SDC: Scalable Description Coding for Adaptive Streaming Media

Jason Quinlan [1], Ahmed Zahran [2], Cormac Sreenan [1]

[j.quinlan, cjs]@cs.ucc.ie, azahran@eece.cu.edu.eg

[1] Department of Computer Science, University College Cork, Ireland.

[2] Electronics and Electrical Communications Dept., Cairo University, Egypt.

***Abstract*—Video compression techniques enable adaptive media streaming over heterogeneous links to end-devices. Scalable Video Coding (SVC) and Multiple Description Coding (MDC) represent well-known techniques for video compression with distinct characteristics in terms of bandwidth efficiency and resiliency to packet loss. In this paper, we present Scalable Description Coding (SDC), a technique to compromise the tradeoff between bandwidth efficiency and error resiliency without sacrificing user-perceived quality. Additionally, we propose a scheme that combines network coding and SDC to further improve the error resiliency. SDC yields upwards of 25% bandwidth savings over MDC. Additionally, our scheme features higher quality for longer durations even at high packet loss rates.**

***Index Terms*—Scalable Video Coding, Multiple Description Coding, Scalable Description Coding, Incremental Datagram Delivery, Network Coding, Layered Coding**

I. Introduction

According to the 2011 Cisco Visual Networking Index [1], mobile video traffic exceeded 50% for the first time in 2011. More interestingly, the video traffic will reach 70% of the mobile data traffic at a compound annual growth rate of 90% between 2011-2016. These figures are alarming for mobile network operators, who are already struggling to keep up with data backhaul demands and the management of heterogeneous wireless networks with diverse data rates. At the same time, users expect to view video on devices with vastly different configurations, ranging from tiny smartphones to large-screen high-definition TVs. Hence, the potential benefits of scalable media streaming techniques are apparent by permitting the adaptation of video to match the device resolution and available network resources, without significantly reducing user Quality of Perception (QoP) [2].

Currently, fixed coding dominates and streaming servers will offer either a single stream resolution or a selection of predefined resolutions from which to choose, sometimes known as Simulcast [3]. Simulcast typically requires excessive storage while using single resolution may result in unsuitable video quality for some devices. More importantly, both approaches lack flexibility of adapting to network conditions. Additionally, both approaches may result in inefficient usage of system bandwidth as the number of users increase.

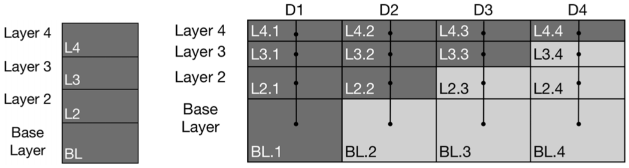
Video coding techniques represent a means to mitigate the aforementioned limitations. More specifically, scalable methods allow devices to dynamically modify their stream quality by adjusting the percentage of received data. Thus a device might commence by selecting to receive the highest resolution it can support, but subsequently alter the quality in response to changes in the achievable network throughput. Scalable Video Coding (SVC) [4], an extension to the H.264/MPEG-4 Part 10 or AVC (Advanced Video Coding) compression standard, achieves streaming scalability by partitioning a video stream into several subsets, commonly called layers. The cumulative receipt of each additional layer offers a progressively higher quality [4], [5]. A serious limitation in SVC is that the loss of a lower layer hinders the device from decoding the dependent higher layer. Hence, Multiple Description Coding (MDC) offers an improved error resiliency to packet loss, but at the cost of increased data rates due to redundancy techniques, such as forward error correction (FEC), as in MDC-FEC [6], [7].

In this paper, we present Scalable Description Coding (*SDC*), a technique that enhances MDC-FEC, henceforth referred to as MDC, with a novel transmission scheme to achieve lower data rates without sacrificing user-perceived quality. SDC operates by redefining the MDC description prior to transmission, to reduce the required bandwidth. Compared to MDC, SDC improves the user-perceived quality with lower bandwidth usage levels, while offering increased robustness against packet loss. Our analysis quantifies the data rate reductions, showing that in some instances the SDC data rates are on par with those of SVC. Furthermore, we propose several optimizations to SDC, including SDC with network coding [8], that further improves SDC performance.

The remainder of the paper is organized as follows. Section II gives background information on SVC, MDC and existing research on MDC optimization. Section III presents SDC, followed by two optimization mechanisms in Section IV. Section V is dedicated to evaluation while Section VI concludes the paper.

II. Background

SVC separates each stream into a Base Layer (*BL*) and one or more enhancement Layers (*EL*) as illustrated in Figure 1a. The BL represents a low resolution version of the stream, normally produced from the key frames, while the ELs contain higher quality information. By receiving the BL and a relevant EL, a higher quality stream resolution can be decoded. A major limitation in SVC is that a higher layer can only be decoded when all lower layers on which it depends are received. It may be the case that the higher layers depend on a subset of the lower layers according to the dimensions being scaled - scaling could be temporal (frame rate), spatial (image dimensions) or quality (signal to noise ratio) [4]. Note that the BL is always required for SVC media decoding.



(a) SVC **(b)** MDC

Figure 1: 4-layer SVC and 4-description MDC Group of Frames (GOF).

MDC creates *N* independent descriptions representing stream vertical slices of *equal importance* using layer partitioning and FEC, as illustrated in Figure 1b [9]. Layer partitioning divides each SVC enhancement layer into a number of equal *sections*, based on its layer order value, e.g., layer two is subdivided into two equal byte sized sections. These sections are then extended across four descriptions using FEC such that receiving any two descriptions would allow for correct decoding of layer two. Note that each description contains a copy of the BL, such that the reception of *any* description would enable decoding of the BL. More generally, *M* descriptions are required for decoding the M^{th} layer. In Figure 1b, the minimum number of sections required to decode and view all streams is shown in dark shade, with the redundancy shown in light shade. Clearly, the quantity of redundant data in MDC is inversely proportional to the level of the layer in SVC, with the BL containing the highest overhead.

Several works in the literature aims to reduce the data redundancy in MDC. One approach is to remove the BL from the MDC description and place it in its own description [10], [11]. Other ideas propose to separate the BL to generate two descriptions [12] or reduce the number of higher layers to suit the requesting devices [13]. In this paper, we focus on reducing the higher layer replication which in turn reduces the number of descriptions required and this reduction offers greater opportunity for better use of available bandwidth.

III. Scalable Description Coding (SDC)

Typically, an SVC stream consists of M layers, and to increase resilience to network loss this stream is encoded as an MDC of N descriptions. Hence, the MDC representation of each Group of Frames (*GOF*) can now be viewed as a matrix of M times N $sections$. In keeping with the relationship between the receipt of an additional MDC description and the incremental increase in stream quality, it is most natural that M and N be equal. Hence, each layer i has (N-i) redundant sections. This redundancy translates to an increased transmission cost and a higher consumption of device computation and battery resources. The following subsection explains how SDC can reduce these drawbacks, followed by a four-layer video example.

1. SDC Overview

The design of Scalable Description Coding (SDC) aspires to reduce the transmission cost of MDC by reallocating a subset of the enhanced layer sections to a new *scalable* description prior to transmission. Hence, SDC decreases the number of N descriptions required to decode a media stream, while maintaining the level of stream quality received by the user. As previously stated, MDC utilizes layer partitioning to create numerous sections per layer and FEC to extend the original section over numerous descriptions. Eq (1) defines the maximum stream quality of an MDC GOF to be the sum of all original sections for all layers. It also defines the per layer partitioning ratio by generating the number of original sections per layer.

TABLE I: Main SDC Notation

Q	The stream quality value
N	The number of SVC layers per GOF
K_j	The byte size of layer j
$S_{i,j}$	The i^{th} section of layer j
$size[arg]$	Function that returns the byte size of arg
D_c	The complete description
D_s	The scalable description
D_r	The redundancy description

$$\max_{Q} = \sum_{j=1}^{N} \sum_{i=1}^{j} S_{i,j} \tag{1}$$

Prior to transmission, SDC will reallocate these pre-FEC original sections, so as to reduce FEC redundant replication and thus the number of transmitted descriptions.

SDC consists of three types of descriptions:

i) A Complete description - D_c: the primary role of this description is the delivery of the lower layer sections of the stream. It is identical to an MDC description in that it contains one section from each layer in a GOF and as such contains a percentage of redundant replication. Similar to MDC, one or more D_c can be transmitted per GOF. Eq (2) defines the byte size and section allocation for a complete description.

$$size[D_r] = size[D_c] = \sum_{j=1}^{N} size[S_{1,j}] \tag{2}$$

ii) A Scalable description - D_s: the role of this description is the delivery of the higher layer sections of the stream. It is formed by combining several sections from the enhancement layers in an iterative downward fashion and as such contains no redundant replication. Initially, one section from the highest layer is added to the scalable description, $S_{N,N}$, this iteration is the base case. Subsequent iterations shall only commence, iff the byte size of one section from the next layer down, $S_{N-1,N-1}$, plus one additional section from every layer added so far, $S_{N-1,N}$, is less than or equal to the byte size of the D_c. In this manner, every iteration above the base case will reduce the number of D_c required to decode the stream by one. Eq (3) defines the section allocation for the scalable description.

$$size[D_s = \sum_{i=1}^{N} \sum_{j=1}^{i} S_{N-i+1,N-i+j}] \leq size[D_c] \tag{3}$$

One benefit from this manner of D_s creation, is that the value of the layer, with only one section added to the D_s, is the combined number of D_c and D_r required by SDC. Thus allowing us to rewrite Eq (1) as Eq (4).

$$\max_{Q} = D_c * (N - i) + D_s + D_r \tag{4}$$

The initial design of SDC is such that only one D_s shall be transmitted per GOF and that the byte size of the D_s is comparative to the byte size of the D_c. Future work will investigate attaining further benefits from other design alternatives for the D_s.

iii) A Redundancy description - D_r: the role of this description is to reduce the effects of network loss. The D_r is either formed by utilizing FEC (to extend the original sections) or by employing network coding (to combine the previously transmitted descriptions), as highlighted later in the paper. It is identical to the D_c in that it contains one section from each layer in a GOF, but is similar to D_s as only one D_r can be transmitted per GOF. Eq (2) defines the byte size and section allocation for a redundancy description.

The transmission scheme for an SDC stream is comprised of one or more D_c, one D_s and one D_r, transmitted in that order.

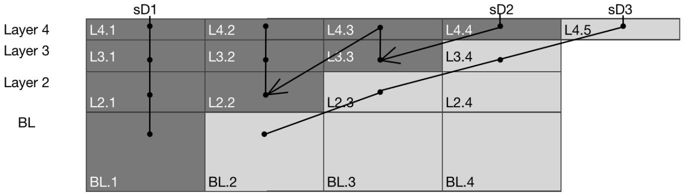


Figure 2: An example of four descriptions being reduced to three descriptions using SDC.

2. A four-layer video Example

To better appreciate the benefits of SDC we provide an example in which we assume that each SVC layer is 300 bytes. This number is chosen purely to simplify the example by allowing each layer/description to be transmitted within the size of a typical un-fragmented IP datagram. By assuming a value of 300 bytes per layer, this yields an SVC GOF of 1,200 bytes - 4 layers x 300 bytes. Note that in a case where the layers are not the same byte size, but incrementally larger/smaller, the same mechanism is employed but additional performance gains may be attained, due to the changes in section size. Let the corresponding MDC representation have four descriptions, each consisting of 625 bytes (300 + 150 + 100 + 75), determined using Eq 2, thus totaling 2,500 bytes, 650 x 4, for the MDC GOF. Clearly, to improve the stream reliability, MDC adds significant overhead to SVC, providing a sufficient motivation for SDC to offer efficiencies by reducing bandwidth demands but without sacrificing user-perceived quality.

By utilizing Eq (1) to Eq (3), SDC can re-packetize the sections from Figure 1b into the descriptions in Figure 2; note that the black lines through the sections define the contents of each description. It can be seen that sD1, D_c, combined with sD2, D_s, now contains all of the data required to decode the original stream to its highest quality, i.e. four layers in the example, and is on par with the transmitted byte size of SVC. Owing to the manner in which the D_s is created, one additional section, L4.5, is required. This additional section is mandatory, as the D_r, sD3, must contain one section of layer four. This section can be computed by replicating L4.1, as the D_s, sD2, already contains the other three sections from layer four. In this manner, we have reduced the number of descriptions transmitted over the network from four to three, thus yielding bandwidth savings of 25% over MDC. Increased savings can be achieved with SVC streams of larger number layers.

Table II highlights the number of sections allocated to each type of description in SDC, byte size of each section per description and number of sections required to decode a layer. Note that to decode layer *N*, all sections for layers 1 to *N* must have been received. As can be seen, a D_c contains one section from each layer, whereas the D_s can contain numerous sections for a given layer. Note that the byte size difference between MDC, 625, and SDC, 575, could be utilized by SDC as a construct for message handling between server and device. SDC has similar dependency hierarchies to both SVC and MDC. Like SVC, the D_s is of a higher priority due to the potential of reduced bandwidth and increased stream quality but similar to base layer loss in SVC, the dependency hierarchy in SDC, also increases the possibility of frame loss where only D_s is received. Whereas like MDC, if any combination of D_c or D_r is received, the system performs exactly as an MDC system, in which a reduced quality version of the stream is decodable and the device/network incurs approx. 52% inherent bandwidth loss. Furthermore, SDC does not introduce any additional constraints on media coding/decoding beyond those for MDC. Clearly, SDC yields significant bandwidth savings in comparison to MDC as shown in Table III. However, its bandwidth requirements relative to SVC are certainly not negligible. This issue will be addressed by the optimization mechanisms presented in the following section.

TABLE II: Number of sections allocated to each type of description in SDC, byte size of each section per description and number of sections required to decode a layer

	Sections Required:								
	MDC/SDC Complete Desc				SDC *Scalable* Desc				To Decode
Layers	4 Layer		6 Layer		4 Layer		6 Layer		a Layer
6	n/a	n/a	1	50	n/a	n/a	4	200	6
5	n/a	n/a	1	60	n/a	n/a	3	180	5
4	1	75	1	75	3	225	2	150	4
3	1	100	1	100	2	200	1	100	3
2	1	150	1	150	1	150	0	0	2
BL	1	300	1	300	0	0	0	0	1
Total		625		735		575		630	

IV. SDC Optimisations

In this section we present two additions to the basic SDC approach, each of which serves to offer an improvement on performance.

1. SDC-NC: Network Coding for SDC

We propose incorporating network coding (NC) [14], [15] into SDC to improve the overall performance. In SDC-NC, the redundancy description, D_r, as outlined in Section III, is replaced by a network coded (NC) description, D_{r-nc}, which is the exclusive disjunction, or XOR, symbolized by $\oplus$, of all the leading descriptions for a given GOF, $D_c \oplus D_s$ in our example. As with most NC implementations, should the byte size of the D_s be smaller than a D_c, then the D_s shall be padded with trailing 0s [16], to maintain a description of equal size and balance the impact of the NC mechanism mechanism.

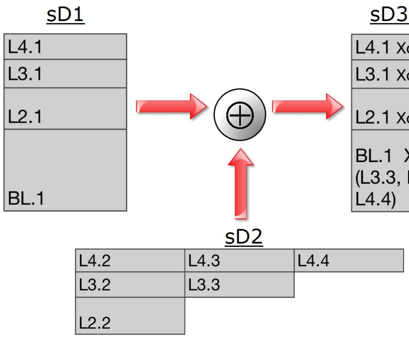


Figure 3: Network Coding design for SDC-NC.

By utilizing SDC-NC, individual devices can recover from the loss of any description (scalable or complete) by receiving the NC description. In this manner, if *any* two descriptions are correctly received by the client device in the four-layer example, the full quality stream can be decoded.

While NC significantly improves the resiliency of SDC to errors and losses, it is important to note that while the quality of the decodable stream generally increases, the number of scenarios in which no frame can be decoded also increases. In this manner, the NC description can also be seen as a prioritized description, such that receiving only that description increases frame loss and as a result decreases stream quality, but as can seen in the evaluation in Section V this loss rarely occurs.

2. I-SDC: SDC with Incremental Datagram Delivery

Due to the inter-dependence of layers/descriptions in adaptive streaming, it is of benefit to the receiver to ascertain without delay which specific descriptions have been received during transmission. Hence, the device can infer what has been lost and what is ultimately required to maximize stream quality - all remaining descriptions or just a subset. SDC is ideally positioned to benefit from this type of inference, as its architecture consists of a combination of prioritized (D_s and D_{r-nc}), equal importance (D_c) and redundancy (D_r) descriptions, and it offers the potential of maximum stream quality with only a subset of descriptions. With this benefit in mind, SDC can be further improved by optimizing the transmission of its descriptions.

Incremental Datagram Delivery (IDD) is proposed as a dynamic transmission protocol, which is based on an incremental delay in the initial transmission of each description in a GOF, with higher priority descriptions being offered transmission precedence. Thus increasing the time available for retransmission of the important descriptions and for the receiving device to infer its ongoing requirements based on the descriptions received so far. In this manner, IDD offers a prioritized transmission scheme which can offer benefits by reducing the occurrence of transmission obstacles such as delivery latency, datagram re-ordering, and by increasing QoS.

By employing IDD-based SDC (I-SDC), and by implementing the transmission schedule is as outlined in section III, the device is able to disregard the final description should all previous description be received. Table III shows that I-SDC would attain significant processing savings over SDC, a 35% reduction, by eliminating the need for decoding the redundant SDC description, D_r. As can be seen, the current implementation of I-SDC while decreasing resource usage on the device, is not reducing network consumption, as the final description is only being dropped after being received at the device. Future work will consider IDD transmission options to reduce network costs.

TABLE III: GOF byte size processed at the device, using an example of 300 bytes per SVC layer

	Number of bytes processed at the device			
	4 Layer		6 Layer	
	Bytes * (Layers or Desc)	Total	B * (L or D)	Total
SVC	300 * 4	1,200	300 * 6	1,800
SDC	625 * 2 + 575 * 1	1,825	735 * 3 + 630 * 1	2,835
I-SDC	625 * 1 + 575 * 1	1,200	735 * 2 + 630 * 1	2,100
MDC	625 * 4	2,500	735 * 6	4,410

It is important to note that while SDC-NC and I-SDC are separate optimization techniques, they combine quite naturally, to further increase SDC performance.

V. Evaluation

In this section, we assess the performance of SDC and SDC-NC and compare it with both SVC and MDC. The theoretical improvements in reduced device processing from the use of IDD were quantified in the previous section, but in the simulations we chose not to implement IDD because the benefits offered by SDC for bandwidth reduction and increased stream quality are the focus of this paper. In our evaluation, we consider four and six layered streams transmitted over a lossy network with different loss rates. The lossy medium is simulated using Network Simulator 2 (ns-2) [17] and a 300 frame layered video is obtained using the JSVM framework [18]. Figure 4 provides an overview of the adaptive streaming topology that our evaluation simulates.

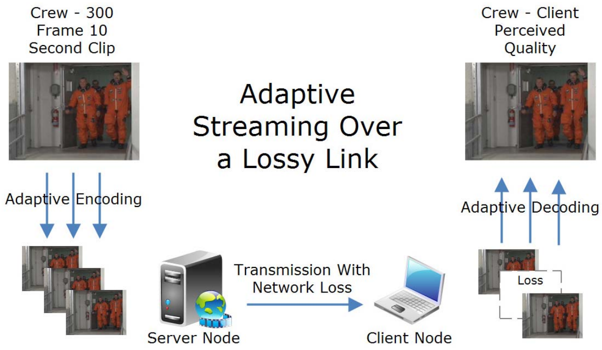

Crew - 300 Frame 10 Second Clip
Adaptive Streaming Over a Lossy Link
Crew - Client Perceived Quality
Adaptive Encoding
Adaptive Decoding
Transmission With Network Loss
Server Node
Client Node
Loss

Figure 4: Overview of adaptive streaming topology used to determine bandwidth savings and PSNR.

TABLE IV: Example of average percentage of viewing time at a specific layer for 4 and 6 layers with 10% network loss

	4 Layer				6 Layer			
	SDC-NC	SDC	MDC	SVC	SDC-NC	SDC	MDC	SVC
Layer 6	n/a	n/a	n/a	n/a	95.7%	89.2%	52.7%	51.2%
Layer 5	n/a	n/a	n/a	n/a	0%	0%	36.2%	5.8%
Layer 4	97.8%	90.8%	63.4%	64.6%	0%	0%	9.4%	7.8%
Layer 3	0%	0%	31.3%	7.3%	0%	5.6%	1.7%	8.6%
Layer 2	0%	6.5%	5.1%	8.3%	1.2%	2.5%	0%	8.9%
Layer 1	.8%	1.7%	.2%	8.8%	2.5%	2.7%	0%	8.9%
No Viewable Layer	1.4%	1%	0%	11%	0.6%	0%	0%	8.8%

1. Methodology

NS-2 is used to simulate a two node pair (server-client) communicating over UDP with a duplex unreliable link corresponding to the path between a server and a client. An ns-2 Errormodel is utilized to simulate a percentage of packet loss. Constant Bit Rate (CBR) is used to simulate transmission of the video trace, with the size of the datagram for SVC being 300 bytes, with MDC and SDC at 625 bytes. A streaming model is defined, such that the datagrams need to arrive within a specific timeframe, so as not to reduce the perceived quality of the stream. SVC and MDC require four datagrams per GOF, with SDC requiring three datagrams, thus decreasing the delivery rate of SDC.

The simulation design implemented a layer ID and GOF ID, such that each datagram is allocated to a specific layer/description in a specific GOF. This sign allows the simulation to track the datagrams between server and client. Datagrams arrive at the client in the order they are sent from the server but if datagrams arrived out of order, then the GOF ID and layer ID are utilized to determine what is decodable by the client.

Extensive simulations are run to determine the mean percentage value for viewable layers in SVC, MDC, SDC and SDC-NC. To coincide with the length of the YUV media streams, a stream size of 300 frames was chosen for both the four and six layer simulations, thus creating an averaged trace file for each streaming model, which is then analyzed to determine the highest viewable layer per frame, as a valuable metric for assessing QoP.

2. Experimental Results

First, we assess the instantaneous video quality using different schemes as highlighted in Table IV and Figure 5.

Table IV provides the percentage of viewing time at the different possible quality levels for the compared algorithms at a packet loss rate of 10%. SDC shows a better performance in comparison to SVC and MDC. The network coding component in SDC-NC result in a further improvement of approximately 6% of the highest quality viewing time, thus providing an increased QoP for the user. It is important to note that the marked change between the viewable layers of SDC can be quite high, especially as the number of layers increases, i.e. in the six layer example the transitions from highest quality to the next available quality was from layer six to layer three in SDC and from layer six to layer two in SDC-NC. Also there is a noticeable percentage of stream transition to the lowest level in SDC-NC and SDC that is not evident in MDC (as MDC combines all received description, per GOF, which increases higher layer decoding), but it is felt that the jump from higher layer to lower layer happens so infrequently that it can be easily and efficiently addressed, such as with a simple retransmission scheme.

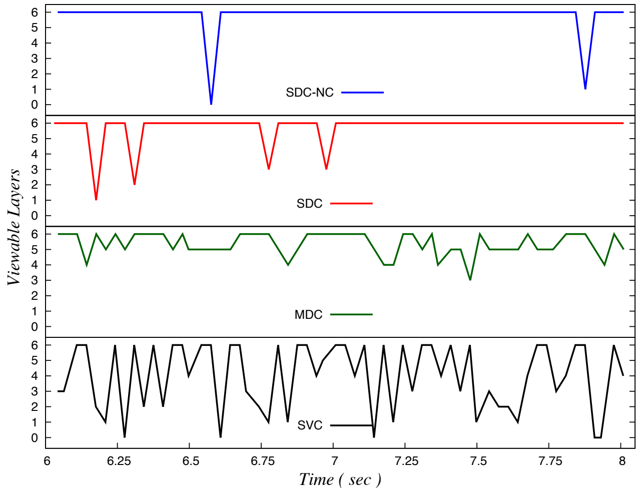


Figure 5: Two second example of movement in viewable quality in SDC, SDC-NC, MDC and SVC with six viewable layers and 10% packet loss.

Figure 5 presents a two second snapshot of the data from Table IV and illustrates the viewable layers during the snapshot interval for the considered schemes. Additionally, the figure also illustrates the frequency of transitions between different video layers. Clearly, the figure shows that SDC and SDC-NC have far fewer transitions in comparison to MDC and SVC, which has the worst performance. The figure also illustrates that SDC-NC results in higher performance gains as the medium becomes more lossy. Hence, the inclusion of the network coding component would be of great benefit to less reliable mediums such as wireless technologies.

It is important to clarify that once the highest layer (e.g. layer six) is decodable this does not mandate the client to view the highest layer but permits the client to view any layer between the highest SDC decodable layer (layer six) and penultimate decodable layer (layer three), i.e the client can view any layer between four and six.

3. Measured Impact on PSNR (peak signal-to-noise ratio)

PSNR [19], a widely used pixel-difference based objective metric, is utilized to assess the quality of YUV video streams and our results reference the Y-PSNR luminance values. The steps that we adhere to for H.264 PSNR evaluation are as follows. First, we create a multilayered H.264 video using JSVM v9.19 [18]. In our evaluation, we create crew.264 from the raw 4CIF 30fps YUV data stream crew.YUV, which consists of a crew of astronauts waving while walking down a corridor, obtained from the Leibniz Universität Hannover video library [20]. Next to determine the per frame SVC layer value, the transmission of this video over lossy links is simulated using NS-2.

Then JSVM is used to extract each layer from the original crew.264 and reconstruct a YUV stream for each layer. Note that we also up-sample lower resolution streams for PSNR calculation using the normative up-sampling JSVM method

"DownConvertStatic", which is based on a set of integer-based 4-taps filters derived from the Lanczos-3 filter. JSVM does not contain a dependable mechanism to simulate packet loss, so to simulate this loss, we developed our own code, "modPSNR.exe", based on the JSVM code base, which creates a modified YUV stream, mod_crew.YUV, from the layered YUV streams based on the SVC layer values per frame in the ns-2 generated trace file. Finally JSVM calculates the corresponding PSNR.

We also adopt a basic error concealment strategy based on frame duplication to compensate for non-decodable frames. A more advanced error concealment could be used to increase the PSNR values for the decoded streams, but is outside the scope of this paper.

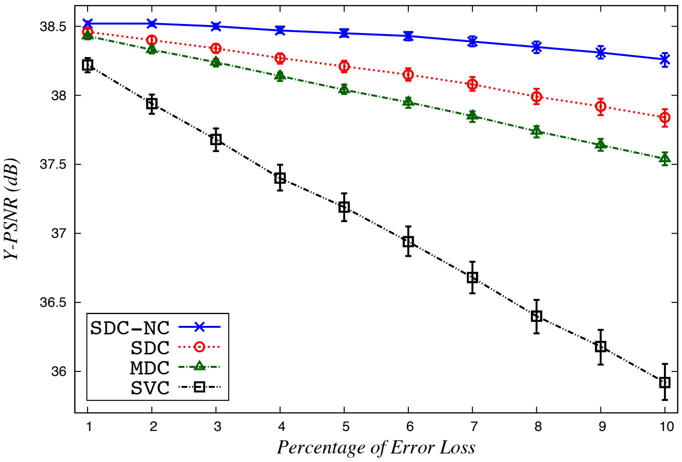


Figure 6: Mean Y-PSNR values and 95% Confidence Interval error bars for a 4 Layer scheme with incremental packet loss.

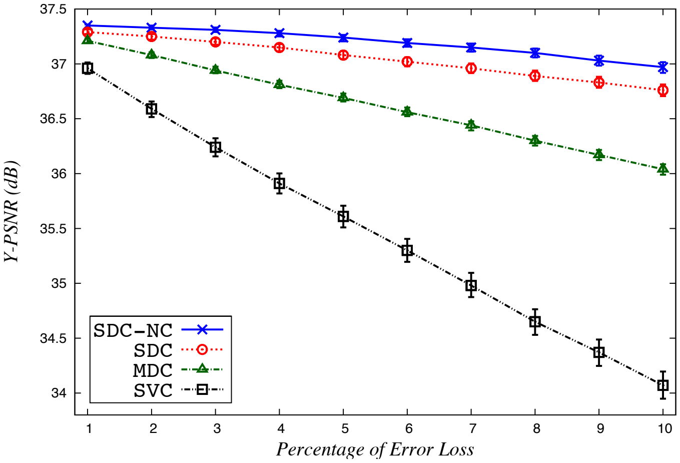


Figure 7: Mean Y-PSNR values and 95% Confidence Interval error bars for a 6 Layer scheme with incremental packet loss.

Figure 6 (4-layer) and Figure 7 (6-layer) plot the Y-PSNR values with 95% confidence interval error bars, for the adaptive mechanisms versus percentage of error loss. The figures show that SDC and SDC-NC result in a noticeable improvement in PSNR with lower bandwidth usage than MDC. This improvement is proportional to the loss rate and reaches up to a 9% increase in the average PSNR at the 10% loss rate.

VI. Conclusion

This paper presented Scalable Description Coding (SDC), a novel approach for the transmission of redefined MDC-encoded video. SDC shows a noticeable superiority in comparison to MDC in terms of bandwidth requirements, without sacrificing user-perceived quality. We believe that SDC is especially significant for mobile networks where bandwidth over-the-air and in the backhaul continue to be insufficient to satisfy the growing demand of video applications. SDC yields upwards of 25% bandwidth savings over MDC and also delivers higher quality streams for longer periods of time. Future work includes the comparative evaluation of SDC in a real media testbed, as well as determining the benefits offered by SDC to other versions of MDC [7].

VII. Acknowledgement

The authors thank the anonymous reviewers for their valuable feedback and acknowledge the support of Science Foundation Ireland (SFI) under Research Grant 10RFP/CMS2952.

References

[1] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast, 2011-2016. Available at:, "http://www.cisco.com/en/US/solutions /collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.pdf."

[2] G. Ghinea and J. Thomas, "Quality of perception: user quality of service in multimedia presentations," *IEEE Trans. on Multimedia*, vol. 7, no. 4, pp. 786–789, 2005.

[3] G. Conklin et al., "Video coding for streaming media delivery on the Internet," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, pp. 269–281, March 2001.

[4] T Schierl et al., "Mobile Video Transmission Using Scalable Video Coding," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1204 – 1217, 2007.

[5] H. Schwarz et al., "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 17, pp. 1103–1120, Sept. 2007.

[6] R Puri et al., "Forward error correction (fec) codes based multiple description coding for internet video streaming and multicast," *Signal Processing: Image Communication*, vol. 16, pp. 745–762, May 2001.

[7] V. Goyal, "Multiple description coding: Compression meets the network," *Signal Processing Magazine*, vol. 18, pp. 74–93, Sept 2002.

[8] R. Ahlswede et al., "Network information flow," *IEEE Trans. on Information Theory*, vol. 46, pp. 1204–1216, July 2000.

[9] J Heide et al., "Cooperative media streaming using adaptive network compression," *2008 Int. Symp. World of Wireless, Mobile and Multimedia Networks, WoWMo. '08*, pp. 1 – 7, June 2008.

[10] P. Chou and H. Wang, "Layered multiple description coding," *Proc. Packet Video Workshop*, 2003.

[11] L. Kondi, "A rate-distortion optimal hybrid scalable/multiple-description video codec," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '04)*, 2004.

[12] Z. Zhao et al., "Multiple description scalable coding for video streaming," *2009 10th Workshop Image Analysis for Multimedia Interactive Services. WIAMIS '09.*, pp. 21–24, 2009.

[13] T. Berkin Abanoz and A. Murat Tekalp, "SVC-based scalable multiple description video coding and optimization of encoding configuration," *Signal Processing: Image Communication*, vol. 24, no. 9, pp. 691–701, 2009.

[14] M. Halloush and H. Radha, "Practical Network Coding for scalable video in error prone networks," *Picture Coding Symposium, 2009. PCS 2009*, pp. 1–4, 2009.

[15] S. Katti et al., "XORs in the Air: Practical Wireless Network Coding," *IEEE/ACM Trans. Networking*, vol. 16, pp. 497–510, June 2008.

[16] C. Fragouli et al., "Network coding," *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 63–68, Jan. 2006.

[17] The Network Simulator - ns-2, "http://nsnam.isi.edu/nsnam/index.php."

[18] J. Reichel, H. Schwarz, and M. Wien, "Joint scalable video model 11 (JSVM 11)," *Joint Video Team, doc.*, Jul. 2007.

[19] D. Salomon, *Guide to Data Compression Methods.* Springer, 2002.

[20] Video traces for network performance evaluation. Available at:, "ftp://ftp.tnt.uni-hannover.de/pub/svc/testsequences/."