Distributed Delivery of Popular Videos over Ultra-dense Networks

Chia-Cheng Yen
Department of Computer Science,
National Tsing Hua University,
Hsinchu, Taiwan
ccyen@vc.cs.nthu.edu.tw

Jia-Shung Wang
Department of Computer Science,
National Tsing Hua University,
Hsinchu, Taiwan
jswang@cs.nthu.edu.tw

***Abstract*—The application of video streaming is expected to shift to mobile broadband as soon as the Ultra-dense Networks (UDN) and High Efficiency WLAN (HEW) become popular. The motivation of this paper is to propose a distributed delivery scheme for streaming popular videos over UDN wireless environments. The H.264 SVC scalable source coding and the LT codes of rateless channel coding are both considered and integrated to provide reliable and scalable video services wirelessly.**

In the allocation phase, a hot video clip is both SVC encoded and LT encoded. Then, the coded data are randomly distributed among small-cell stations (SCs) or related attached servers. In the serving phase, one nearby SC immediately decodes its own LT coded data once a request issued and broadcasts them to the requesting user and the neighboring SCs to continue the undone decoding process in parallel. Thus, the clients can receive data from multiple SCs to achieve the goal of bandwidth aggregation. Besides, once the decoding process being completed, the repairing process is ignited to recover the possible node failure if exists. Two data allocation schemes are considered to balance the storage space, transmission bandwidth and fault tolerance requirements. In the ultra-dense scenarios, our simulations show that each request can be served with at least acceptable quality and with one more enhancement layer in average under communication loss rate less than 10% and possible node failures.

Keywords—Distributed decoding, H.264 SVC, LT codes, Ultra-dense networks (UDNs), Bandwidth aggregation

I. Introduction

According to the CISCO's VNI Mobile Forecast report, mobile data traffic will grow 11-fold from 2013 to 2018, a compound annual growth rate of 61%. Mobile video is the largest and fastest growing segment; it is forecasted to account for over 69.1 percent. Qualcomm is now providing the developing solutions to meet "1000x challenge."

To bring the network close to the clients to offer unprecedented capacity, Ultra-dense Networks (UDN) or Hyper-dense Heterogeneous and small cell networks (HetSNets) [1] are driven to change the network deployment principle. Thus, the deployment of ultra-dense small cells (SCs) plays a key role of the promising approach to meet the unprecedented demands. Mobile Content Delivery Network (MCDN) [2] is gaining increasing attention due to huge demand and popularity of mobile video traffics. As CDN, MCDN takes full advantage of caching in the network edge (i.e., base station or end-user devices) in such a way that the buffered video can be delivered with less network latency and traffic load. The motivation of this paper is to study distributed delivery technologies for streaming popular videos over UDN or HetSNets wireless environments.

In this paper, we propose a distributed solution for providing popular videos streaming to clients in UDNs, where all the SCs are randomly deployed in a fixed area. They are equipped with computational apparatus and storage space, which means each node, has ability to communicate with neighboring SCs and decode received data. The proximity discovery is employed so as to search neighboring nodes dynamically. Therefore, the clients only need to download the video data when the nodes are decoding encoded data.

LT codes of rateless channel coding [3] and the H.264 scalable video coding (SVC) [9] are both considered and integrated to provide reliable and scalable video services over UDNs. Data allocation of hot video clips is also an essential issue to be considered to balance the storage space, transmission bandwidth and fault tolerance. In our solution, LT encoded data of these hot videos are randomly distributed among the UDN nodes (small cells or attached servers). The benefits of random allocation are three folds: (1) The participating nodes can be randomly chosen instead of selecting the specific nodes during the decoding process; (2) Since the data are randomly allocated among nodes, the source data can be recovered with high probability when multiple node failures occur; and (3) The surviving nodes can be used to repair the failed nodes by regenerating the encoded data if available. In the delivery process, the distributed decoding and broadcasting is proposed. The clients can receive and aggregate the required data through the broadcasting of those participated nodes simultaneously to get the benefit of bandwidth aggregation.

The contributions of our work include the following.

(i). A feasible solution for streaming hot videos over ultra-dense wireless environments is given. Initially, hot videos are both SVC encoded and LT encoded, and the coded data are randomly distributed among nodes.

(ii). Two random allocation schemes are considered to balance the storage space, transmission bandwidth and fault tolerance requirements.

(iii). A distributed decoding and delivery scheme to achieve

978-1-4673-7194-0/15/$31.00 ©2015 IEEE

the goal of bandwidth aggregation is proposed.

(iv). The source data can be recovered with high probability when transmission errors and/or node failures occur. And the surviving nodes can be used to repair the failed nodes.

The rest of this paper is organized as follows. In Section II, we briefly describe the concept of LT codes, SVC coding, belief propagation, and related works. In Section III, our proposed solution: distributed decoding and delivery is described and discussed. In Section IV, simulation results are presented. Finally, the conclusions and future works are given in Section V..

II. BACKGROUND AND RELATED WORKS

A. Fountain Codes and LT codes

The basic idea of Fountain codes [3] is that receivers can successfully recover K original source symbols with high probability when they receive enough (little larger than K) encoded packets which generated from a given set of source symbols by the senders. Fountain codes are a class of rateless code which means the number of encoded packets that can be generated limitless and determined on the fly. There are a variety of applications of fountain codes, including robust distributed storage [4], wireless sensor networks, peer-to-peer applications [5], and delivery of streaming content [6], [7], [8].

Luby Transform codes (LT codes) [3] are the first class of practical fountain codes. LT codes employ the particularly simple operation XOR, thus the coding complexity is quite low. In LT codes, the decoder is able to recover K source symbols from any subset of $K + O(\sqrt{K} ln^2 (\frac{K}{\delta}))$ encoded symbols with probability 1-δ. Both encoding and decoding complexity are of the order $O(Kln(K/\delta))$, which are directly affected by on the average degree of Soliton distribution [3]. The performance of LT codes is dominated by the degree distribution. *Ideal Soliton distribution* is quite fragile since the expected size of the ripple is one. The *Robust Soliton distribution* ensures the expected ripple size large enough so that the ripple never disappears in the process of decoding with high probability. Figure 1 shows some typical degree distributions.

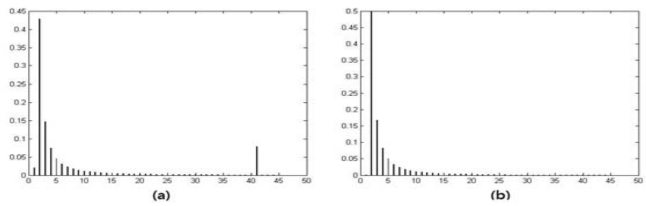


Figure 1: The LT degree distribution for K = 1000, (a) Robust Soliton distribution, c=0.1 and δ=0.5. Note that spike at K/R = 41; (b) Ideal Soliton distribution.

B. Belief Propagation

Belief Propagation (BP) is a well-known efficient algorithm for decoding LT encoded data [3]. The Figure 2 shows how BP algorithm is applied to decode data. The BP decoding chooses encoded data of degree one as the *ripple* set. In this example, the ripple set is {2} initially. The edges connecting to this symbol 2 are removed. After that, the new degree one symbols (5 in this case) can be released, and included in the ripple set. Decoding process can be done by performing BP approach iteratively until ripple set is empty.

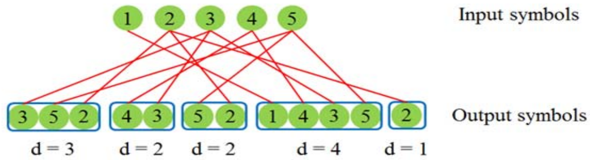


Figure 2: An example of LT encoding.

C. Scalable Video Coding

Scalable Video Coding (SVC) [9] had been developed as an extension of the well-known H.264/AVC standards. SVC provides a coding scheme of bitstreams with multilayer (one base and several enhancement layers) for three scalable capabilities: temporal, spatial, and quality. Then such a bitstream can be adapting to various client devices in error-prone heterogeneous networks. In other words, a client device that cannot receive all coded packets is tolerable; the SVC video decoder is still functioning with some level of quality degradation.

D. Channel Coding and Video Coding

In [7], after partitioning the video file into several data blocks, they are organized into different class according to the importance of layer which they belong to. Then, data blocks are distributed to Network Coding (NC) nodes. Each NC node can perform receiving, transferring, and encoding operations. In [7], the optimal rate allocation algorithm was proposed to achieve maximum video quality and find the appropriate distribution. Besides, applying channel coding as data protection when transmitting video data in error-prone networks, the multilayer property of SVC is adopted as well. The advantages of combining SVC and rateless codes will be discussed in the following sections. In [8], the received video quality of each client is influenced by the number of gateway (server) connections. In other words, a client is able to access higher layer of video data and get better video quality if the amount of gateway connections becomes large.

E. Data Allocation

The allocation problem of SVC layers in distributed storage of gateways has been discussed in [8]. Let α denote the layer of the SVC encoded video and let l_α be the amount of video data of layer α. Their allocation plan is: each gateway contains l_α/α share of data for layer α, i.e., each gateway has the base layer, but only half of the 1st enhancement layer data, one-third of the 2nd enhancement layer data, and so on.

F. Unequal Error Protection and Expanding Window

The concept of Unequal Error Protection (UEP) is to provide various levels of redundancy to different importance data block, i.e., the base layer can obtain more redundancy than any of enhancement layers. The related works of UEP and rateless codes are discussed in [10], [11], [12], and [13]. The idea of Expanding Window for UEP was proposed in [11]. All of N input symbols are partitioned into k classes according to their level of important, $n_1 + n_2 + \ldots + n_k = N$, where n_i is the number of input symbols in i^{th} class for i = 1, …, k. The i^{th} expanding window contains data from class 1 to class i.

III. THE RPOPOSED SOLUTION

In this section, the proposed data allocation and delivery scheme of hot videos are presented and discussed, as well the distributed decoding (belief propagation) algorithm of LT codes.

Our scheme is designated for UDNs. Assumed that the number of nodes serving a request in UDN can be represented by the following Poisson distribution [15], indicating that the probability of k nodes which can stream video to the client simultaneously.

$$P(N(A) = k) = \frac{e^{-\lambda|A|}(\lambda|A|)^k}{k!} \quad \text{Eq. (1).}$$

Where k is the number of nodes, λ represents the density of nodes per unit area, and $|A|$ is the area size. For example of λ=5 and $|A|$=1, the probability of the amount of nodes, $n = N(A)$, can be used to serve a request are: $P(n \geq 2)$=0.8599, $P(n \geq 3)$=0.8262, $P(n \geq 4)$=0.7420, $P(n \geq 5)$=0.6016, and so on.

A. Distributed Belief Propagation

To achieve parallel decoding from distributed storage, we proposed the scheme of distributed BP decoding. Once a request from a client issued, some nearby SC immediately responds and decodes its own LT coded data, then broadcasts the degree-one symbols to the requesting client and the neighboring SCs to continue the undone decoding process in parallel. As shown in Figure 3, the source data are encoded and allocated in two nodes. Node 2 begins to decode data when the request from a client was received. In Node 2, the symbol 2 is the only degree one symbol. After symbol 2 is released, it will be broadcasted to the neighboring nodes. After Node 1 received symbol 2, symbol 5 is released first and also symbol 3. Again, these symbols will be broadcasted to Node 2. The distributed decoding process is terminated as all of ripple sets are exhausted.

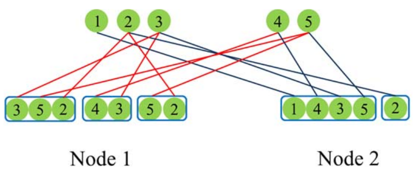


Figure 3: The decoding process of distributed BP.

B. Data distribution and Allocation Schemes

In data allocation scheme, after both SVC encoded and LT encoded, the coded data are randomly distributed among nodes (SCs or related attached servers). The random distribution has three benefits as listed in Section 1.

<u>Data Allocation</u>

According to the concept of UEP, we provide various levels of redundancy to data blocks with different importance. Of course, the base layer should have more redundancy than any of enhancement layers. The consideration of allocation should balance the storage space, transmission bandwidth and fault tolerance requirements. In our implementation, two allocations $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ and $\left(\frac{1}{2},\frac{1}{3},\frac{1}{5}\right)$ are suggested and evaluated (see Section IV). For each node (SC or related server), $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ means: the base layer is deployed with half of amount, the enhancement layer 1 is deployed with 1/3 amount and the enhancement layer 2 is deployed with 1/4 amount, respectively. In other words, the probability of any encoded symbol appearing in a node is 1/2 for the base layer, and 1/3 for the enhancement layer 1, and so on.

<u>Random Allocation with LT Encoded Packets</u>

The first allocation scheme employs the original LT coding for source data. First, video is SVC encoded. Then, each sub-bitstream (video layer) is independently LT encoded to generate encoded packets as shown in Figure 4. Then, the packets of a specific layer are randomly distributed to nodes with a limited amount, $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ or $\left(\frac{1}{2},\frac{1}{3},\frac{1}{5}\right)$or others according to the distribution in Eq. (1).

<u>Random Allocation with Expanding Window Packets</u>

The second allocation applies the Expanding Window technique [11] to encode source data. Similarly, video is SVC encoded, all of the sub-bitstreams (video layers) are combined into a single file in order of importance level. Then, we encode the combined file as illustrated in Figure 5.

C. Encoding with LT Codes

In our implementation, the degree distribution follows the robust Soliton distribution and the parameters c and δ are set as 0.02, and 0.01, respectively. For c=0.02 and δ=0.01, the probability of generating degree one symbols becomes extremely low, but the probability of degree two symbols increases instead. Those parameters are not suitable for the small number N of source symbols, which would cause the high probability of decoding failure. However, the large size of N is appropriate for applying them. In our experiments, the number of source symbols N is set as 1,000.

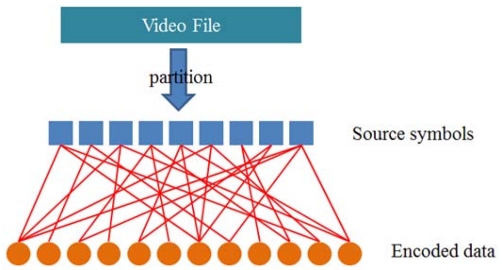


Figure 4: LT encoding process.

D. Delivery Scheme

By using distributed BP decoding, the degree one data from other nodes are needed. As a result, the degree one data are transferred not only for clients but also for the nodes requiring for new ripple symbols to prevent early decoding termination. Therefore, each node broadcasts its own self-generated degree one symbols to the neighbors participating in the decoding process. The distributed BP has advantage of parallel processing so as to reduce the execution time of LT decoding.

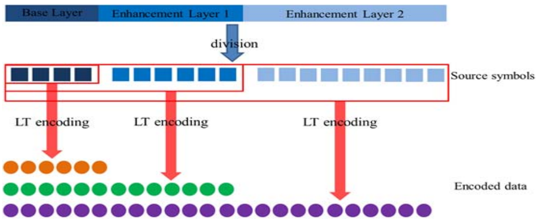


Figure 5: LT encoding process with Expanding Window.

Distributed Decoding and Broadcasting

As mentioned before, each node is allocated a limited amount of encoded data for base and enhancement layers. Thus, they all can perform decoding process by exchanging degree one symbols among nearby nodes sharing the same coverage of the radio station. See Figure 6.

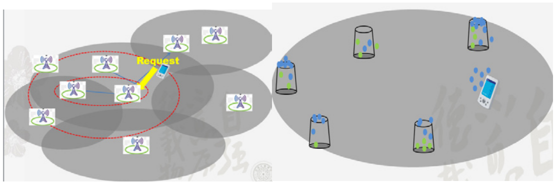


Figure 6: The illustration of distributed decoding.

Due to the random deployment (allocation), the decoding process can be initiated at any node. In other words, it is not necessary that a node act as the master server to control other nodes. Any node can be an initiator or a participator asynchronously. The flow of the distributed decoding process is depicted in Figure 7.

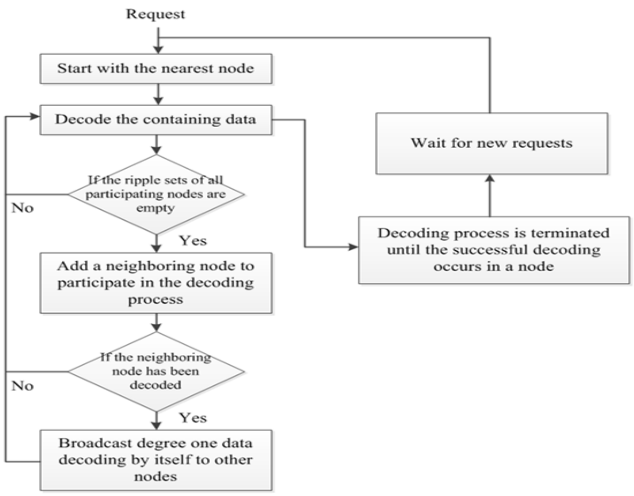


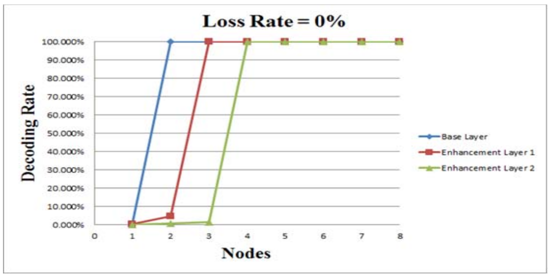
Figure 7: The flow chart of distributed decoding process.

IV. EXPERIMENTAL RESULTS

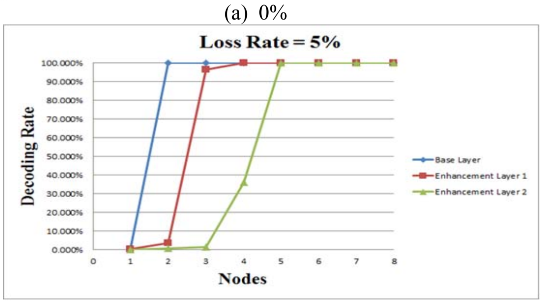
Our simulation was implemented on the environment of cloud platform with ITRI Cloud OS Solution (http://openstackdays.com/files/A5_ITRI.pdf). Each VM is equipped with Intel E5645 2.40 GHz, 2G RAM, and Win7 64bits. The test video is "Vidyo.yuv" with resolution 1280X720, 60 frames per second and coded as three-layer SVC. Our goal is to achieve average performance of quality: one base layer plus one enhancement layer.

A. Decoding Rate

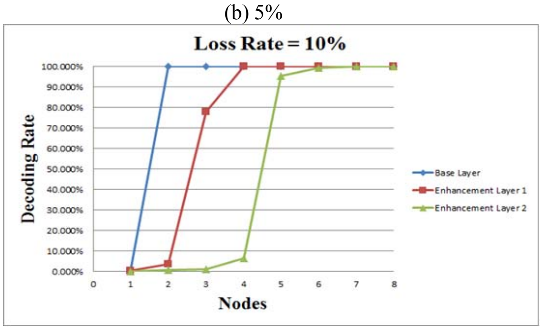
The successful decoding rates of random allocation with parameters $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ of loss rates 0%, 5%, and 10% are shown in Figure 9(a)-(c). The base layer can be completely decoded as the number of nodes down to 2. And 3 nodes and 4 nodes are needed for enhancement layers 1 and 2, respectively. As for loss rates 5% and 10%, more nodes will participate in decoding due to the loss. Consider Figure 9(d), the Expanding Window case with loss rate 10%. Comparing to random allocation, more number of nodes are going to participate in decoding because data of different layers are mixed in encoding. The higher layers have to wait for the lower layers. This situation becomes worst as the loss rate increasing.



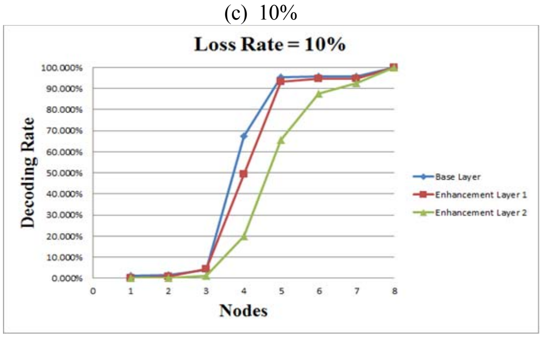
(a) 0%



(b) 5%



(c) 10%

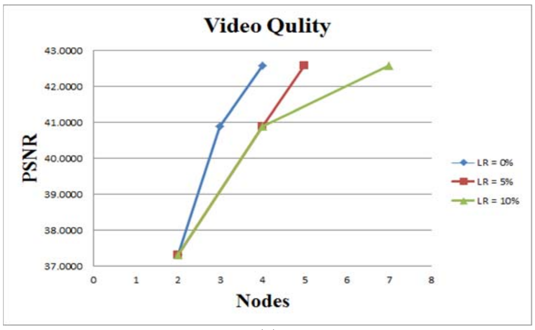


(d) Expanding Window with loss rate 10%

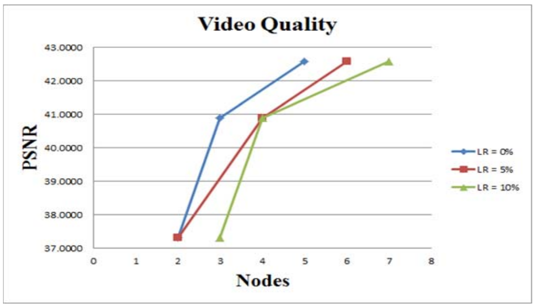
Figure 8: The decoding rates of random allocation $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ with loss rates (a) 0%, (b) 5%, (c) 10% and the Expanding Window with loss rate (d) 10%.

B. Video Quality

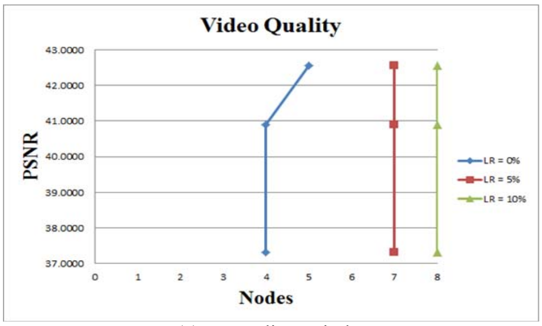
The video quality is illustrated in Figure 9 with different loss rates. We can observe that the random allocation with parameters $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ has the better quality with less number of service nodes than that of other allocations. As mentioned, the goal is to achieve one base layer plus one enhancement layer for clients. The allocation $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ achieves this goal with all three loss rate 0%, 5%, and 10%. In the case of no loss, all the layers can be completely decoded with four nodes. The allocation $\left(\frac{1}{2},\frac{1}{3},\frac{1}{5}\right)$ also achieve the goal with 4 nodes. However, the allocation with Expanding Window fails to achieve since the drawback mentioned above.



(a)



(b)

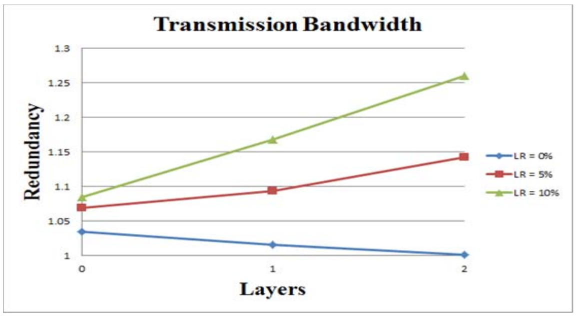


(c) Expanding Window

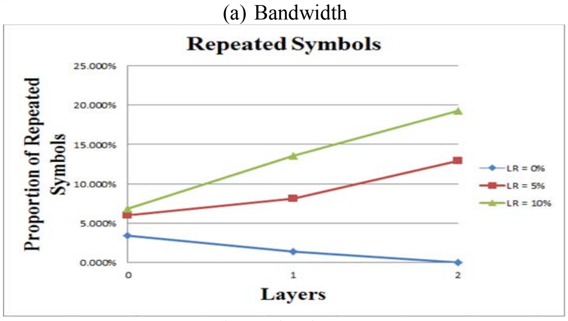
Figure 9: PSNRs of SVC layers with random allocations, (a) $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$, (b) $\left(\frac{1}{2},\frac{1}{3},\frac{1}{5}\right)$, and (c) Expanding Window.

C. Transmission Bandwidth

The transmission bandwidth of the randomly allocation with parameters $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ is depicted in Figure 10(a). The transmission bandwidth is slightly larger than the source data when the loss rate is 0%. Due to the data loss, the transmission bandwidth is enlarged to ensure the lost data can be successfully received. It is easily seen that the enhancement layer 2 required more bandwidth than others because more nodes to participate in decoding. Moreover, the required bandwidth of the 2nd enhancement layer drops when the loss rate is 0%. The situation seems odd; however it can be explained by the proportion of repeated symbols as shown in Figure 10(b). Consider the allocation of $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$, each base layer has half of encoded symbols, thus some degree one symbols may be generated by two nodes simultaneously, thus the proportion of repeated symbols is higher than that of enhancement layers 1 and 2.



(a) Bandwidth



(b) Redundancy

Figure 10: The transmission bandwidth and the rate of repeated symbols (transmission redundancy) of random allocation $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$.

D. Storage Space

Table I lists the storage space requirement and proportion of each video layer in different allocation schemes. The amount of data allocated in each node, random allocation with $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ requires the largest storage space than others while achieves the desired video quality with less nodes. The allocation with Expanding Window requires least storage space, but it fails to achieve the goal of video quality.

Table I. Storage space of allocation schemes.

Allocation	Storage Space	Proportion		
		Base Layer	EN Layer 1	EN Layer 2
$\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$	2.7303x	0.4615	0.3076	0.2307
$\left(\frac{1}{2},\frac{1}{3},\frac{1}{5}\right)$	2.3468x	0.4838	0.3225	0.1935
Expanding Window	2.0603x	0.5000	0.3333	0.1666

For unbiased comparison, we extend the storage space of Expanding Window to 2.5799x. Unfortunately, as the loss rate is 10%, it cannot achieve as well, see Figure 11.

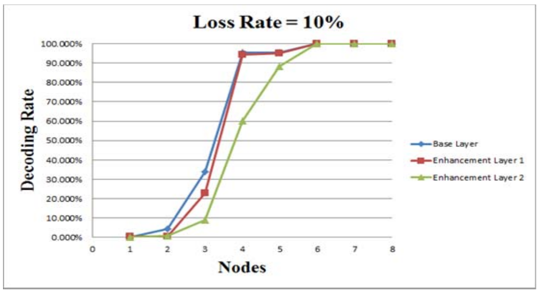


Figure 11: The decoding rate of Expanding Window of loss rate 10% with storage space 2.5799x.

E. Bandwidth per Storage Unit

Table II lists the bandwidth requirements for different schemes. We observe that allocation with $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$needs least bandwidth to decode video since the amount of participating nodes is less than that of others. As mentioned, allocation with Expanding Window requires more nodes than others; it needs more bandwidth obviously.

Table II. Bandwidth of allocation schemes.

	Total Bandwidth		
	Loss Rate=0%	Loss Rate=5%	Loss Rate=10%
$\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$	7.9108 MB	8.9009 MB	9.7333 MB
$\left(\frac{1}{2},\frac{1}{3},\frac{1}{5}\right)$	7.9177 MB	9.1533 MB	9.8143 MB
Expanding Window	7.9956 MB	9.0236 MB	9.7013 MB

V. CONCLUSION AND FUTURE WORK

Mobile video is the largest and fastest growing segment, and MCDN is a feasible platform to serve huge demand of mobile popular video traffics. As CDN, MCDN takes full advantage of caching in the network edge in such a way that the buffered video can be delivered with less network latency and traffic load. This paper evaluated the distributed delivery techniques for streaming hot videos over ultra-dense wireless environments. Initially, in the allocation phase, a hot video clip is both SVC encoded and LT encoded. Then, the coded data are randomly distributed among nodes (SCs or related attached servers). Two random allocation (distribution) schemes, LT codes and LT codes with Expanding Window are considered to balance the storage space, transmission bandwidth and fault tolerance requirements. The video quality (one base layer plus one enhancement layer) for clients is guaranteed. The allocation $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ achieves this goal with all three loss rates 0%, 5%, and 10%. The successful decoding of random allocation with parameters $\left(\frac{1}{2},\frac{1}{3},\frac{1}{4}\right)$ of loss rates 0%, 5%, and 10% are also proved. The transmission bandwidth is slightly larger than the source data when the loss rate is 0%. Due to the data loss, the transmission bandwidth is enlarged to ensure the lost data can be successfully received. Finally, the storage space requirement and proportion of each video layer in different allocation schemes are presented and discussed.

The wireless P2P streaming solution with hybrid caching could be considered in the future work to reduce the cache memory size and enhance the bandwidth efficiency. And, to further improve the cache placement and reduce the backhaul congestion, the concept of proactive caching to develop a popularity-based pre-caching scheme is a possible solution

REFERENCES

[1] Insoo Hwang, Bongyong Song, and Samir S. Soliman, "A Holistic View on Hyper-Dense Heterogeneous and Small Cell Networks," *IEEE Communications Magazine*, June 2013, pp. 20-27.

[2] Afif Osseiran, et al., "Scenarios for 5G mobile and wireless communications: the vision of the METIS project," *IEEE Communications Magazine*, Vol. 52, No. 5, May 2014, pp. 26 - 35.

[3] M. Luby, "LT Codes," in Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science, November 2002, pp. 271–280.

[4] Henry C. H. Chen, Yuchong Hu, Patrick P. C. Lee, and Yang Tang, "NCCloud: A Network-Coding-Based Storage System in a Cloud-of-Clouds," *IEEE Transactions on Computers*, Vol. 63, No. 1, January 2014.

[5] Valerio Bioglio, Rossano Gaeta, Marco Grangetto, and Matteo Sereno, "Rateless codes and random walks for P2P resource discovery in Grids," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 25, No. 4, April 2014.

[6] Hsu-Feng Hsiao, and Yong-Jhih Ciou , "Layer-aligned Multi-priority Rateless Codes for Layered Video Streaming," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. PP, No. 99, January 2014.

[7] Nikolaos Thomos, Jacob Chakareski, and Pascal Frossard, "Prioritized Distributed Video Delivery With Randomized Network Coding," *IEEE Transactions on Multimedia*, Vol. 13, No. 4, August 2011.

[8] Mohsen Sardari, Eun-Seok Ryu, Faramarz Fekri, and Nikil Jayant, "Multilevel Diversity Coding via Rateless Codes for Reliable and Scalable Video Multicasting," *IEEE Communications Letters*, Vol. 17, No. 5, May 2013.

[9] Heiko Schwarz, Detlev Marpe, and Thomas Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 17, No. 9, September 2007.

[10] Shakeel Ahmad, Raouf Hamzaoui, and Marwan M. Al-Akaidi, "Unequal Error Protection Using Fountain Codes With Applications to Video Communication," *IEEE Transactions on Multimedia*, Vol. 13, No. 1, February 2011.

[11] Dino Sejdinovic, Dejan Vukobratovi ´ c, Angela Doufexi, Vojin Šenk, and Robert J. Piechocki, "Expanding Window Fountain Codes for Unequal Error Protection," *IEEE Transactions on Communications*, Vol. 57, No. 9, September 2009.

[12] Kai-Chao Yang and Jia-Shung Wang, "Unequal Error Protection for Streaming Media Based on Rateless Codes," *IEEE Trans. on Computers*, Vol. 61, No. 5, pp. 666-675, 2012.

[13] Kuo-Kuang Yen, Yen-Chin Liao, Chih-Lung Chen, John K. Zao, and Hsie-Chia Chang, "Integrating Non-Repetitive LT Encoders With Modified Distribution to Achieve Unequal Erasure Protection," *IEEE Transactions on Multimedia*, Vol. 15, No. 8, December 2013.

[14] Ghid Maatouk, and Amin Shokrollahi, "Analysis of the second moment of the LT decoder," *IEEE International Symposium on Information Theory*, June 28 2009-July 3 2009, pp. 2326–2330.

[15] Benyuan Liu , and Don Towsley, "A Study of the Coverage of Large-scale Sensor Networks," *IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, Oct. 25-27 2004, pp. 475–483.