International Journal of Communication Networks and Security

Volume 2 | Issue 1 Article 11

January 2013

VOD STREAMING WITH A NETWORK CODING EQUIVALENT CONTENT DISTRIBUTION SCHEME

P. PRATYUSHA
QCET, Nellore,India, P.PRATYUSHA@gmail.com

SD.AFZAL AHMAD
QCET, Nellore,India, SD.AFZALAHMAD@gmail.com

P. BABU
QCET, Nellore,India, P.BABU@gmail.com

VOD STREAMING WITH A NETWORK CODING EQUIVALENT CONTENT DISTRIBUTION SCHEME

P.PRATYUSHA[1], SD.AFZAL AHMAD[2], P.BABU[3]

[1]PG Student, [2,3]Associate Professor, QCET, Nellore

Abstract- Although random access operations are desirable for on-demand video streaming in peer-to-peer systems, they are difficult to efficiently achieve due to the asynchronous interactive behaviors of users and the dynamic nature of peers. In this paper, we propose a network coding equivalent content distribution (NCECD) scheme to efficiently handle interactive video-on-demand (VoD) operations in peer-to-peer systems. In NCECD, videos are divided into segments that are then further divided into blocks. These blocks are encoded into independent blocks that are distributed to different peers for local storage. With NCECD, a new client only needs to connect to a sufficient number of parent peers to be able to view the whole video and rarely needs to find new parents when performing random access operations. In most existing methods, a new client must search for parent peers containing specific segments; however, NCECD uses the properties of network coding to cache equivalent content in peers, so that one can pick any parent without additional searches. Experimental results show that the proposed scheme achieves low startup and jump searching delays and requires fewer server resources. In addition, we present the analysis of system parameters to achieve reasonable block loss rates for the proposed scheme.

INTRODUCTION:

MULTIMEDIA streaming is now a popular Internet service. However, efficient streaming to a large client population is hampered by server bandwidth constraints and the fact that IP-layer multicast is not universally supported. Peer-to-peer (P2P) collaborative streaming is a promising solution to the problem of efficiency.In a P2P system, each peer requests multimedia content from specific supplying peers.Then, after receiving the data, the peer caches it in local storage so that the (receiving) peer can now become a new supplier for other peers. An important challenge in a P2P collaborative video-ondemand (VoD) streaming system is to develop an effective content distribution scheme that can support a dynamic network among peers, where autonomic peers can join or leave the system at any time and any place in the network. The situation is further complicated by the need to support random access, such as the trick plays of pause/resume, jump, fast forward (FF), and rewind. Such trick plays may occur frequently. Most existing approaches require at least O(log(N)) time to locate the requested segment, where N is the number of segments of the requested video. The schemewe propose in this study can offer a more efficient approach (and one that supports trick plays) to the P2P-based interactive VoD systems. The "cache-and-relay" technique used and keeps recently played data in the cache of the receiver so that it can be forwarded to other peers. BitTorrent allows users to stream videos and watch them, even during download. This approach requires clients to cache the entire video file, even though they have already viewed it, thus wasting storage space. The cache-and-relay technique has difficulty with trick play operations. For instance, a parent peer might jump to another play point in the video. This would prevent it from forwarding a continuous stream to its child peers, thus requiring all its child peers to search for a new parent. As a result, it will cause propagation delay for the child peers. The proposed scheme avoids these problems by adopting the additional static local storage used instead of sliding window playback buffering, to efficiently support users' interactive operations and decrease complexity. The advantage of using additional storage is that any user interactivity on the part of the peer does not affect its children from continuing to receive its stored media data. Moreover, observations from a large number of user requesting logs indicate that random seeking is frequently performed by most users. This is reasonable, as users usually jump directly to the scene of interest and skip boring segments. Therefore, it would be favorable if the system could guarantee peers the ability to jump to any play point in the requested video without searching for new parent peers that possess specific segments. In this paper, we propose a novel network coding equivalent content istribution (NCECD) scheme for a multisource, P2P-based, interactive VoD so as to 1) enable child peers to link to these parents with partial and not duplicated data for the complete video and 2) tackle the problem of parent departure,weuse linear network coding to generate an encoded block by encoding all blocks in one segment. If enough encoded blocks are received by a child peer, the child peer can decode the original segment. Therefore, linear network coding, combined with interleaving block distribution, results in a situation in which a child peer only needs to find a sufficient number of parent peers to be able to view any given segment of the requested video; the child peer does not have to search for new parent peers to view the next segment or to perform interactive operations (e.g., jump or rewind). Furthermore, when a parent peer leaves the network, the child peer can still receive some encoded blocks from other parents so as to decode the original segment. Besides, the child peer can locate any peer that caches encoded blocks of the requested video as a parent peer in the P2P networks, thereby obtaining good video segment search performance. The main contributions of this study are as follows:

1. Using interleaving block distribution and linear network coding techniques, we propose an efficient media data distribution scheme for P2P-based, interactive VoD streaming.

2. The proposed content distribution scheme is cost effective as it does not need to maintain an index data structure and network topology to locate parent peers that cache target segments.

3. P2P-based trick plays are supported effectively and naturally.

4. The study analyzes practical system parameters such as block size, peer cache capacity, required number of parents, linear network coding implementation, and packet loss rates from parent peer departure.

Existing System:
An important challenge in a P2P collaborative video-on-demand (VoD) streaming system is to develop an effective content distribution scheme that can support a dynamic network among peers, where autonomic peers can join or leave the system at any time and any place in the network. The situation is further complicated by the need to support random access, such as the trick plays of pause/resume, jump, and fast forward (FF), and rewind. Such trick plays may occur frequently. Most existing approaches require at least O (log (N)) time to locate the requested segment, where N is the number of segments of the requested video. The scheme we propose in this study can offer a more efficient approach (and one that supports trick plays) to the P2P-based interactive VoD systems.

Proposed System:
The proposed scheme avoids these problems by adopting the additional static local storage instead of sliding window playback buffering, to efficiently support users' interactive operations and decrease complexity. The advantage of using support random access, such as the trick plays of pause/resume, jump, and fast forward (FF), and rewind. Such trick plays may occur frequently. Most existing approaches require at least O (log (N)) time to locate the dditional storage is that any user interactivity on the part of the peer does not affect its children from continuing to receive its stored media data. Moreover, observations from a large number of user requesting logs indicate that random seeking is frequently performed by most users. This is reasonable, as users

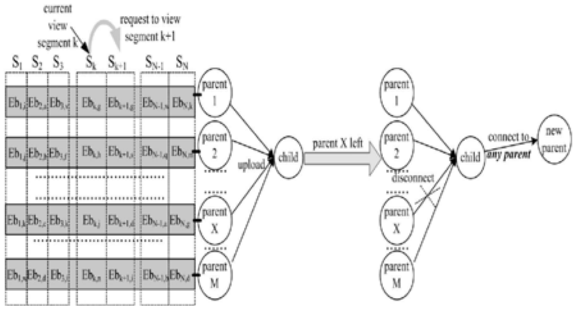

Fig: The proposed NECED scheme

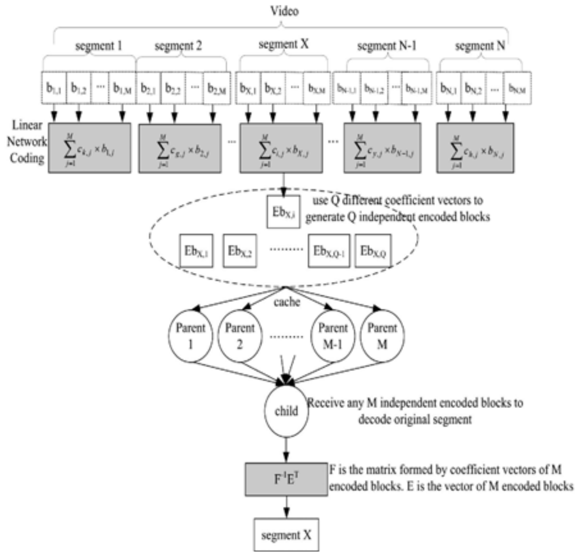

Fig: The network coding technique is used to encode video blocks. One video is divided into N segments and each segment is divided into M blocks. Via network coding, M blocks of a segment become an encoded block. In this way, different encoding coefficients are used to generate Q independent encoded blocks. Each parent peer caches one encoded block; one child peer connects to M parent peers to receive M independent encoded blocks to decode original segment X.

Software Requirement Specification:
Software Specification
Operating System: Windows XP
Technology : JAVA 1.6, JMF

Hardware Specification
Processor : Pentium IV
RAM : 512 MB
Hard Disk : 80GB

Modules:

- Admin

It receives the request from clients and processes it, sends the requested video corresponding to client bandwidth.

- Client

It send the request to the server, request contains requested file, available bandwidth of client and receives the video based on its available bandwidth.

Simulation Results:

Here,we will consider buffering time and decoding time in order to estimate the complete startup and jump delays in all schemes. So as to reduce the decoding delay from decoding network coding blocks, we adopt the Gauss-Jordan elimination implemented in the decoding process.Using progressive decoding, the decoding time is almost completely covered by the buffering time, and thus, is almost negligible. Fig. 5a shows the startup delay including searching,buffering, and decoding delay (only for the proposed scheme used to decode network coding blocks to receive the original segment) for a variety of average peer buffering. As the peer population increases, it is most likely that child peers in systems locate closer parent peers, thus reducing the average segment searching delay. VMesh performs the worst since it uses a DHT search to locate the segment of interest at the startup stage. Note that for VMesh, as the peer population increases, the startup searching delay should also increase. However, a child peer is also more likely to find closer parent peers, and therefore, decrease the startup searching delay. Thus, in VMesh, the startup delay is reduced when the peer population increases.

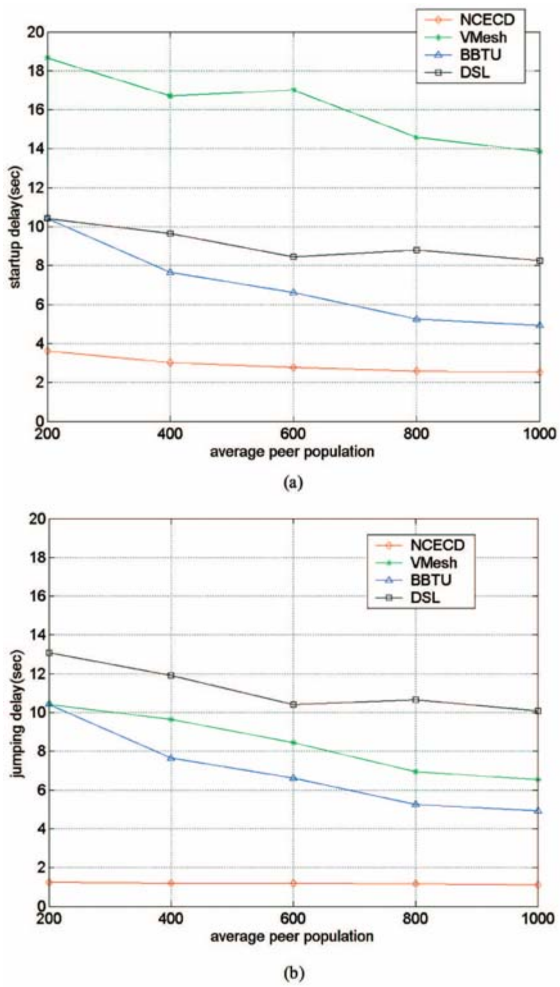

Fig. 5. (a) Startup delay under different average peer populations.
(b) Jump delay under different average peer populations.

The NCECD scheme performs better BBTU, DSL, and VMesh, by, on average, 56.7, 68.3, and 82 percent, respectively. Similarly, jump delay incorporates the seeking segment location latency, buffering delay, and decoding delay (in the proposed scheme). Fig. 5b plots the jump delay under various average peer populations. The trend is similar to that of the startup delay observed in other competing schemes. Note that the proposed NCECD scheme does not generate searching delay, only buffering and decoding delays, when a child peer requests a jump operation.

This is because a child peer only needs to connect to a sufficient number of parent peers to view the whole video; thus, the jump searching delay is zero. 2) The relationship between segments supplied to and demanded by peers: To evaluate the required server resources and bandwidth for this kind of P2P system, we first explore the relationship between supply and demand among peers.

When supply and demand are balanced, most requests of child peers can be served by parent peers, making server stress low. Fig. 6a shows the duplication number of each segment distributed to all peers in a system.

The segment popularity model from [1] is indicated by “popularity model” in Fig. 6a. For a balanced relationship of supply and demand, the segment duplication number should be proportional to its popularity. Therefore, if the duplication number of one distribution scheme can more closely match the “popularity model,” it suggests that the distribution scheme can achieve a more balanced relationship between supplying peers and segments demanded by other peers.

As shown in Fig. 6a, each parent peer in the proposed NCECD scheme is equivalent, in other words, one parent peer can be replaced by any other parent peer. Since one peer is a demander and also a supplier, if cached content of each peer is equivalent, it implies that the relationship of supply and demand is balanced. Hence, the duplication number most closely matches the “popularity model” for the proposed NCECD scheme. In some existing systems, to reduce system complexity, a peer downloads one random segment for caching; thus, the number of duplications of each segment in a video is almost the same in such a system.

The uniform distribution scheme cannot completely balance the relationship between supply and demand. In VMesh, the popularity-aware distribution scheme is proposed to ascertain the popularity of each segment. The popularity of each segment is estimated by exchanging information between neighbor peers. The more the exchange of information, the more precise will be the estimated popularities of the segments.

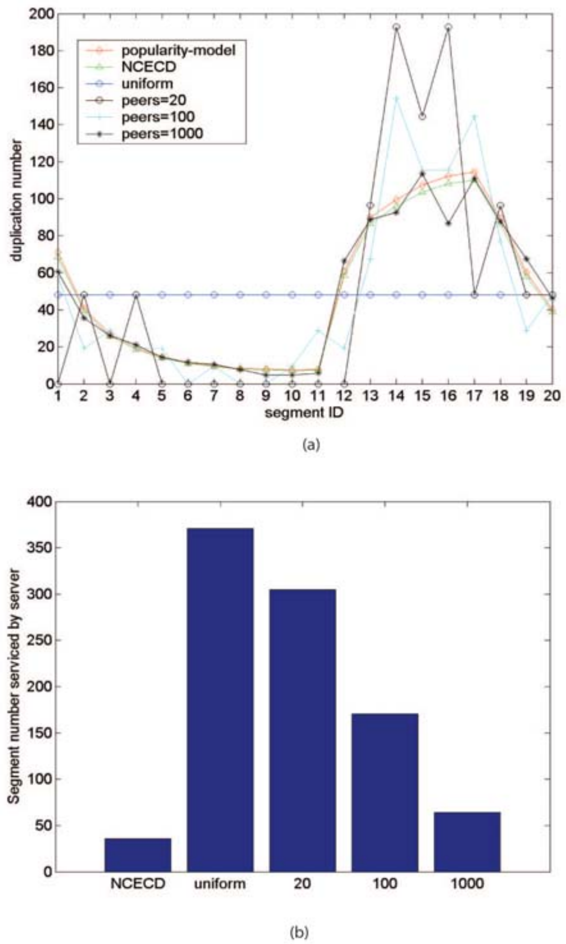

Fig. 6. (a)The duplication number of segments under different schemes.
(b) The number of segments serviced by the server.

As shown in Fig. 6a, the lines indicated by 20, 100, and 1,000 peers present the duplication number of each segment when the peer number for exchanging information is 20, 100, and 1,000, respectively. As the amount of information being exchanged increases, the result gets closer to the "popularity mode;" however, it is costly for a number of peers to periodically exchange and update popularity information for segments. Fig. 6b plots the difference of requested segments by child peers and supplied segments by parent peers. The server needs to handle requests for segments whose demand exceeds supply. Thus, the larger the gap, the higher is the server stress. The proposed NCECD scheme outperforms the uniform and the popularity-aware schemes at 20, 100, and 1,000 peers by 90.3, 88.2, 78.9, and 43.9 percent, respectively.

CONCLUSION:

In this paper, we proposed a novel data distribution scheme called NCECD to provide interactive VoD services in a P2P network. In the NCECD scheme, videos are divided into smaller segments, which are further divided into blocks. The NCECD scheme applies network coding technology to generate several encoded blocks by combining the encoding of all blocks in one segment. These encoded blocks are distributed to peers on the system.A child peer needs only to find and link to a sufficient number of parent peers to view the entire video, thus eliminates the search for new parent peers. In this way, interactive functionality can be supported efficiently. Network coding techniques naturally provide failure-tolerant streaming services as a client on the system connecting to multiple parent peers who have stored equivalent media data and are able to stream media data in parallel and collaboratively. An appropriate number of extraparent peers is found to provide low block loss probability. Experimental results show that the NCECD scheme substantially relieves server stress by optimally matching supply of and demand for segments in a P2P network. Simulation and analyses demonstrate that the proposed scheme outperforms other competing schemes such as VMesh, BBTU, and DSL in terms of startup delay, jumping delay, and server stress. Additionally, NCECD can achieve very low block loss probabilities under various system parameters by connecting to an appropriate number of extra parent peers, allow for failure-tolerant streaming services in a P2P network.For future development we can add multiple servers.

REFERENCES:

[1] W.P.K. Yiu, X. Jin, and S.H.G. Chan, "VMesh: Distributed Segment Storage for Peer-to-Peer Interactive Video Streaming,"IEEE J. Selected Areas in Comm., vol. 25, no. 9, pp. 1717-1731, Dec.2007.

[2] C. Xu, G.M. Muntean, E. Fallon, and A. Hanley, "A Balanced Tree-Based Strategy for Unstructured Media Distribution in P2PNetworks," Proc. IEEE Int'l Conf. Comm. (ICC '08), pp. 1797-1801,May 2008.

[3] C. Zheng, G. Shen, and S. Li, "Distributed Prefetching Scheme forRandom Seek Support in Peer-to-Peer Streaming Applications,"Proc. ACM Workshop Advances in Peer-to-Peer Multimedia Streaming,pp. 29-38, Nov. 2005.

[4] D. Wang and J. Liu, "A Dynamic Skip List-Based Overlay for On-Demand Media Streaming with VCR Interactions," IEEETrans. Parallel and Distributed Systems, vol. 19, no. 4, pp. 503-514,Apr. 2008.

[5] H.V. Jagadish, B.C. Ooi, and Q.H. Vu, "BATON: A Balanced Tree Structure for Peer-to-Peer Networks," Proc. Int'l Conf. Very Large Data Bases (VLDB '05), pp. 661-672, Aug. 2005.

[6] S.Y.R. Li, R.W. Yeung, and N. Cai, "Linear Network Coding," IEEE Trans. Information Theory, vol. 49, no. 2, pp. 371-381, Feb. 2003.

[7] M. Wang and B. Li, "R2: Random Push with Random Network Coding in Live Peer-to-Peer Streaming," IEEE J. Selected Areas in Comm., vol. 25, no. 9, pp. 1655-1666, Dec. 2007.

[8] P.J. Wu, J.N. Hwang, C.N. Lee, C.C. Gau, and H.H. Kao, "Eliminating Packet Loss Accumulation in Peer-to-Peer Streaming Systems," IEEE Trans. Circuits and Systems for Video Technology, vol. 19, no. 12, pp. 1766-1780, Dec. 2009.

[9] E.W. Zegura, K.L. Calvert, and S. Bhattacharjee, “How to Model an Internetwork,” Proc. IEEE INFOCOM, vol. 2, pp. 594-602, Mar. 1996.

[10] G. Wu, B. Li, and Z. Li, “Dynamic Bandwidth Auctions in Multioverlay P2P Streaming with Network Coding,” IEEE Trans. Parallel and Distributed Systems, vol. 19, no. 6, pp. 806-820, June 2008.

[11] H. Chi, Q. Zhang, J. Jia, and X. Shen, “Efficient Search and Scheduling in P2P-Based Media-on-Demand Streaming Service,” IEEE J. Selected Areas in Comm., vol. 25, no. 2, pp. 119-130, Jan. 2007.

[12] C. Gkantsidis, J. Miller, and P. Rodriguez, “Comprehensive View of a Live Network Coding P2P System,” Proc. ACM SIGCOMM Conf. Internet Measurement (IMC ’06), pp. 177-188, Oct. 2006.

[13] S. Annapureddy, S. Guha, C. Gkantsidis, D. Gunawardena, and P. Rodriguez, “Exploring Vod in P2P Swarming Systems,” Proc. IEEE INFOCOM ’07, pp. 2571-2575, May 2007.

[14] K. Nguyen, T. Nguyen, and S.-C. Cheung, “Peer-to-Peer Streaming with Hierarchical Network Coding,” Proc. IEEE Int’l Conf. Multimedia and Expo, pp. 396-399, July 2007.

[15] M. Wang and B. Li, “Lava: A Reality Check of Network Coding in Peer-to-Peer Live Streaming,” Proc. IEEE INFOCOM, pp. 1082- 1090, May 2007.

[16] C. Feng and B. Li, “On Large-Scale Peer-to-Peer Streaming Systems with Network Coding,” Proc. ACM Int’l Conf. Multimedia, pp. 269-278, Oct. 2008.

[17] A. Habib and J. Chuang, “Service Differentiated Peer Selection: An Incentive Mechanism for Peer-to-Peer Media Streaming,” IEEE Trans. Multimedia, vol. 8, no. 3, pp. 610-621, June 2006.

[18] http://www.bittorrent.com, 2011.

[19] C.S. Chang, T. Ho, M. Effros, M. Medard, and B. Leong, “Issues in Peer-to-Peer Networking: A Coding Optimization Approach,” Proc. IEEE Int’l Symp. Network Coding, June 2010.

[20] A. Sharma, A. Bestavros, and I. Matta, “dPAM: A Distributed Prefetching Protocol for Scalable Asynchronous Multicast in P2P Systems,” Proc. IEEE INFOCOM, Mar. 2005.

[21] Y. Cui, B. Li, and K. Nahrstedt, “oStream: Asynchronous Streaming Multicast in Application-Layer Overlay Networks,” IEEE J. Selected Areas in Comm., vol. 22, no. 1, pp. 91-106, Jan. 2004.

[22] T.T. Do, K.A. Hua, and M.A. Tantaoui, “P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment,” Proc. IEEE Int’l Conf. Comm. (ICC), June 2004.

[23] C. Gkantsidis and P.R. Rodriguez, “Network Coding for Large Scale Content Distribution,” Proc. IEEE INFOCOM ’05, Mar. 2005.

[24] M. Kim, L. Lima, F. Zhao, J. Barros, M. Medard, R. Koetter, T. Kalker, and K.J. Han, “On Counteracting Byzantine Attacks in Network Coded Peer-to-Peer Networks,” IEEE J. Selected Areas in Comm., vol. 28, no. 5, pp. 692-702, May 2010.

[25] F.H.P. Fitzek, M. Katz, and Q. Zhang, “Cellular Controlled Short- Range Communication for Cooperative P2P Networking,” Wireless Personal Comm., vol. 48, no. 1, pp. 141-155, 2009.

[26] F.H.P. Fitzek, M.V. Pedersen, J. Heide, and M. Me´dard, “Network Coding: Applications and Implementations on Mobile Devices,” Proc. Fifth ACM Workshop Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks (PM2HW2N), pp. 83- 87, Oct. 2010.

◈◈◈