A Scalable Peer-to-Peer IPTV System

Meng-Ting Lu[ac], Hung Nien[ac], Jui-Chieh Wu[b], Kuan-Jen Peng[ac],
Polly Huang[ade], Jason J. Yao[ac], Chih-Chun Lai[f] and Homer H. Chen[acd†]
Department of Electrical Engineering[a]
Department of Computer Science and Information Engineering[b]
Graduate Institute of Communication Engineering[c]
Graduate Institute of Networking and Multimedia[d]
National Taiwan University[e]
Institute for Information Industry[f]
[†]E-mail: homer@cc.ee.ntu.edu.tw

Abstract—**HotStreaming is an overlay peer-to-peer (P2P) based IPTV system. It integrates the innovations in both overlay networking and video coding for optimal user experience. The HotStreaming system is composed of three key components: partnership formation, data request scheduling and multiple description coding (MDC). In the partnership formation component, we propose two policies to reduce the time of disconnection and the number of isolated peers. In the MDC component, we adopt MDC with spatial-temporal hybrid interpolation (MDC-STHI), which makes the peers in a P2P network to adjust the streaming traffic according to their bandwidth limitation and capability of devices. The experimental results show that the HotStreaming system improves the video quality over a lossy and dynamic networking environment.**

Keywords-component; IPTV; video broadcast; video streaming; error recovery; overlay peer-to-peer network; multiple description coding

I. INTRODUCTION

With the rapid advances in multimedia entertainment, broadband network, and semiconductor manufacturing technologies, high-speed multimedia communication is coming of age. However, the key to a successful deployment of IPTV services lies in the quality of services (QoS). Our objective in this work is to develop a scalable networking and video coding platform for quality video broadcasting over the Internet.

The distribution of video streams over IP is a quasi-reliable multicast problem in nature. Solutions at the network layer have long existed. Multicast routing protocols such as DVMRP [1] and PIM-SM [2] are IETF RFCs and widely implemented in commercial routers. Reliable multicast solutions at the network layer such as SRM [3] have also been proposed early in 1997. The problem with the network-layer solutions is deployment. All routers in between the multiple senders and receivers need to run the necessary mechanisms to work.

HotStreaming, the proposed IPTV system, takes 1) the overlay peer-to-peer networking approach, with which the system can exploit the resources, such as storage and bandwidth, available among the peer users to minimize the need of server infrastructure, and 2) the MDC approach, with which heterogeneous peers can flexibly subscribe to streams of different resolutions and enjoy the best quality their available resources permit.

To achieve scalability through adjustment in the application layer, several designs were recently proposed, including Application-level Multicast [4], ZigZag [5] and CoolStreaming [6]. All of them exploit the deployment flexibility at the application level. Our HotStreaming system is closest in spirit to CoolStreaming, but different in the design of the partnership formation mechanism and in the use of video coding scheme.

MDC is the component that impacts the user perception of the video. Among so many P2P streaming systems, only a small number of them apply MDC. Padmanabhan et al. [8] introduced data redundancy to CoopNet using MDC. Zink et al. [9] discussed the feasibility of MDC for P2P networks and compared it to hierarchically layered encoded video. Khan et al. [10] compared the performance of MDC over P2P networks to that of Content Delivery Networks (CDN). In the above three papers, MDC proved to be effective over P2P networks. However, conventional MDC schemes are too complex and usually use non-standard video codecs, so they are seldom used in practice. Multiple state video coding, proposed by Apostolopoulos [11][12], as a form of MDC, is easy to implement, and may still use standard codecs as basis. Zhang et al. [13] also discussed the error concealment method of multiple state video coding when parts of the descriptions are lost. On the basis of the multiple state video coding, our proposed MDC with Spatial-Temporal Hybrid Interpolation (MDC-STHI) [14] adds two streams of lower-resolution pictures to enhance the scalability and error resilience. In order to improve the video quality when a description stream is missing, we introduce a spatial-temporal hybrid interpolation scheme that exploits both spatial and temporal information at the source. This feature allows the decoder to conceal the lost frames more effectively and provide better video quality.

The rest of this paper is organized as follows. Section II introduces the system architecture of HotStreaming. Section III presents the partnership formation of the overlay network. Section IV describes the spatial-temporal hybrid interpolation scheme. The integration of the whole system is given in Section V. Experimental results and discussions are presented in Section VI. Finally, Section VI concludes this paper.

II. HotStreaming Architecture

The HotStreaming architecture is composed of three major technical components: partnership formation, video segment request scheduling, and multiple description coding. The partnership formation component is responsible for finding the peers to serve video segments to the user. The video segment request scheduling algorithm manages how and where to request the video segments. The multiple description coding determines the encoding and playback strategy for satisfying user perception. As far as we know, HotStreaming is one of the pioneering, if not the first, IPTV systems that integrate innovations in both overlay networking and video coding for optimal user experience.

The partnership formation component is essential for the overlay P2P based video broadcasting. In this overlay P2P network paradigm, a video stream is divided into segments that are delivered from peer users to peer users. More specifically, a user chooses a number of peers to form a partner group, from which the user receives video segments and requests for retransmissions if necessary. A user might also be chosen by other peers in the network to be their partners. The construction of this partner/consumer relationship greatly affects the efficiency and stability of the network. In fact, load balancing and stability of the partnership relationship in the presence of peer dynamics, i.e., joining and leaving the network randomly, are the main design considerations in the partnership formation component of HotStreaming. Results from the simulation [7] show that, compared to a recently published mechanism for the same function [6], the two partnership formation schemes of HotStreaming are able to reduce the instability of the P2P partnership by 52% and 89% respectively, and the stability gain is achieved without compromising the system load balance. In the following sections, we focus our discussions on the partnership formation and multiple description coding components. The terms used in this paper are defined in Table I, and they are defined relative to the peer node A.

TABLE I. NOMENCLATURE

Partner Peer	Peer that serves data to the peer node A
Consuming Peer	Peer that receives data from the peer node A
Contact Node	Node that a newly joined peer first connect to
PartialView	The list of partner peers
InView	The list of consuming peers

III. Partnership Formation

The partnership formation component determines for each user a list of partner peers that can serve video segments to the user. Because the peers may leave or join the overlay network at any time, the reliability of the transmission is affected, and it is necessary to select substitute partners in the event of partners leaving the network.

The PartialView is intended to capture the candidate partners that are alive and preferred by the receiving peer. It is dynamically adjusted based on peer advertisements, messages in which each peer advertises its existence. When a peer receives the advertisements, it decides whether to select the sending peer as a partner according to the algorithm used in the partnership formation component. A balanced and robust partnership formation policy can achieve both load balance and stability. In this part, we survey the gossip-based partnership mechanism and propose possible improvements.

A. Gossip-Based Partnership

SCAMP (Scalable Membership protocol) [15] is the state-of-the-art peer-to-peer partnership formation protocol that operates in an unstructured and fully decentralized manner. It is the mechanism used in CoolStreaming, a well-known overlay P2P TV broadcasting system. In SCAMP, each peer node maintains a PartialView and an InView. The system is self-organizing in the sense that the size of a PartialView naturally converges to the value required to support a gossip algorithm reliably.
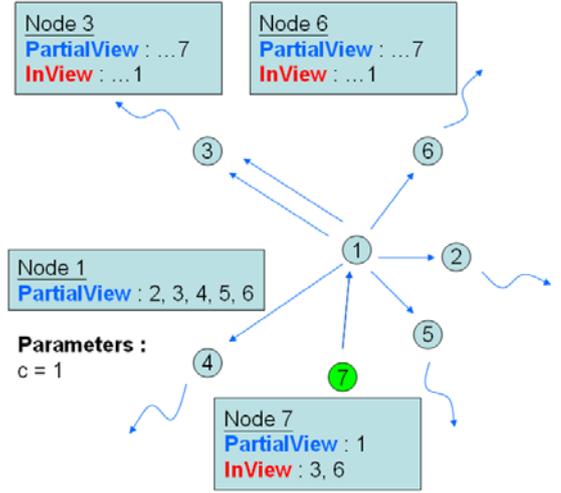

Figure 1. Node 7 enters the system and chooses node 1 as the contact node.

The process of a new node joining the system is illustrated in Figure 1. When a peer node 7 joins the system for the first time, it adds the contact node 1 in its PartialView. The contact node then advertises the identifier of this newly joined peer node to all its partner peers and a predetermined number of additional peers randomly selected from its PartialView. These advertisements are also referred to as subscription messages. When a peer receives a subscription message, it adds the newly joined node to its PartialView of size M with a probability of $1/(1+M)$, and the subscription message is dropped. If the newly joined node is not added to the PartialView, the subscription message will be randomly forwarded to another peer in the PartialView.

To avoid staleness, the nodes in the PartialView are all timed and each peer node should send subscription messages periodically to reset the expiration time. These advertisement messages are also used to update the partner list as other nodes join and leave the system dynamically.

In SCAMP, subscription forwarding and partner selection processes are purely random. Although this achieves good load balance, the network can be unstable in dynamic environments. In the presence of node dynamics, SCAMP system is often disconnected, and this results in video interruption and overall network instability defined in [7]. When a node discovers that there are no more nodes in the partial or in view, it leaves and re-joins the network. To improve the stability, we propose two preferential random policies for the subscription message advertising and partner selection.

B. Improved Preferential Gossip Partnership

When a node leaves, it may result in the isolation and disconnection of some nodes, which means that these nodes have no partners and they are not partners of any other node. The proposed two preferential policies aim to reduce the time duration of disconnection and number of isolated nodes caused by node dynamics.

The first policy is to allow preferential random forwarding of the subscription messages, referred to as the Preferential Random Forwarding approach. When each node re-subscribes itself, partners with larger partnership are favored and have a higher chance to be chosen. To balance the partnership, the subsequent forwarding of the subscription messages will be sent to partners with smaller partnership.

The second policy is to allow preferential selection of partners. In SCAMP and Preferential Random Forwarding, a node is selected as a partner when its subscription is received by the probability $1/(1+M)$. In this further improvement, referred to as the Preferential Random Selection approach, a peer with small partnership size selects all nodes from all subscriptions it has received until reaching the default partnership size limit.

IV. MULTIPLE DESCRIPTION CODING WITH STHI

The multiple description coding is a critical component of HotStreaming that directly influences the user satisfaction. We decompose a video stream in the temporal dimension for diversity gains, and encode the odd frames based on the previous odd frames and the even frames based on the previous even frames. As opposed to coding the frame-to-frame differences in MPEG-4, we code the interleaving frame differences. Then the two sub-streams can be transmitted separately over the network and played back with synchronization at the receiver. Such a design is more fault tolerant than conventional video coding. When one sub-stream is lost due to network congestion, the decoder can still independently play the remaining sub-stream with controlled degradation.

Based on the concept described above, we propose a new coding scheme named MDC-STHI (Spatial Temporal Hybrid Interpolation) [9], which generates four streams as shown in

Figure 2. Streams E_f and O_f represent the even and odd streams, respectively, encoded from the original full-size video. E_q and O_q denote the down-sampled version encoded at a lower bit rate. Our implementation uses 2:1 down sampling in each dimension of the video frame, resulting in one quarter of the original resolution. While E_q and O_q require extra bits to encode, we will show in the next section that the combination of the four streams has many desirable features for video streaming over P2P networks such as HotStreaming.

Figure 3 illustrates a possible P2P networking scenario using MDC-STHI. Generally, Internet users with various computational capabilities are connected through networks of different bandwidths, and in a P2P environment like HotStreaming, they may be randomly grouped as peers for video streaming. In Figure 5, the number of circles indicates roughly the amount of resources at the node. Thus, N_3 has enough bandwidth and computing power to retrieve both streams E_f+O_q and O_f+E_q from N_1 and N_2. In turn, N_4, N_5, and N_6 can receive appropriate combinations of streams from N_3 according to their available resources. For example, N_4 may retrieve only E_q or O_q from N_3 because it uses a mobile device with a narrowband wireless connection and a small-sized screen that best displays quarter-sized videos. On the other hand, N_6 can accommodate all streams from N_3 and may further distribute the video streams intelligently to other peers. Compared to other MDC designs, MDC-STHI offers superior flexibility for P2P video streaming as the four streams are independently encoded and a node may adaptively deliver suitable combinations according to the peers' capabilities.

The combination of E_f+O_q or O_f+E_q poses an interesting challenge as it contains two streams of different resolutions. During transmission, the bits from O_q may piggyback in the same IP packet with E_f of the previous frame, and vice versa for O_q. This saves the overhead and ensures same time arrival of adjacent frames. We also developed an innovative hybrid interpolation scheme for such combinations, fully explained in [14].
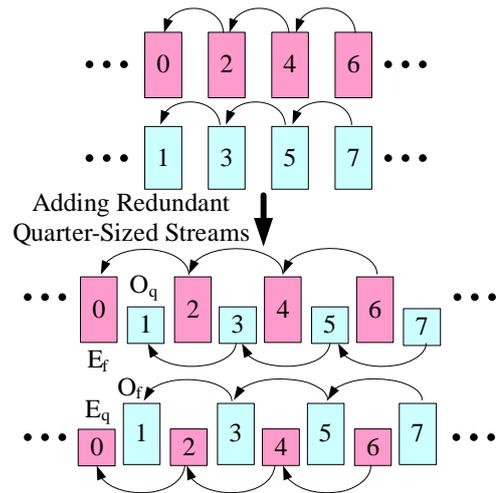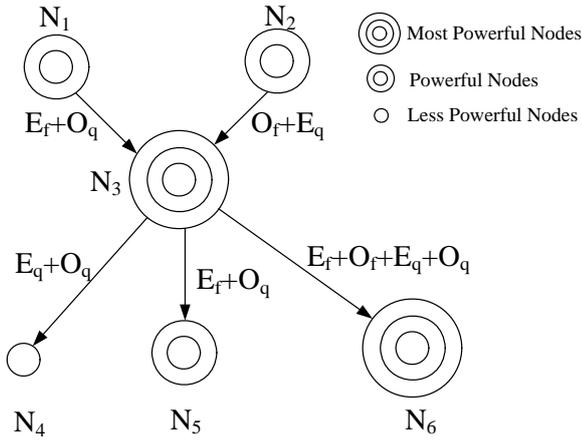

Figure 2. Video Streams in MDC-STHI.
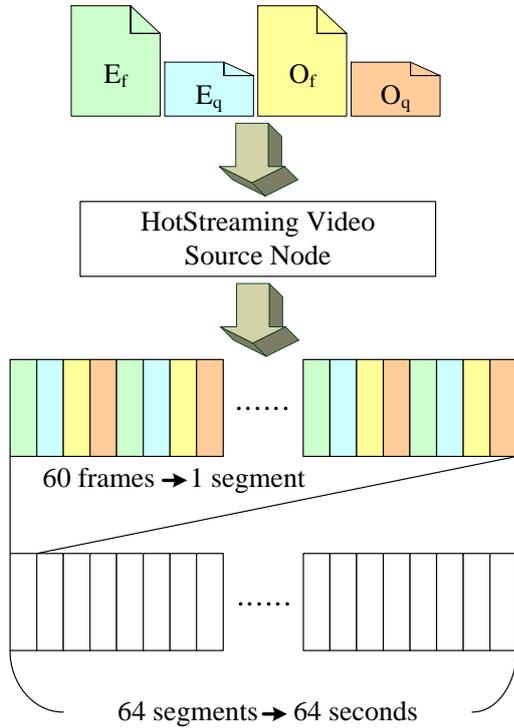
Figure 3.   MDC-STHI over a P2P network.

Figure 4.   HotStreaming video source node.

V.   INTEGRATION

Figure 4 shows the integration of MDC-STHI with the HotStreaming video source node. The four independently-encoded MDC-STHI bit-streams are read into the video source node and segmented. Each segment contains 60 frames, 15 frames for each bit-stream, thus there are 64 segments in the buffer, equivalent to 64 seconds of video data.

Figure 5 shows the video data flow of a HotStreaming node. Each node can request different combinations of full-sized and quarter-sized frames. When receiving a segment, the MDC-

STHI decoder processes the data using the corresponding MPEG-4 decoder based on the type of the segment. The decoded frames are then put into the STHI buffer. If the node only receives part of the description due to packet loss or bandwidth constraint, the STHI module will perform spatial-temporal hybrid interpolation on the lost frames. Finally, the interpolated frames are sent to the display buffer.
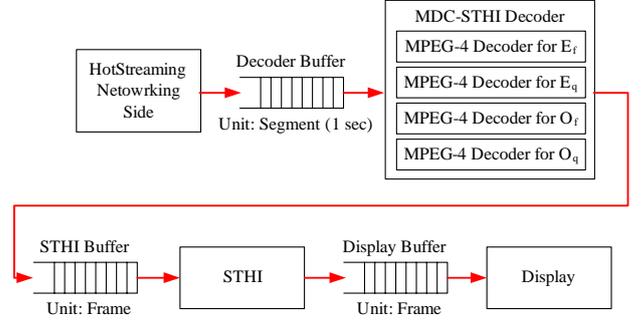

Figure 5.   Video data flow of a HotStreaming node.

VI.   EXPERIMENTAL RESULTS

In this section, we show the simulation results using the Football sequence of the CIF resolution, which are typical among the extensive experiments conducted. The simulations are based on the network trace collected from the execution of the HotStreaming system. Since the trace data is more than 300 seconds long, we only demonstrate 23 seconds of it with representative characteristics. The segment loss statistics of two descriptions are shown in Figures 6 and 7. In these figures, 0 means the segment is lost, while 1 indicates the segment is successfully received.
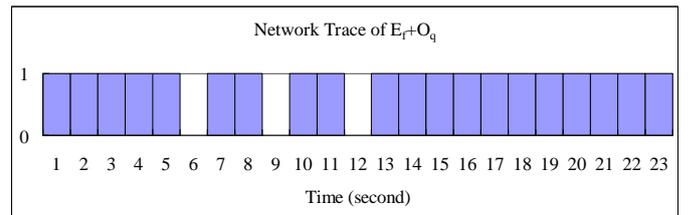

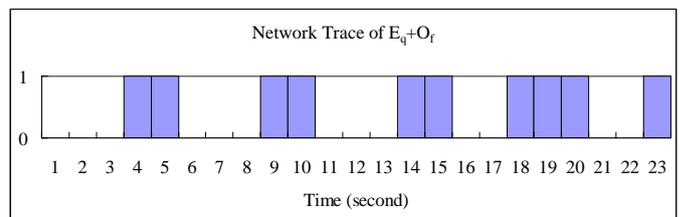Figure 6.   HotStreaming network trace data of E_f+O_q.

Figure 7.   HotStreaming network trace data of E_q+O_f.

For conventional streaming systems, the player just repeats the previous frame when data loss occurs. The PSNR values for such a scheme are shown in Figure 8. The PSNR values for MDC-STHI are shown in Figure 9. The PSNR drops in these figures correspond to the segment loss in Figures 6 and 7. By comparing Figures 8 and 9, we can see that the PSNR values for frame repetition are usually below 20dB when segments are lost. However, the PSNR values for MDC-STHI are usually over 30dB. The PSNR values drops below 20dB only when both descriptions are lost.
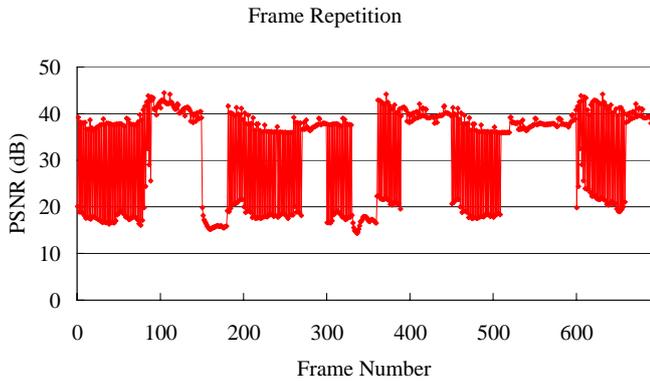
Frame Repetition

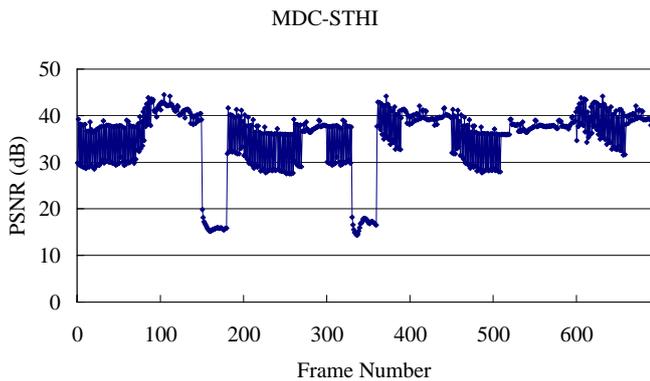Figure 8. PSNR valus for frame repetition.

MDC-STHI

Figure 9. PSNR values for MDC-STHI.

VII. CONCLUSION

In summary, we have presented the networking and video coding parts of HotStreaming, and explained the integration of both parts. The experimental results indicate that the combination of the P2P streaming with the multiple description coding provides significantly improved robustness. For future work, the core designs of the networking and video coding parts can be further integrated with better video segment request scheduling. The mechanism of video segment request scheduling in the state-of-the-art CoolStreaming is rather heuristic and simple. A policy of requesting the rarest segment first could be inadequate and unsuitable for heterogeneous networks. Our premise is to allow each node in a heterogeneous network to request segments of particular resolutions or temporal streams based on the available resources on the partner peers or itself. As a result, the system should be able to better utilize locally available resources and achieve better overall user satisfaction.

REFERENCES

[1] D. Waitzman, C. Patridge, and S. Deering, "Distance vector multicast routing protocol," *Request For Comments 1075*, IETF Network Working Group, Nov. 1988.

[2] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, "Protocol independent multicast-sparse mode (PIM-SM): Protocol specification," *Request for Comments 2362*, IETF Network Working Group, June 1998.

[3] S. Floyd, V. Jacobson, C. Liu, S. McCanne and L. Zhang., "A reliable multicast framework for light-weight sessions and application level framing," *IEEE/ACM Trans. on Networking*, vol. 5, no. 6, pp. 784-803, Dec. 1997.

[4] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, "Application-level multicast using content-addressable networks," *Proc. of Third International Workshop on Networked Group Communication*, London, England, pp. 14-29, Nov. 2001.

[5] D. A. Tran, K. A. Hua, and T. T. Do., "A peer-to-peer architecture for media streaming (Zigzag)," *IEEE Journal of Selected Areas of Communications*, pp. 121-133, Jan. 2004.

[6] X. Zhang, J. Liu, B. Li, and T.-S. P. Yum, "DONet/CoolStreaming: A data-driven overlay network for live media streaming," *IEEE INFOCOM*, vol. 3, pp. 2102-2111, Miami, FL, USA, Mar. 2005.

[7] J.-C. Wu, K.-J. Peng, M.-T. Lu, C.-K. Lin, Y.-H. Cheng, P. Huang, J. Yao, and H. H. Chen, "HotStreaming: Enabling scalable and quality IPTV services," *Proc. of IPTV Workshop in conjunction with the 15th International World Wide Web Conference*, Edinburgh, Scotland, UK, May 2006.

[8] V. N. Padmanabhan, H. J. Wang, and P. A. Chou, "Resilient peer-to-peer streaming," *Proc. of IEEE International Conference on Network Protocols*, pp. 240-247, Atlanta, Georgia, USA, Nov. 2003.

[9] M. Zink and A. Mauthe, "P2P streaming using multiple description coded video," *Proc. of Euromicro Conference*, pp. 240-247, Rennes, France, Sep. 2004.

[10] S. Khan, R. Schollmeier, and E. Seinbach, "A performance comparison of multiple description video streaming in peer-to-peer and content delivery networks," *Proc. of International Conference on Multimedia and Expo*, pp. 503-506, Taipei, Taiwan, June 2004.

[11] J. G. Apostolopoulos, "Error-resilient video compression via multiple state streams," *Proc. of International Workshop on Very Low Bitrate Video Coding*, pp. 168-171, Kyoto, Japan, Oct. 1999.

[12] J. G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and Path diversity," *Proc. of Visual Communications and Image Processing 2001*, pp. 392-409, San Jose, CA, USA, Jan. 2001.

[13] G. Zhang and R. L. Stevenson, "Efficient error recovery for multiple description video coding," *Proc. of International Conference on Image Processing*, pp. 829-832, Singapore, Oct. 2004.

[14] M.-T. Lu, C.-K. Lin, J. Yao, and H. H. Chen, "Multiple description coding with spatial-temporal hybrid interpolation for video streaming in peer-to-peer networks," *Proc. of IEEE Packet Video Workshop*, Hangzhou, Zejiang, China, Apr. 2006.

[15] A. J. Ganesh, A. M. Kemarrec, L. Massoulie, "Peer-to-peer membership management for gossip-based protocols," *IEEE Trans. on Computers*, vol. 52, no. 2, pp. 139-149, Feb. 2003.