

Fair Rate Allocation of Scalable Multiple Description Video for Many Clients

Jacco R. Taal and R. (Inald) L. Legendijk
Information & Communication Theory Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, The Netherlands

ABSTRACT

Peer-to-peer networks (P2P) form a distributed communication infrastructure that is particularly well matched to video streaming using multiple description coding. We form M descriptions using MDC-FEC building on a scalable version of the “Dirac” video coder. The M descriptions are streamed via M different application layer multicast (ALM) trees embedded in the P2P network. Client nodes (peers in the network) receive a number of descriptions $m < M$ that is dependent on their bandwidth. In this paper we consider the optimization of the received video qualities, taking into account the distribution of the clients’ bandwidth. We propose three “fairness” criteria to define the criterion to be optimized. Numerical results illustrate the effects of the different fairness criteria and client bandwidth distributions on the rates allocated to the compressed video layers and multiple descriptions.

Keywords: multiple description coding, scalable coding, erasure codes, peer-to-peer networks, error resilience, path diversity.

1. INTRODUCTION

Peer-to-Peer (P2P) networks and their file swapping P2P applications have become popularized in the past years because of their high download bandwidths and inherent server off-loading. This success has stimulated research into using peer-to-peer networks as infrastructures for streaming video over the Internet.^{1,2} In addition to being clients, the peers in the network then also serve as *application-level* multicast nodes. Although IP-level multicast offers efficient distribution from server to clients using routers, this mechanism is not widely spread nor often used. For application-level multicast (ALM), however, no infrastructural changes are required, making fast and flexible deployment possible. Furthermore, the way the multicast is carried out can be specifically tailored to video streaming applications.

If a P2P network implements ALM, we can use intermediate nodes in the network to forward data to other nodes. The data forwarding model that we consider to be most suitable for video streaming over peer-to-peer networks is known as “bartering”. Chunks of data are exchanged between nodes in a “as fair as possible” manner. Bartering has two advantages. First, the clients exchange the data without server intervention, thus off-loading the server. Second, the clients are encouraged to participate in sharing the downloaded data with other peers, which effectively combats the freeriding problem.³ The underlying P2P network takes care of downloading all missing chunks from other nodes, and forwarding available chunks of data to nodes that still miss that data.

To apply the bartering model to streaming video, two possibilities exist. The first is to chop up the compressed video stream into data packets, and assign these packets to different network chunks. For example, chunk 1 contains video data packet 1, 3, 5, . . . , and chunk 2 contains video data packets 2, 4, 6, An important drawback of this approach is that since all packets are crucial for proper video decoding, missing chunks will result in major video degradation.

The second – and much more attractive – approach is to apply bartering on multiple descriptions (MD) of the (compressed) video source rather than straightforwardly chopping up a compressed bit stream. In this case, a video encoder generates several (multiple description) streams, that are all independently decodable. The different descriptions are put into the different chunks. If more chunks, and hence more descriptions, are received, the decoded video quality improves. The advantages of the MD approach is that video streaming becomes robust against P2P bandwidth variations or failing P2P nodes, which in both cases cause random chunks of data to be unavailable to a video decoder.

Send correspondence to J.R.Taal@ewi.tudelft.nl. This work has been funded by the Freeband projects CACTUS (www.cactus.tudelft.nl) and I-SHARE (www.freeband.nl)

In this paper we use the multiple description coding approach MDC-FEC proposed by Puri and Ramchandran⁴ (see Figure 1). First, the video is encoded using an M -layer video encoder. Each layer k is encoded at a rate R_k and is then protected by an (M, k) (Reed-Solomon) erasure code. The thus obtained protected data is evenly distributed over the M descriptions. Finally, the individual descriptions form the chunks that are bartered by the P2P nodes. If any m out of M descriptions are available, the first m layers can successfully be decoded, resulting in a quality Q_m .

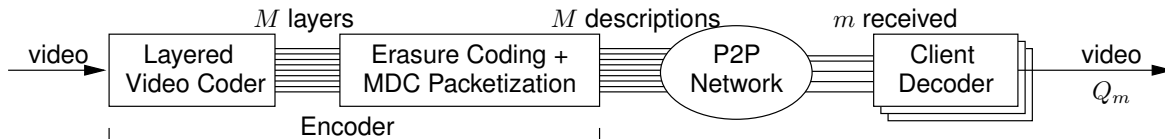


Figure 1. Block Diagram of the multiple description video distribution over a P2P network

In the practical application of the above MDC-FEC approach, several (encoding) parameters have to be chosen, namely the number of layers (and descriptions) and the bit rate at which the individual layers are encoded. These choices depend strongly on the quality that is desirable and achievable for clients with different bandwidth. In our earlier work⁵ we have proposed to use a probabilistic client-bandwidth distribution model that describes clients bandwidth due to physical network limitations (e.g. the various ISDN, cable, ADSL, and LAN connections) and due to network congestion. In this way, the probability for a P2P peer receiving a certain number of descriptions can be calculated as a function of the above mentioned parameters. Using an overall criterion, such as the average compression distortion over all clients, the optimal number of layers/descriptions and the bit rate per layer can be found.

In the next section we will discuss related work. We present our approach to MDC video streaming in P2P network in Section 3, and derive a model for the behavior of our MDC-FEC system. Also we derive an RD-model for the layered Dirac codec. In Section 4 we first present three different criteria to find the optimal encoding parameters. We will introduce these criteria from the perspective of “fairness” to the clients. After all, multiple description coding is inherently less efficient than single description coding, hence some clients “pay” bandwidth or quality for the benefit of others. For instance, rather counter-intuitive, a criterion that minimizes the average MSE distortion results in mainly minimizing the distortion for clients with low bandwidth, whereas clients with high bandwidth gain little. We will also discuss the optimization of the encoding parameters for the selected fairness criteria. Finally, in Section 5, we show results of optimizing the fairness criteria with specific settings and assumptions on the client distribution. We conclude our paper with a discussion.

2. RELATED WORK

2.1. Peer-to-Peer Networks

A peer-to-peer network consists of a subset of all nodes present on a network (Internet). Just by having a connection to one or multiple nodes that are part of the P2P network, these nodes are (virtually) connected to the whole P2P network. The actual IP connection to these nodes may go through other nodes or routers that themselves are not part of the P2P network. Therefore the term *overlay network* is also often used in place of P2P network. Often there is no central registration site or server which controls or tracks the peers connected to the P2P network. This means that all functionalities of the P2P network have to be implemented in a distributed fashion. A commonly observed property of P2P networks is that the larger the P2P network, the more efficient, or the higher the performance. This is contrary to the single server solutions where the number of clients that can be handled, is often bounded.

A commonly used data exchange technique in P2P networks is “bartering”. Chunks of data are bartered (traded) amongst peers. This greatly reduces the fan-out of a server. BitTorrent,⁶ a system that supports a large number of parallel file downloads, splits up the file into a number of chunks. Clients that already have downloaded certain chunks, forward these chunks to other clients, such that the server is relieved. Furthermore, the clients that are forwarding chunks are allowed to download more chunks simultaneously, resulting in a higher download speed. This tit-for-tat approach is good for the P2P network as a whole. Since peers are encouraged to forward chunks for other clients, this greatly simplifies control of the network.

2.2. Multiple Description Coding

In multiple description coding (MDC) a certain amount of redundancy is added to transmitted (compressed) data, such that when one or more descriptions are lost, we are still able to recover the source data with an acceptable amount of distortion. Where error-correcting codes are typically used to correct bit and burst errors, MDC is used to handle situations in which losing entire packets or descriptions is likely.

There is a substantial body of literature on MDC of images. The MDC approaches can be divided into four categories, namely:

- MDC-Q, in which nested quantizers or lattice vector quantizers are used to generate the multiple description. If one description is lost, the decoder effectively uses a coarser reconstruction,⁷
- MDC-TC, in which uncorrelated signal components are transformed using a correlating transform, introducing a controlled amount of redundancy. If a correlated component is lost, the remaining component(s) can be used to recover the source data with an acceptable distortion,⁸
- MDC-SB. Source-based MDC: The autocorrelation naturally present in the source data is exploited to generate correlated descriptions,⁹
- MDC-FEC. This is combination of layered (or progressive) encoding and erasure coding. First a number of layers are generated. Then erasure codes of different strength are applied to each layer, such that, if m out of M descriptions is received, m layers can be successfully decoded.¹⁰

The MD encoding of video is more complicated than MD encoding of random i.i.d. dat. This is due to the complex autocorrelation structure of video, but also to the non-Gaussian distribution of the data. Furthermore, special care needs to be taken to prevent accumulation of errors in inter-frame decoding. An example of a MDC-SB like video encoder is the following.¹¹ The encoder generates an “even” stream, containing even video frames 0, 2, 4, . . . and an independently decodable “odd” stream containing all odd video frames. If a description is missing, it can be estimated from the other description(s) because of the (inherent) correlation between successive video frames. A clear disadvantage of this (and any MDC) approach is the less efficient compression as correlation between odd and even frames is not exploited.

2.3. Application Layer Multicast (ALM)

The combination of multiple description coding and application layer multicast has been addressed by Castro *et al.*² Their *Splitstream* system splits the video stream into two descriptions, similar to the above presented example where one description contains the even frames and the other description contains the odd frames. Splitstream constructs two different multicast trees with the same server and containing the same clients, but with different routes. When one of the nodes temporarily fails, each sub-trees is deprived of one description, but since the two subtrees are based on disjoint paths, it is very likely that the other description can still be received. Other proposals^{12–15} also describe streaming of media P2P networks, but these works concentrate mainly on P2P tree formation and handling of joins and leaves.

3. MDC VIDEO STREAMING IN PEER-TO-PEER NETWORKS

3.1. Peer-to-Peer TV Architecture

In our previous work³ we presented the basic design of a peer-to-peer television (P2P-TV) system. In P2P-TV, a separate P2P layer takes care of peer-finding, building multicast trees, handling joins and departures of nodes and handling (temporarily) failing peers. Furthermore, the P2P layer measures bandwidth latencies, peer failures etc, such that the video always can be efficiently distributed through the multicast tree.

The P2P video distribution system within P2P-TV supports any peer to be the server of a video stream. At the same time, any peer in the network can subscribe to the stream. Figure 2 depicts an example of such a P2P network. The Server S is part of the P2P network and produces several descriptions (D_1, \dots, D_N) of the video data. Client C_i receives these through peers $P_A, \dots, P_Z, P_1, \dots, P_N$. Another client C_j also receives these descriptions from another set of peers. In this example C_i forwards one or more descriptions to C_j . This general model enables simple packet forwarding and bartering to distribute the video descriptions through the P2P network to all clients. We assume that the total of incoming

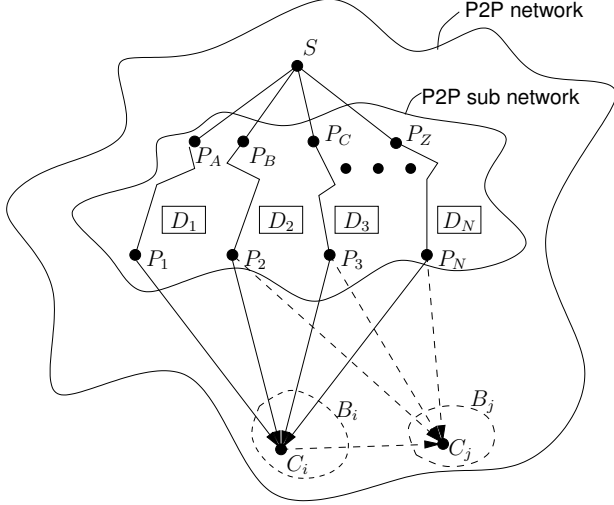


Figure 2. Example of an Overlay P2P network.

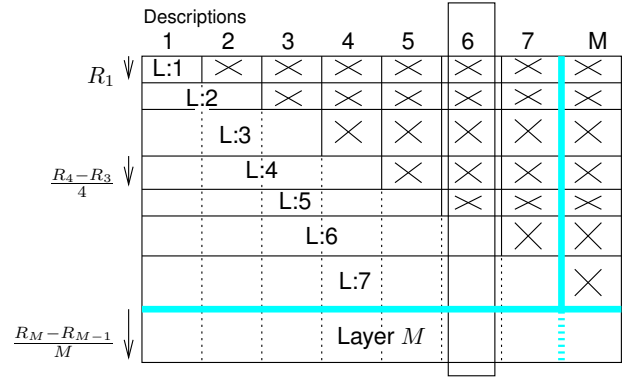


Figure 3. MDC-FEC Packetization scheme. Each layer k is protected by an (M, k) erasure code (horizontal direction). The parts marked with a cross are parity codes. Each column is one description (packet)

and outgoing bandwidth for each node C_i is limited to B_i . Usually, B_i forms the bandwidth bottleneck, because the rest of the network is normally able to support a total flow of B_i to node C_i . Since the total flow is split up in separate description, following different paths through the network, this is often a valid assumption. We also assume that the total bandwidth for server S , B_S is sufficient to distribute at least one copy of each description.

3.2. Layered MDC Video Encoder

As we mentioned in the introduction of this paper, we use the Multiple Description Coding approach proposed by Puri and Ramchandran⁴ for creating the descriptions that are streamed over the application layer multicast trees in the P2P network. We prefer their MDC-FEC approach over other MDC approaches for two reasons. First, this MDC approach is flexible in the number of descriptions to generate. If an efficient layered encoder is present, any number of descriptions can be generated. Second, the behavior of the MDC-FEC system is easy to model if the rate-distortion performance of the layered encoder are known. Hence, end-to-end rate-distortion optimization of the video streaming systems becomes possible.

The video coder we use generates an inter-coded base layer. The $(M - 1)$ enhancement layers do not employ temporally predictive coding. On one hand this overcomes the effect of accumulating errors, since missing enhancement data has no effects on the next frame. On the other hand, we lose the efficiency of temporal predictive coding for the enhancement layers.

For decoding layer l , first all lower layers $1, \dots, l - 1$ should be decoded successfully. The layered coder is followed by a packetizer, as depicted in Figure 1. We generate multiple descriptions by combining erasure codes of each layer into packets. This way we ensure that when a client receives only a subset of the descriptions, it can still successfully decode a number of layers.

In general, an (M, k) block-code is able to correct $\frac{M-k}{2}$ random errors and $M - k$ erasures of which the positions are known. As shown in Figure 3, each description contains information of all layers using erasure codes. The data of layer l is first split up in l equal sized data blocks and then the remaining $M - l$ blocks are filled with the respective erasure codes. Any number l of these blocks is then sufficient to recover the original l data blocks of this layer. As a special, case layer 1 is in fact just copied in all descriptions. The last layer is only split up such that all descriptions are needed to reassemble that layer. The total number of layers M is equal to the number of descriptions. We have control over the redundancy added by erasure codes, since the rate of each layer is still free to choose. For instance, by using a small base layer rate and a large enhancement layer rate, we can induce a low redundancy.

The base layer is encoded at a rate R_1 . The total rate of the first n layers is denoted by R_n . Hence, each successive layer i

has a rate $R_i - R_{i-1}$. The rate R_D of each of the M descriptions is then given by:

$$R_D = \sum_{l=1}^M \frac{R_l - R_{l-1}}{l} \quad (1)$$

$$R_D = \sum_{l=1}^M \alpha_l R_l, \quad (2)$$

where $R_0 = 0$, and

$$\alpha_l = \frac{1}{l(l+1)} \quad \text{for } l = 1, 2, M-1, \text{ and } \alpha_M = 1/M. \quad (3)$$

A node or client in the P2P network with bandwidth R_c receives $m = \lfloor \frac{R_c}{R_D} \rfloor$ descriptions. After decoding these m descriptions, the client experiences a compression distortion of $D(R_1, R_2, \dots, R_m)$ for $m = 1, \dots, M$. In principle, the rate-distortion function for layer m depends on *all* rates R_k of the individual layers $k = 1, \dots, m$ as layered compression induces losses relative to a single layer coder that are dependent on the rates of the (lower) layers. Hence, for each layer we may have a different rates-distortion function. By definition, $D(0) = \sigma_x^2$.

3.3. Layered Dirac Video Encoder

In our work we used *Dirac*, an open-source video coder developed by the BBC.¹⁶ *Dirac* is wavelet-based and its performance is comparable to H.264. In Figure 4 the bit rate - distortion (RD) curve of *Dirac* is shown for the Foreman sequence. We also plotted several RD-curves for two-layer encoding. The branch-point in each curve is the rate R_1 for the first layer. The efficiency loss for the first enhancement layer is clearly visible. For higher layers (not shown) we have observed an insignificant loss of coding efficiency. Consequently, the rate distortion function becomes only dependent on the rate of the base layer R_1 and the rate of the layer m under consideration, i.e., $D(R_1, R_2, \dots, R_m) = D(R_1, R_m)$.

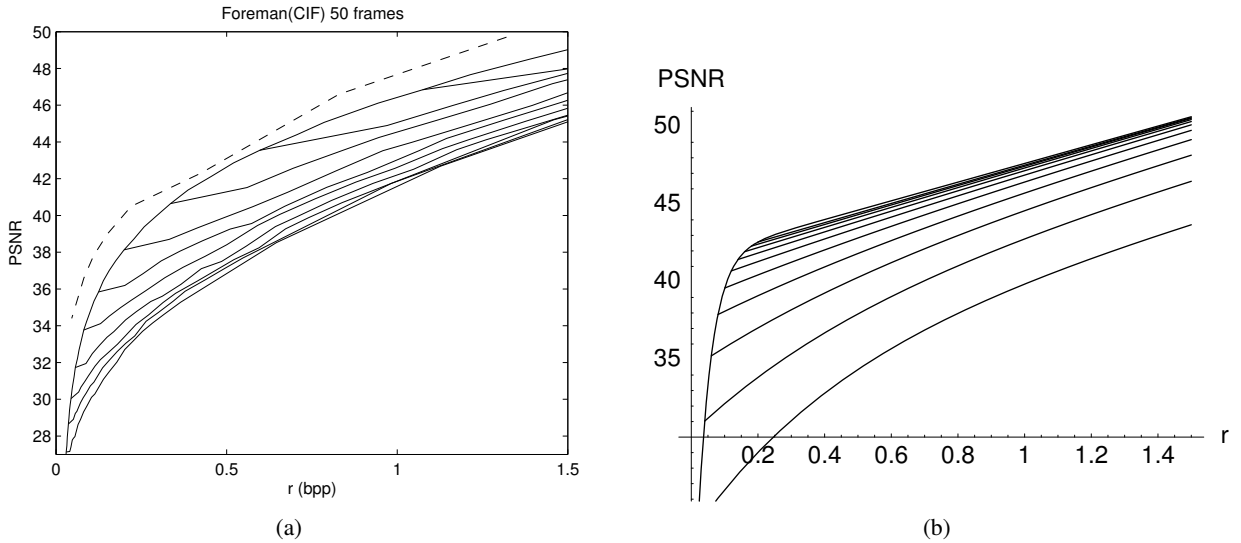


Figure 4. (a) Rate-Distortion curves for the layered *Dirac* encoder with the 30Hz CIF-Format Foreman sequence. The dashed line is for the H.264 encoder. (b) Rate-Distortion model

In order to be able to find an optimal rate allocation, we need to have an analytic model of the RD curves. First we model the single layer curve.

$$D(R) = \sigma_x^2 2^{-2(R+a_0(1-2^{-b_0 R}))}. \quad (4)$$

This model embeds the information-theoretical bound for Gaussian i.i.d. sources ($a_0 = 0$), but also models the enhanced coding efficiency for autocorrelated sources such as video. For $R \rightarrow \infty$, the slope of the curve becomes the well-known

6dB per bit. For smaller R , the curve has a larger slope, which slowly decays to the 6dB bound. The top curve in Figure 4(b) shows the resulting model.

We see in Figure 4(a), that the slope of the two-layer curves $D(R_1, R_2)$ is smaller than for a single layer curve $D(R_1)$. By letting model parameters a and b depend on R_1 , we can model the RD behavior for any combination of R_1 and R_2 . Since we observed no additional loss in efficiency for layers 3,4,..., the model for the m -layer encoder is based on the 2-layer encoder:

$$D(R_1, R_2 \dots, R_m) = D(R_1, R_m) = \sigma_x^2 D(R_1) 2^{-2((R_m - R_1) + a(R_1)(1 - 2^{-b(R_1)(R_m - R_1)}))} \quad \text{for } m \leq M. \quad (5)$$

Functions $a(R_1)$ and $b(R_1)$ are fit to the experimentally observed rate-distortion behavior of the coder.

4. FAIR MDC STREAMING

4.1. Rate Allocation Problem

If we have to serve a large number of different clients, all with a different bandwidth R_c , we have to trade-off quality and redundancy of the descriptions. Making an optimal trade-off is not trivial as clients having different bandwidth will receive a different number of descriptions, and hence experience a different quality and effective bandwidth. On one hand, we wish to offer every client a quality that is as high as possible. On the other hand because of scalability of the system, we cannot offer each client individually an optimal stream. Furthermore, clients have to accept the fact that they have to forward packets to other peers, i.e. that they have to donate bandwidth to the P2P network, especially when we introduce redundancy by using MDC.

In order to deal with this dilemma in a fair way, two ingredients are needed. In the first place we need to know (or model) the distribution of the bandwidths R_c available to the clients. The distribution is modeled by the probability density function (PDF) $f_{R_c}(R_c)$. Secondly, we need to establish a criterion that expresses what we mean by ‘‘fair’’, and which also lends itself for optimization. The fairness criterion will generally be a function of the clients bandwidth R_c and the bit rates R_k allocated to the layers of the video encoder. The fairness criterion is therefore denoted by $FC(R_c; R_1, R_2 \dots, R_M)$. An in-depth discussion on the somewhat socio-economical question of how to distribute resources over a heterogeneous population is, however, outside the scope of this paper. In the following subsections we will introduce and discuss different examples of fairness criteria.

Given the distribution of the clients bandwidth and the fairness criterion, the optimal compression parameters M, R_1, \dots, R_M can then be found by maximizing

$$\{M, R_1, R_2 \dots, R_M\} = \arg \max_{M, R_1, R_2 \dots, R_M} \int_0^\infty f_{R_c}(r) FC(r; R_1, R_2 \dots, R_M) dr \quad (6)$$

In most cases the rate allocation problem (finding optimal values for $R_1, R_2 \dots, R_M$ given R_D and M) can only be solved numerically. The values of R_D and M , however, can either be optimized numerically but often are chosen practically. The P2P network, for instance, may only support a limited number of descriptions. Or, if we observe that most of the client bandwidths are smaller B_{\max} , we could choose to select $R_D = B_{\max}/(M + 1)$ so that the client-bandwidth spectrum is equally divided. In the remainder of this paper we only discuss optimizing the rates $R_1, R_2 \dots, R_M$ and we assume that we either have already selected values for M and R_D , or that these values are to be found in an outer optimization loop.

When R_D and M are fixed, the maximization function becomes:

$$\{R_1, R_2 \dots, R_M\} = \arg \max_{R_1, R_2 \dots, R_M} \sum_{j=0}^M \int_{b_j}^{b_{j+1}} f_{R_c}(r) FC(r; R_1, R_2 \dots, R_j) dr \quad (7)$$

where,

$$\begin{aligned} b_i &= i R_D \quad \text{for } i = 0, 1, \dots, M \\ b_{M+1} &= \infty. \end{aligned}$$

4.2. Minimal MSE Fairness Criterion

The first – and the most straightforward – criterion we consider is to average the distortion $D(R_1, R_2, \dots, R_m) = D(R_1, R_m)$ over all clients:

$$\hat{D}(R_1, R_2, \dots, R_M) = \sum_{j=0}^M \int_{b_j}^{b_{j+1}} f_{R_c}(r) D(R_1, R_j) dr = \sum_{j=0}^M C_j D(R_1, R_j), \quad (8)$$

where the number of clients receiving $i + 1$ out of M descriptions is C_i , computed as:

$$C_i = \int_{b_i}^{b_{i+1}} f_{R_c}(r) dr \quad \text{for } i = 0, 1, \dots, M. \quad (9)$$

Given a fixed number of description M and a fixed rate per description R_D , this criterion can be solved numerically using the Lagrange multiplier method, as discussed in the work of Puri *et al.*⁴

$$L(R_1, R_2, \dots, R_M, \lambda) = \sum_{j=0}^M C_j D(R_1, R_j) + \lambda \left(\sum_{j=1}^M \alpha_j R_j - R_D \right) \quad (10)$$

After equating the partial derivatives to zero, we obtain a set of equations for which the roots can be found numerically:

$$\frac{1}{\alpha_1} \sum_{j=1}^M C_j \frac{\partial D(R_1, R_j)}{\partial R_1} + \lambda = 0 \quad (11)$$

$$\frac{C_i}{\alpha_i} \frac{\partial D(R_1, R_i)}{\partial R_i} + \lambda = 0 \quad \text{for } i = 2, \dots, M \quad (12)$$

$$\sum_{l=1}^M \alpha_l R_l - R_D = 0 \quad (13)$$

In Ref. 4, a method is presented to solve these simultaneous equations such that after optimization $R_i < R_{i+1}$ holds for all rates. Unfortunately, the required conditions can only be verified for a single layer coder that is described by a single RD-function. In our case, however, we have to deal with a layered coder that is described by multiple RD-functions. Consequently, the derivative of $D(R_1, R_2, \dots, R_m)$ with respect to R_i also depends on R_j , $1 \leq j < i$. For that reason, we cannot use method in Ref. 4, and as a result we cannot guarantee that after optimization all $R_i < R_{i+1}$. Currently we deal with monotonicity of R_i as a postprocessing step *after* optimization. The resulting optimization procedure is illustrated in Figure 5.

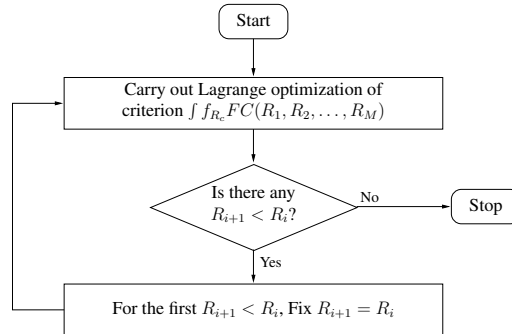


Figure 5. Optimization Algorithm which ensures an monotonic increasing sequence R_i

4.3. Maximal Average PSNR Fairness Criterion

A drawback of the MSE metric described above is that the (inverse) magnitude of the MSE is not a good measure of quality. When the *average* MSE is minimized, most effort is put in minimizing the highest MSE values. Little effort is put in minimizing the lower MSE values, although these still can gain significant amounts of quality. An obvious alternative choice for the fairness criterion is to use the performance measure often used in video compression, namely peak-SNR. When we average the PSNR over all clients, the following criterion is obtained:

$$\widehat{\text{PSNR}}(R_1, R_2, \dots, R_M) = \sum_{j=0}^M \int_{b_j}^{b_{j+1}} f_{R_c}(r) \text{PSNR}(R_1, R_j) dr \quad (14)$$

where $\text{PSNR}(R_1, R_j) = 10 \log_{10} \frac{255^2}{D(R_1, R_j)}$. We can optimize the above criterion in a similar way as the minimum MSE criterion. The Lagrangian function is then given by

$$L(R_1, R_2, \dots, R_M, \lambda) = \sum_{j=0}^M C_j \text{PSNR}(R_1, R_j) + \lambda \left(\sum_{j=0}^M \alpha_j R_j - R_D \right). \quad (15)$$

After partial differentiation to R_i we obtain the following set of equations:

$$\frac{1}{\alpha_1} \sum_{j=1}^M C_j \frac{1}{D(R_1, R_j)} \frac{\partial D(R_1, R_j)}{\partial R_1} + \lambda = 0 \quad (16)$$

$$\frac{C_i}{\alpha_i} \frac{1}{D(R_1, R_i)} \frac{\partial D(R_1, R_i)}{\partial R_i} + \lambda = 0 \quad \text{for } i = 2, \dots, M \quad (17)$$

$$\sum_{l=1}^M \alpha_l R_l - R_D = 0 \quad (18)$$

The procedure outlined in Figure 5 can also be used for solving the above set of equations.

4.4. Weighted PSNR Loss (WPL- p) Fairness Criterion

From the client's point of view, the client is paying for a certain bandwidth R_c and wishes to use this bandwidth as efficient as possible, in particular, the client wishes to obtain maximal quality of the received compressed video stream. Unfortunately, because of the multiple description coding, there is an inherent reduction in quality. As long as there is a balance between the MDC-induced quality reduction and the benefit for the entire P2P video distribution system (of which the client is an integral part), the client is willing to take part in the system.

The third proposed fairness criterion reflects the MDC-induced quality reduction for the individual clients. We consider the system to be fair when the (psnr) quality of all clients is close to the performance obtained *if* a single description coding system (SDC) had been used. The following *weighted PSNR loss* (WPL) criterion measures the difference between MDC and SDC performance. The p -parameter controls the way in which differences are weighted:

$$\text{WPL}_p(R_1, R_2, \dots, R_M) = \left| \text{PSNR}_{\text{SDC}}(R_c) - \text{PSNR}_{\text{MDC}}(R_1, R_2, \dots, R_{\lfloor \frac{R_c}{R_D} \rfloor}) \right|^p$$

After substitution of this criterion in Eq. (7) and simplification of the resulting expression, we obtain:

$$\widehat{\text{WPL}}_p(R_1, R_2, \dots, R_M) = \sum_{j=0}^M \int_{b_j}^{b_{j+1}} f_{R_c}(r) \left| 10 \log_{10} \frac{D(r)}{D(R_1, R_j)} \right|^p dr. \quad (19)$$

With $p = 1$, this criterion becomes identical to the maximal average PSNR criterion Eq. (14). For $p > 1$ we put more emphasis on larger quality reductions. Minimizing the criterion then results in a solution where the difference between

MDC case and the SDC case is more balanced over all clients. Note that for $p \rightarrow \infty$, we effectively minimize the maximum quality reduction.

The Lagrangian for the weighted PSNR loss with parameter p (WPL- p) criterion is

$$L(\lambda, R_1, R_2, \dots, R_M) = \sum_{j=0}^M \int_{b_j}^{b_{j+1}} f_{R_c}(r) \left| 10 \log_{10} \frac{D(r)}{D(R_1, R_j)} \right|^p dr + \lambda \left(\sum_{j=1}^M \alpha_j R_j - R_D \right), \quad (20)$$

resulting in the following set of equations:

$$\frac{1}{\alpha_1} \sum_{j=1}^M \frac{-p \left(\int_{b_j}^{b_{j+1}} f_{R_c}(r) \log \frac{D(r)}{D(R_1, R_j)}^{p-1} dr \right)}{D(R_1, R_j)} \frac{\partial D(R_1, R_j)}{\partial R_1} + \lambda = 0 \quad (21)$$

$$\frac{1}{\alpha_i} \frac{-p \left(\int_{b_i}^{b_{i+1}} f_{R_c}(r) \log \frac{D(r)}{D(R_1, R_i)}^{p-1} dr \right)}{D(R_1, R_i)} \frac{\partial D(R_1, R_i)}{\partial R_i} + \lambda = 0 \quad \text{for } i = 2, \dots, M \quad (22)$$

$$\sum_{l=1}^M \alpha_l R_l - R_D = 0. \quad (23)$$

Similar to the other two criteria, the procedure outlined in Figure 5 is used for solving the above set of equations.

5. RATE OPTIMIZATION RESULTS

In this section we present the results of the optimization of above criteria given several different client-bandwidth distributions and different choices for R_D and M . In these simulations we used the general RD model as derived in Section 3.3.

The first distribution we consider is shown in Figure 6(a). We expressed the clients bandwidths in bit-per-pixel, also note that $f_{R_c}(r > 1.5) = 0$. Furthermore we fix $M = 4$ and $R_D = 0.3$. Using the MSE criterion, this resulted in rate assignment $\mathbf{R} = \{R_1, R_2, R_3, R_4\} = \{0.15, 0.15, 0.18, 0.74\}$. The resulting quality depending on the number of descriptions that is received (and therefore depending on the client bandwidth) is shown in Figure 6(b). This plot shows MDC quality depending on the number of received descriptions in the lower curve. For reference we also plotted the SDC quality $D(r)$ (upper curve). A first observation is that the first layer already obtains a fairly high quality. This has two reasons: a) the RD-function is very steep at lower rates, making it “cheap” to obtain low distortions at low rates; b) since the RD-model correctly models the penalty in coding efficiency in the enhancement layers, a small base layer implies lower qualities for the higher layers as well. Another observation is that only for clients with bandwidth $R_c > 1.2$, the system really increases the base quality.

When we use the PSNR criterion, the optimal assignment is $\mathbf{R} = \{0.14, 0.14, 0.14, 0.78\}$ (Fig. 6(c)). Although barely noticeable, this criterion favors the higher bandwidths with a little bit higher quality. When we apply the WPL-4 criterion however, $\mathbf{R} = \{0.15, 0.15, 0.32, 0.69\}$, this results in a more “fair” distribution of the quality over all clients. Clients receiving three descriptions do obtain a little higher quality than when receiving only two descriptions, making it worthwhile to participate in de P2P network.

Now we consider the distribution in Figure 6(e). This distribution puts emphasis on clients having bandwidths between 0.3 and 0.6. Since $R_D = 0.3$, all clients with $R_c < 0.3$ receive no descriptions at all. For the MSE criterion ($\mathbf{R} = \{0.3, 0.3, 0.3, 0.3\}$) gives a high quality to all clients, but effectively results in a single-layer, single-description solution. With the WPL-4 criterion on the other hand, $\mathbf{R} = \{0.20, 0.31, 0.45, 0.45\}$ and even clients with a $R_c > 0.6$ still gain quality at a small expense of quality for the lower bandwidth clients (Fig. 6(g)).

For the distribution shown in Figure 6(h), we fixed $M = 6$ and $R_D = 0.2$. To gain insight in the effect of the p -parameter, we varied p from 1 (effectively the PSNR criterion) to $p = 16$. Figures 6(i)-6(l) display the results. Remember that a higher p -value puts more emphasis on minimizing larger offsets between the SDC and MDC quality. Using a large p , the criterion goes to a greater extent to prevent large offsets, which results in a system that really offers different qualities for different bandwidths.

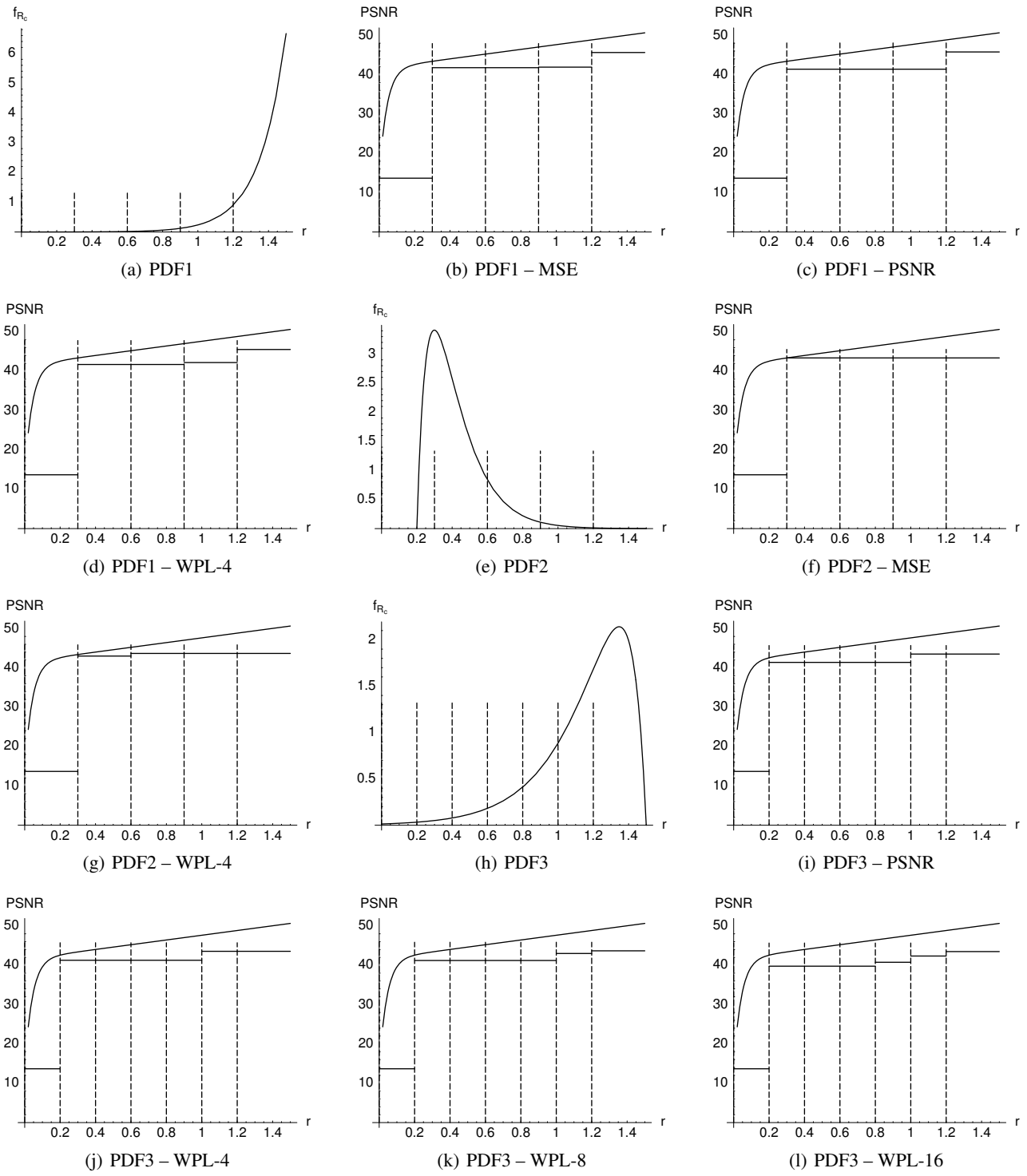


Figure 6. Results for the optimization of the MSE, PSNR and WPL- p criteria for several client distributions (PDF1,2,...)

5.1. Streaming experiment

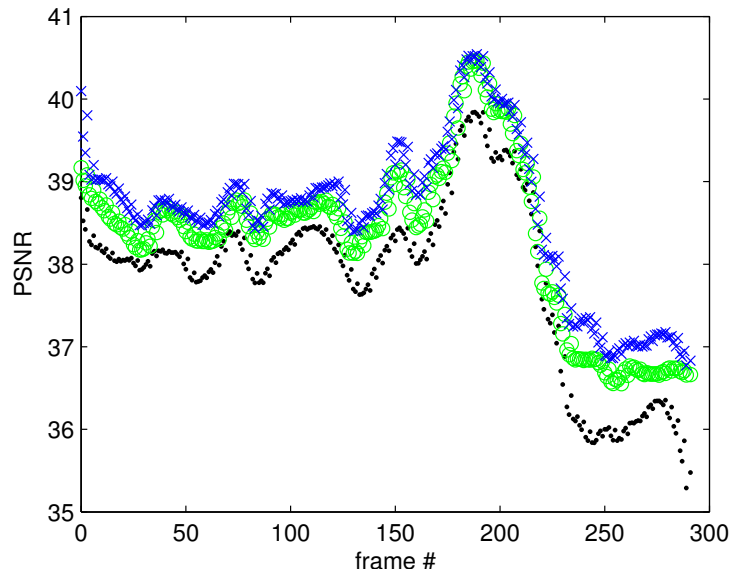


Figure 7. Video streaming experiment with $\mathbf{R} = \{0.20, 0.31, 0.45, 0.45\}$. Each curve shows the obtained PSNR per frame, when one (\cdot), two (\circ), three or four (\times) descriptions are received.

We have performed a streaming experiment on the short Foreman sequence. For distribution PDF1 and fairness criterion WPL-4 we obtained after optimization the rate-setting $\mathbf{R} = \{0.20, 0.31, 0.45, 0.45\}$. We encoded the sequence with our layered-Dirac coder and obtained the results as plotted in Figure 7. Note that our implementation does not employ an inner rate-control loop, hence the greatly varying PSNR quality of different frames. On average however, we clearly obtain different quality levels as requested by our rate setting.

6. CONCLUSIONS

In this paper we have described a multiple description video compression technique that lends itself well for streaming video over P2P networks. The MDC coder is based on a layered video coder in combination with forward error correction. We considered the optimization of the bit rates at which the layers of the video encoder should work, to obtain an optimal and fair performance. We have introduced three fairness criteria, which in combination with the distributions of the P2P clients' bandwidth distribution, define the criterion to be optimized. The rate-distortion functions on which the results are based are taken from a layered version of the Dirac video coder. The results we have obtained show that by choosing an appropriate fairness criterion we can obtain a fair mapping between the bandwidth of a client and the quality it will receive. This is very important in a P2P network, where clients will decide on the basis of this mapping, whether they will participate in the P2P streaming network, and how many descriptions they want to receive.

REFERENCES

1. S. Sheu, K. A. Hua, and W. Tavanapong, "Chaining: A generalized batching technique for video-on-demand systems," in *Proc. of the International Conference on Multimedia Computing and Systems*, (Ottawa, Canada), June 1997.
2. M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "Splitstream: high-bandwidth multicast in cooperative environments," in *Proc. of the 19th ACM symposium on Operating systems principles*, pp. 298–313, 2003.
3. J. A. Pouwelse, J. R. Taal, R. L. Lagendijk, D. H. J. Epema, and H. J. Sips, "Real-time video delivery using peer-to-peer bartering networks and multiple description coding," in *Proceedings International Conference on Systems, Man and Cybernetics*, pp. 4599–4605, (The Hague, The Netherlands), 2004.

4. R. Puri and K. Ramchandran, "Multiple description source coding through forward error correction codes," in *Proc. Asilomar Conference on Signals, Systems and Computers*, (Asilomar, CA), Oct. 1999.
5. J. R. Taal, J. A. Pouwelse, and R. L. Lagendijk, "Scalable multiple description coding for video distribution in p2p networks," in *Proc. Picture Coding Symposium*, (San Francisco, CA), 2004.
6. J. Pouwelse, P. Garbacki, D. Epema, and H. Sips, "A measurement study of the bittorrent peer-to-peer file-sharing system," Tech. Rep. PDS-2004-007, Delft University of Technology, Apr. 2004.
7. V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Trans. on Information Theory* **39**, pp. 821–834, May 1993.
8. V. Goyal and J. Kovacevic, "Generalized multiple description coding with correlating transforms," *IEEE Trans. on Information Theory* **47**, pp. 2199–2224, September 1999.
9. J. Apostolopoulos, "Error-resilient video compression through the use of multiple states," in *Proceedings Int. Conf. on Image Processing, 2000*,
10. V. N. Padmanabhan, H. J. Wang, and P. A. Chou, "Resilient peer-to-peer streaming," in *Proceedings of the 11th IEEE International Conference on Network Protocols*, p. 16, IEEE Computer Society, 2003.
11. J. Apostolopoulos, T. Wong, W. tian Tan, and S. Wee, "On multiple description streaming with content delivery networks," in *Proc. IEEE INFOCOM*, July 2002.
12. V. Pai, K. Kumar, K. Tamilmani, V. Sambamurthy, and A. E. Mohr, "Chainsaw: Eliminating trees from overlay multicast," in *Proc. IPTPS*, 2005.
13. Y. Guo, K. Suh, J. Kurose, and D. Towsley, "P2Cast: peer-to-peer patching scheme for VoD service," in *Proc. of the 12th Int. Conf. on World Wide Web*, pp. 301–309, 2003.
14. T. Do, K. A. Hua, and M. Tantaoui, "P2VoD: Providing fault tolerant video-on-demand streaming in peer-to-peer environment," in *to appear in the Proc. of the IEEE Int. Conf. on Communications (ICC 2004)*, jun 2004.
15. D. A. Tran, K. Hua, and T. Do, "A peer-to-peer architecture for media streaming," *IEEE Journal on Selected Areas in Communications, Special Issue on Advances in Service Overlay Networks* **22**, pp. 121–133, jan 2004.
16. "The DIRAC project." <http://www.bbc.co.uk/rd/projects/dirac/>.