in signal processing and communications [from the guest editors]," *IEEE Signal Process. Mag.*, vol. 26, no. 5, pp. 17 –17, Sep. 2009.

[13] F. Meshkati, H. Poor, and S. Schwartz, "Energy-efficient resource allocation in wireless networks," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 58 –68, May 2007.

[14] A. Leshem and E. Zehavi, "Game theory and the frequency selective interference channel," *IEEE Signal Process. Mag.*, vol. 26, no. 5, pp. 28–40, Sep. 2009.

[15] D. Schmidt, C. Shi, R. Berry, M. Honig, and W. Utschick, "Distributed resource allocation schemes," *IEEE Signal Process. Mag.*, vol. 26, no. 5, pp. 53 –63, Sep. 2009.

[16] S. Milani and G. Calvagno, "Improving quality-of-experience for multiple description video transmission in peer-to-peer networks," in *Proceedings of the 2010 ACM workshop on Advanced video streaming techniques for peer-to-peer networks and social networking*, ser. AVSTP2P '10. New York, NY, USA: ACM, 2010, pp. 65–70. [Online]. Available: http://doi.acm.org/10.1145/1877891.1877907

[17] S. Milani, M. Gaggio, and G. Calvagno, "3dtv streaming over peer-to-peer networks using fec-based noncooperative multiple description," in *Proceedings of the 2011 IEEE Symposium on Computers and Communications (ISCC 2011) - Worshop MediaWin 2011*, ser. ISCC '11. Washington, DC, USA: IEEE Computer Society, Jun. 28 - Jul. 1, 2011, pp. 13–18. [Online]. Available: http://dx.doi.org/10.1109/ISCC.2011.5984018

[18] T. Wiegand, "Version 3 of H.264/AVC," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6),* 12^{th} *Meeting*, Redmond, WA, USA, Jul. 17 – 23, 2004.

[19] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Service," IETF, Tech. Rep., RFC 2475, Dec. 1998, internet Draft. [Online]. Available: http://www.ietf.org/rfc/rfc2475.txt

[20] IEEE 802.11/D13.0, Part 11, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, IEEE Std. 802.11e, Jan. 2005.

[21] J. Heinanen and R. Guerin, "A Two Rate Three Color Marker," IETF, Tech. Rep., RFC 2698, Sep. 1999, internet Draft. [Online]. Available: http://www.ietf.org/rfc/rfc2698.txt

[22] S. Milani and G. Calvagno, "A low-complexity rate allocation algorithm for joint source-channel video coding," *Signal Processing: Image Communications*, vol. 24, no. 5, pp. 368–383, May 2009. [Online]. Available: http://dx.doi.org/10.1016/j.image.2009.02.005

[23] R. Bernardini, M. Durigon, R. Rinaldo, P. Zontone, and A. Vitali, "Real-Time Multiple Description Video Streaming Over QoS-Based Wireless Networks," in *Proc. of International Conference on Image Processing (ICIP 2007)*, San Antonio, TX, USA, Sep. 2007, pp. 245 – 248.

[24] I. V. Bajić, "The Effects of Channel Correlation on the Performance of Some Multiple Description Schemes," in *Proc. of* 10^{th} *Canadian Workshop on Information Theory (CWIT 2007)*, Edmonton, Alberta, Canada, Jun.6–8, 2007, pp. 93 – 96.

[25] M. J. Osborne, *An Introduction to Game Theory*. Oxford University Press, 2004.

[26] D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*. Wiley, 1998.

[27] S. Milani, G. Calvagno, R. Bernardini, and P. Zontone, "Cross-Layer Joint Optimization of FEC Channel Codes and Multiple Description Coding for Video Delivery over IEEE 802.11e Links," in *Proc. of the 2008 IEEE International Conference on Future Multimedia Networks (FMN 2008) (co-located with NGMAST2008)*, Cardiff, Wales, GB, Sep.17 – 18, 2008, pp. 472 – 478.

[28] S. Milani. (2011) Simone Milani's Home Page. web site. [Online]. Available: http://www.dei.uniud.it/∼sim1mil

[29] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600 – 612, Apr. 2004.

[30] F. Xiao. (2000, Winter) DCT-based Video Quality Evaluation. MSU Graphics and Media Lab (Video Group). [Online]. Available: http://compression.ru/video/quality_measure/vqm.pdf