Asian Journal of Computer Science and Technology
ISSN: 2249-0701 (P) Vol.6 No.2, 2017, pp.43-49

DOI: https://doi.org/10.51983/ajcst-2017.6.2.1781

Performance Improvement for Fast Video Streaming in Peer-to-Peer Network

Gajanand Sharma[1] and Dinesh Goval[2]
[1]Research Scholar, [2]Professor, Department of CEIT
Suresh Gyan Vihar University, Jaipur, Rajasthan, India
E-Mail: Gajanan.sharma@gmail.com, dinesh8dg@gmail.com
(Received 11 September 2017; Revised 30 September 2017; Accepted 12 October 2017; Available online 19 October 2017)

Abstract **- Peer-to-peer (P2P) systems are becoming increasingly popular due to their ability to deliver large amounts of data at a reduced deployment cost. While P2P systems faster the development of novel media applications, they also represent an interesting alternative paradigm for media streaming applications that can benefit from the inherent self organization and resource scalability available in such environments. In this paper, we are improving the packet delivery ratio for the Peer to Peer network. We are using Head-Tail Video Streaming methodology. As we can show from the results session, the comparison for the Enable Adaptive Solution and Proposed Methodology (Head-Tail Video Streaming). As we can see from the results Peer-to-Peer Packet Delivery Ratio is Low for the Enable Adaptive Solution as compare to Proposed Methodology (Head-Tail Video Streaming).**
Keywords**: Quality adaptive streaming, Peer-to-peer networks, Congestion control, Layered encoding**

I. INTRODUCTION

The P2P (Peer-to-peer) technology of content sharing namely Kazaa, Gnutella, and Napster are the applications which have been surprisingly successful on World Wide Web (www) or Internet. The P2P technology has achieved the large attention of public that through the Napster which usually the system supporting to music sharing on Internet. It's an interesting and emerging research technology having promising product-base.

The Intel P2P functioning (working) group gave P2P definition as, "The sharing of computer resources and services by direct exchange between systems" (by Kan, 2001). Hence, provides the P2P systems with the important two key characteristics:

1. Scalability: There's no technical limitation and algorithmic of system size, for example, system complexity would be the constant regardless the nodes number in system.
2. Reliability: The malfunction of any node provided won't affect the entire system (or can be even different nodes).

The networks of file sharing such as the Gnutella are a good example of reliability and scalability. In the Gnutella, peers get associated initially to the flat-overlay network, in that each peer found to be equal. The peers get associated directly that without master server's requirement malfunction and arrangement of any node doesn't cause different system nodes to the malfunction.

The P2P is able to be classifying into the two groups categorized by model type: The hybrid P2P and pure P2P. The pure model of P2P, like free-net and Gnutella, doesn't have the central server. The Hybrid P2P models like Magi, Grove and Napster uses the central server to gain the meta-data like peer identity on that the data get stored or need to verify the security credentials. In the hybrid model, the peers usually contact the central server, before straightly contacting the different peers.

A. P2P Networks Topologies

And according to (Peter, 2002), entire topologies of P2P, no matter how dissimilar they will be, will have the single feature. Whole file transfers usually made in between the peers are constantly done straightly through the connection of data connection which is made in between the file sharing peer as well as the that is peer for their requesting. Before the file transfer control process, on the other hand, it can be executed in number of different ways. As it stated by (Minar, 2001), the P2P networks of file sharing may be categorized into the four basic functions: the ring, hierarchical, decentralized, and centralized systems. Though, the certain topologies may exist, it's normally the distributed systems practice that to have the extreme complex topology by joining the number of basic systems to form, that is called now the hybrid systems. We are going to provide the concise introduction to all four basic systems and then delve deeper into hybrid system topic.

B. Centralized Topology

The centralized topological concept has been illustrated in the Fig.1; it is extremely depended on the model of conventional server/client model. The centralized server should exist that is utilized to manage user databases and files of the multiple numbers of peers which have log onto them (Peter, 2002). The server of client contacts needs to inform current IP-address of it as well as the names of entire files which it's desirous to share. It gets done every time with launched application. The peers data collected will further be utilized by a server in order to create the

centralized dynamical database, which maps the file names in order to set the IP-addresses.

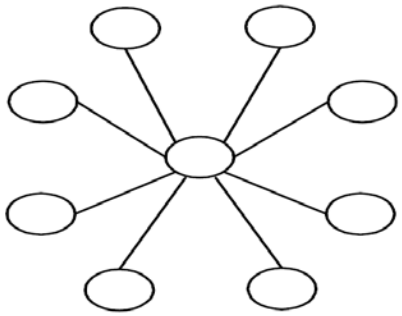

Fig1. The Centralized Topology Representation [16]

C. Ring Topology

It get cleared from the centralized topology drawback is which a central server will become the bottle-neck (when the load gets heavy) as well as the single failure point. There're different contributing factors that led to surfacing the ring topology observed in the Figure No.2. It's formed of the machine clusters which are arranged in different ring in order to act the distributed server (the Minar, 2001). The machines cluster will function together to give the good balancing of load balancing and the higher availability. The topology is normally utilized when entire machines are correspondingly close to network that means it is the most suitably owned by sole organization; where problem is not anonymity. The Figure No.2 presents the easy ring topology description.

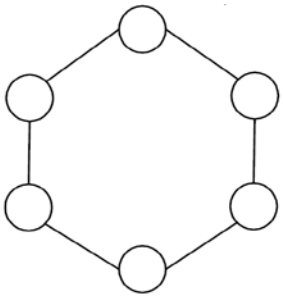

Fig.2 The Ring Topology Reorientation[16]

D. Hierarchical Topology

The hierarchical type systems come to an existence with human civilization beginning. Every matter from the easy family to the businesses as well as the Government gradually operates in the hierarchical manner. Today, number of the Internet applications operates in the hierarchical surroundings. The best hierarchical system example on an Internet is the DNS (Domain Name Service) (Minar, 2001). The authority directs from the root name-servers to registered name-servers and etc. Particular topology is very appropriate for the systems which need governance form which consists the delegation authority or rights. Other example of the system which makes hierarchical topology use would be the CAs (Certification Authorities) which certify internet of entity validity. The CA root is actually delegate different authoritative rights of it in order to companies which subscribe, so which all those companies are able to, in the grant certification turn to all those reside their underneath. The Figure No.3 gives the brief observation of how the hierarchical system observes like.

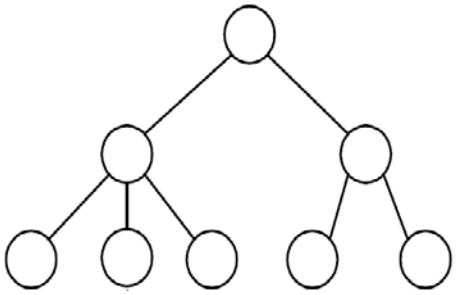

Fig.3 The Hierarchy Topology Observation[16]

E. Decentralized Topology

In the architecture of pure P2P, there are no centralized servers. Whole peers are nearly equal, therefore producing the unstructured, flat network topology (by Peter, 2002). Just refer the Figure No.4 for observation. So to join the network, all peer should initially, contact to the node of bootstrapping (node which usually online), that provides combining of IP-address peer to single or more available peers, officially it is making the ever dynamical network part. Every peer, wherever, will just have the data regarding the neighbors that are peers which have the direct edge in order to it in network.

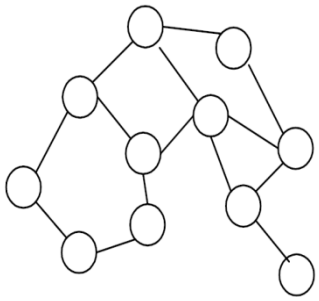

Fig.4 The Decentralized Topology Observation[16]

As there are no different servers for searches management, the files queries get flooded throughout network (by Kurose, 2003). The query flooding activity is not accurately the best ever solution, as it requires the great overhead network traffic. The application example are uses particular Gnutella model. It details that how it shares and searches the files in the pure network of P2P that will be observed in the section of Gnutella.

F. Hybrid Topology

And having discussed basic P2P networks topologies, we are now come to extreme complex systems of real world which normally gathers the several essential topologies into the single system. It is called as 'Hybrid Architecture' (by Yang, 2002). We are going discuss the number of examples that in section only to provide concise idea of architecture. In the different system, the nodes usually function more than single role.

G. Centralized and Ring Topology

Particular hybrid topology is extremely general sight in web hosting world (by Minar, 2001). As stated before in a section of ring topology, the web servers heavily loaded normally have the servers' ring which specializes in the failover and load balancing. Therefore, the servers preserve the ring topology. However the clients are associated to servers ring through the centralized topology (that is, server/client system) as illustrated in the Figure No.5. Hence, the whole system is equally hybrid; the mixture in between sturdiness of the ring topology that with centralized system simplicity.

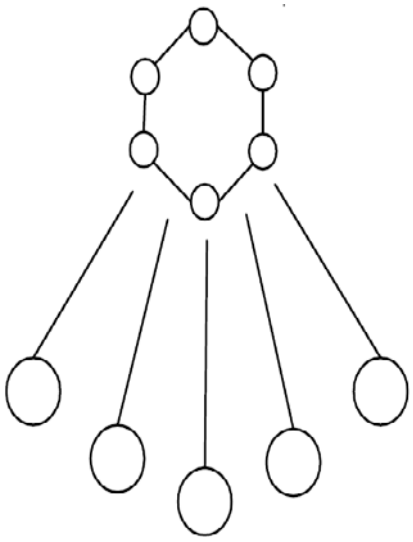

Fig.5 The Centralized & Ring Topology Representations [16]

H. Centralized and Centralized Topology

It is nearly a case where network server a client (itself) of the largest network (by Minar, 2001). Particular kind of hybrid topology observed in the Figure No.6 is the extreme general practice in the organizations which give the web services. The simple example which will assist to illustrate particular point would be, when the contacts of centralized servers and web browser. And the web server can function as well as format entire results in which they will be presented in the HTML-format and in a function of doing these, particular servers may contact (themselves) to different servers (for example, the Database Server) to gain the required data (by Minar, 2002).

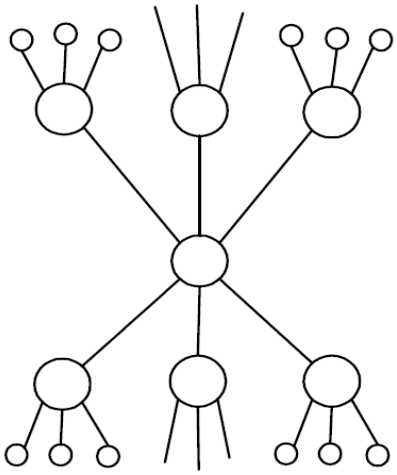

Fig.6 The Centralized Topology Representation [16]

I. Centralized and Decentralized Topology

In a topology, the peer which operates as the group leaders are represented (by Kurose, 2003). They recognized by number of names. Some of them call all them as the "Group Leader Nodes", Even Ultra Nodes or Super Nodes. In order to maintain the things consistent and simple with below sections regarding the Fast track (by Sharma 2002), we are going to refer all them as the "Super Nodes". Certain Super Nodes usually functions the centralized server tasks as that in a centralized topology; but just for peer's subset. All the Super Nodes usually tied with each other with the decentralized manner. Therefore, the particular type of hybrid topology introduces the two numerous control tiers. The initial is where the basic ordinary peers get associated to a "Super Nodes" with the fashion of centralized topology. And the second is that the Super Nodes get associated to each other in the fashion of decentralized topology as illustrated in the Figure No.7). And with centralized topology, all Super Nodes keeps the database which maps the file names to IP-addresses of entire peers which get assigned to it (by Yang, 2002). It require to be note here that database of Super Node's just keeps the peers track within own group of it. It largely minimizes the peer's scope which it requires to serve. So that any common peer with the connection of high-speed will be eligible for the Super Node. The best P2P example application which uses particular topologies should be the Kazaa/FastTrack. Different better example of particular topology should be the general Internet email. The Mail clients require the relationship decentralized to specific servers of mail. Such as the Super Nodes, particular mail servers need to share the emails in decentralized fashion along themselves.

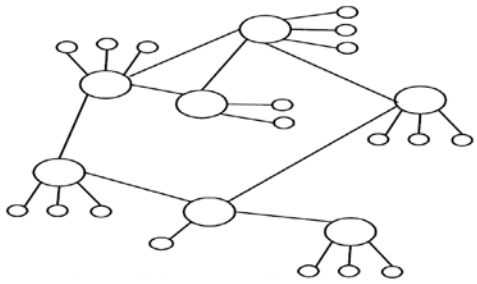

Fig.7 The Description of Centralized & Decentralized Topology [16]

J. Other Potential Topologies

Just note that the topologies mentioned here i.e. the hybrid topologies are only general ones. As it may be observed, there are large deals numerous combinations of the hybrid topologies which are able to be gained from general topologies. Whereas, if one required to make number of combinations; the resulting topology will definitely become so much complex, hence they are making all them very hard to manage.

II. PROPOSED METHODOLOGY

The greater element of systems (mesh-based) is the scheduling. A scheduler should be accurate in order to prevent the delayed and duplicate data. Almost entire system based on the mesh, as mentioned before employing the comparable method of scheduling. Typical scheduler schedules the layers chunks (or pieces) in the sequential order. For an example, it assigns the chunk 1 & 2 to the sender 1 and chunk 3 to the sender 2 and etc. Certain types of schedulers are that mainly behave the complex, when the node occur failure.

If the sender 1 failed and then left a network, the sender 2 or sender 3 will like to give up the packets that used to be sent by the support sender 1. Of course the scheduler need to consider precisely its preferred data where the other senders give up all the packets or need to continue the own sending packets. Hence, the scheduler's particular systems largely make the scheduling mistake. And by losing the scheduling, particular mistake will be resolved; as well as the bandwidth would be enhanced as well.

The task becomes difficult, when recently added peer-to a sender list can't meet receiver requirements. For an example, the sender has the bandwidth of lower upload as compared to the failed node. As the current codec video that utilized in the internet that is layered encoded, they are enough sensitive to the loss of sensitive packet. For an example, the LC is the codec based on stack that of one layer completeness is constrained to entire lower layers. In particular condition, the scheduler decisions made are extremely critical due to the scheduling mistake that may lead to single or number of layer losses.

The suggested Head Tail method, usually schedules the chunks easily while it is able to recover the temporal node failures. The Head-Tail is depended on certain fact in which every stream of bit has the two ends. And having the two ends usually assists the scheduler to the schedule chunks from the two points of data stream. Its similar bubble operation and the Quick type. In particular methods of sorting, the operation gets done from the starting as well as the array end that results to the better execution of certain type's algorithms.

The pair of Head-Tail of every two appropriate peers and then assigns every layer end to one of them. The peers begin to send the chunks from every end that until they reach to the rendezvous point. As the peer bandwidths are heterogeneous in nature as well as pairing their results in order to unequal entire bandwidth, the point of rendezvous can vary from pair to- pair. The pairing assists the scheduler to select greatest pair depended on bandwidth aggregated. The Head-Tail operation is simply illustrated in the Fig.8

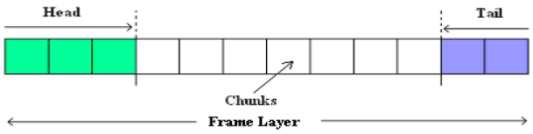

Fig.8 Simple of Head-Tail Operation [19]

The simplicity itself the first ever advantage of the method Head-Tail. As it expressed in the Fig.8, after selecting and then pairing peers, the single peer of every pair is selected as the Head and other one as the Tail. The head is generally with extreme stability and bandwidth. The head usually sends the chunks from layer starting as well as the Tail from layer end. The operation usually continues until that they reach to similar layer chunk. In particular point the scheduler can be the next layer or similar layer in a next frame by using pair. The smoothly dealing of node failures is the different advantage of certain method. Single node failure (best-case) may affect one end flow. The scheduler that without any generous effort can continue receiving process of chunks from another end that until the substitution for a failed node, need to be found. Or in low bandwidth case of different peer scheduler can select different peer for pair. At the worst case the scheduler can drop a layer.

A. Head-Tail Streaming System Architecture

In the general schema, Head-Tail elements of the system and the relationships of them have depicted in the Figure No. 9 There are number of architecture points in the system that needs to attend:

1. Video Codec
2. Methods of Accessing to Video Information
3. Streaming Architecture & Protocol,
4. Node Replacement Policies,
5. Scheduling

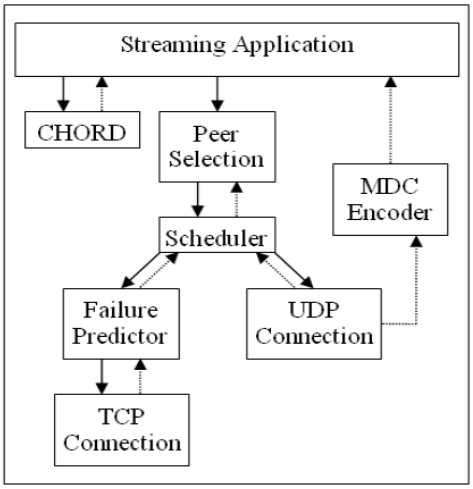

Fig.9 Architecture of Head-Tail System[19]

B. Video Coded

Entire formats of video that are utilized on web are based on layer that it is not to interrupt the packet of sending, if sender & receiver bandwidth aren't that enough. In particular the codec, the layers need to be sent accordingly to bandwidth in between the two nodes. The MDC codec usually acts better than that of networks LC with the rate of high-failing. In the MDC, the LC contrary, if general problem in the single layer receiving occurred, it won't affect the higher layers. However, the certain format of video requires high bandwidth. Because of high node rate that failing in the P2P-networks; the MDC reduces the quality video. Due to certain issue in the single layer it doesn't affect on other layers, the MDC is utilized for the system of Head-Tail. It's assumes that in every frame architecture, all layers size is nearly equal.

C.Methods of Accessing To Video Information

The CHORD is utilized as the technique of searching and maintaining the Head-Tail data system. Three distinct features which made us to utilize the Chord instead of number of different P2P (Peer-to-Peer) lookup protocols that were the simplicity of it, the provable correctness, as well as the provable execution even in concurrent node face departures and arrivals. And by constructing the overlay network of it is simple as well as defects of it are nearly removable. On different side, it is able to be utilized to gain the stability node and search Free Riders due to CHORD which has the periodical messages in order to find the failings [19].

The CHORD are maintaining the information pieces of data nodes; but the suggested system usually estimated the available nodes bandwidth that to be considered. Hence, the available bandwidths of every node get added to hash-table of the Chord network. Certain change reduces exchanged messages number in between the nodes to find the appropriate nodes.

As like other system of video distributing, the resource hardware can't be utilized for the special purposes in the system as well as the current internet algorithms that should be noticed. But one of the certain resources, that are the output allocation nodes bandwidth, is able to be controlled by the system. The bandwidth output may be allocated to every node; on different bandwidth, it is saved for the certain receiver. The updated existing bandwidth is that stored in the CHORD network. By utilizing the stored information in the CHORD network, every node may estimate the bandwidth of it via the suggested techniques in the [18] & [19].

D.Streaming Architecture

And after network joining, the receiver can request the peers that have the certain network file. On the list of receiving peers, the Head-Tail selects the peers set depended on download bandwidth of it and then the senders aggregated bandwidth. In the executed technique of Head-Tail selects the peers that based on greatest fit and in order to keeps different nodes as the backup. The best nodes get selected depended on available bandwidth of them, availability and packet loss. Hence, the nodes which meet the best selected criteria of main senders and different have kept in the backup list nodes in node failure case.

E.Streaming Protocol

Application of every network has the own protocol of it in order to communicate with different networks nodes. In the Head-Tail TCP/IP is that has been used utilized as the network underlying protocol as well as build the own protocol above certain layer. The Head-Tail utilizes the bound signaling in order to control whole session of streaming. One connection is get utilized for the control messages sending while different one need to transfer the data video. The control messages that get sent data packets and by TCP are sent by using the UDP. The TCP assists to send the reliable messages, that while UDP assists to have the extreme control on video data flow. The message controls are very small that not enough to congest or delay and they are to be hold in the single TCP-message. The UDP usually suffers from the unreliable data transfer. Hence, we have added the easy method of ARQ in order to protocol for requesting the lost packets from the senders. In the method the scheduler usually estimates the RTT of every connection depended on the connection history and then requests the probable lost packets especially from sender. The RTT get estimated depended on the algorithm of Jacobson/Karels [18]. The control protocol's associations are the two-way handshake. Initially, the receivers are willing to establish the connection that announced to the sender by the message control, and then the sender will send the message of ACK. And after receiving particular message by the receiver, it unlocks the UDP port.

F.Node Replacement Policy

The node leaving is depended on the two types: (1) aware & (2) unaware. In the aware leaving the sender tells the receiver that leaving network. The receiver doesn't let the sender to leave until the current receiving layers get downloaded completely. And during the current transmission layer the scheduler has so much time select the new peer as well as replace one leaving.

The node failure is refers to the unaware leaving. The node failure can occur by the failure of link or to close application streaming without the notification. In particular case, the last requested chunk get lost or delayed. Depended on estimated RTT of connection; the scheduler requests the last chunk that against the second peer is still sending the chunks assigned. IN the next RTT if last chunk get received, the scheduler finds out the delay. Chunk isn't received during the next RTT then the scheduler finds out the node failure and exchanges a failed peer that with the other. And during the new peer finding as well as replacing the failed node, the scheduler used to make the effort in order to recover lost chunks due to the second peer that is sending chunks lost from different layer end. The single technique is to enhance this performance remedy in order to select the peer that replacing and make out the required connection of it that during the retransmission of the lost chunk. Hence in the node failure case, the peer is that replacing in order to take an immediate part. And by using certain remedy, by initializing the connection as well as preparing peer replacing get overlapped with node failure discovery.

The replacing peer failure is the major concern in the technique of node replacement. When the replacing peer gets failed, the system experienced the double failure, just one for the initial failed node as well as other different failed node. Hence, the maximum lost time, every time the delay occurs. The system usually predicts the required failure. Then, it tries to search appropriate peer replacing. If the replacing node gets failed then a system has the enough time to find other peer because of long failure find out phase-delay.

G.Scheduling

As the packet loss as well as retransmission of data occurs frequently, the scheduling in the mesh-based system of distribution is the extreme critical part in the designing. It means that the data get distributed and fragmented in between the nodes, there's no mapping in between nodes and data. And because of special Head-Tail System Design, the scheduling is simple and then make system of self-supporter. In certain system, the scheduler assigns the packets data from head-tail of the layer. Hence, the propagation lose is repetitive and zero that receiving as well as transferring the zero propagation. The high performance particular system is due to the overlapping in between the assigned chunks to- senders. The nodes usually try contact each other, and not to pass each other. It's possible to send the chunk only twice at node point that meeting similar chunk that usually happens rarely. The first delay chunk has sent with the single node which causes particular problem. Moreover, the second-node retransmit chunk. The most crucial differences are in between the Head-Tail scheduling system as well as different systems that is the packet retransmitting path. The conventional system utilizes the method of one way one node for the transmission. In a system of Head-Tail though we've the system of two ways transmitting in that the nodes are moving forward to each other, until they are meeting to point of meeting (rendezvous point). So if the delay happens in any of node there's no requirement to modify scheduling as well as there's no requirement for to consider another nodes' packet delay.

III. RESULTS

In the part of results, we have illustrated the comparison for Proposed Methodology (Head-Tail Video Streaming) and Enable Adaptive Solution. As we are able to observe results of P2P PDR (Peer to Peer Packet Delivery Ratio) is Low especially for the Enable Adaptive Solution as castrated to the Methodology Proposed (Head-Tail Video Streaming).

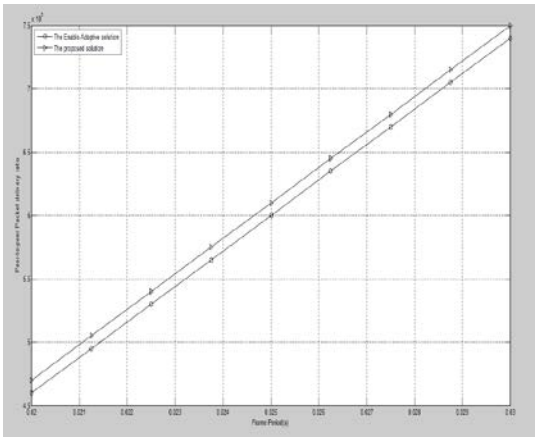

Fig.10 The PDR (Packet Delivery Ratio)

In order describe the comparison; the Frames are then transmitting the duration of fixed time. At the duration of fixed time, the frame of Video Data is transmitting.

In order to describe the comparison, we may contrast the PDR (Packet delivery Ratio) for the Proposed Methodology (Head-Tail Video Streaming) and Adaptive Solution. We get PDR (Packet Delivery Ratio) is high enough for Methodology Proposed (Head-Tail Video Streaming) for similar duration.

IV. CONCLUSION

The P2P systems are becoming popular day by day because of the ability of them in order to deliver the large data amounts at the cost of minimized deployment. While the P2P systems usually foster novel media applications

development, they further represent the interesting paradigm alternative for the application of media streaming which is able to benefit from inherent self-organization as well as the resource scalability that exist in particular environments. Base paper represents the exact overview of the application as well as network-layer mechanism which enable the successful frameworks of streaming in the P2P (Peer-to-Peer) systems. Further they describes the architecture of media delivery which need to be deployed over the P2P-networks, so to address certain streaming applications needs. We demonstrate how the application of video streaming may benefit to the diversity offered that to P2Psystems, and perform the scheduling solutions and distributed streaming with the multi-path packet transmission.

In the Research, we have demonstrated the comparison for Proposed Methodology (Head-Tail Video Streaming) and Enable Adaptive Solution. We get PDR (Packet Delivery Ratio) is high enough for Methodology Proposed (Head-Tail Video Streaming) for similar duration.

REFERENCES

[1] X. Zhang, J. Liu, B. Li, and T.-S. P. Yum, "Coolstreaming/DONet: A data-driven overlay network for efficient live media streaming," in *Proceedings of IEEE INFOCOM*, vol. 3, pp. 13-17 March 2005.

[2] V. N. Padmanabhan, H. J. Wang, and P. A. Chou, "Resilient peer-to peer streaming," in *Proceedings of IEEE ICNP*, Atlanta, Georgia, USA, 2003.

[3] N. Magharei and R. Rejaie, "Understanding mesh-based peer-to-peer streaming," in *Proceedings of ACM NOSSDAV*, Newport, Rhode Island, USA, 2006.

[4] Y. Shen, Z. Liu, S. Panwar, K. Ross, and Y. Wang, "Peer-driven video streaming: Multiple descriptions versus layering." *Proceedings of IEEEICME Amsterdam*, The Netherlands, 2005.

[5] A.Albanese, J. Bl¨omer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission," *IEEE Transactions on Information Theory*, vol. 42, pp. 1737–1744, Nov. 1996.

[6] L. Golubchik, J. Lui, T. Tung, A. Chow, and W. Lee, "Multi-path continuous media streaming: What are the benefits?" *ACM Journal of Performance Evaluation*, vol. 49, no. 1-4, pp. 429–449, Sept 2002.

[7] J. Apostolopoulos, T. Wong, W. Tan, and S. Wee, "On multiple description streaming with content delivery networks," in *Proceedings of IEEEINFOCOM*, vol. 3, 23-27 June 2002, pp. 1736–1745.

[8] A.C. Begen, Y. Altunbasak, O. Ergun, and M. H. Ammar, "Multipath selection for multiple description video streaming over overlay networks," *Signal Processing: Image Communcation*, vol. 20, pp. 39–60, 2005.

[9] T. Nguyen and A. Zakhor, "Multiple sender distributed video streaming," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 315–326, April 2004.

[10] D. Jurca and P. Frossard, "Distributed Media Rate Allocation in Overlay Networks," *Proceedings of IEEE ICME Toronto*, Canada, 2006.

[11] J. Chakareski and P. Frossard, "Distributed Streaming Via Packet Partitioning", *Proceedings of IEEE ICME*, Toronto, Canada, July 2006.

[12] E. Setton, J. Noh, B. Girod, "Rate-Distortion Optimized Video Peerto- Peer Multicast Streaming", *Workshop on Advances in Peer-to-PeerMultimedia Streaming at ACM Multimedia* , pp 39-48.

[13] J.-P. Wagner, J. Chakareski, and P. Frossard, "Streaming of scalable video from multiple servers using rateless codes", *Proceedings of IEEEICME*, Toronto, Canada, July 2006.

[14] C. Gkantsidis and P.R. Rodriguez, "Network coding for large scale content distribution", *Proceedings of IEEE INFOCOM*, Miami, Florida, USA, March 2005.

[15] C. Wu and B. Li, "Diverse: Application-Layer Service Differentiation in Peer-to-Peer Communications", *IEEE Journal on Selected Areas in Communications*, Vol. 25, No. 1, pp 222-234, January 2007.

[16] Choon Hoong Ding, Sarana Nutanong, and Rajkumar Buyyam ," Peer-to-Peer Networks for Content Sharing".

[17] Dan Jurca, Jacob Chakareski, Jean-Paul Wagner," Enabling Adaptive Video Streaming in P2P Systems" 2012.

[18] Dinesh Goyal *et.al* "Analysis & Design of Visual Cryptography using Moving Image", in *International Journal of Information and communication Technology Research* (ISSN: 2223 – 4985) Vol.3, No. 12, Nov.-Dec. 2013.

[19] Ali Sianati *et.al*, "Head-Tail Video Streaming Over Peer To Peer Systems", *International Journal of Computer Networks & Communications (IJCNC)*, Vol.3, No.2, March 2011 DOI : 10.5121/ijcnc.2011.