An Adaptive Cross layer Resource Allocation Scheme for Correlated Wireless Video Sources

Dorna Bandari*, Pascal Frossard†, Gregory Pottie*
*University of California, Los Angeles, USA
†École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
Email: dorna@ucla.edu, pascal.frossard@epfl.ch, pottie@icsl.ucla.edu

***Abstract*—In this work we study adaptive resource allocation for uplink transmission of correlated video sources. We consider a framework where multiple wireless sources transmit correlated information via a common base station. We introduce an optimization problem for sources to maximize the weighted sum of received quality of all videos, when source correlation can be used at the decoder in case of missing data. Each source finds its respective Multiple Access (MAC) parameters and performs packet selection. This is done with minimal information exchange with the base station. We model the quality of a decoded video as a piecewise linear function of qualities of the most correlated views, and verify the validity of the piecewise linear quality model for a two source case. We use this model to simplify the resource allocation method. We then compare performance of our correlated resource allocation to optimal resource allocation for independent sources, as well as to a baseline method. The simulations show that our proposed method results in higher average Y-PSNR than the optimal independent resource allocation in most channel conditions, without significative complexity increase in the base station or the source nodes.**

***Index Terms*—resource allocation, cross-layer design, correlated videos.**

I. Introduction

In this work we envision a scenario where several cameras capture a live event and stream it simultaneously to a base station. The applications are live coverage of concerts, conferences, or political events. With free-view and multiview video becoming more popular, and with limited resources in wireless transmission, we expect an increased demand for more efficient streaming of correlated videos.

We consider the scenario of N video sources transmitting correlated video streams through a shared wireless channel to a common base station, before delivery to the decoder. In this work we consider a regular H.264 video encoder and a simple decoder which uses correlation between sources for concealment of missing frames. The goal is to maximize the weighted sum of received video qualities given the resource constraints. The decoder may use source correlation to decode the N videos in case of insufficient bandwidth or loss, as shown in Figure 1. We consider Code Division Multiple Access (CDMA) as the MAC scheme, however the methods developed in this paper can be applied to other access schemes. We formulate an optimization problem that selects the best code, power assignment, and packet selection at each wireless source, with minimal feedback from the base station.

We introduce a piecewise linear model for the quality of each video decoded using correlated information and show the validity of the model for a two node case. This model enables us to split the global optimization problem into separate problems that are solved at each station, and to iteratively converge to the global optimum using updates from the base station. We apply a cross-layer algorithm for the sources to find their individual optimum MAC parameters and packet scheduling with minimal information exchange with the base station.

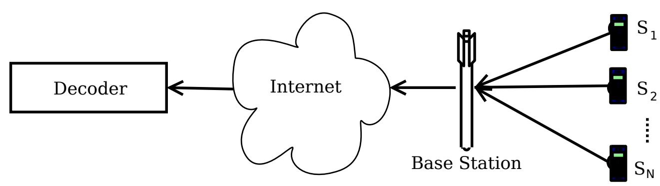


Fig. 1. General framework: N sources stream live video to the base station on a shared bottleneck channel, before it is forwarded to the decoder.

When transmitting dependent videos, it is desirable to use correlation characteristics to increase communication efficiency. An area of research that considers correlated video streams is multiview video coding (MVC). In MVC multiple cameras capture the same scene from different views, producing correlated video sequences that are encoded jointly in order to increase the overall compression performance [1], at the price of inter-camera communication. Alternatively, distributed coding techniques have been proposed for the independent encoding of correlated camera sources; the complexity is shifted to the joint decoder that exploits the inter-view correlation for effective reconstruction [2]. DSC has been studied jointly with the resource allocation problem in [3]. However, the resource allocation method does not take advantage of the developments in cross-layer design.

In general, the wireless resource allocation for correlated video sources has not been studied extensively in the literature. In the area of multimedia communication for independent sources, it has been thoroughly demonstrated that there is significant benefit in using cross-layer design as compared to traditional layered design for live video transmissions [4] [5]. The works in [6] and [7] propose optimal solutions for a utility-based cross-layer resource allocation. These methods however consider downlink communications, which does not present the same challenges as uplink, in particular the fact that each sources' individual constraints (e.g., power) also needs to be considered.

Among uplink optimum resource allocation solutions, the following three papers are most related to our work. In [8] an uplink OFDM system is considered and a utility-based objective function is optimized, although not specific to video sources. In [9] optimal resource allocation is performed for uplink transmission of video sources; specifically algorithms are developed to assign MAC resources to each video source in a centralized manner, optimizing overal received quality. In [10] optimum rate allocation and packet scheduling for video sources is found for uplink tranmission, with the objective of maximizing the overall video quality. However the individual MAC parameters are not found, which does not allow for adaptation to changes in availability of individual resources. None of the works explicitly consider the sources' correlation in the rate allocation problem.

The rest of the sections of this paper are as follows. In Section II we introduce the framework of our proposed method and formulate the problem. In Section III we discuss the piecewise linear model and give details on the optimization solution. In Section IV we present simulations for validation of the piecewise linear quality model, as well as overall resource allocation results.

II. Problem Settings

A. Framework

We consider the transmission of N correlated video sources through one common base station. We consider that the sources have no possibility to precisely adapt the video encoding to the actual transmission conditions, so that the rate allocation problem becomes equivalent to a packet scheduling optimization. A joint decoder reconstructs each of the views. It uses the inter-view correlation information to compensate missing information due to frames that have been lost or dropped. We consider here a very simple concealment strategy where missing frames are replaced by the corresponding data in the most correlated views. Note that the error concealment can be chosen differently without affecting the resource allocation framework proposed in this paper.

Our objective is to maximize the weighted sum of qualities of all received videos. In case of concealment we model the quality of each video as the weighted sum of qualities of its neighbours, where the weights depend on video characteristics, correlation between sources, and packet loss probability. Then the global optimization problem is split into local optimization problems solved at each source, and the global optimum is reached iteratively by using updates from the base station. As we choose CDMA as the Multiple Access scheme, the variables to be found are the optimum code and power assignment and packet scheduling for each individual source.

B. Problem Formulation

First, we note that the transmission rate can be written as a function of the MAC parameters. In a Gaussian multiple-access channel, the achievable rate for user *i* in terms of CDMA parameters is given by [11]

$$R_i = k_i.B.log(1 + \zeta_i.\frac{p_i}{k_i}), \tag{1}$$

where p_i is the power allocated to user *i* and k_i is the number of orthogonal codes assigned to it. B is the total available bandwidth, and $\zeta_i = \beta_i \frac{g_i}{N_0.B}$. The parameter β_i is called the SNR gap, which accounts for the difference between the theoretical achievable rate and the achievable rate in a real system. g_i is the channel gain for user *i*, and N_0 is the noise floor. We name $\frac{N_0.B}{g_i}$ the *Normalized Noise* for user *i*.

The resource allocation problem consists in finding the optimal code choice, power assignment and packet selection at each of the CDMA sources, so that the overall quality is maximized. The problem can be formulated as follows.

$$\begin{aligned} \underset{(p_i,k_i)}{\text{minimize}} \quad & -Q_T(\boldsymbol{R}) \\ s.t. \quad & p_i \le P_{iMAX} \qquad\qquad \forall i \in [1:N] \\ & \sum_{i=1}^{N} k_i \le K \\ & \sum_{i=1}^{N} p_i \le P_T \end{aligned} \tag{2}$$

$Q_T(R)$ represents the objective function, with the aggregate decoding quality. It is a function of $\boldsymbol{R}$, the vector of rates of each source as given by Eq (1). P_{iMAX} is the maximum power for source *i*, K is the total number of available codes, and P_T is the total power that can be used by the network. In the next section we propose a solution to optimize this problem with a simple iterative algorithm.

III. Optimal Resource Allocation

A. Aggregate Utility Model

In general, the quality of a decoded video stream, when decoded using correlation irrespective of the decoding scheme, is a function of its own data rate as well as data rates of videos that are correlated to it. When the rates are denoted by the vector $\boldsymbol{R}$, the quality of video *i* becomes

$$Q_i^c(\boldsymbol{R}) = f(R_1, R_2, R_3, ...) \tag{3}$$

A concave, monotonically increasing quality-rate function can be constructed for any video stream. When the stream is pre-encoded, rate adaptation can be achieved by packet filtering. A packet ordering algorithm such as one proposed in [12] can be used to order the frames so that the least important packets are dropped first when resources become scarce.

That being said, our resource allocation method does not require all nodes to order their frames, and if complexity is strictly constrained, each node can choose to use a known model for its Q-R function. Of course, using a model results in a sub-optimal solution. One such function is given in [13]. Whether a known model is used or an optimum algorithm utilized, the resulting quality-rate function is bijective. Therefore we can replace the rate in equation (3) by the quality from the estimated Q-R function, formulating the decoding quality as a function of quality of the correlated video sequences, i.e.,

$$Q_i^c(\boldsymbol{R}) = f(Q_1(R_1), Q_2(R_2), Q_3(R_3), ...)$$

We now propose a simple piecewise first order approximation of the above function in the decoder, with a linear combination of qualities:

$$Q_i^c(\boldsymbol{R}) = \sum_{j=1}^{N} \alpha_{ij}^m . Q_j(R_j) + \delta_i^m, \quad \boldsymbol{Q}_N \in C_m^N. \tag{4}$$

$\boldsymbol{Q}_N$ is the array of quality-rate functions, i.e. $[Q_1(R_1)...$-$Q_N(R_N)]$. The model parameters, α_{ij}^ms and δ_i^ms are estimated for the range of values of $\boldsymbol{Q}_N$ that belong to the N dimensional space C_m^N. In this work C_m^Ns are constructed by partitioning the range of possible values for qualities of each video into 2dB sections. Therefore any value for the array of qualities, $\boldsymbol{Q}_N$, belongs to one such space.

The model parameters are initially calculated at the decoder by decoding videos in two ways; regular decoding, which results in videos with qualities $\boldsymbol{Q}_N$, and correlated decoding, which is used to find values for $Q_i^c(\boldsymbol{R})$s. Then, for each range of video quality values, C_m^N, the model parameters, $\boldsymbol{\alpha}^m$ and $\boldsymbol{\delta}^m$, are found by fitting the video quality values to the model from Equation (4), using LMS.

Depending on the correlation level and the qualities of the correlated videos, it is possible for the correlated decoding method to decrease the quality of a decoded video when compared to regular decoding, i.e., for some *i* and some *m*,

$$Q_i^c(\boldsymbol{R}) < Q_i(R_i), \quad \boldsymbol{Q}_N \in C_m^N.$$

To ensure that we always decode using the method that achieves the higher decoded quality, the decoder can simply set the model parameters to a row of the identity matrix in such cases.

$$\text{if } \sum_{j=1}^{N} \alpha_{ij}^m . Q_j(R_j) + \delta_i^m < Q_i(R_i), \text{ set } \begin{cases} \alpha_{ii}^m = 1, \\ \alpha_{ij}^m = 0, \ \forall j \neq i \\ \delta_i^m = 0. \end{cases}$$

The aggregate quality of the network finally becomes

$$\begin{aligned} Q_T(\boldsymbol{R}) &= \sum_{i=1}^{N} \gamma_i . Q_i^c(\boldsymbol{R}) \\ &= \sum_{j=1}^{N} \left(\sum_{i=1}^{N} \gamma_i . \alpha_{ij}^m . Q_j(R_j) \right) + \sum_{i=1}^{N} \delta_i^m . \gamma_i \\ &= \sum_{j=1}^{N} \eta_j . Q_j(R_j) + \delta_j^m . \gamma_j, \end{aligned} \tag{5}$$

where $\eta_j = \sum_{i=1}^{N} \gamma_i . \alpha_{ij}^m$ and γ_i is a parameter set by network administrators as a measure of relative importance of each video stream in the aggregate quality function. It can be shown that the above is a concave function of p_i and k_i [14], which makes (2) a convex optimization problem with linear constraints.

B. Optimization solution

In order to solve the optimization problem presented in Section II-B we first formulate it as an unconstrained optimization problem. The Lagrangian cost function is given by

$$\begin{aligned} L(\boldsymbol{k}, \boldsymbol{p}, \boldsymbol{\lambda}, \nu, \omega) = &-Q_T(\boldsymbol{R}) + \boldsymbol{\lambda}.[\boldsymbol{p} - \boldsymbol{P_{MAX}}]^T \\ &+ \nu.(\sum_{i=1}^{N} k_i - K) + \omega.(\sum_{i=1}^{N} p_i - P_T), \end{aligned}$$

where $\boldsymbol{\lambda} = [\lambda_1 \lambda_2 ... \lambda_N]$, ν and ω are the dual variables, $\boldsymbol{p} = [p_1, p_2, ..., p_N]$ is the vector of power assignments, $\boldsymbol{k} = [k_1, k_2, ..., k_N]$ is the vector of number of codes assigned to each user, and $\boldsymbol{P_{MAX}} = [P_{1MAX}, P_{2MAX}, ..., P_{NMAX}]^T$ is the vector of power limits of each user. *K* is the total number of codes, and P_T is the maximum total power allowed in the network. Then taking the infimum over $\boldsymbol{k}$ and $\boldsymbol{p}$ will result in the Lagrange dual function,

$$g(\boldsymbol{\lambda}, \nu, \omega) = \inf_{\boldsymbol{k}, \boldsymbol{p}} (L(\boldsymbol{k}, \boldsymbol{p}, \boldsymbol{\lambda}, \nu, \omega))$$

In this problem strong duality holds, therefore solving the Lagrange dual function will solve the primal problem. To solve the unconstrained concave dual problem, the partial derivatives of Lagrangian function with respect to **p** and **k** are set to zero. For each station we get the following two equations,

$$\begin{aligned} B.\eta_i. \left[\log(1 + \zeta_i . \frac{p_i}{k_i}) - \frac{p_i . \zeta_i}{(k_i + p_i . \zeta_i) . \ln(10)} \right] . \frac{dQ^{(i)}(R_i)}{dR_i} \\ = \nu \end{aligned} \tag{6}$$

$$\frac{B.\eta_i.\zeta_i}{\ln(10)} . [\frac{k_i}{k_i + p_i . \zeta_i}] . \frac{dQ^{(i)}(R_i)}{dR_i} = \lambda_i + \omega \tag{7}$$

Once the above two equations are solved for p_i and k_i in each station, the dual optimal variables can be found iteratively using the sub-gradient method. Starting from $\boldsymbol{\lambda}^0, \omega^0, \nu^0$, repeat,

$$\lambda_i^{k+1} = (\lambda_i^k + \theta . (p_i^* - P_{iMAX}))^+ \tag{8}$$

$$\nu^{k+1} = (\nu^k + \delta . (\sum_{i=1}^{N} k_i^* - K))^+ \tag{9}$$

$$\omega^{k+1} = (\omega^k + \varepsilon . (\sum_{i=1}^{N} p_i^* - P_T))^+ . \tag{10}$$

$\theta, \delta,$ and ε are small constants, and $(x)^+$ is 0 for $x \leq 0$ and x otherwise. Since source nodes do not have access to information about other stations' power and code assignment, the variables ν^{k+1} and ω^{k+1} have to be computed at the base station or the receiver. Their value is periodically broadcasted back to the stations. Algorithm 1 describes this method.

At each source, the system of equations, (6) and (7) is solved by defining a new variable, $X_i = (1 + \frac{p_i}{k_i} . \zeta_i)$, and dividing the two equations in order to eliminate $\frac{dQ^{(i)}(R_i)}{dR_i}$. After rearranging, we get the following:

$$X_i . (\log(X_i) - \frac{1}{\ln 10}) = \frac{\nu . \zeta_i}{(\lambda_i + \omega) . \ln 10} - \frac{1}{\ln 10} . \tag{11}$$

Algorithm 1: Optimal Resource Allocation

Input: Constraints: $K, P_T, \boldsymbol{P}_{MAX}$; Channel parameters: $B, \boldsymbol{\eta}, \boldsymbol{\zeta}$; C_m^Ns found by 2dB partitions of video quality values.

Initialization: $\nu^0 = \omega^0 = 0.5, \lambda_i^0 = 0.5 \quad \forall i,\ k = 0$, flag $= 0, \boldsymbol{\alpha}^n = I_{N\times N} \quad \forall n, and m = 0$.

Repeat:

At each source, i**:**

1) Capture and encode W_k video frames.
2) Arrange frames in the order of contribution to video quality, or assume a model for the Q-R plot, as in Section III-A.
3) Using ν^k, ω^k and λ_i^k find p_i^*, k_i^* and optionally the optimum set of frames to transmit by solving Equations (11) through (13).
4) Transmit the optimum set of frames using MAC parameters, p_i^*, k_i^*.
5) Update λ_i^{k+1} using Equation (8).

At base station:

6) Receive video frames from all sources, forward to the decoder.
7) Find ν^{k+1} and ω^{k+1} using (9) and (10).
8) Broadcast ν^{k+1} and ω^{k+1}, and if flag = 1, $\boldsymbol{\alpha}^m$.

At the decoder:

In the initial phase: Find $\boldsymbol{\alpha}^m$ parameters using method described in Section III-A.

Otherwise:

9) Using the rate vector, $[R_1...R_N]$, calculate the quality vector, $[Q_1(R_1), ...Q_N(R_N)]$.
10) Find m' such that $[Q_1(R_1), ...Q_N(R_N)] \in C_{m'}^N$. If $m' \neq m$, then $m = m'$ and flag $= 1$.
11) Decode the videos, with error concealment method depending on values of $\boldsymbol{\alpha}^m$.

There is always a unique solution for X_i. This can be solved by a simple table look up at each source. Once X_i is found, each station finds its own optimum $\frac{dQ^{(i)}(R_i)}{dR_i}$ using Equation (7). Then, using its corresponding Q-R function, the station finds the rate at which the slope of Q-R plot matches the found $\frac{dQ^{(i)}(R_i)}{dR_i}$, which is its optimum rate, R_i. If a packet ordering algorithm was used to create the Q-R function, the station also simultaneously finds its packet selection. The following equations are finally solved to determine the parameters p_i and k_i:

$$k_i = \frac{R_i}{B.log(X_i)} \tag{12}$$

$$p_i = \frac{(X_i - 1).k_i}{\zeta_i}. \tag{13}$$

This method has low complexity for sources and the base station, i.e. a few operations per iteration. A packet ordering algorithm can be used at a source, with complexity depending on the choice of the user. Alternatively a user can use a known Q-R model as given in [13]. At the decoder only the initial phase has added complexity compared to decoding of independent sources. In the initial phase each frame is decoded twice, and LMS is used to find the correlation parameters. The overall number of iterations for the algorithm convergence is as the sub-gradient method, $1/\epsilon_T^2$, where ϵ_T is the distance to the optimum.

IV. Simulations

A. Simulation Setup

Simulations are performed using Matlab and a modified version of the H.264 reference software [15]. We simulate the performance for videos from two sets of correlated sequences, BreakDancing and BookArrival [16] [17]. In each case a network with one receiver and three transmitting video sources is considered, with two correlated sources, and one source which does not use correlated decoding. We encode each video using H.264, creating a stream of RTP packets. Each video packet consists of exactly one video frame. The GoP size is 5, with one I and 4 P frames. The frame rate is set to 5 fps. For decoding we use the H.264 [15] decoder that has been modified to replace lost frames with either the corresponding frames from a correlated neighbour, or the previous frame in the same sequence, depending on the algorithm input. The wireless channel is Gaussian, with the channel parameters given in Table I. We vary the number of orthogonal codes in order to vary the normalized rate.

TABLE I
Channel Parameters in simulations

Parameter	Values
Max Power per user	10W
Bandwidth	40 kHz
Max total power	20 W
β (SNR gap) for each user	0.9

B. Correlated Video Model Validation

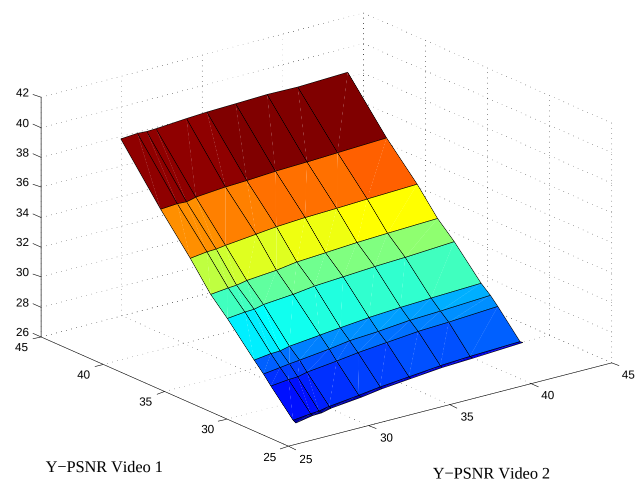


Fig. 2. Breakdancing sequence. Q_1^c vs $Q_1(R_1)$ and $Q_2(R_2)$

To verify the piecewise linear quality model for correlated decoding of Eq (4), we use two sets of video sequences, BreakDancing [17] and Book Arrival sequences [16]. The aim is to construct a plot relating changes in the quality of a video decoded using correlated error concealment, i.e., $Q_1^c(R_1, R_2)$, to qualities of videos that are used in the decoding, $Q_1(R_1)$ and $Q_2(R_2)$, as defined in Section III-A. We remove frames from each video at random. We then decode the sequence using two methods. One, by frame replacement with the previous frame; this gives the values $Q_i(R_i)$. Second, by frame replacement with corresponding frames from the correlated video source, which gives values of $Q_1^c(R_1, R_2)$. Figure 2

TABLE II
MSE VALUES OF THE PIECEWISE LINEAR MODEL.

Sequence	C_m^2 size	MSE
BreakDancing	$2dB \times 2dB$	0.43 dB
Book Arrival	$2dB \times 2dB$	0.88 dB

demonstrate this 3D function for the BreakDancing sequence set, averaged in order to smooth the function. To visualize it better we project the 3D function on both axes in Figures 3 and 4.

We model the found $Q_1^c(R_1, R_2)$ function as a piecewise linear approximation of $Q_1(R_1)$ and $Q_2(R_2)$. We create the C_m^2 spaces By partitioning the possible values of quality vector $[Q_1(R_1), Q_2(R_2)]$ into 2dB by 2dB sections. For each range we use LMS to find the respective model parameter matrix, $\boldsymbol{\alpha}^m$. The MSE of the model is found to be 0.43 dB for video sequence sets of BreakDancing, and 0.88 dB for the Book Arrival sequence sets, as given in Table II.

The gain from using correlated decoding versus regular decoding can be observed in Figure 3. We observe that when $Q_1(R_1)$ is on average 27 dB and the average $Q_2(R_2)$ is 40 dB we get about 1dB gain from using correlated decoding. We should point out that this gain is from decoding only, and does not include gains from the correlated resource allocation.

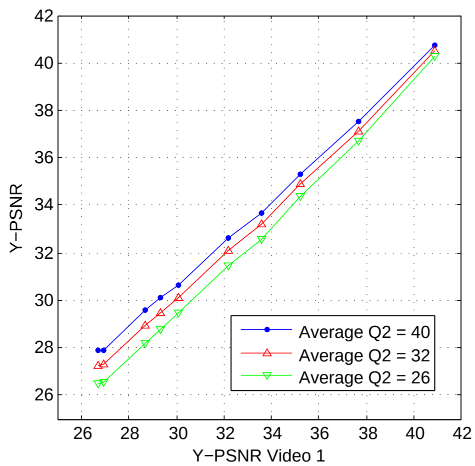


Fig. 3. Breakdancing sequence. Q_1^c vs $Q_1(R_1)$

C. Resource Allocation Performance

We compare the performance of three algorithms. One is our proposed correlated resource allocation scheme described in Algorithm 1. The second also uses our algorithm but the correlation parameter matrix is set to $I_{N\times N}$, which makes it an optimal resource allocation scheme for independent sources. We use this comparison to demonstrate the utility gain solely due to use of correlation. We refer to this algorithm as No-Correlation (NC). The third algorithm is a basic resource allocation, where code and power are divided equally among the nodes, but sources optimally order the frames as in the previous two schemes. We refer to this algorithm as the Basic method.

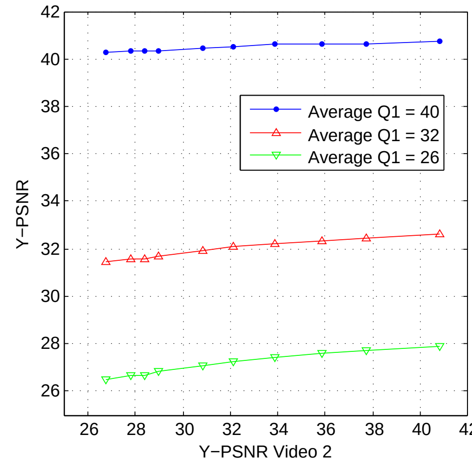


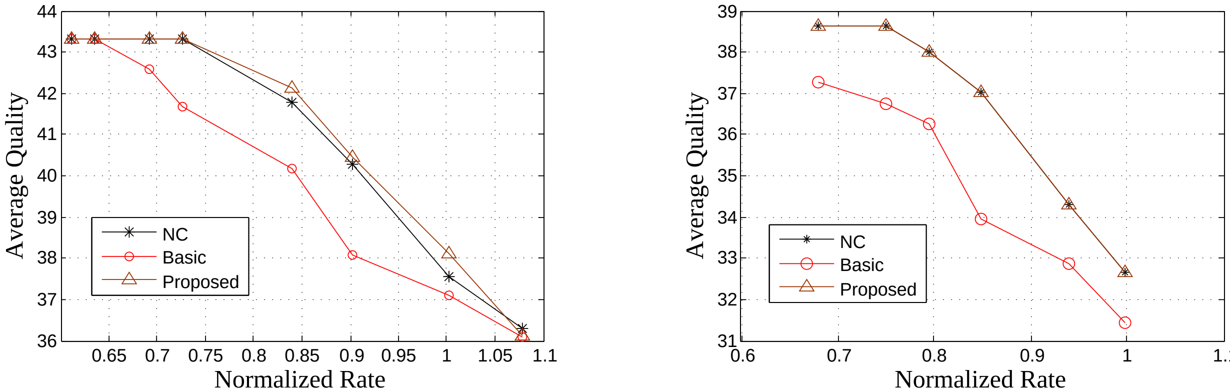
Fig. 4. Breakdancing sequence. Q_1^c vs $Q_2(R_2)$

We plot average quality of videos in terms of Y-PSNR versus the normalized rate. The normalized rate is calculated by dividing the total load by the channel capacity. We consider two cases for channel conditions. In the first case the two correlated sources experience 23 and 17 dBm *Normalized Noise*, as defined in Eq. 1, and the uncorrelated source has normalized noise of of 23 dBm. The resulting performance for each video sequence set is given in Figures 5(a) and 5(b). We observe that the proposed algorithm has up to 0.5 dB higher average Y-PSNR than the NC method for the BreakDancing sequences, and at worst case it performs equally well. For the BookArrival sequence our perposed method has the same performance as the NC.

We then consider the normalized noise of 27 dBm and 17 dBm for the correlated sources and 20 dBm for the independent source. The results are presented in Figures 6(a) and 6(b). In this case for both videos we see gain, up to 1.75 dB in average video quality, in our proposed method compared with the NC method. The basic method performs poorly since the quality found for the user with the worst channel drops to zero when normalized rate is at 0.87 for BreakDancing and at 1 for BookArrival sequence sets.

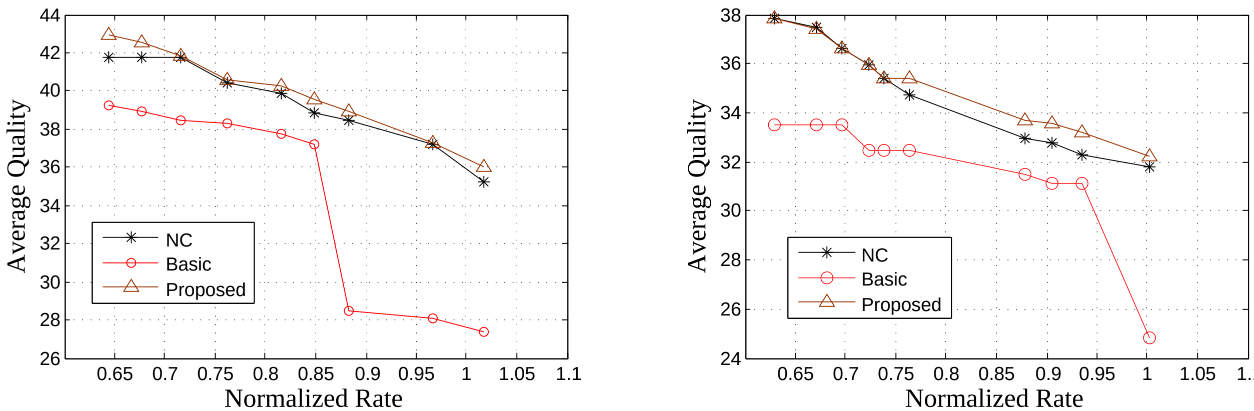
V. CONCLUSION

In this work we propose a method to find the optimum resource allocation for uplink transmission of correlated video sources, such that the total received video quality is maximized. We develop a model for relating the quality of a video decoded using correlation to quality-rate characteristic functions of videos that are used in its decoding. Based on this model, we formulate an optimization problem and develop an algorithm to solve it with little information exchange with the base station. From the simulations we observe that even with a simple correlated error concealment scheme, our proposed resource allocation method results in up to 1.75 dB gain over the optimum resource allocation with independent decoding, which provides a lower-bound on the performance of our



(a) BreakDancing (b) BookArrival

Fig. 5. Average quality for 3 sources, Normalized Noise power (dBm) = [23, 17, 23].



(a) BreakDancing (b) BookArrival

Fig. 6. Average quality for 3 sources, Normalized Noise power (dBm) = [27, 17, 20].

algorithm. The small additional complexity at decoder resides in the estimation of the source correlation parameters, which is generally only performed once in static setting. A more precise correlation model with improved quality estimation that explicitly considers the geometry of the scene is currently under study.

References

[1] R.S. Wang and Y. Wang, "Multiview video sequence analysis, compression, and virtual viewpoint synthesis," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 10, no. 3, pp. 397–410, 04 2000.

[2] Chuohao Yeo and K. Ramchandran, "Robust distributed multiview video compression for wireless camera networks," *Image Processing, IEEE Transactions on*, vol. 19, no. 4, pp. 995 –1008, 2010.

[3] C. Khirallah, V. Stankovic, L. Stankovic, and S. Cheng, "Multiterminal source coding for multiview images under wireless fading channels," *Multimedia Tools and Applications*, vol. 48, pp. 457–470, 07 2010.

[4] S. Shakkottai, T.S. Rappaport, and P.C. Karlsson, "Cross-layer design for wireless networks," *Communications Magazine, IEEE*, vol. 41, no. 10, pp. 74–80, oct. 2003.

[5] M. van Der Schaar and N Sai Shankar, "Cross-layer wireless multimedia transmission: challenges, principles, and new paradigms," *Wireless Communications, IEEE*, vol. 12, no. 4, pp. 50–58, aug. 2005.

[6] V.G. Subramanian, R.A. Berry, and R. Agrawal, "Joint scheduling and resource allocation in cdma systems," *Information Theory, IEEE Transactions on*, vol. 56, no. 5, pp. 2416–2432, may 2010.

[7] Guocong Song and Ye Li, "Cross-layer optimization for ofdm wireless networks-part i: theoretical framework," *Wireless Communications, IEEE Transactions on*, vol. 4, no. 2, pp. 614–624, march 2005.

[8] J. Huang, V.G. Subramanian, R. Berry, and R. Agrawal, "Joint scheduling and resource allocation in uplink ofdm systems," 2007, pp. 265–269.

[9] Yi Su and M. der van Schaar, "Multiuser multimedia resource allocation over multicarrier wireless networks," *Signal Processing, IEEE Transactions on*, vol. 56, no. 5, pp. 2102–2116, may 2008.

[10] J. Chakareski, "Distributed media cooperation for enhanced video communication," *Journal of Zhejiang University - Science A*, vol. 7, pp. 773 – 783, 2006.

[11] Andrea Goldsmith, *Wireless Communications*, Cambridge University Press, New York, NY, USA, 2005.

[12] J. Chakareski and P. Frossard, "Distributed packet scheduling of multiple video streams over shared communication resources," in *Multimedia Signal Processing, 2005 IEEE 7th Workshop on*, oct. 2005, pp. 1–4.

[13] Klaus Stuhlmuller, Niko Farber, Michael Link, and Bernd Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1012–1032, 2000.

[14] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge Univ Press, Cambridge, U.K., 2004.

[15] "H.264 reference software, version 16.2," http://iphome.hhi.de/suehring/tml/.

[16] I. Feldmann and M. Mueller et. al, "Hhi test material for 3d video," ISO/IEC JTC 1/SC 29/WG 11, Doc. N15413, april 2008.

[17] C. L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," New York, 2004, SIGGRAPH, pp. 600–608.