Received: March 10, 2016

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Comparative Analysis of Utilization Based Hybrid Overlay for Live Video Streaming in P2P Network

Kunwar Pal[1*], Mahesh Chadra Govil[1], Mushtaq Ahmed[1]

[1]*Department of Computer Science and Engineering,*
Malaviya National Institute of Technology Jaipur, India
* Corresponding author's Email: kunwar.11mar@gmail.com

Abstract: With a sudden surge in demand for live video streaming using peer to peer (P2P) network, there has been a rise of traffic in media transmission applications that employ this P2P approach. P2P is a decentralized media communication method which is more cost effective, and scalable. However, the traditional peer to peer network architecture has several limitations, and overlay construction is one of them. The two different unstructured overlay approaches like tree and mesh have already been discussed. This paper gives a detailed comparison of various overlay approaches in P2P networks. We have also proposed and implemented a utilization based hybrid overlay approach for the P2P network. The two crucial parameters which are primarily considered while overlay construction is: peer upload bandwidth and peer resource utilization. For the purpose of verifying our approach, we have done a comparative analysis of the proposed utilization based approach with the popular DenaCast approach. The different QoS parameters like start-up delay, playback delay, end to end delay, frame loss ratio are used for verifying our approach. The simulation results presented in this paper provide better network performance and quality of video at receiver side.

Keywords: Peer to Peer network, Live video streaming, Resource utilization, Content Delivery Network, ISP traffic.

1. Introduction

In the last couple of years, an exponential increase has been witnessed in the number of users that has contributed to a steep rise in the requirement of online video streaming. Factors like easy accessibility and availability of different platform for video streaming have also added to the increase in popularity of online video streaming. Some popular user applications of online video streaming are YouTube [1], NetTv [2], IPTV [3] etc. According to a study conducted by Cisco [4], it has been estimated that till 2019, 80-90% of data transmission over the internet will be only due to video transmission. Video streaming can be classified into, live video streaming and video on demand. The maintenance and complexity of video on demand is less as compared to live video streaming. No real time constraints are applicable to video on demand as they are applicable in live video streaming. Video can be enjoyed by the user only at a specific broadcast time in live video streaming while; the user can enjoy the video according to their convenience in the video on demand [5]. Complexity and cost of the server side increases due to the exponential increase of users for video streaming in traditional client-server architecture. In traditional client-server architecture, the server is the only source who can fulfil the demand of each peer. The upload capacity of the server is limited, as the number of user increases this upload capacity decreases and performance of the whole network as a unit also decreases due to the same reason [6].

Peer to Peer (P2P) computing can be a solution to the issues in traditional client-server architecture. It is a distributed computation mechanism. Due to its properties like scalability, easy to maintain, low cost and complexity at the server side, the popularity of P2P network is increasing rapidly. Resources of receiver peers are also used for the improvement of the network, and thus overall resource utilization of network is improved. The addition of a new peer doesn't directly affect the server capacity. The availability of neighbour peers in the P2P network makes it scalable and it also helps to decrease the congestion at the server side. P2P network is more reliable and fault tolerance system as compared to traditional client-server architecture. A combination of both traditional client-server and the P2P network is more efficient and provides the solution to the problems of both the architectures [7]. This integration of the P2P network approach with live video streaming has been implemented in CoolStreaming [8]. It is a milestone work in the field of live video streaming for P2P networks. This approach using CoolStreaming for content distribution is the same as in BitTorrent [9]. Content distribution and overlay construction are some of the principal issues in video streaming; CoolStreaming uses swarm based approach and gossiping protocols for content distribution and overlay construction respectively. The basic idea with design issues and implementation of CoolStreaming are discussed in detail by SusuXie et al. [10]. An improvement of CoolStreaming, its design theory, and practical implementation is discussed by [11], where churn is described as a crucial factor in P2P live streaming.

In this paper, we would discuss all the issues related to the overlay construction. Our primary concern is that how the end devices are connected to each other? The Peer should be connected in such a way that they can get the appropriate data without missing the frame deadline. We have also proposed and implemented a utilization based hybrid overlay approach for the P2P network. The two crucial parameters which are primarily considered while overlay construction is: peer upload bandwidth and peer resource utilization. For the purpose of verifying our approach, we have done a comparative analysis of the proposed utilization based approach with the popular DenaCast approach. The simulation results presented in this paper provide better QoS parameters like start-up delay, playback delay, end to end delay, and quality of video at receiver side.

2. Related Work

A network of computers that is constructed on the already present network is called an overlay network. Peers in the overlay network are connected by a logical or virtual link. But, the actual or the underlying network is connected using physical links. Traditionally, overlay construction in the P2P network can be classified into structured overlay and unstructured overlay [12]. Structured overlay construction uses data to key assignment procedure to maintain a record for each peer that is available in the network. A graph based map is created for matching peer to data key. Some of the examples of structured overlay are Pastry [13], Chord [14], Viceroy [15], Tapestry [16] and Content Addressable Network (CAN) [17]. The structure is reliable, but the maintenance of structured overlay is complex. The graph formed from this approach is fixed, and rigidity of this approach is maximized. This overlay construction approach is also not able to support the complex queries. The unstructured overlay is more popular in P2P network as compared to the structured overlay. It is less complex to maintain unstructured overlay than to a structured overlay. According to the current condition of the network; a peer can form the overlay instead of following the pre-defined graph map. Traditionally, unstructured overlay construction in the P2P network can be classified into; tree overlay and mesh overlay.

2.1 Tree overlay

Data transfer between the peer is the prime responsibility of network layer. The basic approach that is used for data transfer in the P2P network is IP multicast. It is a very simple, efficient and popular approach to transfer data between the peers. This approach is easy to implement. However, there are some issues in IP multicast which are yet to be solved like scalability, congestion, Network Address Translation (NAT) that can also lead to degradation of network performance [18]. A solution to the above issues can be achieved by using the application layer multicast which has been gaining popularity over the last few years. Application layer multicast is also known as end system multicast. In Application layer multicast, the end devices have the authority to forward the data further to other peers available in the network whereas, in IP multicast the end devices work only as the receiving devices. The end devices don’t have the forwarding ability in IP multicast [5]. Tree overlay is a simple and efficient approach that supports multicasting. It is less complex and easy to maintain compared to the other overlay approaches available in the P2P network. This is because a parent-child relationship is maintained in the tree overlay. This parent-child relationship avoids the loops in the tree overlay, and it also provides the data without replication. Source or video server is available at the zero level, and a hierarchy of peers is created that comprises of multiple levels. As the number of peers in the network increases; the number of levels in the tree overlay also increases. A peer can only connect to one parent at a time, and a parent peer can have any number of children in a simple tree overlay. So, parent peers forward the video content to all its available child peers in the tree overlay. The child peer can also do the same for their children, so the media content is reachable to all the peers available in the network. The push-based scheduling scheme is used in tree overlay. In this scheme, the parent peers send the data to all their available children in the network. Some of the examples of tree-based overlay are ESM and NICE [19].

The structure of tree overlay is fragile, and each child peer is fully dependent on its parents for media transmission. Parent-child relationship between the peers in tree overlay decreases the start-up delay which is not the case in other overlay structures in the P2P network. The leaf peers can only download the media content from their immediate parents, but they cannot upload the content in the network so their upload bandwidth is not used and the overall performance of the network drops. Whenever the connection between the parents and its children is weak, then the child peers have to find new parent peers again. This affects the performance of the network. Peers in the P2P network are mobile in nature. Whenever parent peers leave the network, it will affect all its children (immediate or hierarchical children). The position of the dynamic peer in a tree overlay also affects the performance of the network, if dynamic peers are higher in the hierarchy, then the complexity and maintenance of the parent-child relationship are very high.

2.2 Mesh overlay

A solution to the tree overlay problem can be given by using the mesh-based overlay. Peer in mesh-based overlay follow the property of complete mesh. In the complete mesh approach, each peer is connected to every other peer available in the network. Push based scheduling scheme is not used in the mesh-based overlay. Push based scheduling with mesh-based overlay provides the replication of media content in the network. Thus, peers can receive the media content from more than one peer. So, if the peer is dynamic in nature and leaves the network frequently, in that case, the child peer can still retrieve the media content from the other peers. The impact of a dynamic peer is less in mesh overlay as compared to that in tree overlay. This makes the mesh overlay network more reliable in comparison to the tree overlay network. Peers who are lower in the hierarchy in the mesh-based overlay can also upload the content. So the bandwidth utilization of the peer lower in the hierarchy is higher as compared to that in tree overlay. This affects the overall resource utilization of P2P network [20]. Due to the frequent exchange of notifications mesh overlay suffers from efficiency latency trade-off. Chainsaw, CoolStreaming, Bullet, and Anysee, are the examples of mesh-based overlay [21]. Bandwidth-bottleneck and content-bottleneck are the two main problems which arise while using live video streaming in a P2P network. P2P live video streaming using mesh-based overlay is discussed in PRIME in detail [22]. A solution to the bandwidth and content-bottleneck problems is given in this PRIME approach. For content-bottleneck, efficient pattern delivery approach is used, and for bandwidth-bottleneck, the bandwidth degree approach [23] is used. Push based scheduling scheme doesn't work in mesh overlay. So, pull based scheduling scheme is more efficient due to lack of replication of media content. Pull based scheduling scheme provides a near to optimal solution for throughput and bandwidth utilization issues in the mesh based overlay. The assumptions made for the approach are that the server bandwidth is three times more than the raw streaming rate and the group containing the number of peers should have less than 10,000 [24]. The approach also provides the new push-pull based hybrid scheduling scheme in mesh overlay. A novel P2P video streaming approach, TURINstream is defined by A. Magnetto et al. [25] to provide a solution to the QoS problems. The clusters are formed, and different overlay is used for different packets like control and media packets. The clusters are created using a distributed algorithm, so there is full utilization of upload bandwidth of peers. PlanetLab is used for creating and simulating the prototype of TURINstream. For testing the approach, a scenario is created using the dynamic nature of peer and flash crowd with limited upload bandwidth of each peer.

2.3 Hybrid overlay

Traditionally, the unstructured overlay can be classified into tree and mesh overlay. Maintenance of tree overlay is easy, control overhead of tree overlay is minimized, start-up delay and transmission delay are also less in tree overlay as compare to mesh overlay. But maintenance of tree overlay increases with an increase in the number of dynamic peers in the P2P network as they are placed higher in the hierarchy. While mesh overlay is more reliable and scalable, resource utilization of mesh overlay is more compared to that for tree overlay. And thus, mesh overlay is more complicated. Both the overlay approaches do not provide an optimal solution to the overlay construction problem in the

P2P network. So, some authors have tried to combine both these overlay approaches and provide a new aspect to the overlay construction problem. Q. Huang et al. [26] defined a novel approach for hybrid overlay construction. This approach is a combination of both the tree and mesh overlay approaches, tree overlay is used for control packet transmission while mesh overlay is used for transfer of media packets. For tree overlay, geographical location of the peer is used and for mesh overlay layered peer selection mechanism is used. The efficiency of each peer is calculated periodically, and inefficient peers are removed from the overlay. Only geographical location of a peer is used in the approach, and different crucial parameters like stability and bandwidth of the peer are not considered.

mTreebone is an approach which takes stability into prime consideration for creating the overlay. Stability of a peer is calculated using the amount of time for which the peer stays in the network and the peers that stay for more time as compared to optimal age threshold value in the network are called stable peers. Stable peers are the backbone of an overlay; tree overlay is created using the stable peers. The rest of the peers form a mesh overlay by connecting stable peers with the unstable peers. A hybrid push-pull scheduling scheme is used for media transmission between the peers. Scheduling and partitioning schemes provide a minimum transmission delay and control overhead for the approach; the simulation results also prove the same [27]. Only stability is not sufficient for overlay construction, the other parameters which play a crucial role in overlay construction are upload bandwidth, geographical location etc. An another approach which uses similar bandwidth range to create a tree overlay is discussed in HyPO [21]. In a geographical place, the peers that are in the same bandwidth range create the tree overlay. The depth of the tree plays a significant role in tree overlay. An even distribution of peers is done so that the depth of the tree is minimized. Further, a mesh is created between the peers that are at the same geographical location. Transmission time taken by the approach is less as compared to the other approaches due to the depth of the tree. The control overhead of the approach is also minimized. There is no clear criterion to distinguish between using the tree or mesh overlay in this approach. Also, the upload bandwidth of stable peers is not fully utilized in the approach. A group based CDN-P2P hybrid architecture (GCP2P) as defined in [28] is a combination of both the architectures of CDN (Content Distributed Network) and P2P. GCP2P leverages the properties of both the architectures and provides scalability, reliability, reduces control overhead and has less interrupt latency. Peer, which is physically near to CDN server, is considered as a super peer. Location based peer selection mechanism is used for creating the super peer. Area and channel for peer are also considered for creating a physical group named as sub-overlay. Simulation analysis of GCP2P provides less start-up delay compared to that for P2P and CDN individually. Bandwidth utilization of the peers in the P2P network is a prime consideration. To solve this issue, a Hybrid Live P2P Streaming Protocol (HLPSP) is described by the author to solve the problem of hybrid overlay network [20]. Overlay creation uses the upload bandwidth of each peer. So the peer that have maximum upload bandwidth should be higher in the hierarchy. Source peer is at the level *0* with the highest bandwidth. Simulation analysis of the approach is done with the DenaCast approach (an enhanced version of CoolStreaming). Through the simulation, authors show that the HLPSP provides better results compared to the DenaCast approach.

HLPSP consider the upload bandwidth of peers for overlay creation, other crucial factors like the stability of each peer, geographical location are not considered. A new hybrid overlay creation is discussed in our previous work [29], which consider all different crucial factors which are discussed above for overlay construction. This approach is a hybrid overlay combination of both tree and mesh overlay. Tree overlay is created using the stability of peers and mesh is created only at same level according to the upload bandwidth, age, and geographical location. Simulation result of the approaches shows that start up delay, playback delay, and the end to end delay is less for the approach as compared to that for DenaCast approach [30].

3. Utilization Based Overlay

The above discussion shows that the upload bandwidth of the peers is a prime factor in the P2P overlay network. The upload bandwidth of peer plays a significant role but if a peer doesn't have good connectivity to other peers or peer doesn't upload sufficient amount of data to other peers then its upload bandwidth doesn’t remain as efficient. So, not just the upload bandwidth of the peer but also the bandwidth utilization of each peer in the overlay also plays a significant role. In this paper, we will provide a new utilization-based approach for creating an overlay in the P2P network. The peers that utilize their resources more are given priority instead of those peers that have more upload bandwidth but properly don't utilize their resources. This overlay is a hybrid overlay, and it leverages the properties of both tree and mesh overlays.

Procedure to be followed when a new peer enters the network is described in Algorithm 1.

***Algorithm 1[A1]*: Procedure for New peer i**

1. $REQ_i^T < B_{TU}^i, C_{id}, G_i >$
2. If Request is new goto 3 else 4.
3. Position find for new peer P_i (B_{TU}^i, G_i)[A2]
4. Neighbour/Parent List CreationL_i. [A3]
5. $RES_T^i(L_i)$
6. Check update Periodically
7. $C\_req_i^j$ where (j $\subseteq L_i$)
8. $C\_res_j^i$(Positive/Negative)
9. $if(C_res_j^i)$goto 10 else 7 for new j from L_i
10. According to RTT time finds the best response
11. If continue step (12) else leave the network.
12. Periodically update status to Tracer goto (2).

When a new peer enters in the P2P network, it will follow the procedure as shown in algorithm 1. The new peer i sends the request to tracker T. This request is a tuple which consists of three variables $< B_{TU}^i, C_{id}, G_i >$ where B_{TU}^i is the Total upload bandwidth of peer i, C_{id} is the content id of media which peer i wants to view, G_i is the geographical location of peer i. Tracker receives the requested tuple and calculates the level for peer i in the overlay, which is shown in Algorithm 2.

Algorithm 2[A2]: Level Find (B_{TU}^i, G_i)

1. Find B_{TU}^S
2. Find Range R of network
3. Search location according to (C_{id}, G_i)
4. L_vel_i = Ceil[{ $B_{TU}^S - B_{TU}^i$ } / R] ;
5. Return L_vel_i;

Algorithm 2 finds the level of peers in P2P overlay by using the geographical location and upload bandwidth. After finding the level for peer i, tracker finds the list of best available peers for peer i. From this list, peer i can choose an appropriate neighbours/parent. Algorithm 3 is used for list creation by the tracker.

Algorithm 3:Neighbour/Parents Creation [A3]:

1. if (L_S<Max (S))
2. L_i.insert(S)
3. level = 1
4. While(level<= L_vel_i&&L_i<Th_1)
5. { For(i=0; i<n; i++)
6. { If(P_U^i>Th_2&& $P_o^i != P\_active_{max}^i$)
7. L_i.insert(P_U^i) }
8. If(L_i<Th_1)
9. { For(i=0; i<n; i++)
10. { If(If(P_U^i>Th_2 and pi has one peer at level less than L_vel_i and pi has space in L_i)
11. L_i.insert(P_U^i) }
12. Level++
13. } }
14. Level = 1
15. While(level <L_vel_i and L_i<Th_3)
16. { For(i=0; i<n; i++)
17. { If(P_U^i has at least one place left)
18. L_i.insert(P_U^i) }
19. If(L_i<Th_3)
20. { For(i=0; i<n; i++)
21. { If(P_U^i has at least one place less than L_vel_i and P_U^i has one place left)
22. L_i.insert(P_U^i) }
23. Level++
24. } }

For creating the list of possible parents, tracker uses the level of each peer and utilization of each peer. The server is considered at the highest level in the overlay. So priority is given to the server (line 1-2, algorithm 3) but if the peer is lower in the overlay, then update also takes place (line 11-20, algorithm 3). The list contains the peers only for lower levels (server is at the 0 level and as the height of the tree increases the level also increases). And for each level, only those peers are added to the list that has utilization value greater than the threshold (Th_2). Tracker adds the new peer to the list, if there is sufficient space in the list (Th_1) and if the level of the old peer is less than the level of new peer (line 4-13, algorithm 4). For calculating the utilization of each peer, only the data of its last three transactions is considered. And for a new peer that comes for the first time in the network, the average utilization value of the network is assigned to that peer. The overall overlay is created in such a way that the peers which have sufficient upload bandwidth are placed above in the hierarchy. For parent selection only those peers are considered which have a higher utilization ratio so; a new peer can receive the media content as quickly as possible.

After creating the list of possible parents/neighbour, tracker sends that list to requesting peer i. Peer i sends the connection request to neighbour peer from the list. Peer, i can send the connection request to more than one peer, and according to RTT (Round Trip Time) of the responding peer, it chooses the parent peer (line 5-10, Algorithm 1). Tracker provides the list according to the utilization, upload bandwidth, and geographical location; however, RTT time gives the idea of real-time congestion in the network. The nomenclature used in the paper is defined in Table 1.

In this paper, we simulate the approach and provide the results. Different parameters like start-up delay, the end to end delay, playback delay, frame loss ratio and packet drop due to destination unreachable are used to verify the result.

4. Simulation and Result

4.1 Simulation setup

This Utilization based overlay approach is implemented using the OverSim simulator. OverSim is a P2P network simulator and open source overlay for OMNET++ simulation environment (OS). For exchanging and processing network messages Discrete event simulation (DES) is used in OverSim [31]. Two different modules are employed in OverSim and processing between the modules is done using the C++ language and for topology creation, the Network Descriptor (NED) language is used. The parameters which are used in our simulation are given in Table 2.

4.2 Simulation results

Utilization-based overlay affects the different QoS parameters in P2P live video streaming. Start-up delay plays a significant role in QoS of P2P live video streaming. Start-up delay is directly affected by the overlay. In the utilization based overlay, peers form an overlay according to their bandwidth utilization. So, peers that are using their upload bandwidth more are getting higher priority, and there is a high probability of them being selected as parents to other peers. This directly affects the start-up delay. If a peer chooses those peers (utilized peers) as their parents then start-up delay is less compared to the DenaCast approach (an enhanced version of CoolStreaming). Initially, start-up delay for both the approaches is approximately same, but as the number of peers increases, start-up delay for DenaCast approach becomes more as compared to that for utilization based overlay as shown in Fig. 1.

End to end delay between the peers is also a crucial parameter for P2P networks. If the parent peers are not uploading the content, then the end to end delay at receiver side increases and that degrades the quality of media content which receiver peer wants from its parent. In the utilization based overlay, the parent peers are chosen according to their upload bandwidth and bandwidth utilization; so the peers that are active and have sufficient amount of bandwidth are chosen. This is the reason

Table 1. Parameter used in algorithm.

S. No	Parameter	Meaning
1.	Max (S)	Maximum number of peer that can connect to server
2.	$\mathrm{P\_active}^{i}_{max}$	Maximum number of active peers
3.	Th_1	Maximum passive peers of new peer = B_u/frame size
4.	Th_2	Average Utilization of Network
5.	$\mathrm{L\_vel}_i$	Level of peer i
6.	Th_3	90% of maximum possible neighbours
7.	P^{i}_{U}	Utilization ratio of peer i, {Uploaded bandwidth / total bandwidth }
8.	L_i	list for peer i
9.	$B_{u[pk]}$	Total upload bandwidth of peer k
10.	$RES_T^i(L_i)$	Response from Tracker to peer i
11.	$\mathrm{C\_req}^{j}_{i}$	Connection request from peer i to j

Table 2. Simulation parameters and values.

S. No	Parameter	Value
1.	Simulation Duration	500s
2.	Buffer Map Exchange period	1 s
3.	Average Video Bit Rate	512 Kbps
4.	Source Number	3
5.	Video Codec	MPEG4 Part I
6.	Chunk Size	5 Frames
7.	Average Chunk Length	130Kb
8.	Number of Runs	10
9.	Neighbour Notification Period	2s
10.	Maximum Number of levels	6

Received: March 10, 2016

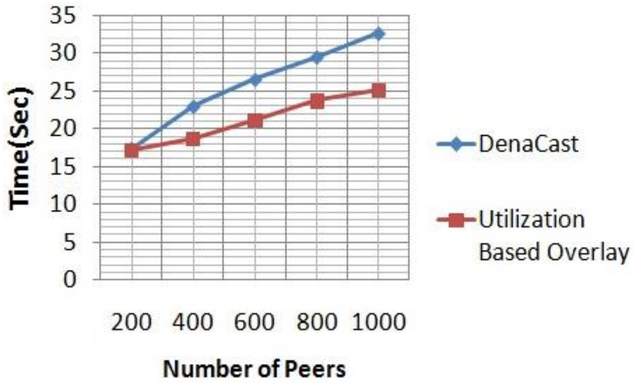

Figure.1 Average start-up delay

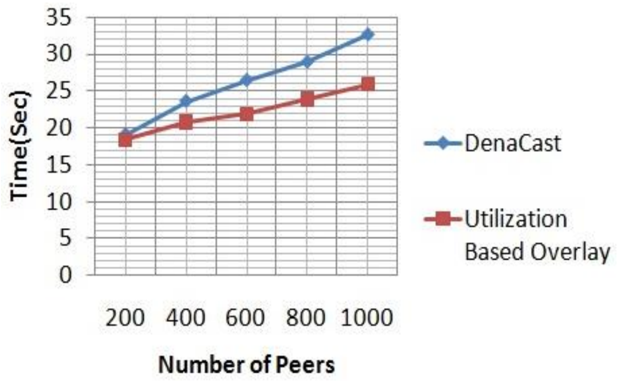

Figure.2 Average end to end delay

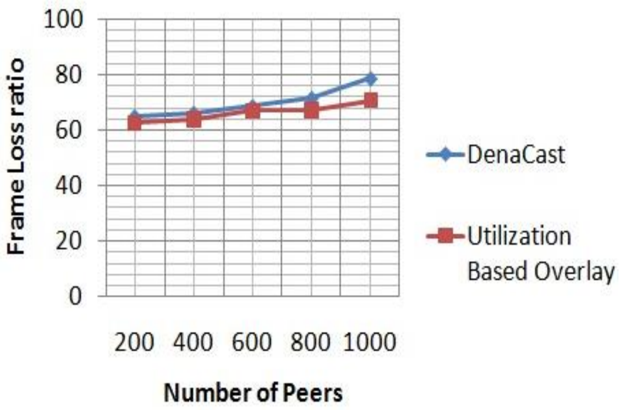

Figure.3 Average frame loss ratio

that why the end to end delay in utilization-based approach is more as compared to that in the DenaCast approach. Figure 2 shows a comparison of the end to end delay for both the approaches.

Frame loss ratio of DenaCast approach is more as compared to that for utilization based hybrid overlay approach. As the number of peers increases the frame loss ratio of DenaCast based overlay approach increases while the frame loss ratio in utilization based hybrid overlay approach is less. The reason of frame loss ratio in the P2P network is the number of frames which are lost in the network or are dropped due to late arrival.

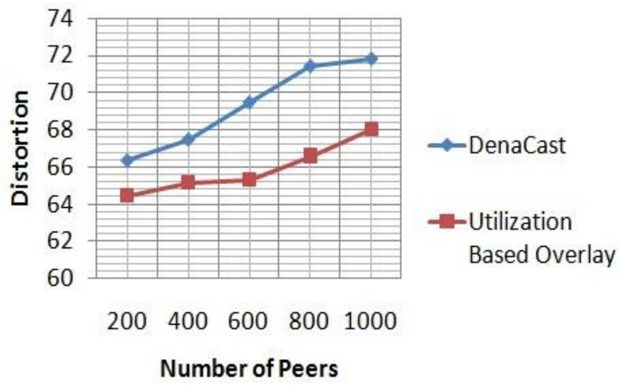

Figure.4 Average distortion

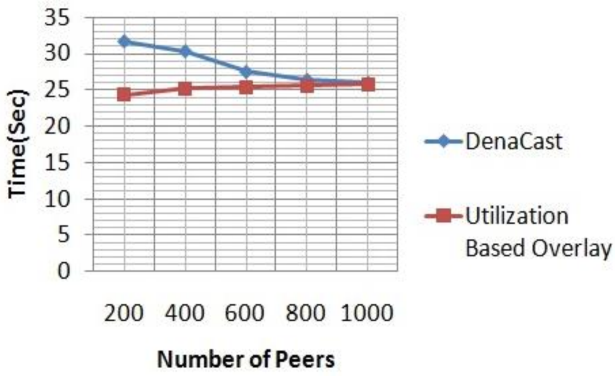

Figure.5 Average playback delay

Frames which arrive at the receiver side after the deadline are useless and just increase the congestion in the network. Utilization-based hybrid overlay approach gives priority to peers that have greater utilization, so frames are not lost, and the overall frame loss ratio is also less. A comparison of frame loss ratio to DenaCast and utilization based overlay approaches is shown in Fig. 3.

As the frames are lost or are unreachable to the receiver at the destination due to the expiry of the deadline of the frame, the distortion increases. Due to this reason for frame loss ratio, the distortion of video in utilization based hybrid overlay approach is less compared to that in the DenaCast approach.

Playback delay of utilization-based approach is a bit more as compared to that of DenaCast approach. As the number of peers in the network increases it is approximately same for both the approaches and the result is shown in Fig. 5. So overall QoS due to utilization based hybrid overlay increases and the overall performance of the network also increases.

5. Conclusion and Future Work

Peer to peer network is gaining popularity for live video streaming transmission due to its scalability, resource utilization, low complexity and cost efficient architecture. However, there are

 DOI: 10.22266/ijies2017.0630.30

various limitations in the traditional P2P network approaches and; one of them being overlay construction. In this paper, a comparison of different unstructured hybrid overlays for the P2P network is covered, and a new utilization based overlay construction in the P2P network is proposed. Upload bandwidth and resource utilization of each peer are used for overlay creation. Both the parameters are very useful for overlay creation because, whenever the highly utilized and high upload bandwidth peers are near to the source then it can also provide better media quality to the other peers. Different levels are created for the highly utilized peers in the mesh overlay. Due to these properties of utilization based overlay the QoS of P2P network also improves, and the different QoS parameters also prove the same. Network performance parameters like start-up delay, end to end delay and playback delay are improved while video quality at the receiver side is also improved due to the less frame loss ratio and less distorted frames. So utilization based overlay improves the overall experience of receiver in P2P live streaming. The future work aims at providing a detailed comparison of this utilization based hybrid overlay approach with the other different hybrid overlay approaches.

References

[1] “YouTube.” [Online]. Available: https://www.youtube.com/. [Accessed: 11-Jan-2017].

[2] “NetTv.” [Online]. Available: http://nettv.com.np/nettv/. [Accessed: 11-Jan-2017].

[3] “IPTV.” [Online]. Available: https://www.iptvonline.ca/. [Accessed: 11-Jan-2017].

[4] E. Summary, “Cisco Visual Networking Index: Forecast and Methodology, 2014-2019 White Paper - white_paper_c11-481360.pdf,” 2015.

[5] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. W. O’Toole Jr., “Overcast: reliable multicasting with on overlay network”, In: *Proc. of the 4th conference on Symposium on Operating System Design & Implementation*, p. 14, 2000.

[6] V. Venkataraman, P. Francis, and J. Calandrino, “Chunkyspread: Multi-tree Unstructured Peer-to-Peer Multicast”, In: *Proc. 14th IEEE Int. Conf. Netw. Protoc. - ICNP’06*, pp. 2–11, 2006.

[7] S. M. Y. Seyyedi and B. Akbari, “Hybrid CDN-P2P architectures for live video streaming: Comparative study of connected and unconnected meshes,” In: *Proc. of the 2011 International Symposium on Computer Networks and Distributed Systems, CNDS 2011*, pp. 175–180, 2011.

[8] X. Zhang, J. Liu, B. Li, and T. S. P. Yum, “CoolStreaming/DONet: A data-driven overlay network for efficient live media streaming”, In: *Proc. of the IEEE Infocom*, Vol. 3, No. C, pp. 13–17, 2005.

[9] L. D’Acunto, T. Vinko, and J. Pouwelse, “Do BitTorrent-like VoD systems scale under flash-crowds?”, In: *Proc. of the 2010 IEEE 10th Int. Conf. Peer-to-Peer Comput. P2P 2010 - Proc.*, no. Section IV, pp. 1–4, 2010.

[10] S. Xie, B. Li, S. Member, G. Y. Keung, X. Zhang, and S. Member, “Coolstreaming : Design , Theory, and Practice”, *IEEE Trans. Multimed.*, Vol. 9, No. 8, pp. 1661–1671, 2007.

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

[12] J. Crowcroft, M. Pias, R. Sharma, and S. Lim, “A survey and comparison of peer-to-peer overlay network schemes”, *IEEE Commun. Surv. Tutorials*, Vol. 7, No. 2, pp. 72–93, 2005.

[13] A. Rowstron and P. Druschel, “Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems”, *Design*, Vol. 11, No. November 2001, pp. 329–350, 2001.

[14] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan, “Chord: A scalable peer-to-peer lookup protocol for Internet applications”, *IEEE/ACM Trans. Netw.*, Vol. 11, No. 1, pp. 17–32, 2003.

[15] D. Malkhi, M. Naor, and D. Ratajczak, “Viceroy: A Scalable and Dynamic Emulation of the Butterfly,” In: *Proc. of the 21st annual ACM symposium on Principles of distributed computing*, pp. 183–192, 2002.

[16] B. Y. Zhao, L. Huang, J. Stribling, S. C. Rhea, A. D. Joseph, and J. D. Kubiatowicz, “Tapestry: A resilient global-scale overlay for service deployment”, *IEEE J. Sel. Areas Commun.*, Vol. 22, No. 1, pp. 41–53, 2004.

[17] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, “A Scalable Content-Addressable Network”, In: *Proc. of the 2001 Conf. Appl. Technol. Archit. Protoc. Comput. Commun.*, Vol. 31, No. 4, pp. 161–172, 2001.

[18] Y. H. Chu, S. G. Rao, S. Seshan, and H. Zhang, “A case for end system multicast”, *IEEE J. Sel. Areas Commun.*, Vol. 20, No.8, pp. 1456–1471, 2002.

[19] S. Awiphan, Z. Su, and J. Katto, “ToMo: A

two-layer mesh/tree structure for live streaming in P2P overlay network", In: *Proc. of the 2010 7th IEEE Consumer Communications and Networking Conference, CCNC 2010*, pp. 1–5, 2010.

[20] C. Hammami, I. Jemili, A. Gazdar, A. Belghith, and M. Mosbah, "Hybrid live P2P streaming protocol", *Procedia Comput. Sci.*, Vol. 32, pp. 158–165, 2014.

[21] H. B. H. Byun and M. L. M. Lee, "HyPO: A Peer-to-Peer based hybrid overlay structure", In: *Proc. of the 2009 11th International Conference on Advanced Communication Technology*, Vol. 1, pp. 840–844, 2009.

[22] N. Magharei and R. Rejaie, "Understanding Mesh-based Peer-to-Peer Streaming", In: *Proc. of the 2006 international workshop on Network and operating systems support for digital audio and video.*, p. 10, 2006.

[23] N. Magharei and R. Rejaie, "PRIME: Peer-to-Peer Receiver-Driven Mesh-Based Streaming", *IEEE/ACM Trans. on Networking*, Vol.17, No.4, pp. 1052 - 1065, 2009.

[24] M. Zhang and Q. Zhang, "Understanding the Power of Pull- based Streaming Protocol : Can We Do Better ? Presented by Rabin Karki Background", *IEEE J. Sel. Areas Commun.*, Vol. 25, No. 9, pp. 1678–1694, 2007.

[25] A. Magnetto, R. Gaeta, M. Grangetto, and M. Sereno, "TURINstream: A totally pUsh, robust, and effIcieNt P2P video streaming architecture", *IEEE Trans. Multimed.*, Vol. 12, No. 8, pp. 901–914, 2010.

[26] Q. Huang, H. Jin, and X. Liao, "P2P live streaming with tree-mesh based hybrid overlay", In: *Proc. of the Int. Conf. Parallel Process. Work.*, no. 60433040, 2007.

[27] B. C. Canada, "mTreebone : A Hybrid Tree / Mesh Overlay for Application-Layer Live Video", In: *Proc. of the Distributed Computing Systems, 2007. ICDCS'07. 27th International Conference on*, p. 49, 2007.

[28] T. N. Kim, S. Jeon, and Y. Kim, "A CDN-P2P hybrid architecture with content/location awareness for live streaming service networks", In: *Proc. of the Int. Symp. Consum. Electron. ISCE*, pp. 438–441, 2011.

[29] K. Pal, M.C. Govil, and A. Mushtaq, "A New Hybrid Approach for Overlay Construction in P2P Live Streaming", In: *Proc. of the ICACCI*, pp. 431–437, 2015.

[30] K. Pal, M.C. Govil, and A. Mushtaq, "Comparative Analysis of New Hybrid Approach for Overlay Construction in P2P Live Streaming", In: *Proc. of the ERCICA*, 2016.

[31] I. Baumgart, I. Baumgart, B. Heep, B. Heep, S. Krause, and S. Krause, "OverSim: A Flexible Overlay Network Simulation Framework", In: *Proc. of the 2007 IEEE Glob. Internet Symp.*, pp. 79–84, 2007.