

Client clustering and joint multistream FEC rate allocation in IPTV systems

Jacob Chakareski and Pascal Frossard

Signal Processing Laboratory - LTS4, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

Abstract—This paper addresses the problem of clustering heterogeneous clients in IPTV services over lossy networks. The delivery of the same stream to clients with different capabilities or access networks is surely suboptimal in terms of average quality for the population of receivers. Instead, we propose that the streaming servers deliver distinct multicast streams to different subsets of clients. We formulate an optimization problem where the receivers are clustered depending on the quality of their connection so that the average video quality in the IPTV system is maximized. Then we propose a novel algorithm for determining optimally the clusters, as well as the source and channel rate allocation in each of the clusters. Simulation results show that the proposed algorithm is able to maximize the average quality in the system when each of the servers transmits information to a distinct cluster. In particular, we show that the proposed solution outperforms baseline schemes that serve all clients with the same multicast stream, as it is commonly the case in practical systems.

Index Terms—IPTV, FEC, rate allocation, error resiliency, video streaming

I. INTRODUCTION

Internet Protocol Television (IPTV) is an emerging multimedia application that is expected to gain a significant share of the broadcast media market [1]. IPTV systems deliver conventional TV content via IP multicast over privately owned and managed broadband networks. Customers access the service from a local switching office using their last mile Internet connection. In order to combat packet loss experienced during the streaming of the content over the access network of the customers the provider includes forward error correction (FEC) packets in the multicast distribution. A subscriber can thus recover the missing data on its end by decoding together the received media and FEC packets.

The rate of the FEC packets in an IPTV distribution is typically fixed and determined ahead of time. It usually corresponds to a conservative estimate of the loss rate that the media packets could experience during streaming, as assessed by the IPTV provider. However, despite its appeal due to its simplicity this approach in general reduces the efficiency of the content delivery process. In particular, customers whose access networks exhibit loss rates below the protection level of the FEC packets will be unnecessarily penalized with a drop in video quality. That is because the extra amount of FEC packets can be replaced with additional media data rate sent to these clients. This in turn will effectively increase the video quality of the IPTV content for such subscribers. On the other hand, customers experiencing packet loss rates exceeding the correction capability of the FEC data will observe a significant

degradation in video quality. That is because on the average they will not be able to recover the lost media packets using FEC decoding on their end.

The prospects of the above scenario are quite real at present as the receiving clients of an IPTV multicast typically exhibit a variety of packet loss rates. Today, we employ a plethora of broadband technologies for our Internet access links starting from DOCSIS (cable) or DSL, to Ethernet or FTTx (fiber), and to wireless (WI-FI, WiMAX, and cellular). Each of them is characterized with a markedly distinct profile in terms of bandwidth, packet loss, and delay. The IPTV clients' heterogeneity in terms of packet loss is further amplified by the fact that customers frequently rebroadcast the content to different TV sets in their homes again using a range of different networking technologies. Even if some IPTV systems implement some feedback mechanisms, a proper allocation of FEC redundancy minimizes the requests for expensive packet retransmissions.

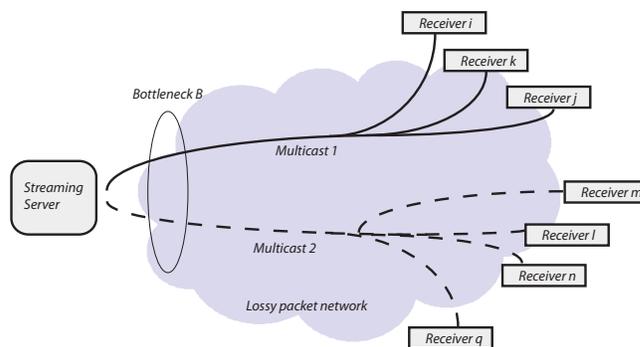


Fig. 1. IPTV system under consideration.

In this paper, we consider the framework illustrated in Figure 1, where a streaming server or a farm of streaming servers delivers concurrent IPTV multicast streams to a population of heterogeneous clients. The overall bandwidth resources are limited to a bottleneck capacity B . We consider that this capacity can be split equivalently between concurrent multicast sessions for the same video asset, where the source and FEC rate allocations in the different sessions can be defined adaptively. We formulate an optimization problem whose objective is to determine clusters of clients and the corresponding rate allocations for each multicast stream, such that the overall quality of service is maximized in the system. We propose a novel iterative algorithm to solve this optimal resource allocation problem and we show that the algorithm clearly outperforms common solutions where the allocation is

uniform across the whole client population.

Client partitioning in multicast streaming systems has been studied in [2] where a dynamic programming algorithm is proposed for computing the rate allocation that maximizes the sum of receivers' utilities. The authors in [3] address the problem of multicast server selection for adapting to the dynamics of streaming networks. The design of multicast solutions based on layered video streams has also been discussed in [4] and a comparison of delivery solutions based on versions or layers is provided in [5]. The case of lossy scenarios is studied in [6], where clients are partitioned into two classes in order to maximize the decoding quality of a layered multiple description coding scheme. Finally, in [7] the authors design a system for IPTV multicast over wireless LANs where a proxy server adapts the amount of FEC protection sent on the wireless link based on client feedback. Unfortunately, this limits the scalability of the system due to the related feedback explosion problem. Moreover, none of the works above addresses the joint problem of client clustering into concurrent sessions and source and FEC rate allocation to the individual clusters for a truly end-to-end optimized performance.

II. RESOURCE ALLOCATION PROBLEM

We characterize the subscribers of an IPTV service with the function $f(e)$ denoting the number of clients experiencing packet loss rate e during the multicast. Furthermore, let B denote the overall system bandwidth allocated for serving the content to the customers. In particular, the system allocates a portion of its bandwidth to the actual content, denoted as r , while the remainder, $B - r$, is used for sending the associated FEC packets. Finally, let (k, n) signify the FEC code that was used to generate the parity packets, where k denotes the number of source (media) packets and n denotes the block size of the code in packets. Then, the media data rate can be written as $r = (k/n)B$.

We are interested in serving the client population via a number of simultaneous multicast sessions M such that the overall video quality is maximized. In particular, each session $i = 1, \dots, M$ will employ a bandwidth B/M to deliver the content of interest at a different media data rate r_i . Using the notation introduced earlier this will correspond to employing different FEC codes (k_i, n) for the IPTV content multicast in session i . In the rest of the section we employ the terms session and server interchangeably though the issue of how each session is actually served by the IPTV system is beyond the scope of the present paper.

We will assign the clients to the various sessions according to their distribution over packet loss rates $f(e)$ such that the average video quality for the entire client population is maximized. In achieving this goal, we have three degrees of freedom over which we can operate: (i) The allocation of different clients to different sessions, (ii) The allocation of different FEC protection levels n/k_i for each session $i = 1, \dots, M$, and (iii) The overall number of sessions M .

Now, let $Q(r_i) = Q(k_i)$ denote the video quality of the content when its data rate is $r_i = (k_i/n)(B/M)$. Furthermore,

let $[e_{i-1}, e_i]$ denote the range of loss rates associated with the clients subscribed to the multicast session i . Then, the cumulative expected video quality for the clients in this packet loss rate range can be computed as

$$\int_{e_{i-1}}^{e_i} E[Q(k_i), e] f(e) de, \quad (1)$$

where $E[Q(k_i), e]$ denote the expected video quality for the IPTV content when multicast in the presence of packet loss rate e and protected with an FEC code (k_i, n) . This quantity can be computed as

$$E[Q(k_i), e] = \sum_{j=k_i}^n \binom{n}{j} (1-e)^j e^{n-j}. \quad (2)$$

Finally, using (1) we can characterize the expected video quality $E[Q]$ for the entire client population as

$$E[Q] = \frac{1}{\int_{e_0}^{e_M} f(e) de} \sum_{i=1}^M \int_{e_{i-1}}^{e_i} E[Q(k_i), e] f(e) de. \quad (3)$$

As described earlier, we are interested in optimizing (3) over the configurable system parameters of the IPTV multicast. In particular, let $\mathbf{k} = (k_1, \dots, k_M)$ denote the vector of source rates for the multicast sessions, let $\mathbf{e} = (e_0, \dots, e_M)$ denote the boundaries of the client groups as assigned to individual sessions, and lastly let M denote the number of sessions. Then, the optimization problem of interest can be written as

$$\max_{\mathbf{k}, \mathbf{e}, M} E[Q], \quad (4)$$

where the constraints of the optimization on the vectors \mathbf{e} and \mathbf{k} are explained in the next section.

III. OPTIMIZATION ALGORITHM

Here, we design an iterative coordinate descent algorithm for solving (4). In particular, each of the variables to be optimized, i.e., $\mathbf{k}, \mathbf{e}, M$, can be considered as one coordinate dimension of the optimization problem in (4). Hence, we propose to optimize over one coordinate while keeping the other two fixed, in an iterative fashion, until convergence. Moreover, it should be mentioned that the normalization constant $\int_{e_0}^{e_M} f(e) de$ in (3) can be omitted, as it does not affect the obtained solution. Finally, without lack of generality the first and last edges of the client population's binning can be selected as $e_0 = 0$ and $e_M = 1$.

In the rest of this section, we describe in detail the proposed optimization algorithm. Let $k_i^{(0)} \in \{k_{min}, \dots, n\}$, for $i = 1, \dots, M$, denote the initial values of the source rate at which the content is multicast by the individual servers. Here, k_{min} denotes the minimum acceptable source rate below which a server cannot multicast the content. Otherwise, it would result into customer dissatisfaction with the quality of the provided IPTV service. This is a system parameter that can be determined ahead of time. Similarly, let $e_1^{(0)} < e_2^{(0)} < \dots < e_{M-1}^{(0)}$ denote the initial loss rate values employed for

assigning the clients to different multicast sessions. Finally, the initial number of sessions M is selected to be sufficiently large.

Then, at every iteration $j = 1, 2, \dots$ we run two loops consecutively. In the first one, we adjust the packet loss rate bins as follows. For $i = 1, \dots, M - 1$ we solve

$$e_i^{(j)} = \arg \max_{e_i \in (e_{i-1}^{(j-1)}, e_{i+1}^{(j-1)})} \left(\int_{e_{i-1}^{(j-1)}}^{e_i} E[Q(k_i^{(j-1)}), e] f(e) de + \int_{e_i}^{e_{i+1}^{(j-1)}} E[Q(k_{i+1}^{(j-1)}), e] f(e) de \right). \quad (5)$$

Subsequently, we recompute the media data rates in the second loop. In particular, for $i = 1, \dots, M$ we solve

$$k_i^{(j)} = \arg \max_{k_i \in \{k_{min}, \dots, n\}} \int_{e_{i-1}^{(j)}}^{e_i^{(j)}} E[Q(k_i), e] f(e) de. \quad (6)$$

Next, we check whether the recomputed source rates satisfy $k_i^{(j)} \neq k_{i+1}^{(j)}$, for $i = 1, \dots, M - 1$. If this condition is satisfied, then the algorithm proceeds. Otherwise, the number of different sessions in the system needs to be reduced first. For instance, let $k_i^{(j)} = k_{i+1}^{(j)}$ for one specific $i \in \{1, \dots, M\}$. Then, session i and session $i + 1$ multicast at the same source rates to two adjacent segments of the client population in terms of packet loss rate. Therefore, we merge these two segments together and assign all corresponding clients to session i only. Session $i + 1$ is removed from the system as unnecessary. Hence the number of active session is reduced for one, i.e., $M = M - 1$. Similarly, we adjust the vector of source rates $\mathbf{k}^{(j)} = (k_1^{(j)}, \dots, k_i^{(j)}, k_{i+2}^{(j)}, \dots, k_M^{(j)})$ and the vector of loss rate values employed for the client binning $\mathbf{e}^{(j)} = (e_1^{(j)}, \dots, e_i^{(j)}, e_{i+2}^{(j)}, \dots, e_M^{(j)})$.

Finally, at the end of iteration j we compute the expected video quality $E[Q]^{(j)}$ using (3). If $E[Q]^{(j)} = E[Q]^{(j-1)}$ we have converged and the algorithm exits. Otherwise, the optimization proceeds with iteration $j + 1$. In Figure 2 below, we provide a formal algorithmic description of the optimization.

```

(0) Initialize  $\mathbf{k}^{(0)}$ ,  $\mathbf{e}^{(0)}$ ,  $M$ ,  $E[Q]^{(0)} = 0$ 
(1) for  $j = 1, 2, \dots$  do
    Compute  $\mathbf{e}^{(j)}$  using (5)
    Compute  $\mathbf{k}^{(j)}$  using (6)
    If  $\exists i$  such that  $k_i^{(j)} = k_{i+1}^{(j)}$ 
        Update  $M$ ,  $\mathbf{k}^{(j)}$ ,  $\mathbf{e}^{(j)}$ 
    Compute  $E[Q]^{(j)}$  using (3)
    If  $E[Q]^{(j)} = E[Q]^{(j-1)}$ 
        BREAK
    
```

Fig. 2. Proposed optimization algorithm.

Convergence of the proposed algorithm is guaranteed as the objective function is bounded from above and at every

subsequent iteration of the algorithm it is monotonically increasing (or at least non-decreasing). We have observed in our numerical experiments that the optimization rapidly converges, typically within a very small number of iterations. Unfortunately, a global convergence of gradient ascent (or descent) algorithms of this type is very difficult to prove because of the complexity of their objective functions. Lastly, it should be mentioned that in our experiments we usually initialize the client bins $(e_1^{(0)}, e_2^{(0)}, \dots)$ to linearly span the possible range of packet loss rates that the client population exhibits. We similarly select the initial source rates $(k_1^{(0)}, k_2^{(0)}, \dots)$ for the various client segments at the onset of the optimization.

IV. SIMULATION RESULTS

Here, we examine the performance of the proposed optimization algorithm for IPTV multicast to heterogeneous clients. The video content of interest comprises the CIF sequence Foreman encoded at 30 fps and multiple bitrates using an H.264 codec. Then, depending on the specific source rate computed by the optimization, a multicast server delivers one of these encodings to its target audience. The overall system bandwidth B available for multicasting the content to the clients was set to 10 Mbps. The FEC block size that we employed in our experiments is 64 packets. The maximum allowed rate for the corresponding FEC packets was set to 20% of the overall available data rate (B/M) . Note that this quantity corresponds to the system parameter k_{min} introduced in Section III denoting the minimum acceptable source rate for a multicast.

For our numerical experiments, we synthesized a client population characterized with an exponential distribution function number of clients versus packet loss $f(e)$, as introduced earlier. There are in total more than two 200K clients in the population. The minimum packet loss rate encountered among the clients is 0.5% while the maximum is 20%. In Figure 3, we show the function $f(e)$ together with the computed source rates for the client clusters corresponding to different multicasts. The vertical lines in Figure 3 delineate the different client segments according to their packet loss rate, as described in Section III.

We can see from Figure 3 that the number of required sessions that the algorithm computed is four. The corresponding source rates are $k_1 = 59$, $k_2 = 56$, $k_3 = 53$, and $k_4 = 51$. As expected, the optimization progressively reduces the source rate assigned to a multicast session as the corresponding packet loss rate increases. Specifically, in order to compensate for the increasing packet loss the algorithm decides to gradually increase the FEC packets' rate for the associated multicast thereby maintaining video quality at the maximum achievable level. Lastly, the expected video quality (Y-PSNR) for the client population, as computed by the algorithm, is 42.78 dB.

Next, we compare the multicast performance of our algorithm against those of two conventional schemes for delivering multimedia content in an IPTV context. Both of these techniques employ a single error protection level for the entire client population. The only difference between the two is the

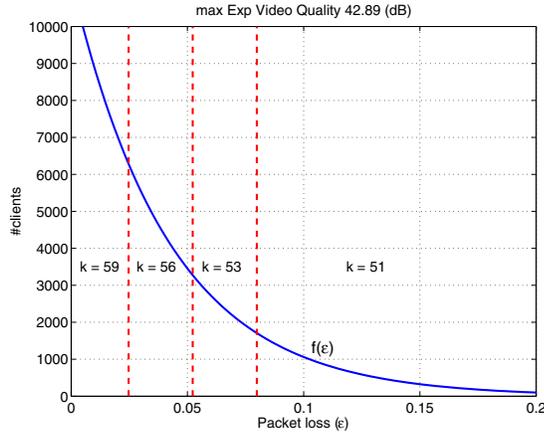


Fig. 3. IPTV multicast of CIF Foreman via four sessions: Client bins (marked with red vertical dashed lines); Corresponding source rates k_i (denoted in text); Client distribution function $f(\epsilon)$ (blue solid line).

rate of their FEC packets that they employ to this end. The first scheme, denoted henceforth *UnifMean*, selects the rate of its parity packets to correspond to the middle point of the range of packet loss rates exhibited by the client population. The second technique, denoted henceforth *UnifCentr*, selects the rate of its parity packets to correspond to the weighted (by $f(\epsilon)$) average of all possible packet loss rates. In other words, the rate of the FEC packets in this case corresponds to the centroid of the distribution of loss rates weighted by the distribution of number of clients as a function of packet loss, i.e., $\int \epsilon f(\epsilon) d\epsilon / \int f(\epsilon) d\epsilon$.

In Figure 4 below, we show a bar graph representing the video quality performance (Y-PSNR) of all three schemes for each of the four client clusters. In addition, we overlay in Figure 4 horizontal lines representing the average performance (in dB) of each scheme across the whole client population. The performance comparison between the three techniques is quite interesting and reveals a lot about their inner working, as discussed in the following. First, it should be noted that our optimization algorithm, denoted henceforth *Opt*, only slightly (0.04 dB) under performs relative to *UnifMean* in the case of the first client cluster, while it slightly (0.15 dB) outperforms *UnifMean* on the second client cluster. The performance gap between the two techniques, at the expense of *UnifMean*, then increases further to 2.7 dB and 17.7 dB in the cases of the third and fourth client segments, respectively.

Furthermore, the comparison between *UnifMean* and *UnifCentr* is even more interesting. By selecting the rate of its FEC packets to correspond roughly to the mid range of observed packet loss rate *UnifMean* is able to perform practically as well as *Opt* on the first cluster, while only slightly under performing on the second one. Contrarily, *UnifCentr* chooses its FEC packet rate to correspond to the far left of the packet loss rate axis, as necessitated by the distribution $f(\epsilon)$. This causes *UnifCentr* to non-trivially (0.9 dB) under perform the other two schemes even on the first client cluster, as the

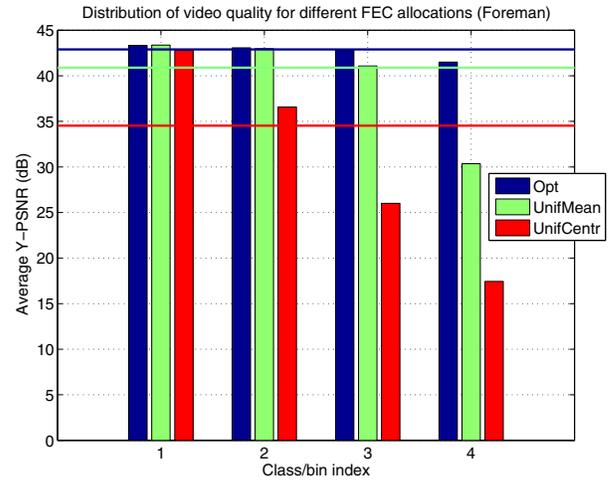


Fig. 4. CIF Foreman: Y-PSNR (dB) per client segment and overall for three multicast schemes: *Opt* (dark blue), *UnifMean* (green), and *UnifCentr* (red).

rate of its FEC packets is quite small and appropriate only for the clients exhibiting very small packet loss rates. The consequences of the imbalance between source and parity rates in the case of *UnifCentr* are accentuated even further when one moves on to the subsequent client clusters. This is illustrated well in Figure 4 by the continuously increasing degradation in video quality for *UnifCentr* over client cluster indices two, three, and four. For instance, for the second client segment the performance of *UnifCentr* has already dropped for 9 dB relative to its own performance on the first cluster. Similarly, the gap in performance between *Opt* and *UnifMean* on one hand and *UnifCentr* on the other has also increased for the same amount between the two client segments.

Next, it should be mentioned that the expected video quality over all clients in the population is 40.89 dB in the case of *UnifMean* and 34.53 dB in the case of *UnifCentr*. Therefore, the proposed optimization provides an average gain of 2 dB and 8.4 dB relative to *UnifMean* and *UnifCentr*, respectively. Finally, in Figure 5 we show the cumulative distribution function (CDF) of the expected video quality for the client population in the case of each of the three approaches. It can be seen that *Opt* substantially outperforms the other two techniques by providing close to maximally possible expected video quality for most of the clients in the population. In particular, the ratio between the standard deviation and the mean value of the video quality distribution across the clients is 0.0242, 0.1290, and 0.2736 for *Opt*, *UnifMean*, and *UnifCentr*, respectively. Therefore, the proposed optimization not only improves video quality on the average but also dramatically reduces its variation across the heterogenous clients.

In the remaining part of this section, we briefly go over the analogous results for another content, the CIF sequence Mother & Daughter (M & D). As in the case of Foreman, we encode M & D using an H.264 codec at 30 fps and a range of bit-rates. We then run our algorithm to determine the optimal number of multi-cast sessions as well as the associated source

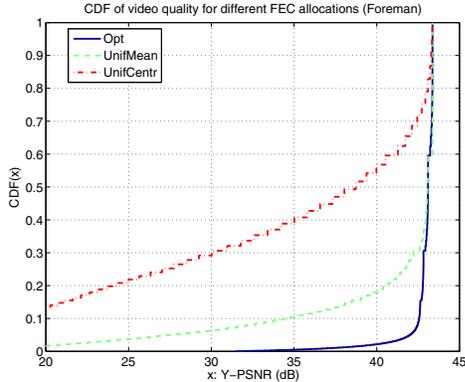


Fig. 5. CDF of video quality for CIF Foreman: *Opt* (solid, dark blue), *UnifMean* (dashed, green), and *UnifCentr* (dashed-dotted, red).

rates and client population bins. These results are shown in Figure 6. We can see that for IPTV multicast of M & D using three sessions is optimal for the selected client distribution $f(e)$. The optimal rates for the three sessions are 57, 54, and 51, respectively, as also denoted in Figure 6. It should be mentioned that due to the much smaller encoding rates for this content the overall system bandwidth B in our experiments was set to 1.25 Mbps.

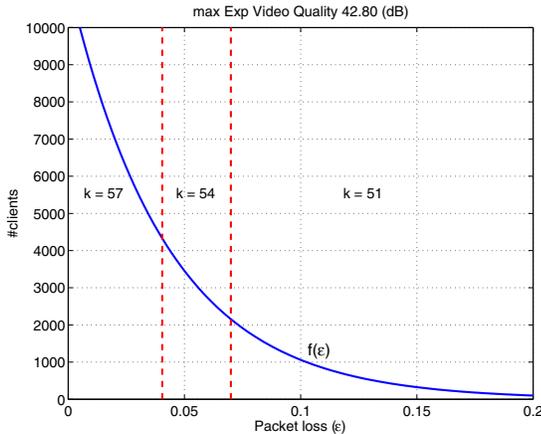


Fig. 6. IPTV multicast of CIF Mother & Daughter via three sessions: Client bins (marked with red vertical dashed lines); Corresponding source rates k_i (denoted in text); Client distribution function $f(e)$ (blue solid line).

Lastly, the corresponding per session average video quality values for the three FEC allocation techniques examined in this paper are shown in Figure 7. It can be seen that again *Opt* provides a significantly improved over the two techniques where a fixed FEC packet rate is assigned to the whole client population. For instance, similarly to the case of Foreman, average gains of 2 db and 8.5 db are recorded over *UnifMean* and *UnifCentr*, respectively. Due to space constraints, we do not include here the CDFs of video quality for the three FEC allocation approaches. There results were quite similar to those shown in Figure 5 for Foreman thereby exhibiting

quite analogous values for the ratio σ/μ for the video quality distribution.

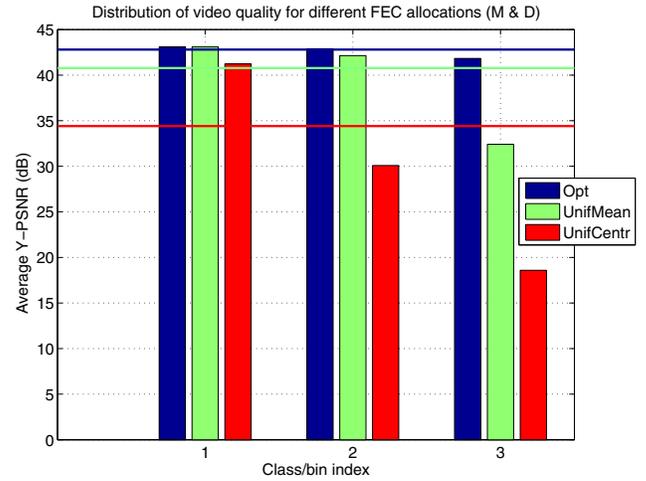


Fig. 7. CIF Mother & Daughter: Y-PSNR (dB) per client segment and overall for three multicast schemes: *Opt* (dark blue), *UnifMean* (green), and *UnifCentr* (red).

V. CONCLUSIONS

Current IPTV systems are often designed in a conservative manner and deliver the same multicast stream to heterogeneous clients that are subscribed to the same TV channel. We show in this paper that this strategy is suboptimal when the clients experience different loss rates. Under the assumption that the IPTV system can build several multicast streams for the same video asset, we propose in this paper a novel resource allocation algorithm that is able to select the best stream for each client, and an effective source and channel rate allocation for each of the multicast streams. Such a strategy outperforms baseline solutions that only deliver one single stream per asset, even if the source and channel rate allocation are optimized for a maximum average quality over the whole population of receivers.

REFERENCES

- [1] Multimedia Research Group, Inc., “IPTV Global Forecast 2009 to 2013,” <http://www.mrgco.com>.
- [2] Y. Yang, M. Kim, and S. Lam, “Optimal partitioning of multicast receivers,” in *Proceedings of IEEE ICNP*, Nov 2000, pp. 129–140.
- [3] Z. Fei, M. Ammar, and E. Zegura, “Multicast server selection: problems, complexity, and solutions,” *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 7, pp. 1399 – 1413, Jan 2002.
- [4] S. McCanne, M. Vetterli, and V. Jacobson, “Low-complexity video coding for receiver-driven layered multicast,” *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 6, pp. 983–1001, Aug 1997.
- [5] T. Kim and M. Ammar, “A comparison of layering and stream replication video multicast schemes,” in *Proceedings of ACM NOSSDAV*, Jun 2001.
- [6] P. A. Chou, H. J. Wang, and V. N. Padmanabhan, “Layered multiple description coding,” in *Proceedings of Packet Video Workshop*, Nantes, France, Apr. 2003.
- [7] M. Wu, S. Makharia, H. Liu, D. Li, and S. Mathur, “IPTV multicast over wireless lan using merged hybrid ARQ with staggered adaptive FEC adaptive systems for improved media streaming experience,” *IEEE Trans. on Broadcasting*, vol. 55, no. 2, pp. 363–374, June 2009.