Abstract—Multi-source streaming is essential for the design of a large-scale P2P streaming architecture. In this paper, we focus on improving the dependability of multi-source video streaming as an alternative to the more prevalent point-to-point streaming. We have designed and implemented a dynamic FEC (D-FEC) protocol for multi-source video streaming. Our D-FEC protocol has the ability to dynamically switch between 4 different FEC techniques to adapt to varying network conditions. A comprehensive performance evaluation was performed with a full-scale system prototype that gives insights into the behavior of the different adaptive FEC schemes, and shows the feasibility of the concept of switching FEC schemes during a streaming session. Guidelines are given in order to best design a FEC protocol switching strategy taking into account the experienced loss rate, the loss burstiness, and the original video stream rate.

I. INTRODUCTION

INTERNET Protocol television (IPTV) is rapidly emerging as a new TV platform. As part of IPTV offerings, the VOD services shift the traditional rigid broadcast video access where the content is selected by the broadcaster and pushed to passive users, into an on-demand distribution model where end-users are free to browse and instantly access video titles from a large content library.

As VOD continues to grow, one of the key aspects operators have to pay attention to is how to scale up both streaming capacity and storage capacity effectively. On-demand television requires significantly more network delivery resources as opposed to the broadcast video distribution model. In this context, the Vidas\textsuperscript{1} Project aims to developing very scalable VOD distribution platform for broadband operators [1]. The architecture is based on P2P streaming where the content is stored in the Set-Top-Boxes (STBs) that act as peers. This VOD distribution model relies on multi-source streaming architecture — any single VOD session requested by a given requesting STB consists in several complementary sub-streams transmitted by other contributing STBs in the network.

In this paper, we address the multi-source streaming reliability issues by investigating several adaptive FEC (Forward Error Correction) techniques suitable for a multi-source streaming case. Many FEC techniques have been proposed in the literature to address the video streaming reliability. XOR-based FEC techniques present the advantage of low processing complexity for a slightly reduced recovery efficiency compared to FEC codes; it also offers a limited flexibility in adjusting the redundancy rate [4]. On the other hand, FEC Interleaving technique is commonly used in video streaming systems to reduce the effects of clustered loss that typically appear with congestions [5]. The disadvantage of interleaving, however, is that it increases the latency due to additional buffering at the sender. Finally, Pro-MPEG is matrix-based XOR technique, which has been primarily developed by broadcasters for unidirectional video distribution. It is a very efficient error recovery technique with a limited processing power requirement, although more adapted for high-quality high-data-rate video streams where the stream buffering at the source is not constrained [6].

In this paper, four different adaptive FEC protocols are designed, adapted, implemented, and evaluated in our multi-source streaming testbed. Most important, we design and implement an integrated signaling protocol for a dynamic FEC protocol (D-FEC) that supports the online switching between different adaptive FEC protocols during a streaming session. The D-FEC protocol is tailored to instantaneously switch between different adaptive FEC protocols to best face the network conditions. Based on extensive performance evaluation using a varying packet loss pattern scenario and different video data rates, we highlight important observations and findings are concerning the performances of the considered adaptive FEC protocols. These performance observations are exploited to adjust the D-FEC switching strategy taking into account detailed characteristics exhibited by the network loss process such as average loss rate, packet loss correlation (burstiness), loss variation volatility, etc.

II. DYNAMIC FEC SWITCHING

Figure 1 illustrates our adaptive FEC protocols switching architecture at a high-level. The adaptive FEC protocols switching consists in switching between different adaptive FEC mechanisms based on the detailed conditions exhibited by the network and other characteristics intrinsic to the video content being streamed. The FEC packets are generated and conveyed within the same RTP session of the sub-stream.

---

\textsuperscript{1} This work is partially funded under the Enterprise Ireland VidAs project, CPTD/07/203. The project focuses on developing a scalable VOD system.
The FEC encoding being used at the contributing peer along with the FEC parameters are all signaled through an in-bound signaling channel using the RTP header and an additional RTP payload header as it is commonly practiced by IETF standards [7]. In conjunction with usual RTP inter-media (Timestamp) and intra-media (Sequence Number) synchronization tools, the above signaled FEC information allows the receiving peer to reconstruct the original FEC blocks without ambiguity (see Figure 1). A single adaptive FEC protocol operates independently at each sub-streaming session to best address the specific network conditions between the receiving peer and each contributing peer involved in the multi-source streaming session.

The focus of our current study will be on the dynamic FEC switching mechanism (D-FEC). Our objective is to thoroughly study the performances of 4 different FEC mechanisms so as to design the best strategy in switching between these FEC schemes in real-time basis during a streaming session. D-FEC should ultimately rely on statistics reported from the receiving peer to appropriately switch between FEC modes, but also to continuously adjust the FEC redundancy rate during the sub-streaming session.

In the above introduced FEC architecture, the contributing peers continuously adjusts the FEC redundancy rate based on feedback information received through the RTCP back-channel. This feedback report (RTCP-APP message) describes precisely the experienced loss rate before FEC recovering (loss rate induced by the network), the loss rate after FEC recovery (as an indication of the FEC efficiency), and the details of the loss pattern exhibited by the network using the metric Means Burst Length (MBL) [2].

### III. RELATED WORKS

Initial analysis and simulation of large-scale P2P streaming systems for broadband operators [1] has revealed that the reliability issue is one of the main inhibitors for a viable high-quality multi-source streaming that is an enabler for the P2P-based streaming model in general. Adaptive Forward Error Correction (FEC) techniques are, therefore, required to control the level of packet loss and keep the overall video streaming quality at acceptable levels. As an error control technique, FEC has the advantage to prevent the network loss avoiding as much as possible additional latencies associated with ARQ (Automatic Retransmission Request) techniques.

Adaptive FEC techniques need to be independently applied at each sub-streaming session between each contributing peer and the receiving peer to best address the specific fluctuating characteristics of each network path crossed by each sub-stream session.

There are different FEC encoding methods used to mathematically generate redundancy data from source data. Each of these techniques has a different complexity level and different error recovery efficiencies. The XOR (eXclusive-OR) encoding is the simplest FEC technique requiring very low processing complexity for a reduced recovery capability. LDPC (Low Density Parity Codes) codes have been largely adopted in the next generation satellite communications (DVB-S2) and are generally used to transfer very large objects.

The RS (Reed Solomon) encoding technique is the most common form of FEC coding with application ranging from DVD content protection, cable communication (DVB-C), and other content delivery over loss-prone networks. RS is a maximum distance separable (MDS) code, which means that no other FEC coding scheme can recover lost source data symbols from fewer received code symbols.

Substantial work has been carried out under the IETF (Internet Engineering Task Force) initiative to integrate FEC techniques in the standard Internet streaming protocols such as RTP (Real-Time Streaming Protocol). Different extensions and RTP profiles has been defined to support the signaling of FEC parameters and allow for FEC redundancy level adaptation [7].

Besides the diversity in the underlying FEC encoding techniques, there are different networking protocol approaches to improve the efficiency of error recovery. The FEC interleaving [2] is used to improve FEC codes resiliency against clustered and ‘bursty’ losses, while uneven FEC techniques [8] are meant to be more media-aware providing more loss-protection to more relevant parts of the media being conveyed. Also, there exist many other techniques that slightly vary the above concepts to accommodate scalable video coding and receiver-driven adaptation in multicast streaming scenarios [10].

The redundancy rate control is another important aspect of adaptive FEC protocols. Typical packet-level FEC protocols that use $k$ media packets to produce $n$ packets, including $h=n-k$ parity packets, have the capacity to overcome up to $h$ packet losses (when using MDS codes). This provides resiliency against a maximum packet loss rate of $p=h/n$ when considering that even FEC packets may be affected by loss. Thus, based on averaged packet loss rate ($p$) measured at the receiver and reported by the RTCP feedback, it is possible to constantly adjust the redundancy level $h$ as follows:

$$h = \frac{p \cdot k}{(1-p)}$$  \hspace{1cm} (1)

Many approaches use this simple linear model with varying levels of complexity to design adaptive FEC schemes. The maximum acceptable loss rate threshold beyond which the streaming server triggers FEC adaptation may differ depending on the nature of the audiovisual stream and its loss resiliency characteristics [8]. The frequency with which the network loss rate is reported to the sender may also impact the responsiveness of adaptive FEC schemes.

The above FEC redundancy control is effective only in cases where packet losses are uniformly distributed over the transmission. In reality, the packet loss process is often variable over time, with higher loss rates coinciding with transient network congestion events. The adaptation based on averaged loss rate is obviously more useful for a very large FEC block; it is less effective for low bit rate streams with limited buffering margins and a reduced RTCP reporting frequency. To address this issue, more advanced loss characterization models have been designed to capture the correlation between packet losses, where it was shown that a simple Gilbert model [4] can considerably improve the performance of adaptive FEC system over the Internet. In this model, one would use the probability of the loss state, instead of the averaged loss rate, in order to control the redundancy transmission.

In this paper, we focus on extensively evaluating the performances of 4 different adaptive FEC techniques under different realistic conditions and in a full scale testbed. The objective is to gain better understanding of the FEC protocols’ behavior to ultimately design a dynamic FEC (D-FEC) switching strategy able to change between different FEC modes according to the experienced network conditions and the video stream being FEC-protected. The 4 different adaptive FEC protocols to be investigated are further detailed below.

- **a)** The adaptive XOR-based FEC is generic enough and follows IETF recommendations.
- **b)** RS-based adaptive FEC protocol is constructed based on the most common FEC protocols reported in the research literature [9].
- **c)** UI-FEC (Unequally Interleaved FEC) protocol combines a specific uneven FEC interleaving technique with RS codes to overcome correlated packet loss.

$$\text{h} = \frac{\text{p} \cdot k}{(1-\text{p})}$$
d) Pro-MPEG is a technique that has been standardized under the Pro-MPEG forum (led by 8 manufacturers and telcos) to provide superior recovery capabilities using a matrix of XOR-based FEC blocks. This latter technique is as well used to protect IPTV streams in MPLS networks (such as British Telecom network).

It is worth noting that UI-FEC uses a Gilbert model to derive the FEC redundancy rate from the loss probability, while the other three FEC schemes use the loss rate as described in the Equation (1).

### IV. CASE STUDY

#### A. Experimental Network Setup

The performance results presented in this paper are obtained from a testbed developed as part of the VidAs project. The testbed is used to validate the concept of switching between FEC schemes in real-time basis, and to evaluate the performances of 4 different adaptive FEC schemes in respect to varying network conditions.

Our VidAs platform includes all different sub-systems such as the multisource video streaming sub-system, the content injection sub-system, and the video content pre-processing sub-system. Figure 2 illustrates the overall network setup used to evaluate the performances of adaptive FEC schemes.

![Figure 2: Network setup configuration.](image)

A SuperNode is used to initiate the video streaming connection as in the multi-source streaming scenario. For performance reasons, we use a separate computer to visualize the different streaming performances metrics in real-time basis during the streaming session. The Performance Monitoring Interface connects to the contributing peer and retrieves, in real-time, all raw data and logs related to the streaming session. This information is then analyzed to extract meaningful performance metrics that are plotted in graphs as the streaming session progress.

In order to focus on evaluating the different adaptive FEC schemes under the same conditions, we use only a single contributing peer that streams 100% of the aggregated video stream. The experiment is run 5 times to evaluate the 4 different adaptive FEC schemes and compare them with the dynamic FEC switching (D-FEC).

In our experiment a new video session is each time started with a different adaptive FEC scheme; for a fair performance evaluation, the same network conditions are reproduced – we vary both the loss rate and packet loss pattern in the same way during 20 minutes. The network condition is determined by an XML configuration file that defines a sort of state machine with three parameters: duration of a state, loss rate, and the mean burst length in a given state. The mean burst length is also referred to as packet loss run length, which in essence stands for the average number of consecutive lost packets during a loss event.

The Figure 3 illustrates the network conditions variation in terms of packet loss that is used to evaluate the performances of the different FEC mechanisms. This loss rate variation scenario tries to reproduce different realistic loss patterns during a-20-minutes streaming session. While the different states allow us to compare the behaviors of the different FEC schemes in face of different network conditions, the variation between states is also a very valuable parameter that allows one to evaluate the responsiveness capabilities of the considered adaptive FEC schemes.

![Figure 3: Variation of the loss rate and pattern during the experiment.](image)

The overall loss pattern illustrated in Figure 3 is composed of four main parts. The first part consists in steadily increasing the loss rate, which allows us to evaluate the efficiency and proactivity of FEC schemes in face of deteriorating network conditions. The second part consists in quickly alternatively increasing and then decreasing the loss rate, which would put tighter responsiveness requirements on the different FEC schemes. The third part consists in provoking ‘bursty’ packet loss and changing the overall loss rate; introducing correlated packet loss is meant to mimic a network congestion that usually manifests itself via clustered packet loss. Finally, the packet loss rate is steeply increased (up to 12% loss) in order to test the efficiency of the different FEC schemes in a highly loss-prone network.

Finally, we performed the same tests considering two different video contents with different bit rates: 1200 Kbit/s and 2500 Kbit/s. The objective here is to assess the impact of the video streaming rate on the performances of the different FEC schemes. In fact, a higher stream data rate would essentially relax the buffering limits requirement at the contributing peer, which in turn give a bigger margin to construct larger and more efficient FEC blocks.

#### B. Performance Metrics

It is important to define accurate performance metrics to fairly evaluate the relevant aspects of the different adaptive FEC schemes.

The FEC redundancy adaptation rate control algorithm is also decisive in setting the FEC accuracy and efficiency. As the video is streamed at the contributing peer, video packets are aligned and additional FEC redundancy packets generated to form what is known as FEC block: a set of data packet and FEC packets. If a minimum threshold number of packets are received at the receiving peer, then the lost packet can be recovered. Each FEC scheme has a different intrinsic efficiency with a different loss recovery capability [3].

The main parameter to take into consideration when determining the optimal FEC block size and the FEC redundancy rate is the loss rate (p) measured at the receiving peer, and made available to receiving peer through the RTCP backchannel. To calculate the best ratio of redundancy for each FEC block, the loss rate probability information from previous RTCP reports are given more weight compared to reports received earlier – a basic weighted moving average (WMA) is used to give more importance to recent reports while still considering past measurements. This favors the responsiveness of adaptive FEC schemes in face of rapid changes in the network conditions.
Depending on the FEC scheme considered, there will be 2 to 3 parameters reported back through RTCP, and used to re-adjust the FEC redundancy level. The parameters determination depends of the FEC type:

1) **XOR-based FEC**

   Since only one redundancy packet can be added with XOR-based FEC blocks, the main parameter to determine is the number ($k$) of data packets in a given FEC block. This uniquely depends on the measured loss rate ($p$):

   \[ k = \frac{1-p}{p} \quad \text{with} \quad 0 \leq p \leq 1 \quad \text{and} \quad n = k + 1 \]  

   Where:
   - $p$ = lost rate
   - $n$ = number of FEC and data packets
   - $k$ = number of data packets

2) **FEC Reed Solomon**

   Reed Solomon FEC codes require setting two parameters for each FEC block, which are the number of data packets ($k$) and the number of redundancy packets ($h$). Based on the intrinsic RS codes recovery capabilities, the ratio $h/n$ should be somehow matched with the current measured loss rate ($p$):

   \[ h = \frac{k \times p}{1-p} \quad \text{with} \quad 0 \leq p \leq 1 \quad \text{and} \quad n = k + h \]  

   Where:
   - $p$ = lost rate probability
   - $n$ = number of FEC + data packets
   - $k$ = number of data packets
   - $h$ = number of FEC packet

3) **Unequal interleaved FEC (UI-FEC)**

   Besides relying on the two FEC blocks parameters ($k$ and $h$) needed in RS codes, UI-FEC uses an additional interleaving parameter ($MBL$) to scatter originally consecutive packets in separate FEC blocks in order to reduce the effect of bursty (clustered) packet loss. MBL is calculated using an extended Gilbert model to track the average number of consecutive packet loss. MBL is tracked at the receiving peer and then reported back to the contributing peer through the RTCP backchannel.

   \[ h = \frac{K \times p}{1-p} \quad \text{with} \quad 0 \leq p \leq 1 \Rightarrow k = \frac{K}{i} \quad \text{and} \quad n = \frac{K + h}{i} \]  

   Where:
   - $p$ = packet loss probability
   - $n$ = number of FEC and data packets in each FEC block
   - $i$ = interleaving factor
   - $k$ = number of data packet in each FEC block
   - $h$ = number of FEC packet in a sequence of interleaved FEC blocks
   - $K$ = number of data packet in a sequence of interleaved FEC blocks

4) **FEC ProMPEG**

   In the ProMPEG -based adaptive FEC, the receiver requires information regarding the number of rows and columns in the FEC matrix. As we decided to use a square matrix, there will be only one parameter to determine: $k$.

   \[ H = \text{round} \left( \left( 2 \times \frac{1-p}{p} \right)^2 \right) \quad \text{with} \quad 0 \leq p \leq 1 \]  

   Where:
   - $p$ = lost rate probability
   - $n$ = matrix height
   - $k$ = Matrix width
   - $H$ = number of data packets

Below we give the two main performance metrics considered in our work to evaluate the different adaptive FEC protocols:

The Recovery Ratio (RR) is actually the ratio between the number of recovered packets and the total number of lost packets. This metric reveals how efficient and responsive is the adaptive FEC scheme in coping with continuously changing network conditions. There is also an element of FEC rate control accuracy attached to the RR performance metric.

\[ \text{FEC - RR} = \frac{\text{RecoveredDataPackets}}{\text{LossDataPackets}} \]  

The FEC recovery ratio, as given in formula (7), is not a sufficient metric to fully evaluate the performances of the different FEC schemes. An excessive FEC redundancy rate with high bandwidth inefficiency could, indeed, go unnoticed.

It is commonly accepted, from the Service Provider’s point of view, that additional bandwidth consumption is tolerated only for enhanced loss resiliency and video quality at the receiver side. In order to precisely evaluate the efficiency of different FEC schemes we define a FEC efficiency metric (FEC-EF) in the formula (8). The Efficiency Ratio (FEC-EF) is calculated by the receiving peer after completing the FEC.

\[ \text{FEC - EF} = \frac{\text{ReceivedDataPackets} + \text{RecoveredDataPackets}}{\text{TransmittedDataPackets} + \text{TransmittalFECPackets}} \]  

FEC-EF is used as the ultimate metric to compare the efficiency, responsiveness, and accuracy of the different FEC schemes.

**V. EXPERIMENTAL RESULTS**

In the following, we evaluate the performances of the five different adaptive FEC schemes (XOR, Pro-MPEG, RS, UI-FEC, and D-FEC) in terms of FEC recovery capability, responsiveness, and efficiency for varying network conditions. The experimental results are each time generated for two different video contents with different data rates; this essentially means we use different stream buffering margins at the contributing peer.

**A. FEC Recovery**

The Figure 4 and Figure 5 illustrate the recovery capabilities of the different adaptive FEC schemes for a-1200-Kbps and a-2500-Kbps...
videos, respectively. Pro-MPEG has been considered only in the scenario with 2500 Kbps video since it requires an important buffering at the source, which makes it very inefficient for low bit rate streams. Its matrix-based FEC block structuring makes it hard to control the redundancy level with an acceptable granularity.

XOR-based adaptive FEC delivers quite performing recovery capabilities, in overall. Particularly, it converges faster than the other adaptive FEC protocol, as it can be seen from the period with 5% of loss rate over 10 seconds. In contrast, for a longer period of stable loss rate, RS and UI-FEC outperforms the other adaptive FEC protocols. They are more accurate after receiving several RTCP reports since their FEC redundancy rate adjustment model is based on a linear equation.

Since D-FEC was designed to switch to one of the four adaptive FEC protocols to achieve the best performances, we can notice that it performs well only when the network conditions are stable enough. During high network conditions volatility periods (e.g., frequent change of loss rate) the D-FEC protocol keeps switching between different adaptive FEC schemes and ends up producing poor performances. D-FEC is, however, as efficient as the best adaptive FEC protocol when the conditions exhibited by the network remain stable (e.g., 3% of loss rate during 2 minutes).

An important observation is that all adaptive FEC protocols produce better results when using high-data-rate streams. This is mainly due to the fact that there are more stream buffering margins at the contributing peer, allowing the use of larger FEC blocks that deliver a better efficiency for the same redundancy rate. It is particularly prevalent for Pro-MPEG that needs an important number of data packets to construct the FEC matrix. Pro-MPEG is also more effective in face of higher loss rates.

The overall FEC recovery rate for each adaptive FEC protocol over the whole streaming session is given in Table 1. After excluding Pro-MPEG that uses excessive FEC redundancy rate, we can see that D-FEC outperforms for a low-bit-rate video stream, while RS outperforms for a high-bit-rate video stream.

### Table 1: Averaged FEC-RR values over a streaming session.

<table>
<thead>
<tr>
<th>Bitrates</th>
<th>XOR</th>
<th>RS</th>
<th>UI-FEC</th>
<th>ProMPEG</th>
<th>D-FEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1200</td>
<td>0.36888</td>
<td>0.43353</td>
<td>0.41779</td>
<td>0.97828</td>
<td>0.52921</td>
</tr>
<tr>
<td>2500</td>
<td>0.28075</td>
<td>0.48064</td>
<td>0.38490</td>
<td>0.89020</td>
<td>0.44991</td>
</tr>
</tbody>
</table>

The FEC recovery capability is only one aspect of the FEC protocols performances. The amount of bandwidth consumed by the extra FEC redundancy transmission is also very important from service provider perspectives. Extra bandwidth consumption could be acceptable only for an improved loss resiliency.

### B. FEC Redundancy

In both test cases (see Figure 6 and Figure 7), FEC redundancy levels react correctly to match the varying loss rates, staying always above the current experienced loss rate.

Pro-MPEG produces a very high redundancy rate which explains its superior performances in terms of FEC recovery. During the whole low-data-rate scenario, Pro-MPEG was using a 7x7 matrix with 49 video packets and 2x7 (14) FEC packets which puts it at around 28% of FEC redundancy. This FEC redundancy rate is somehow reduced for the high-data-rate scenario. There is no variation of the FEC redundancy level with Pro-MPEG since the experienced loss rate reach a maximum of 12%.

Again, from the below results we can confirm that XOR-based adaptive FEC is the most responsive and reactive to loss rates changes. This good performance in terms of FEC recovery rate is due to the fact that we can more easily adjust the FEC block size to instantaneously much the network conditions. Based on this observation, the D-FEC can be improved to switch and stay on the XOR-based mode when the network exhibits volatile conditions.
Both the consumed bandwidth and FEC recovery ratio should be tied up together in another metric to measure the efficiency and accuracy of each adaptive FEC considered. For this reason, the efficiency factor is measured to clearly quantify the effectiveness of each adaptive FEC protocol.

C. FEC Efficiency ratio

Efficiency ratio is calculated with the same method for all FEC models, as revealed earlier. FEC efficiency factors are illustrated in Figure 8 and Figure 9. Table 2 illustrates the average FEC efficiency measurement over the whole test scenario for the two video streams.

Clearly Pro-MPEG produces very poor performances in terms of bandwidth utilization. There is an excessive redundancy rate that while it allows recovering packet loss, it leads to bandwidth overutilization. Pro-MPEG’s performances improve greatly with a higher video data rate and loss rate. Pro-MPEG is more adapted for very high data rates which could be the case for HD-quality video multi-source streaming. In this case, there will be much more flexibility in terms of buffering to produce large FEC matrices and overcome lower loss rates.

<table>
<thead>
<tr>
<th>Bitrates Kbit/s</th>
<th>XOR</th>
<th>RS</th>
<th>UI-FEC</th>
<th>ProMPEG</th>
<th>D-FEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1200</td>
<td>0.94606</td>
<td>0.97166</td>
<td>0.97284</td>
<td>0.79231</td>
<td>0.96137</td>
</tr>
<tr>
<td>2500</td>
<td>0.92020</td>
<td>0.97845</td>
<td>0.97375</td>
<td>0.85176</td>
<td>0.96818</td>
</tr>
</tbody>
</table>

It is clear that UI-FEC outperforms the other adaptive FEC protocols in a bursty environment where packet loss events are correlated. However, this advantage is lost with a high data rate video stream where the construction of very large FEC blocks in RS-FEC provides inherent immunity against clustered packet losses. FEC interleaving is less effective in this case, and for almost all network conditions.

Figure 8: FEC Efficiency Ratio for a 1200Kbps video.

Figure 9: FEC Efficiency Ratio for a 2500Kbps video.

D. Discussion

The very specificity of multi-source streaming makes it very hard to assume high-data rates at each contributing peer, even in case of HD-quality video streaming. Clearly, it is not possible to stream large video volumes from individual contributing peers due to limited uplink bandwidth in asymmetrical broadband networks. The most likely scenario to stream HD-quality video streams is to use more contributing peers. Therefore, the FEC protocols performances in the scenario with low video data rate should the most likely one. Also, very high loss rates are not very likely to be experienced since there will be other fail-over mechanisms that overcome this by switching contributing peers in prevention.

Based on the above observations, it appears that D-FEC should switch between adaptive FEC protocols not only based on the network conditions (loss rate and burstiness) and the video data rate, but also based on the volatility of the network conditions. The speed and amplitude with which the network conditions change are very important factors to consider, as well. Indeed, some adaptive FEC protocols (XOR, RS, and UI-FEC) are more resilient to very volatile network conditions where a great deal of responsiveness is required, while other equation-based redundancy control FEC protocols need more time to converge and achieve better performances. While the weight averaged loss rate maybe tuned to increase the responsiveness of UI-FEC and RS-FEC, it will also lead to an important oscillation in the performances. A new metric should be derived to characterize the network conditions volatility by using the frequency and amplitude of network condition changes.

VI. CONCLUSION

This paper aims to designing the most efficient adaptive FEC strategy. Accordingly, four different adaptive FEC schemes have been implemented and their performances thoroughly compared under varying network conditions. After identifying the specific conditions where each considered adaptive FEC yields superior performances, a new Dynamic FEC (D-FEC) protocol has been designed and implemented to switch between the different adaptive FEC based on the varying conditions exhibited by the network. The concept of dynamically switching between adaptive FEC schemes has been validated via a full-scale system testbed.

The performance evaluation results suggests that each FEC scheme is more effective in specific conditions, and ideally a FEC protocol (D-FEC) should dynamically switch between different adaptive schemes based on network conditions, loss rate volatility, and the video stream traffic pattern.

REFERENCES


