On The Performance of Scalable Video Coding in P2P Live Video Streaming

Zoran Kotevski and Pece Mitrevski
Faculty of Technical Sciences/Department of Computer Science and Engineering, Bitola, R. Macedonia
zoran.kotevski@uklo.edu.mk, pece.mitrevski@uklo.edu.mk

Abstract - The two basic concepts of scalable video coding that are widely used in P2P video streaming are: layered video coding (LVC) and multiple description coding (MDC). With the LVC coding, the video stream is divided in several sub-streams (layers), out of which the first one is the base layer, and all other layers are enhancement layers. The base layer can be decoded independently, while decoding each enhancement layer requires its predecessor layer. MDC coding splits the video stream in several sub-streams (descriptions), where each description can be independently decoded, for the price of a certain coding overhead. The main idea of both techniques is to split the video stream and distribute it over multiple network paths, in order to ensure that at least one sub-stream is received error-free. In this paper, a discrete event simulation model that compares the performance of LVC and MDC coding schemes is developed. The model assumes mesh-based P2P live video streaming system using network path diversity for each of the generated sub-streams. The results obtained imply that MDC exhibits better performance compared to LVC under the same network conditions, but only to a point of 5% introduced coding overhead. When both these techniques are compared to a single description (SD) coding, it appears that SD technique offers better performance than the other two scalable coding techniques, but the downside of SD is that the service degradation is not that graceful compared to MDC or LVC.

I. INTRODUCTION

P2P live video streaming is an intriguing paradigm that has been in active development for almost two decades. Since its beginnings in the mid 90's, large number of techniques and a variety of different approaches were implemented to improve the quality of the offered services. In this manner, two distinctive video coding techniques that are continuously used in such error prone systems are Layered Video Coding (LVC) and Multiple Description Coding (MDC), and represent specific implementations of a scalable video coding technique. In both these concepts, the video stream is split and coded into several sub-streams, which are usually sent through different paths in the P2P network. Each sub-stream contributes to one or more quality characteristics of the video content in terms of temporal, spatial and/or SNR/quality scalability. The multiple path network routing is the main reason for the stream division, because otherwise it would not be of greater use since it would not make much difference whether there is only one or more streams if the data travels through a single network path.

LVC technique divides the video stream in several sub-streams (layers). The first layer is the base layer and all other layers are enhancement layers. The base layer can be decoded independently of the presence of other layers and represents an essential level of quality. Decoding any other layer requires presence of its predecessor layer, i.e. the first enhancement layer requires the base layer, the second enhancement layer requires the first enhancement layer, and so on. The requisite of the base layer forms a very critical part of the scalable video representation which makes the systems that use LVC vulnerable to disruptions since they posses rather single point of failure. LVC technique was used at the very beginnings of P2P live video streaming, when in 1996, S. McCanne et al. [1] presented a P2P video streaming model with tree network structure that implements LVC coding technique. Even though LVC implementation dates at the appearance of P2P streaming systems, it gained higher popularity much later, thus in recent years a lot of P2P streaming protocols, such as [2, 3] use LVC and report substantial results.

MDC, on the other hand, splits the video stream in several description that can be independently decoded. In this case any description is sufficient to play the video, and any additionally received descriptions contributes to the video quality enhancement. In recent years many P2P streaming protocols, such as [4,-6], implement MDC.

If a brief comparison of the two scalable coding schemes is performed, their strengths and weaknesses can easily be inferred. The main advantage of MDC over LVC is that each description can be independently decoded, and the main advantage of LVC is that no coding overhead is produced, that is not the case with the MDC technique.

The research of J. Chakareski et al. [7] deals with performance evaluation of specific implementations of MDC and LVC for video streaming over error-prone packet switched networks. The comparison is performed using different transmission schemes and packet scheduling algorithms, and the main conclusions show that LVC performs better when rate-distortion optimized packet scheduling is implemented. There are several other research arcticles that deal with comparison of LVC and MDC, such as [8,-11], but none of them explores the stochastic aspect of the transmitted sub-streams considering the different nature of LVC and MDC as well as the introduced coding overhead when MDC is employed.

Regarding the MDC overhead, F. Fitzek et al. [12] have experimentally confirmed that the MDC offers solid improvement of video transfer over best effort networks, compared to a single description coded video, but that improvement comes with the price of generated coding overhead that generally depends on the number of generated descriptions as well as the complexity of the video content itself.

In this research we assume that each video sub-stream is sent over different network path, and we take on somewhat different approach compared to other researches that deal with performance comparisons of MDC and LVC coding schemes. Referring to the fore mentioned main strengths and weaknesses of LVC and MDC, there is a specific hypothesis suggesting that if no coding overhead was produced, MDC should perform better than LVC, since any single description is sufficient to play the video representation. Thus, we analyze the performance of both scalable video coding techniques from a pure probabilistic point of view, regardless of the network conditions and packet scheduling schemes. We also investigate the upper limit of MDC coding overhead that would still enable the MDC to perform better or at least equally to the LVC technique, if our forementioned hypothesis is correct.

II. THE DISCRETE-EVENT SIMULATION MODEL

In this section we present the Discrete-Event Simulation (DES) model for performance evaluation of LVC and MDC techniques, and briefly discuss the input parameters used in the simulations.

The considered P2P live video steaming system adopts mesh network topology where the users are organized in certain groups, and each group member communicates with all his neighbors exchanging video chunks. Since the users need to receive multiple sub-streams over different network paths, each of them becomes a member of as many groups as there are video sub-streams. Hence, the user dedicates his Upload Bandwidth (UB) equally to all the groups he has joined. In our previous research [13] we have determined the optimal range of average group sizes, thus in this research we apply an average group size of 60 peers, for each group of peers. The accounted network is asymmetric where peers have infinite download bandwidths. Concerning the peers' UB we implement UB heterogeneity defined by uniform probability distribution in the range from 100 kbps to 1000 kbps. On the basis of the research of K. Sripanidkulchai et al. [14], where it is experimentally confirmed that the arrival of new peers in P2P streaming systems follows an exponential probability distribution, we define peer arrival as a stochastic process with exponentially distributed inter-arrival times ($1/\lambda$), where λ represents the arrival rate. Further more, the research of Z. Ou et al. [15] provides strong arguments that justify the use of exponential distribution for the peer viewing (sojourn) times as well. Even-though in [14] the experiments have shown that peer viewing times follow hevy-tailed probability distribution, the research in [15] confirmed that the use of exponential, Pareto and Weibull distributions for sojourn times in P2P streaming systems exhibit little to no differences at all. Because the exponential probability distribution is the only continuous distribution that has the property of no memory, which characterized many natural phenomena, it emerges as the best choice for representing the peer sojourn times. Since the peer arrival and departure follow an exponential distribution, the number of peers that are concurrently present in a single group has a Poisson probability distribution. This defines the peer churn (the joining and leaving of peers) in a single group as a Poisson process.

We imagine the system from a viewpoint of a single peer who, when joining the system, actually joins 3 groups of peers (one group for each video layer/description), since we have defined the video stream division into 3 sub-streams. Each peer group is independent of the other two and different peers join in these 3 groups (except the single user that we base our evaluation on). This way the concept of the sub-streams transmission over multiple paths is preserved.

In this manner, the system is modeled as a Queueing Network (QN) with 3 independent sub-QNs, each having exponentially distributed arrival and service rates and infinite number of servers.

Fig. 1.presents the DES (QN) model of the P2P video streaming system, from a viewpoint of a single user.

λ_1 – Arrival rate of peers joining group 1

λ_2 – Arrival rate of peers joining group 2

λ_3 – Arrival rate of peers joining group 3

μ_1 – Service rate of peers in group 1

μ_2 – Service rate of peers in group 2

μ_3 – Service rate of peers in group 3

G_1 – Group 1 (distributing sub-stream 1)

G_2 – Group 2 (distributing sub-stream 2)

G_3 – Group 3 (distributing sub-stream 3)

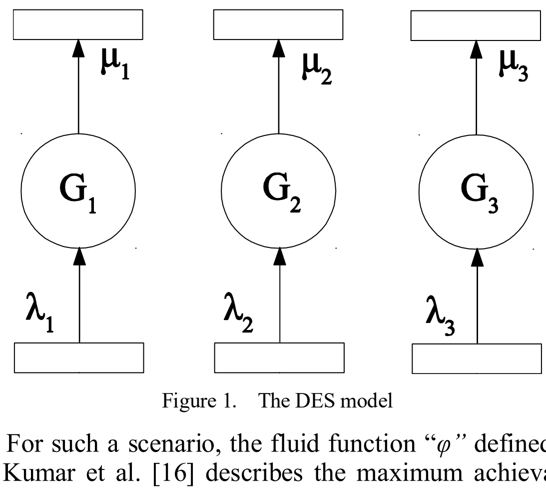

Figure 1. The DES model

For such a scenario, the fluid function "φ" defined by R. Kumar et al. [16] describes the maximum achievable rate that can be streamed to any peer in a certain group at a given time, and in this particular case is given by:

$$\varphi_i = \frac{G_{iUB}}{\#G_i} \tag{1}$$

where:
G_{iUB} – The sum of the UB of all peers in group *i*
$\#G_i$ – the number of peers in group *i*

The performance evaluations reside on the basis of the Probability for Degraded Service (PDS) of the video sub-stream distribution. PDS occurs every time when φ_i drops below the value of sub-stream rate, i.e. when:

$$\varphi_i < V_{SR}$$

where:

V_{SR}– The rate of the sub-stream, which is equal to one third of the video rate (V_R/3).

More specifically, the first performance evaluations are performed on a probability to completely receive 2 out of 3 sub-streams. Clearly, these probabilities are Boolean expressions, described in the following lines:

TABLE I. PDS OF LVC AND MDC TECHNIQUES

	PDS for 2 out of 3 sub-streams
LVC	$\varphi_1 \geq V_{SR}$ and $\varphi_2 \geq V_{SR}$
MDC	$\varphi_1 \geq V_{SR}$ and $\varphi_2 \geq V_{SR}$ or $\varphi_1 \geq V_{SR}$ and $\varphi_3 \geq V_{SR}$ or $\varphi_2 \geq V_{SR}$ and $\varphi_3 \geq V_{SR}$

In the second performance evaluations the probabilities to fully receive 3 out of 3 sub-streams, for both video coding techniques, are compared to the PDS of a Single Description (SD) video stream.

The solution to our model is provided via discrete event simulations (DES), which are performed using SimPy [17]. SimPy DES package based on standard Python programming language [18]. It is an object-oriented, process-based DES language that provides the modeler with simulation components including "Processes" for active components like customers, messages and vehicles, and "Resources" for passive components that form limited capacity congestion points like servers, checkout counters and tunnels. It also provides monitor variables to aid in gathering statistics. Random variables are provided by the standard Python random module.

III. PERFORMANCE RESULTS AND ANALYSIS

In this part we present the results of the performed analyses from which rather interesting conclusions can be obtained. Namely, the chart presented in Fig. 2 plots the probability for degraded service for a P2P video streaming system that receives at least 2 out of 3 video sub-streams. Clearly, as expected, MDC offers best performance, but only if the coding overhead is zero, which is never a case in the reality. Comparable performance of MDC technique, compared to LVC, can be expected if the MDC coding overhead is kept under 5%. For all other percentages of coding overhead higher that 5%, MDC performance degrades gradually.

The results shown in Fig. 3 present performance comparison between single description video coding, on one side, and LVC and MDC techniques for 3 out of 3 received video sub-streams, on the other. As can be seen in the charts, single description video coding offers better performance than the other two techniques for scalable video coding, and the only difference in quality is the more graceful service degradation that scalable video coding offers compared to single description coding. This conclusion strongly supports our proposal, given in [13], that the use of compression throttled scalability and a single description video should be better explored. Namely, video scalability comes in three different concepts. Spatial scalability concept bases on video stream division in several sub-streams where each sub-stream adds to the video resolution. Temporal scalability extracts different frames of the basic video stream and packs each sub-stream with fewer frames per second, depending on the number of sub-streams.

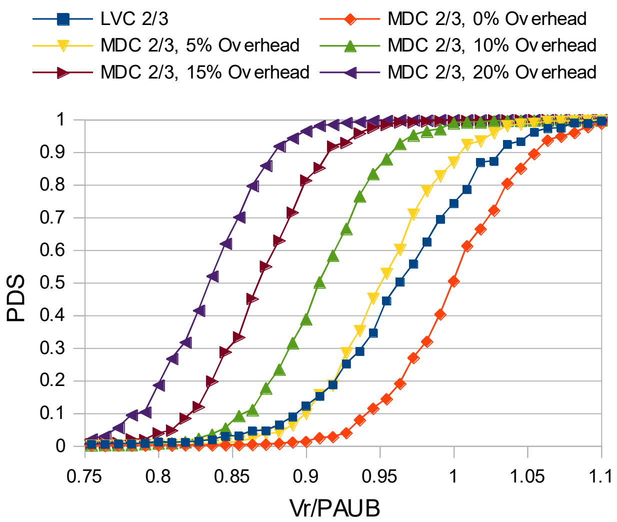

Figure 2. LVC vs MDC performance comparison for 2 out of 3 received video sub-streams

Last but not least, video scalability concept is the concept of SNR (Signal to Noise Ratio)/Quality scalability. This type of video coding regulates the compression intensity of the single description video stream, and hence it produces a single video stream with variable quality. If feedback information about the P2P network conditions is employed, it can easily be used for careful tuning of the compression intensity each and every time a change in the system occurs.

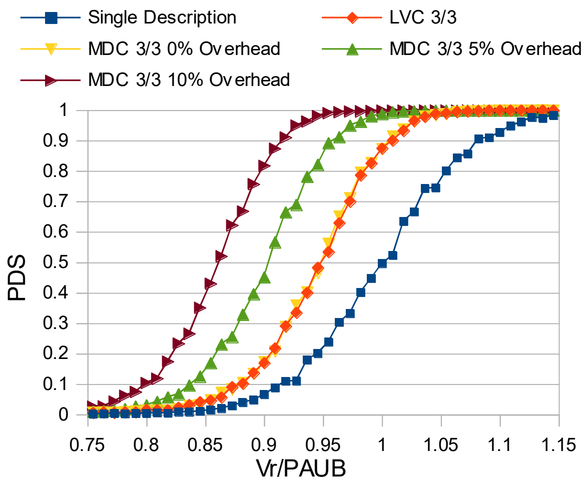

Figure 3. SD vs LVC vs MDC performance comparison for 3 out of 3 received video sub-streams

IV. CONCLUSION

In this research we have evaluated the performance of LVC and MDC techniques for use in P2P video streaming networks, where the separate sub-streams are sent over different network paths. We have also evaluated the performance of these types of scalable video coding techniques against single description video coding. These analyses concentrate on the probabilistic nature of sending multiple streams over multiple paths transmission, which plays an important role for the quality of offered services. The main conclusions imply that MDC technique performs better that LVC, but only to the point of 5% introduced coding overhead. Increasing the overhead (regardless of the nature for its generation, such as the number of sub-streams or the video scene complexity) induces fallback of the MDC performance behind LVC technique. When both techniques (MDC and LVC) are compared to a single description video coding it appears that SD technique offers better performance than the other two scalable coding techniques, but the downside of SD is that the service degradation is not that graceful as when scalable video coding is used.

As mentioned previously, our future research will be concentrated on the exploration of our proposal to implement a SNR/quality scalability to a single description coding and compare its performance to the other scalable video coding techniques implemented in the field of P2P live video streaming networks.

REFERENCES

[1] S. McCanne, V. Jacobson and M. Vetterli, "Receiver Driven Layered Multicast", ACM SIGCOMM Computer Communication Review, vol. 26, issue 4, pp. 117-130, Stanford CA, USA, 26-30 August 1996

[2] R. Rejaie and A. Ortega, "PALS: Peer to Peer Adaptive Layered Streaming", 13th International ACM Workshop on Network and Operating System Support for Digital Audio and Video, pp.153-161, 2003

[3] H. Guo, K.T. Lo, Y. Qian and J. Li, "Peer-to-Peer Live Video Distribution under Heterogeneous Bandwidth Constraints", IEEE Transactions on Parallel and Distributed Systems, vol. 20, issue 2, pp. 233-245, 2009

[4] N. Magharei and R. Rejaie, "PRIME: Peer to Peer Receiver Driven Mesh Based Streaming", Proceedings of the IEEE International Conference on Computer Communications INFOCOM, pp. 1415-1423, Anchorage, Alaska, USA, 6-12 May 2007

[5] H. Guo and K.T. Lo, "Cooperative Media Data Streaming with Scalable Video Coding", IEEE Transactions on Knowledge and Data Engineering, vol. 20, issue 9. pp. 1273-1281, 2008

[6] S. Zezza, E. Magli, G. Olmo and M. Grangetto, "Seacast: A Protocol for Peer to Peer Video Streaming Supporting Multiple Description Coding", Proceedings of the IEEE International Conference on Multimedia and Expo, pp. 1586-1587, Cancun, Mexico, 28 June-03 July 2009

[7] J. Chakareski, S. Han and B. Girod, "Layered Coding vs Multiple Descriptions for Video Streaming over Multiple Paths", Proceedings of the 11th ACM International Conference on Multimedia, pp. 422-431, Berkeley, CA, USA, 02-08 November 2003

[8] R. Singh, A. Ortega, L. Perret and W. Jiang, "Comparison of Multiple Description Coding and Layered Coding based on Network Simulations", SPIE Proceedings of Visual Communications and Image Processing, vol. 3974, pp. 929-939, San Jose, CA, USA January 2000

[9] Y. Shen, Z. Liu, S. S. Panwar, K. W. Ross, and Y. Wang, "Streaming Layered Encoded Video Using Peers", *Proceedings of IEEE International Conference on Multimedia and Expo,* Amsterdam, The Netherlands, 6-8 July 2005.

[10] S. E. Flierl, T. Sikora and P. Frossard, "Coding with Temporal Layers or Multiple Descriptions for Lossy Video Transmission", Proceedings of the 9th international conference on Visual Content Processing and Representation, pp. 137-145, Springer-Verlag, Berlin, Heidelberg, 2006

[11] Y. Zhou, and W- Y Chan, "E- model Based Comparison of Multiple Description Coding and Layered Coding in Packet Networks", European Transactions on Telecommunicationsvol. 18, issue 7 pp. 661-668, 2007

[12] F. Fitzek, B. Can, R. Prasad, and M. Katz, "Overhead and Quality Measurements for Multiple Description Coding for Video Services", Wireless Personal Multimedia Communications (WPMC), vol. 2, pp. 524-528, 2004

[13] Z. Kotevski, P. Mitrevski, "Hybrid Fluid Modeling Approach for Performance Analysis of P2P Live Video Streaming Systems", Peer-to-Peer Networking and Applications, Springer New York, 2013. doi: 10.1007/s12083-013-0205-7

[14] K. Sripanidkulchai, B. Maggs, H. Zhang, "An Analysis of Live Streaming Workloads on the Internet", 4th ACM SIGCOMM Conference on Internet Measurement, pp 41-54, 2004

[15] Z. Ou, E. Harjula, M Ylianttila, "Effects of Different Churn Models on the Performance of Structured Peer-to-Peer Networks", Proceedings of the 20th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, pp 2856-2860 , Tokyo, Japan, 13-16 September 2009

[16] R. Kumar, Y. Liu K. Ross, "Stochastic Fluid Theory for P2P Streaming Systems", Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM), pp 919-927, Anchorage, Alaska, USA, 6-12 May 2007

[17] Simpy (URL: http://simpy.sourceforge.net/)

[18] Python (URL: http://www.python.org/)