Network Coding Efficiency in Live Video Streaming over Peer-to-Peer Mesh Networks

Behrang Barekatain
Faculty of Computer Science & Information System
Universiti Teknologi Malaysia
Johor Baharu, Malaysia
Bbehrang3@Live.UTM.my

Mohd Aizaini bin Maarof
Faculty of Computer Science & Information System
Universiti Teknologi Malaysia
Johor Baharu, Malaysia
Aizani@UTM.my

Abstract- **In the recent years, video and audio streaming have been very popular in the Internet and accounts for excessive traffic not only in the Internet, but also in many private networks. Transferring this amount of traffic needs both reliable and high performance infrastructure as well as efficient approaches to increase bandwidth utilization. Real-time playback deadline and robustness under high peer churning[1] are two important constraints in live video streaming, which can reduce the video quality. Moreover, reliable, flexible and scalable network infrastructures need to be considered, while there are many mobile users who move in the network repeatedly. Wireless Mesh Networks (WMNs) provide this requirement infrastructure. However, routing and forwarding high traffic streams in underlying networks impose high expense due to high-priced devices, routing and forwarding protocols. Overlay network which builds on top of the underlying networks can address these problems. P2P (Peer-to-Peer) live video streaming, WMNs and Network Coding (NC) are three important terms in this paper, which will be described and we show how NC can improve live video streaming quality over WMNs with time-varying channels and P2P networks. Our results show that using NC can reduce jitter delay as well as increase frame diversity and localization, bandwidth utilization and scalability of the network.**

Keywords: Network Coding (NC), Live Video, P2P, WMN

I. INTRODUCTION

In near future, one of the most networks expensive applications will be multimedia distribution systems, especially those which are live and interactive and need to have acceptable quality. Video dissemination is the most important application which not only needs large bandwidth, but also high performance protocols to delivery frames to receivers. Video streams are classified into two categories: Live and Video-on-Demand (VoD). Actually, there are some technical differences between sending live and VoD streams. In live video streaming a synchronized video playbacks on all clients and they have to watch the same frames in an instant. On the other hand, in VoD delivery, users have opportunity to watch whatever video streams whenever they want [1]. In the other words, the playbacks of the same video stream on different clients are not synchronized. As a result, VoD provides more flexibility in comparison with live video streaming. YouTube and on-demand IPTV (Internet Protocol Television) are some good examples of VoD-based systems. Albeit large bandwidth, high performance delivery protocols and some special techniques such as buffering have great influences on video quality, employing good compression methods in the source side can relieve us from these constraints as far as possible. What can be inferred from [2] is that H.264/AVC (Advance Video Coding) is the best video compression standard which has been considered in the recent years.

In H.264 standard, depending on different profiles[2], different order of frames in a GOP[3] can be encoded by the encoder. In fact, each GOP can consist of I, B and P frames. I frame (Intra frame) is the most important frame, which can be used by P or B as a reference and decoded independently. P frame (Predictive inter frame) can use one of the previous I or P frames, while a B frame (Bi-predictive frame) uses both last and next I or P frames as its reference frame [2]. A reference frame can be used by a decoder to recover damaged frames. Figure 1 depicts a classical GOP. In fact, differential frames (B and P) include fewer bits in comparison with I frame. Thus, higher compression rate is achieved by this standard with same quality in comparison with other video compression standards. As a result, this standard can save much more bandwidth than that of others, which is very important in live video streaming, because the source can send more frames at time unit that leads to increase the throughput of the system, more bandwidth utilization and better video quality. Channels in a network are classified into two groups, Constant Bit Rate (CBR) and Variable Bit Rate (VBR) channels.

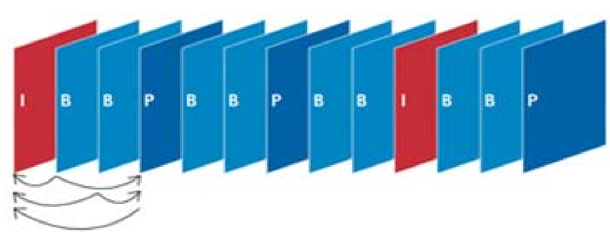


Figure 1. Classical GOP

[1] Peer joining and leaving

[2] H.264 includes 7 profiles and 11 levels

[3] Group Of Picture

The rest of this paper is organized as follows; the evolution of interactive and non-interactive applications and the comparison between CBR and VBR channels are in sections II and III, respectively. In Section IV, different systems in peer-to-peer streaming will be classified. Section V summarizes WMNs as a suitable infrastructure for mobile users who are watching live video streams. In section VI we will explain NC and the advantages of using it in live video streaming over P2P networks and WMNs. Moreover, we bring up future works and remain issues in sections VII. Finally, section VIII concludes the paper.

II. INTERACTIVE vs. NON-INTERACTIVE APPLICATIONS

In interactive applications, deadline is an important constraint in data delivery; otherwise data will not be useful; because of these types of applications are real-time. For example, in live video streaming if a frame arrives too late at the receiver, it will be useless. Imagine that a client received the frame number 23 and after that the frame number 8 arrives. It is quite clear that this frame is useless, because the client is receiving real-time video stream and cannot play backward. On the other hand, non-interactive applications have looser latency constraints [3], for example many seconds or potentially even minutes. A lecture video file is an example of non-interactive application. It is necessary to pay attention that interactive applications require real-time encoding. On the other hand, although non-interactive applications require non-real-time encoding, they may also use real-time method.

III. CBR vs. VBR CHANNELS

What is important in video streaming over time-varying channels is quality. In some channels, CBR (Constant Bit Rate) can be useful. For example, in ISDN we do not need variable bit rate, because of the nature of the channels. However, in many cases such as wireless communications, we cannot encode the video stream with same bit rate during the transmission; because of delay, free bandwidth and some other parameters are not clearly specified before and during the transmission. Thus, we have to encode the video stream with the VBR (Variable Bit Rate) technique to achieve desirable quality in the receiver. Figure 2 shows video quality providing by these two techniques. In figure 2(a), the channel status is time-varying and video stream in encoded by the VBR technique. Consequently, PSNR (Packet Signal-to-Noise Ratio) remains constant, which means continual video quality in the receiver. On the other hand, figure 2(b) depicts a constant bit rate channel which its rate is less than maximum video stream rate. As a result, PSNR is not constant during the transmission and this leads to variable video quality in receivers. These two figures show that VBR technique can be more useful in wireless channels, which suffer from time-varying conditions.

IV. PEER-TO-PEER STREAMING SYSTEMS

Peer-to-Peer systems are the most important systems for disseminating data over the Internet such as file sharing[4] and video streaming. For example, broadcasting of live video streaming over some distributed applications such as PPLive [4], CoolStreaming [5] and SopCast [6] has been very popular in the recent years.

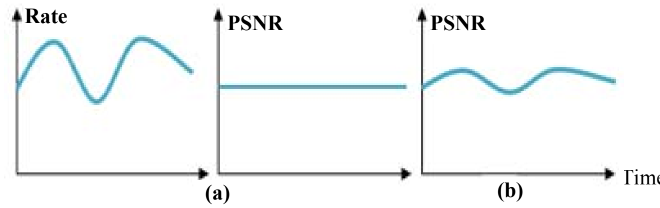


Figure 2. (a) VBR channel (b) CBR channel

BitTorrent [7] and IPTV [8] are other two examples of P2P systems, which are using for file sharing and VoD streaming, respectively. P2P systems are classified into two classes: Tree-based and Mesh-based systems. Peers can join to each other through one of these two approaches and make a virtual network over underlying network, named overlay network. An overlay network is established in Application layer without any change to underlying network. Tree-based approach is divided into Single-Tree and Multi-Tree structures such that peers form a tree and each peer has only one parent, while many children are formed under it. All of these peers upload the received chunks to their children except the leaves. In fact, no leave can participate in uploading video stream and each of them only playbacks received frames, which result in lower overall throughput.

Moreover, if a peer leaves or joins the network (peer churning), it is necessary to reconstruct the tree, which impose onerous expense to the network. To address the first problem, Multi-Tree approach such as SplitStream [16] is introduced. In Multi-tree approach, each peer is an internal node in only one subtree and as a leave in all other subtrees. As a result, all peers can participate in video streaming and upload the received frames to others [9]. Figure 3 shows a Multi-Tree structure. For example, peer 1 is both an internal and leave node in the left and the right subtree, respectively. Both NICE [13] and ZIGZAG [14] are two implemented overlay distribution networks based on multicast-tree.

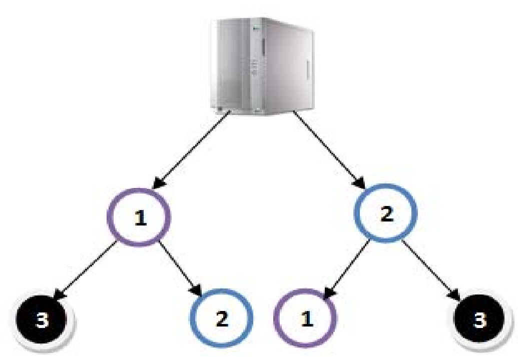


Figure 3. Multi-Tree structure

Although Tree-Based structure is a simple and cost-effective approach, low throughput, high delay, especially in the deeper levels and unacceptable reconstruction overhead due to peer churning are the main drawbacks of it [9][10]. Mesh-based approach is the best method to cope with these

[4] Such as BitTorrent [29] and Gnutella 2 [30]

problems. In mesh–based system, a peer can connect to other peers and upload/download to/from them. Mesh-based systems are more robustness in high peer churning; because of each peer has enough redundant links to others. Moreover, in a mesh-based system, the throughput will be increased due to each peer can pull required frames from or push the new frames to its neighbor immediately. Pull and Push are two main methods to deliver packets in the network. In pull-based methods, each peer can pull needed packets from others based on buffer map technique such as DHT (Distributed Hash Table), while in push-based methods a peer sends packets to other peers base on one of the possible techniques such as LU/DP (Latest Useful chunk/most Deprived Peer) [11]. Push-based methods are suitable for upload-constrained systems, where there are few sources and many receivers. Contrary to push-based system, pull-based systems are useful when there are many sources, but few receivers. It means that pull-based systems are suitable for download-constrained networks. Hybrid method is a new solution consists of both pull and push approaches to provide better performance in data exchanging.

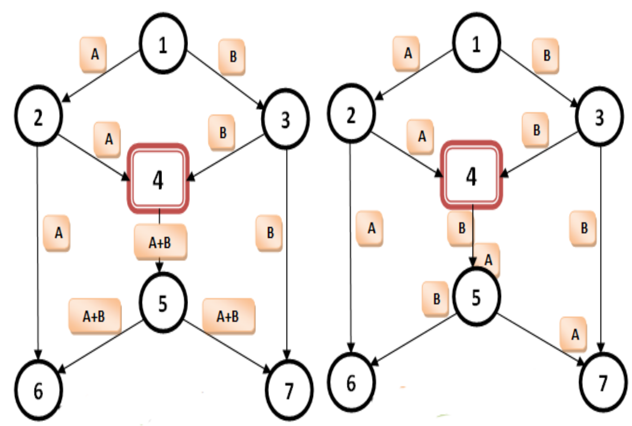
In [12] a perfect practical experiment is done on pure pull and push-pull methods and the results indicate that push-pull-based (Hybrid) method provide higher average delivery ratio in comparison with pure pull-based approaches. In a Tree-Based approach, no receiver gets duplicate packet, because of structured connections between peers. On the other hand, albeit peers in a Mesh-based approach may receive duplicate frames, they can enjoy better video quality, where each peer can use its upload/download bandwidth more effectively. CoolStreaming [15] is one the famous implemented networks, which uses Mesh-based approach. Finally, it is necessary to mention that peers in a mesh-based system share a part of their resources such as bandwidth, memory and CPU power with others.

V. WIRELESS MESH NETWORKS (WMNs)

In the recent years, the benefits of using WMNs in data streaming is considered by many companies. High reliability and robustness due to redundant links, self-organization, self–configuration, low cost scalability, easy in maintenance, nonexistence point-of-failure and reliable coverage by employing multi-hop technique [17] [18] are the most important advantages of using WMNs. Moreover, the integration of different types of wireless networks helps clients to have access to other network services through the use of WMN structure. On the other hand, WMNs like WLANs suffer from some problems such as Time-Varying channel conditions. In addition, more secure algorithm are needed to implement for having a better encryption and data distribution in WMNs. Available aggregated capacity of a WMN can be affected by some factors such as network topology and architecture, load on each node, number of stationary and mobile users, input/output degree of each node, transmission power level and the pattern of the current traffic in the network. All of these factors need to be considered, while we use WMN as our selected infrastructure.

VI. NETWORK CODING: THE THEORY AND APPLICATIONS

Because of data independency, a source can divide the stream into many smaller pieces[5] and send each of them through distinct paths to the destination. In fact, different packets traverse different paths to the destination. Consequently, in live video streaming an intermediate node can create coded frames by combining these distinct frames and the destination extract required frames from them. According to Moore's low that indicates computational processing is become cheaper and therefore the bottleneck has shifted to network bandwidth, network coding which utilizes fair computational power can be used in order to increase throughput, network resource utilization, security on links and resilience to link failure as well as decreasing the absolute delay, especially in live video streaming [19][20][21]. Network coding was first introduced by R. W. Yeung and Z. Zhang as an alternative to routing in 1999. Figure 4 depicts a sample network with and without network coding. In figure 4(b), peers 4 and 5 have to send two bits (A and B) to both 6 and 7 in two time slots. By using network coding, peer 4 can combine[6] these two bits and sends them as one bit to peer 5. Peers 6 and 7 can extract B and A from A+B, respectively. Apparently, peer 4 sends one bit in one time slot instead of two bits in two time slots which means higher bandwidth utilization.



(a) (b)

Figure 4. (a) With Network Coding (b) Without using Network Coding

This figure shows that network coding can improve the delay efficiency in live video streaming. As a result, using network coding in P2P systems can improve the throughput, robustness and efficiency as well as video quality of the video stream, because of sending one chunk equals to send at least two chunks, which are coded in it. Moreover, when a peer receives a chunk which is coded by network coding, the local availability of that frame in the peer will be increased. The required time for searching a frame, startup and end-to-end delay will be decreased due to more local availability of frames. Each link in Figure 5 supports 1bps bandwidth and the propagation delay of all links is α. The source node 1 starts

[5] Chunks/Frames/Packets

[6] XOR operation

sending A and B at T_0. According to tables I and II, peer X will receive the required bit (B) in α second earlier than that of without using network coding, which means lower jitter delay and better video quality.

Delay plays an important role in providing smooth video quality in each peer. In the other words, the higher delay, the lower quality. The best approach to cope with this problem is to increase data localization. As a result, peers can find the required frames in their buffer instead of pulling them from their neighbors. Figure 6 shows how network coding can improve data localization. In this example, a peer needs to have all three frames to provide smooth video quality. Peer 4 requests C from peer 3. Because peer 3 does not have this frame in its buffer, it sends the request message to peers 1 and 2. Peer 1 can send one of C or P (coded frame) in one time slot such that P=A+B+C. In the first case, although peers 2, 3 and 4 will receive C and peer 4 have all required frames, peers 2 and 3 need to receive B and A to provide smooth video quality, respectively.

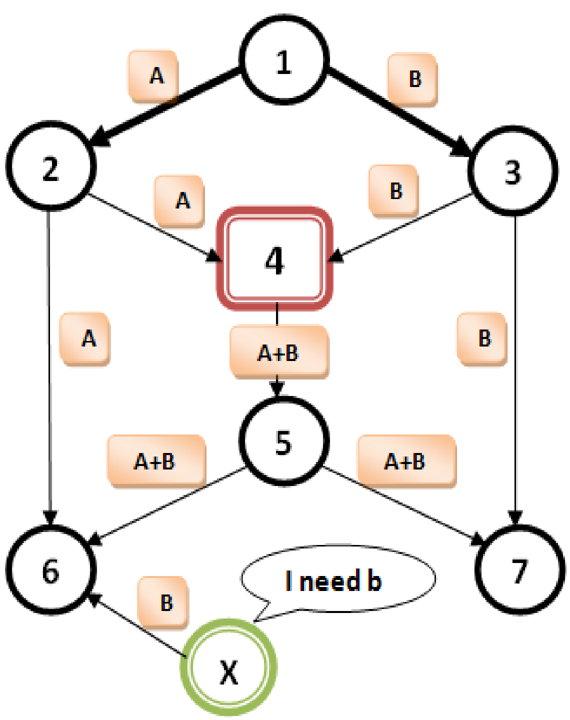


Figure 5. Butterfly network with network coding

Table I. Time progress table with network coding

Step	Time	Source	Destination	Bit
1	T_0	1	2 3	A B
2	$T_0+\alpha$	2 3	4 and 6 4 and 7	A B
3	$T_0+2\alpha$	4	5	A+B
4	$T_0+3\alpha$	5	6 and 7	A+B

Table II. Time progress table without network coding

Step	Time	Source	Destination	Bit
1	T_0	1	2 3	A B
2	$T_0+\alpha$	2 3	4 and 6 4 and 7	A B
3	$T_0+2\alpha$	4	5	A
4	$T_0+3\alpha$	4 5	5 6 and 7	B A
5	$T_0+4\alpha$	5	6 and 7	B

In the other words, if they receive another frame C from a peer, they will discard it because of duplication. In the second case, peer 4 can extract C from P and other peers can save P in their buffers. Certainly, peer 2(3) will have all of three frames, if it receives B (A) or C and this can increase the probability of having all frames in their buffers. Interestingly, in the second case if peers 2 and 3 receive the duplicate frame C, they can use it to extract their required frames, B and A respectively.

Furthermore, if a peer can offer new data to others in most of the time, it means its neighbors can pull their required frames with higher probability. This refers to diversity concept. As a result, in live video streaming, the higher diversity means the higher system overall throughput. In fact, peers can pull required frames from their neighbors in the minimum possible time, which leads to the lower jitter delay in the playback time.

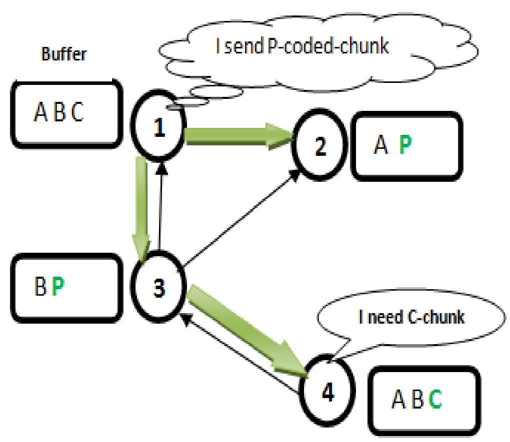


Figure 6. Network coding can increase data locality

Figure 7 depicts that network coding can improve frame diversity in a wireless network. Peer 2 requests C from its neighbors. Because of using network coding approach, it receives coded frames P and P' from two of its neighbors. By employing XOR operation on frames P' and A, the required C will be available. Moreover, D can be achieved from P, A, B and C frames. Then, peer 1 sends a request for D to its neighbors. None of its neighbors have requested this frame before, so they have not it in their buffers, except peer 2 which received it without any explicit request. Finally, peer 2 will send D to peer 1 in the minimum possible delay, which leads to not only lower jitter delay, but also better video quality in peer 1. Definitely, if the neighbors of peer 2 were sent only C, peer 2 had to wait a longer time to receive the required frame.

What can be inferred from figure 8 is that network coding can improve energy consumption of peers in wireless networks such as WMNs. In the other words, peer 2 has to send two distinct frames A and B in the absence of network coding approach. On the other hand, combining the received frames A and B before sending them to peers 1 and 3 not only decrease the consumption energy in peer 2, but also let its neighbors to receive the required frames in less time, which results in better video quality in peers 2 and 3.

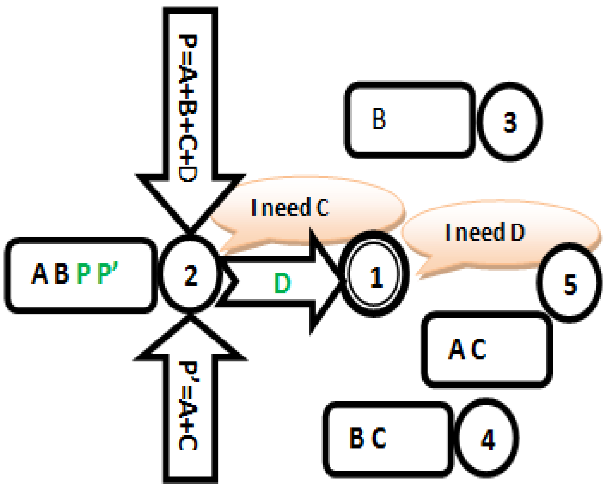


Figure 7. Network Coding can improve frame diversity

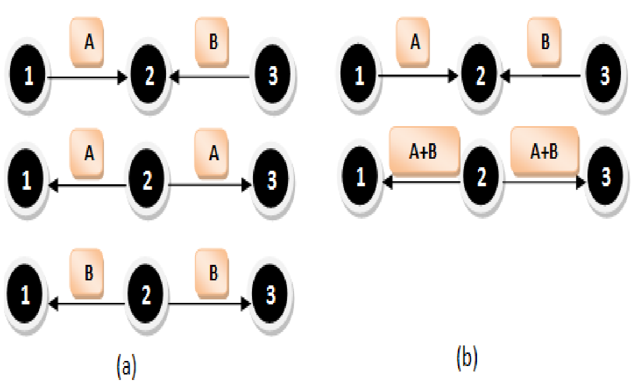


Figure 8. Exchange frames in wireless networks (a) without Network Coding and (b) With Network Coding

Another important issue in WMNs is that all nodes can move from a point to another point, repeatedly. Thus, each peer will visit many other peers, while it interchanges in the network. Thus, it needs to pull/push frames from/to different buffers of different peers. In figure 9, peer 1 is moved from area 1 to area 2. Although peer 2 has frame K in its buffer, this frame is requested by peer 1 after it leaves area 1 and enters area 2. In area 2, no explicit request has been done for K yet. However, peer 3 received it before due to an explicit request for frame C, which resulted in receiving frame P=A+C+K. Consequently, peer 3 sends frame P to peer 1, so it will have K.

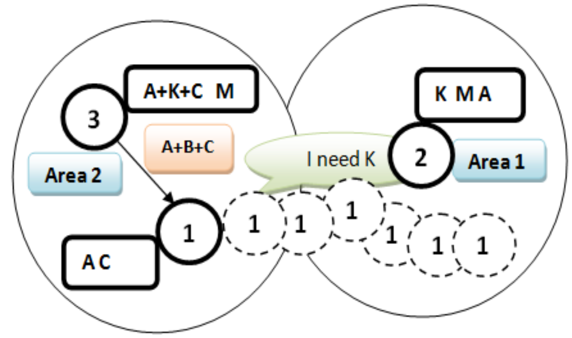


Figure 9. Network Coding efficiency for mobile peers

VII. FUTURE WORKS AND REMAIN ISSUES

In the last decade, many researchers tried to improve the performance of both Tree and Mesh peer-to-peer systems by introducing the new algorithm or enhancing the current methods. MDC is introduced to enhance error resilience over time-varying channels by creating several independent descriptions of a video stream, which can contribute to spatial or temporal resolution and SNR (Signal-to-Noise Ratio) characteristics of the video stream. The more received description, the better video quality will be provided [22]. Utilization of network coding and SVC (Scalable Video Coding) in live video streaming over peer-to-peer networks is discussed in [27]. In [20], the results confirmed the advantages of using the network coding approach in peer-to-peer live streaming, where peers suffer from enough upload bandwidth. In this research, Gauss-Jordan elimination, a progressive decoding method, is used to cope with delay constraint in decoding/encoding of received blocks of a segment in each peer.

In SPANC, a new proposed push-based scheme, video packets are grouped into many substreams and each peer assigns a scheduler to its parents according to SA (Substream Assignment) problem, before they push these substreams to it. Moreover, FRNC (Fast Recovery Network Coding) problem, which is assigned required network coding packets to parents to recover damaged packets in the minimum possible time, is formulated by each peer [19]. A new scheme for peer-to-peer live streaming is introduced in [28] such that unlike other schemes based on network coding, peers can take decisions of data block propagation only base on their local information. In fact, no centralized knowledge of the whole network topology is needed in this scheme. High performance packet exchange algorithm has high effects on live video streaming over peer-to-peer networks. Thus, it is necessary to do research on the scheduling algorithm. For this reason, many Push-Pull based approaches are discussed in [23][24][25] and [26].

Although interesting results have been achieved by these proposed algorithms, future works have to answer numerous ambiguous questions. Proper selection of coefficients has enormous effects on the performance of network coding, especially in live video streaming. In addition, network coding imposes more duplicated packets, if it combines them blindly without pay attention to what neighbor peers are located for in the network. Besides, many proposed solutions remain fairly theoretical, while very few real implementations of network coding algorithms exist. Linear [31], Practical [32] and Random [33] network coding are three important methods, which try to improve network coding efficiency in live video streaming over P2P networks and WMNs with high peer churning, time-varying network channels and a great number of peers. According to our knowledge, there is no perfect investigation on network resource usage, while network coding is employed for multimedia streaming. Actually, long channel and start-up delays, required video quality in case of few peers in an overlay network and lower video quality in comparison with TV services are some important remain limitations in peer-to-peer streaming which need to be considered. Finally, there is no attractive solution for ISPs (Internet Service Providers) to collaborate with those companies which provide peer-to-peer streaming by establishing an overlay networks over the Internet.

In fact, peers use the upload/download bandwidth of ISPs without making any commercial profit for them. All in all, it is necessary to perform much more investigations on network coding efficiency in live and VoD streaming over wireless and P2P mesh networks, because of the extraordinary popularity of video traffics in future networks as well as the necessity of introducing or improving new effective techniques such as network coding to cope with the consequent problems.

VIII. CONCLUSION

Video streaming is going to be the most important streams in the Internet. However, it imposes a large amount of traffic over limited bandwidth in both the Internet and large wide area networks. Although some useful compression methods such as H.264/AVC improve bandwidth utilization, new effective techniques are necessary to provide smooth video quality. In the recent years, network coding has shown that it has enough ability to provide lower delay as well as higher bandwidth utilization, which both of them leads to better video quality in the receiver systems. Moreover, network coding reduces the undesirable effects of time-varying channels, peer churning, link failures and mobile users in live and VoD streaming over P2P and wireless mesh networks. Because of attractiveness of wireless networks, peer-to-peer wireless mesh networks can provide stable infrastructure where peers leave and join the network repeatedly.

All in all, besides high data localization and packet diversity, other benefits such as lower jitter and end-to-end delay as well as less consumption energy for broadcasting live video streaming, make network coding as the best approach to provide smooth video quality, especially in peer-to-peer and wireless mesh networks. Finally, in our opinion, much more investigations must be done on remained issues of network coding efficiency in future work to cope with the current limitations and unanswered questions such as how network coding can make better connections with many other diverse areas, for example, load balancing, security and distributed storage.

ACKNOWLEDMENT

This research is supported by Universiti Teknologi Malaysia (UTM) using **Science Fund (E-Science)** from **MOSTI** with vote number **7928**.

REFERENCES

[1] C. Chen. “Peer-to-Peer Live Video Streaming Services: Past, Current, and the Future,” in CS 219 *Peer-to-Peer Networks with Mobile Applications* Winter 2008.

[2] Iain E. Richardson. “The H.264 Advance Video Compression Standard,” , second edition, 2010, published by Wiley

[3] John G. Apostolopoulos, Wai- tian Tan, Susie J. Wee. “Video Streaming: Concepts, Algorithms, and systems**,”** in HP Laboratories Palo Alto, 2002.

[4] Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross. “Insights into PPLive: A measurement study of a large-scale P2P IPTV system,” In Proc. of IPTV Workshop, International World Wide Web Conference, 2006.

[5] Zhang, J. Liu, B. Li, and T. Yum. “Coolstreaming/donet : A data-driven overlay network for Peer-to-Peer live media streaming,” INFOCOM 2005, IEEE, Vol. 3, pages 2102-2111, 13-17 March 2005.

[6] Sopcast, www.sopcast.com.

[7] Dongyu Qiu, R. Srikant. “Modeling and performance analysis of BitTorrent-like peer-to-peer network,” ACM SIGCOMM Computer Communication Review Volume 34 Issue 4, October 2004.

[8] X. Hei, C. Liang, J. Liang, Y. Liu, and K. Ross. “A measurement study of a large-scale P2P IPTV system,” In IEEE Transactions on Multimedia, 2007.

[9] Chu, Y., Rao, S. and Zhang, “A Case For End System Multicast. In: Proc. Int. Conference on Measurement and Modeling of Computer Systems,” SIGMETRICS, pp. 1–12 (2000).

[10] Jannotti, J., Gifford, D.K., Johnson, K.L., Kaashoek, M.F., O’Toole, J.W., “Overcast: Reliable Multicasting with an Overlay Network,” In: Proc. 4th Symposium on Operating System Design and Implementation (OSDI) (2000).

[11] Thomas Bonald, Laurent Massoulié, Fabien Mathieu, Diego Perino, Andrew Twigg, “Epidemic Live Streaming: Optimal Performance Trade-Offs,” ACM New York, NY, USA -2008.

[12] Meng Zhang, Jian-Guang Luo, Li Zhao, and Shi-Qiang Yang. “A Peer-to-Peer Network for Live Media Streaming – Using a Push-Pull Approach” ACM Multimedia- 2005.

[13] S. Banerjee, B. Bhattacharjee and C. Kommareddy, “*Scalable application layer multicast*,” in Proceedings of ACM SIGCOMM, 2002.

[14] D. A. Tran, K. A. Hua and T. Do, *ZIGZAG: an efficient peer-to-peer scheme for media streaming*,” in Proceedings of IEEE INFOCOM, 2003.

[15] X. Zhang, J.C. Liu, B. Li and P. Yum, “CoolStreaming/DONet: A datadriven overlay network for efficient live media streaming,” In Proceedings of IEEE INFOCOM, 2005.

[16] M. castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron, A. Singh, “SplitStream: High-Bandwidth Multicast in Cooperative Environment,” ACM, 2003.

[17] N. Salta, R. Morla and M. Ricardo, “Improving P2P Video Streaming in Wireless Mesh Networks,” Ad Hoc Networking Workshop (Med-Hoc-Net), 2010 The 9th IFIP Annual Mediterranean, IEEE, pages 1 – 8, 23-25 June 2010.

[18] X. Zhu and B. Girod, “Video Streaming Over Wireless Networks,” Information Systems Laboratory, Stanford University, Stanford, 2008.

[19] K. Chan, S. Chan and A. begen,” SPANC: Optimizing Scheduling Delay for Peer-to-Peer Live Streaming,” IEEE, Vol. 12, page 743-753, 2010.

[20] M. Wang and B. Li, “Lava: A Reality Check of Network Coding in Peer-to-Peer Live Streaming,” INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE, pages 1082 – 1090, 6-12 May 2007.

[21] Y. Liu and Y. Peng, “Network Coding for Peer-to-Peer Live Media Streaming,”, IEEE, 2006.

[22] Y. Liu, “On the minimum delay peer-to-peer video streaming: How realtime can it be?,” in *Proc. 15th Int. Conf. Multimedia*, 2007, pp. 127–136.

[23] B. Li, S. Xie, G. Y. Keung, J. Liu, I. Stoica, H. Zhang, and X. Zhang, “An empirical study of the coolstreaming+ system,” *IEEE J. Sel. Areas Commun.*, vol. 25, no. 9, pp. 1627–1639, Dec. 2007.

[24] G. Zheng, S.-H. G. Chan, X. Luo, and A. C. Begen, “Pattern-push: A low-delay mesh-push scheduling for live peer-to-peer streaming,” in *Proc. IEEE Int. Conf. Multimedia and Expo (ICME)*, New York, 28 Jun.–3 Jul. 2009, pp. 1158–1161.

[25] M. Zhang, Q. Zhang, L. Sun, and S.Yang, “Understanding the power of pull-based streaming protocol: Can we do better?,” *IEEE J. Sel. Areas Commun.*, vol. 25, no. 9, pp. 1678–1694, Dec. 2007.

[26] M. Zhang, J. Luo, L. Zhao, and S. Yang, “A Peer-to-Peer Network for Live Media Streaming – Using a Push-Pull Approach”, MULTIMEDIA '05 Proceedings of the 13th annual ACM international conference on Multimedia, ACM New York, NY, USA, 2005.

[27] S. Mirshokraie and M. Hefeeda, “ Live Peer-to-Peer Streaming with scalable Video Coding and Network Coding”, MMSys’10 10, February 22-23, 2010, Phoenix, Arizona, USA, ACM.

[28] Y. Liu and Y. Peng, “Network Coding for Peer-to-Peer Live Media Streaming”, Fifth International Conference on Grid and Cooperative Computing (GCC’06), 2006 IEEE.

[29] www.BitTorrent.com

[30] M. Portmann, P. Sokavatana, S. Ardon, and A. Seneviratne, “The Cost of Peer Discovery and Searching in the Gnutella Peer-to-peer File Sharing Protocol, Ninth IEEE International Conference on Networks (ICON'01), Bangkok, Thailand October 10-October 12.

[31] S. Robert Li and W. Yeung, “Linear Network Coding”, IEEE Vol. 49, No.2, February 2003.

[32] P P. Chou, Y. Wu, and K. Jain, “Practical Network Coding,” in Proc. Of Allerton Conference on Communication, Control, and Computing, October 2003.

[33] M. Wang and B. Li, “R2: Random Push with Random Network Coding in Live Peer-to-Peer Streaming”, IEEE Vol. 25, Pages 1655-1666, 12 Dec. 2007