A Signaling System for Quality of Service (QoS)-Aware Content Distribution in Peer-to-Peer Overlay Networks

Sasikumar Kandasamy[1], Sivanandam Natarajan[2], and Ayyasamy Sellappan[3]
[1]Department of Computer Science and Engineering, Tamilnadu College of Engineering, India
[2]Department of Computer Science and Engineering, Karpagam College of Engineering, India
[3]Department of Information Technology, Tamilnadu College of Engineering, India

Abstract: *Peers are used to limit and expand the available facilities for different kind of devices, which should able to fetch the data according to the demand of users and available resources. Several factors such as latency, bandwidth, memory size, CPU speed, and reliability can affect the Quality of Service (QoS) of the peer-to-peer(p2p) network. In this paper, we propose a signaling system for QoS-aware content distribution for Peer-to-Peer overlay networks where the signaling system is controlled through a set of data so that it can be operated dynamically. The flow of signal in the system enhances other devices to choose their own way with the requirement of applications. This system is able to reduce the traffic and utilize the available resources.*

Keywords: *Signaling, bandwidth, delay, transmission, buffer, catch.*

Received October 3, 2013; accepted July 6, 2014

1. Introduction

Interest in providing availability of routing through the internet and Quality of Service (QoS), content distribution and file sharing services, enabling multicasting or protecting from Denial of Service (DoS) attacks have been addressed by different application layer overlay design proposals [7]. In recent years, internet usage as a means of content delivery is increased due to the growing popularity of smart handheld devices among content consumers. Available content includes software, smart-phone applications, music and video files available, as well as media streaming applications. Each type of content is associated with a particular desired QoS but low delays between request and reception is good for all types of content [16]. peer-to-peer (P2P) networks were initially designed and considered suitable for huge content distribution across networks. [10]. One of the trends related to the internet is that it is being applied to the transfer of a lot of contents [20]. Nodes in a P2P network, called peers, play a variety of roles in their interaction with other peers. Peers act as clients while accessing the information. Peers acts as servers, when serving information to other peers. While forwarding the information for other peers, peers play a role of router.

1.1. Architecture

P2P networks are distributed systems. The software running on each node in P2P networks provide equivalent functions. P2P networking is a set of technologies that enable the direct exchange of services or data between computers. It is assumed that peers are equals. P2P systems focus sharing among these equal peers. A pure peer-to-peer system runs without the management, any centralized control or hierarchical organization. A hybrid system uses some centralized or hierarchical resources. Peers consist of clients, servers, routers or even networks [4, 15]. Generally, P2P network is divided into three architectural areas, namely, centralized, decentralized and hybrid [1, 7, 8, 22].

P2P networks aggregate a large number of computers and possibly mobile or handheld devices, which join and leave the network frequently. Increasing number of communication software in the area of Audio/Video conferencing and streaming are built on distributed architectures based on the P2P model. Some of the examples are Skype [7] and Peer-Cast [5]. Overlay networks are used increasingly for network-sensitive applications. Some of them are distributed web caching, content dissemination and stream processing [21].

Flooding-based systems do not scale well because of the bandwidth and processing requirements they impose on the network. These systems do not provide guarantees to lookup time or content accessibility [7].

The dynamic boosting of network connections, which occurs commonly during a popular live streaming program, freeloading was identified as a problem for P2P systems because many early P2P protocols allowed non-excludable access to overlay resources [11]. Byzantine faults, Sybil attacks, Eclipse attacks, flash crowds, etc. are challenging issues for large-scale P2P systems. Lack of network layer QoS, inevitably impacts Quality of Experience (QoE) are observed as interruptions to the playback of many currently deployed clients/server applications [3, 12, 13, 17].

1.2. Factors Affecting Quality of Service in P2P Network

The factors affecting the QoS in P2P network are as follows:

- *Latency*: It is also known as time to delivery. Latency is defined as the time from when a publisher publishes an event and a subscriber to that event receives notification that it is available. The overlay network must effectively reduce the overall latency of event notifications. The latency can be defined as delay.
- *Bandwidth*: Bandwidth represents the resources available across a path during event transfer, which is denoted by the number of events transferred between the publisher and subscriber per unit time. If a subscriber does not specify a requirement, then the broker network will assume the default values, which provide the maximum possible bandwidth available along a path.

Some of the contents, which we should be aware at the time of data transmission, are given below:

- *Buffer cache ratio*: Buffer to cache ratio plays a vital role in transmission when a data packet is transmitted through another node [11].
- *Available capacity*: Availability capacity is directly related to the bandwidth. As bandwidth determines the availability of the number of channels at the time of data transmission, it plays a vital role in the content awareness [4].
- *CPU speed*: CPU processing speed is a major need for the data transmission, as it controls the traffic factors at different time.
- *Memory size*: Buffering and caching ratio is a factor of memory present at different nodes. It is important as it plays a vital role in the time delay and heavy traffic.

Several works have been proposed for improving data flow in the physical layer. Some other works concentrates on the video streaming enhancement technique and tries to get an advancement weighting technology called an artificial intelligence is developed for the data transmission, as it is aware of the network. They consider that the node interested in data transmission should have the awareness of the network. If a node has a multiple path to transmit the data, it will cause loss of time and process cycle.

In order to avoid these issues, this paper develops a signaling system for QoS aware content distribution in p2p overlay networks.

2. Related Works

2.1. Content-Aware Caching and Traffic Management in Content Distribution Networks (CDN)

Content-Aware Caching and Traffic Management are proposed for request routing, content placement, and content eviction in CDN. The abstraction of a switch is used to model the CDN. The algorithm would be throughput optimal (stabilize the system) and yield short queues. Their main constraints are finite cache sizes and the periodicities with which content is refreshed in the caches. Two algorithms are developed, one with random evictions and other with Min-Weight evictions and illustrated the superior potential of the latter in causing such drifts. The authors created the iterative versions of both algorithms. The engender large, negative Lyapunov drifts in the system are desirable, since such drifts budget short average queue lengths, created iterative versions of both algorithms are more efficient and hence cause still shorter queue lengths. However, streaming traffic with requests has hard delay constraints. It is dropped, if such a constraint cannot be met [2].

2.2. Networks-Aware Overlays with Network Coordinates(NC)

Large-scale overlays, Distributed Hash Tables (DHTs), are based on a purely logical identifier space, designed for load balancing and routing resilience. Overlays incorporate full network-awareness, where it is a fundamental requirement to understand the physical network topology when constructing the overlay. Network-Aware Overlay (NAO) is used to build applications that optimize for network metrics such as latency, bandwidth and packet loss. Network Coordinates (NCs) offer such powerful potential for overlay networks, which is a fundamental paradigm for dynamic overlay management. By providing an underlying geometric framework, NCs allows application of a rich, unified set of algorithmic tools and techniques to a variety of network problems. Developing coordinates with near-perfect accuracy is a long-term challenge, as this approach was sufficiently accurate for the most applications. It allows tradeoffs between accuracy and measurement overhead for dynamic network-aware overlays [19].

2.3. Network Load-Aware Content Distribution in Overlay Networks

The authors make an in-depth investigation on the issue of client/node selection, which is a fundamental problem in massive content distribution on overlay networks. They envision a hypercube as the overlay network and give the novel server/client selection schemes. The network load of each session is reduced and well balanced across the sessions and the network resource consumption is low. Their schemes do not require measurement of any network performance metrics or the network topology or routing information. This algorithm is scalable due to the absence of network measurement and low implementation complexity. Lower Wireless LAN Services (WLS) implies less network congestion created by concurrent streams, hence, better quality for streaming applications. When the clients are partitioned into disjoint subsets based on the degree of interference criterion, the high network resource usage and the interference among the concurrent connections can occur [10].

2.4. P2P-based VoIP QoS Management in Heterogeneous Networks

IP Multimedia Subsystem (IMS), Session Initiation Protocol (SIP) and Service Oriented Architecture (SOA) are adopted to deploy the most promising P2P technologies for heterogeneous networks based voice services. This ensures the possibility of heterogeneous communication and the scalable Voice over Internet Protocol (VoIP) service in the heterogeneous environment. A QoS-aware scheme deployed on end-points is proposed because P2P VoIP traffic potentially suffers from performance issues like packet loss, delay, jitter, which greatly affect the QoS [6].

2.5. Trust and Reputation Model nConsidering Overall Peer Consulting Distribution

A localized information trust and reputation system is proposed here using securing data transmission with technical procedures, such as cryptography. This method can discern a small difference between real quality of service QoS and other peers' feedback while distinguishing the malicious peers, even when the exaggeration coefficient is high. When one or a group of peers change their QoS, the model exhibits a quick reaction to this change. This response is demonstrated by a rapid decrease in reliability when the QoS change is downward and a slow increase when the change is upward. A slow reaction to the upward QoS change may exclude those peers who frequently change their QoS and encourage consistent reliable service providers [14].

2.6. Quorums for Replication of Multimedia Objects in P2P Overlay Networks

In order to increase the performance of the Multimedia Quorum-Based (MQB) protocol a novel synchronization mechanism is presented where only some replicas, not all replicas are updated even for a write type of operation in a quorum. In the Multimedia Quorum-Based with Read Material (MQB-RM) protocol, replicas in a read quorum are updated after a transactions read the newest replica. The number of materializations of replicas can be reduced in the MQB protocol compared with the other protocols [18].

3. Proposed Methodology

In this paper, a simple signaling system is proposed for content awareness in which every node involved in the transmission should be aware of its own aware content and processing speed. In addition, node has an ability to choose its path of transmission. The proposed method is divided into three steps:

1. Find the quality of service factors.
2. Find the correct data present node.
3. Create a link between the two nodes for data transmission.

3.1. Assumptions

A P2P network consists of individual nodes which have a processor installed in it. A node is able to calculate its quality of service parameter regularly. It passes its quality of signal to every other node at each time it is updating its quality of service factor. Looking the quality of service signal of another node a sender node decides about its data transmission. Looking into the quality of service of every node is able to decide the path of data transmission. Every node is identical; those have identical bandwidth memory size installed on every node. The processing speed may vary node to node. There should be some reference value which is known and common to the entire set of nodes present in the network. A fixed level of maximum processing speed, bandwidth, catching size is calculated for all the nodes present in a network before the deployment of the network elements by the experts. This standard is chosen for the references for the entire set of nodes to make the network to a standard.

3.2. Network Model

In Figure 1 the node 'a' can send data to node d through c. Similarly it can send data to 'g' through 'h'. Then the process has to go through the following steps. The path is determined through which the data packet passes.

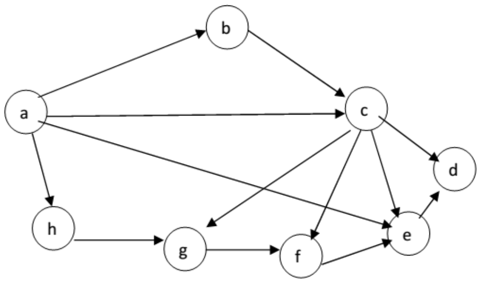

Figure 1. General network topology in peer to peer network.

3.3. Calculation of Quality Factors

To calculate the quality factor, some of the following factors are consideration. They are CPU speed, buffering to caching ratio, bandwidth factor.

3.4. Bandwidth Factor

The bandwidth factor can be calculated as the ratio of bandwidth present at a certain time to the maximum bandwidth present in the network. The bandwidth can be calculated by using the algorithms given in [9].

Bandwidth Factor (BF) =band width present at a certain time/maximum bandwidth (1)

Here, the numerator and denominator are of the same unit. Therefore, the division generates a non-unit fraction that is carrying a value always between 0 and 1.

3.5. CPU Factor

CPU factor is calculated as the ratio of the CPU processing speed at a certain time to the maximum CPU speed can be reached.

CPU Factor (CF) = CPU speed at a certain time/maximum CPU speed (2)

Here, the numerator and denominator are of the same unit. Therefore, the division generates a non-unit fraction that is carrying a value always between 0 and 1.

3.6. Buffer to Cache Ratio (BC)

The occupied buffer size for caching is then a product of the quantity (*nb*) and the average length of cached descriptions (*lb*). Suppose constant s is the unit size of 1 second descriptor:

$$Bsize = nb * lb * s \tag{3}$$

Let Csize be the maximum memory size present at a node. *Csize* should equal for all nodes present in a network.

The quantity of the data generated (*nb*) can be represented as

nb= Data Generated (DG) +Data Available (DA) (4)

Then the buffer to cache ratio is given by

$$BC= (Csize- Bsize) /Csize \tag{5}$$

Caching a long period of a descriptor may decrease the bandwidth utilization. On the other hand, caching more descriptions help to avoid the quality fluctuation and may lead to better flexibility in highly dynamic environments. As a peer usually has a limited buffer size, it is necessary to find the tradeoff between the above two strategies.

Here, the numerator and denominator are of same unit so the division generates a non-unit fraction that is carrying a value always between 0 and 1.

3.7. Quality Factor

Since the quality of p2p network depends on the bandwidth, CPU speed and buffer size, the quality factor is derived by the summation all the 3 metrics CF, BF and BC.

Quality factor(QF)=(w1.CF+w2.BF+w3.BC)/3 (6)

where w1,w2 and w3 are weight factors used as normalization constants, whose value lies between 0 and 1. The weights can be adjusted as per user requirements depending on these factors.

3.8. Determining of Quality Factor Signal

The QF range is divided into four parts. The detail is given below

Table 1. Factor signal according to QF.

QF Range	Assigning quality factor signal
0.81-1.00	00
0.56-81	01
0.31-0.55	10
0.0-0.30	11

Table 1. shows that the quality factors can be divided into four levels. For the signaling purpose, two bits signaling systems are selected. Considering two bits, it gives four states of quality of factor. They are 00,01,10,11.

1. 00 denotes the most quality of service is present
2. 01 denotes a good quality of service is present and data can be easily transmitted with a little delay
3. 10 denote quality of service is lesser, were data can be transmitted with a mere delay.
4. 11 show the least quality of service is present, which implies possibility of transmitting data is almost impossible.

3.9. Stack Management System In Quary Optimization In Peer To Peer Network

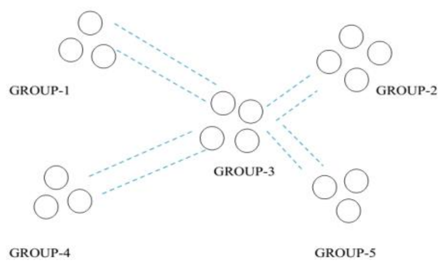

Figure 2. Hybrid P2P network.

The whole peer-to-peer network is divided into groups as shown in Figure 2. The entire groups have unique ids. All the nodes in the network have also a unique id. The combination of the group id with the node id is given a unique identification to the node in a network. Generally, a node has the ability to communicate with all other nodes in the group as well as it can communicate with other group. When a node wants a data, it generates a request packet, which contains the sender id with the group id. It also includes a quality factor signal range (minimum value and maximum value), that includes Bandwidth factor, and CPU factor and buffer to cache ratio. First, it will transmit the request to all the neighbors in the same group. A node can find its neighbors inside the group or in another group. This can be done through GPS.

A data packet carrying quality of service in its data field can be enhanced the quality of service in peer to peer network. The data packet shape is shown in Figure 3.

Sender id	Intermediate ids	Maximum data quality	Minimum data quality required	Request of data	Flag=1

Figure 3. Request data packet.

When a node wants some data, it generates a request packet for it. The node first transfers the message (request packet) to the neighbor nodes keeping the flag on (one), which describes it is a request packet. The data packet contains information about sender node. It carries a stack structure in it for the intermediate groups. The stack structure contains the node ID through which the data packet travels. Whenever the packets travel it pushes the node id to the stack.

Destination id	Sender ID	Sender group	Intermediate ID present	Data quality present	Flag=0

Figure 4. Acknowledgement packet.

When a node identifies the request can be satisfied by it generates the response data packet. The node sends an acknowledge packet to the sender node. Figure 4 shows the structure of acknowledgement packet travels in the opposite way the request data packet has travelled. When an acknowledgement packet is passing through a group of node, it pops it id from the stack and send to the next id present in it. It keeps the flag 0 (one) which indicates the data packet is a response packet. The flag helps the intermediate nodes to choose the push or pop operation. It also sends the quality present in it.

After receiving the acknowledgement, the interested node sends the contract to the node with the data and chosen as best node for communication as shown in Figure 5. The detail data packet structure is given above. It carries the sender id, destination id and intermediate ids through which the data packet travels and contract.

Destination ID	Sender ID	Intermediate group IDs present	Contract for transmission	Flag=1

Figure 5. Data packet structure of contract of data flow from source to destination.

When the node having the correct data gets the contract it starts the data transmission in that service range Figure 6 shows the data packet quality at data transmission phase.

Destination id	Source ID	Intermediate group IDs	Data	Flag=0

Figure 6. Data packet quality at data transmission phase.

3.10. Stack Present of the Intermediate Nodes

Figure 7 shows the data structure present at the field of intermediate node ids. It is clear that sender group id is present at the bottom and the id of the group which contain node which contain the required data.

Group <ID>nearer to destination
…………….
Group<ID>nearer to sender group
Sender group<ID>

Figure 7. Structure of stack present at intermediate node.

3.11. Decision Table to Estimate the Optimum Node

When a node sends a request packet find out a node with the right data in a right quality, the above node may get more than one acknowledgement data packets. At the time, the interested node had to choose the best node for the data communication. To find out the best quality data communication the interested node maintains the following table and takes the decision of the appropriate node for getting the data.

Table 2 .Decision table present at nodes.

Node <IDs>	Quality factor signal	Chosen node
548	00	Yes
……………	………………….	…………….
……………..	………………….	……………….

The Table 2 contains three columns. They contain node id fields which keeps the whole set of node id which sends the acknowledgement packets. The second column of the table contains the corresponding quality factor signal and the third column gives the remark to the chosen node.

3.12. Procedure of Finding the Right Node and the Right Quality of Service Connection

1. Every node has to get its bandwidth factor, buffer to cache factor, CPU factor and data generation ratio at regular interval.
2. Bandwidth factor is calculated by dividing the available bandwidth with maximum bandwidth.

3. CPU factor is calculated by dividing the CPU speed to the maximum CPU speed.
4. Buffer to cache factor is calculated by dividing the differences between cache size and buffer size to the cache size.
5. Considering all above the factors, the node has to set its quality factor in the above given way.
6. Getting all above factor, the node has to determine the quality factor signal.
7. At first, an interested node generates a requesting data packet. In the request packet, one field is reserved for the intermediate groups. This field contains a stack. The stack is used for the path determination.
8. Every group has a group leader. As group leaders have all the node's ids with them, the interested node searches the data in its group. If the data is not found among the own group, then the request data packet is sent to other groups.
9. When the packet is received by any group, which has not generated data packet, it first checks the flag value.
10. If the flag value is one, it will check the query presents it. If it is available, then it will check the quality present.
11. If the quality is in the range of the defined range then it sends the acknowledgement packet. The node sends the data packet keeping the flag zero.
12. The acknowledge data packet also contains the QoS level. At the time of sending the acknowledgement packet, the node copied the intermediate nodes <ids> to the intermediate field of the data packet keeping the order same.
13. When a node gets the data packet and does not find the request data in it, then it pushes the id to the stack of the data packet and sends to other nodes whose group id is not on the top.
14. When a node receives a data packet and the received data packet's flag is zero, then it first checks the destination.
15. If the destination id matches with its own group id, the data packet is transmitted to the specific node. The interested node checks the QOS level. It maintains the QoS levels got from different nodes in a table. Then it finds the best way of getting the data and sends a contract to the node. The sender node kept the flag 1 at the time of transmission of contract.
16. If the node finds the data packet's flag field is zero and the destination field does not match with its own id; then it just pop the stack sends the data packet to the top value group of the stack.
17. After getting the contract, the node with the data makes a path sends the required data to the requested node. At this time, the flag is zero. The data may be of a single unit or of a stream of data.

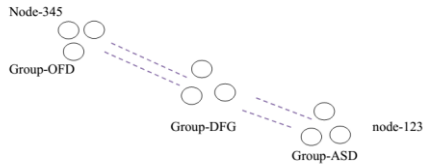

Figure 8 .Groups in peer to peer network.

There are some groups in P2P network as shown in Figure 8. A node having node id 123 is present in the group ASD. So the unique id is ASD123.The node 123 wants a data, but node 123 is unaware of the data where it is present. The requirement of data is defined in the range of quality of service signal. Suppose here the maximum range is 00 and the minimum range is 10. The field in the data packet is set according to the minimum and maximum range. It creates a data packet which is a request packet transferred it to the neighbor nodes. Figure 9 shows the structure of the request data packet. All the groups have a group leader. Group leader keeps the detail of all the nodes. The interested node transferred the data packet to neighbor node through the group leader.

ASD123	…………	00	10	Data information	Flag=1

Figure 9. Request data packet.

When the node asks the request it first traverse through its group (group is present in hybrid models and decentralized models). The group can be detected by the group ID added with the unique ID. If the data is present among the members of the group then it gets data from that node.

ASD123	ASD	00	10	Data information	Flag=1

Figure 10. Data packet flows from group ASD to group DFG.

In case of unavailability of data in own group the data packet is transferred to neighboring groups. At this time the group ID is pushed to the stack present at the intermediate id field. The request packet is transferred to all other groups except the group having the group id ASD. Now the stack contains ASD. The data packet is traversed to the other group whose ID is DFG. The structure of this data packet is shown in Figure 10. When a single node gets the request packet first checks the flag. If the flag is one it tries to find the availability of data in its node. If the data are available it sends the acknowledgement packet to the interested node through the group. The group is detected through the stack. The acknowledgement packet is sent by setting the flag zero. If the node is unable to find requested data then it transmits the message to the nodes present in DFG group.

ASD123	DFG ASD	00	10	Data information	Flag=1

Figure 11. Data flow from DFG to OFT.

If the data is not present in the DFG group then the group DFG is added to the stack and transmitted to other groups except the group ASD as shown in Figure 11.Suppose the other group is OFD and it finds the data is available in its group which is in the node having the unique ID OFD345 in the range of QoS is 01 then the node having the data sends the acknowledgement packet. In the acknowledgement packet the flag is set to zero.

OFD345	OFD	ASD123	DFG ASD	01	Flag=0

Figure 12. Flow of data acknowledgement packet from OFT to DFG.

The data are transmitted to the top value of the stack that is DFG. When a node of the DFG group gets the acknowledgement packet it will first check the flag. Structure of this acknowledgment is shown in Figure 12. If the flag is zero at the time of receiving of the data packet then the node pops the stack. After that the node just identifies the top of the stack and sends it to the group having the top group ID. Now the top ID is present is ASD as shown in Figure 13.

OFD345	OFD	ASD123	ASD	01	Flag=0

Figure 13. Acknowledgement packet from group DFG to ASD.

The destination node is present inside the group. Then the node having the unique id ASD123 stores in a table as mentioned above. Then the node waits for a predefined time. At this time it gets another acknowledgement from a node having the unique id SDF234. This value is also stored in the table for the comparison of the best. Suppose the signal of quality factor is 10.

Node <IDs>	Quality factor signal	Chosen node
OFD345	01	Yes
SDF234	10	No

Table 3. Decision table at ASD123.

As 01 is the best quality the node having the unique id ASD123 sends a contract to the node OFT345 as shown in Figure 14.

ASD123	OFD345	Intermediate group IDs	Contract to send data	Flag=1

Figure 14. Node ASD123 sending the contract to OFT345.

The contract data packet flows in the same way that it flows at the time of the request packet. After getting the contract the node having the group id OFD345 sends the data to the node ASD123 as shown in Figure 15.

ASD123	OFD345	DFG ASD	Data	Flag=0

Figure 15. Data flow between OFT345 to ASD123.

Here the data can be of any length. The data may be a unique data or a stream of data.

4. Simulation Results

4.1. Simulation Setup

This section deals with the experimental performance evaluation of our algorithms through simulations. In order to test our protocol, the NS-2 simulator [18] is used. NS2 is a general-purpose simulation tool that provides discrete event simulation of user defined networks.

We have used the BitTorrent packet-level simulator for overlay networks. A network topology is only used for the packet-level simulator. Based on the assumption that the bottleneck of the network is at the access links of the users and not at the routers, we use a simplified topology in our simulations. We model the network with the help of access and overlay links. Each peer is connected with an asymmetric link to its access router. All access routers are connected directly to each other modeling only an overlay link. This enables us to simulate different upload and download capacities as well as different end-to-end (e2e) delays between different peers. Figure 16 shows the topology of the network.

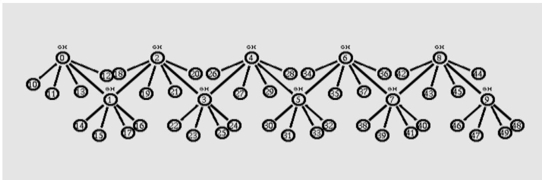

Figure 16. Topology of network.

4.2. Results

We have compared our Signalling System-QoS Aware Content Distribution (SS-QACD) architecture with Flow-level Nework Bandwidth Simulation (FLNBS) [9].

The performance is evaluated in terms of bandwidth utilization, processing delay, signaling overhead and packet delivery ratio.

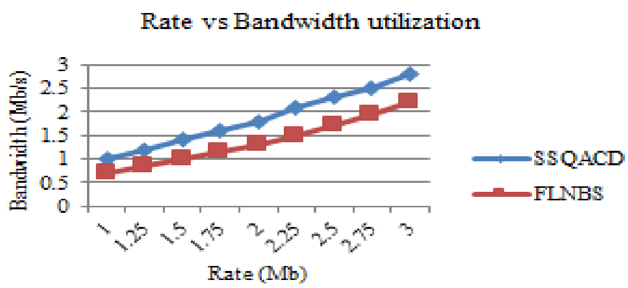

Figure 17. Rate Vs Bandwidth Utilization.

In Figure 17, the X-axis shows the transmission rate varied from 1 to 3Mb and Y-axis shows the corresponding bandwidth utilization (measures in Mb/s) of the peer nodes. Clearly, the utilization increases as the rate increases. The utilization of SSQACD is 27% higher when compared to FLNBS.

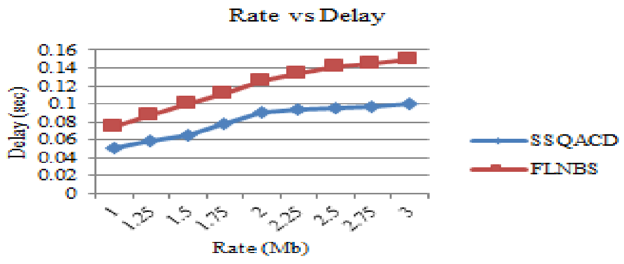

Figure 18. Rate Vs Delay.

In Figure 18, the X-axis shows the transmission rate varied from 1 to 3Mb and Y-axis shows the processing delay (measured in seconds) occurred at the peer nodes. Clearly, the delay increases as the traffic rate increases. But SSQACD has 31% less delay when compared to FLNBS.

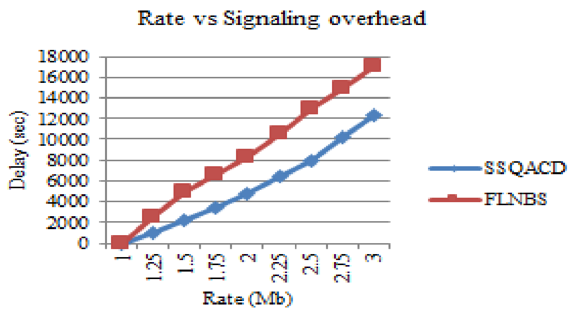

Figure 19. Rate Vs Signaling overhead.

In Figure 19, the X-axis shows the transmission rate varied from 1 to 3Mb and Y-axis shows the signalling overhead occurred (in terms of packets) at the peer nodes due to various messge exchanges. The overhead increases as the traffic rate increases. But SSQACD has 52% lesser overhead when compared to FLNBS.

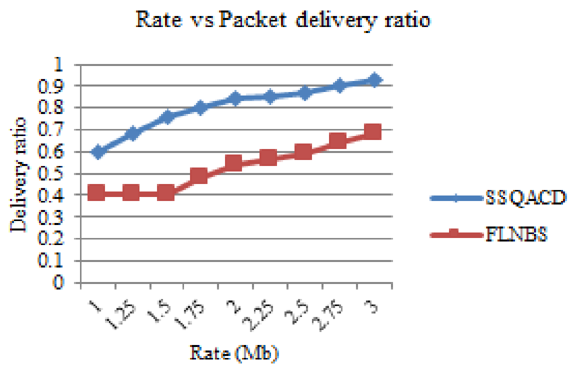

Figure 20. Rate Vs Packet Delivery Ratio.

In Figure 20, the X-axis shows the transmission rate varied from 1 to 3Mb and Y-axis shows the corresponding packet delivery ratio of the peer nodes. The packet delivery ratio is measures as the ratio of packets received successfully to the total number of packets transmitted. The figure shows the SSQACD has 34% higher delivery ratio when compared to FLNBS

5. Conclusions

A sender node need not have to calculate all the factors affecting in the quality of service data transmission at each time, when the node has to transmit the data. Getting the contract the nodes are generating the data. So there is no loss of energy involved in data generation. The delay involved in the data transmission has not any adverse effect on the transmission system as it is tolerable and it is priory known to the sender and destination nodes. A fast processor is present at each node has the ability to calculate the quality factor at a little time difference. So in each favorite scenario the nodes are able to send the best quality of data to the sender.As the transmitted quality factor signal is only about two bits. So the overhead of quality factor signal can be neglected. The quality factor signaling system does not degrade any quality of the contents present, which are affecting the data transmission rate. The node has the choice the correct node for transmission. A node can take decision dynamically for the data transmission. The method is a purely decentralized method. Every node has the freedom to request the query packet and choose the right node from which it can get the right data. This procedure is applicable for all types of data that is a low amount of data with high QoS, high amount of data with high QoS, low amount of data with low QoS and high amount of data with low QoS. It also supports the streaming of data flow.

There should be more technique present for the transmission of data taken place smoothly at time of busy hour. An alternating processor speed can be introduced for this purpose. A detail study of data requirement is needed for different type of files and their busy hours. Buffering to catch ratio also needs a good observation in different scenarios and different time period. The quality factor signal can be modified according to different geographical areas and demand taking into consideration. Query optimization technique can be introduced to it. User's interest should value at time of busy hours.

References

[1] AlLahham Y. and Hassan M., "Scalable Self-Organizing Structured P2P Information Retrieval Model Based on Equivalence Classes," *The International Arab Journal of Information Technology*, vol. 11, no. 1 , pp.78-86, 2014.

[2] Amble M., Parag P., Shakkottai S., and Ying L., "Content-Aware Caching and Traffic Management in Content Distribution Networks," *in Proceedings of IEEE INFOCOM*, Shanghai,

pp. 2858-2866, 2011.

[3] Ayyasamy S. and Sivanandam S., "A QoS-Aware Intelligent Replica Management Architecture for Content Distribution in Peer-to-Peer Overlay Networks," *International Journal on Computer Science and Engineering*, vol. 1, no. 2, pp. 71-77, 2009.

[4] Ayyasamy S. and Sivanandam S., "Trust Based Content Distribution for Peer-To-Peer Overlay Networks," *International Journal of Network Security and Its Applications*, vol. 2, no. 2, pp. 134-145, 2010.

[5] Chanpong S., Kamolphiwong S., and Ngamwongwattana B., "Quality of Service Evaluation for Peer Selection in Peer-to- Peer Streaming Services," *International Journal of Innovation, Management and Technology*, vol. 3, no. 3, pp. 255-258, 2012.

[6] Fu X. and Wang J., "P2P-based VoIP QoS Management in Heterogeneous Networks," *in Proceedings of 5th International Conference on Wireless Communications, Networking and Mobile Computing*, Beijing, pp. 1-3, 2009.

[7] Galán J. and Gazo A., " Overview and Challenges of Ooverlay Networks: a Survey," *International Journal of Computer Science and Engineering Survey*, vol. 2, no. 1, pp. 19-37, 2011.

[8] Gamiochipi R., Incentive-Driven Qos In Peer-To-Peer Overlays, Doctoral thesis, University College London, 2010.

[9] Gkogkas A., Roverso R., and Haridi S., "Accurate and Efficient Simulation of Bandwidth Dynamics for Peer-To-Peer Overlay Networks," *in Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools*, Paris, pp. 352-361, 2011.

[10] Han S. and Xia Y., "Network Load-Aware Content Distribution in Overlay Networks," *Computer Communications*, vol. 32, no. 1, pp. 51-61, 2009.

[11] He Y., Gu T., Guo J., and Dai J., "KQStream: Kindred-Based QoS-Aware Live Media Streaming in Heterogeneous Peer-to-Peer Environments," *in Proceedings of International Conference on Parallel Processing Workshops*, Xian, PP. 56-56, 2008.

[12] Hofmann M. and Beaumont L., *Content Networking: Architecture, Protocols, and Practice*, Morgan Kaufmann, 2005.

[13] Jason A. and Shanmugasundaram R., "Design of QoS Aware Dominating set based Semantic Overlay Network (QADSON) for Content Distribution," *Journal of Computer Science*, vol. 7, no. 10, pp. 1478-1489, 2011.

[14] Jia C., Xie L., Gan X., Liu W., and Han Z., "A Trust and Reputation Model Considering Overall Peer Consulting Distribution," *IEEE Transactions On Systems, Man, And Cybernetics-Part A: Systems And Humans*, vol. 42, no. 1, pp.164-177, 2012.

[15] Mahambre S., Kumar S., and Bellur U., "A Taxonomy of QoS-Aware Adaptive Event-Dissemination Middleware," *IEEE Internet Computing*, vol. 11, no. 4, pp. 35-44, 2007.

[16] Mushtaq M. and Ahmed T., "Hybrid Overlay Networks Management for Real-Time Multimedia Streaming over P2P Networks," *International Federation for Information Processing*, vol. 4787, pp. 1-13, 2007.

[17] Mykoniati E., Landa R., Spirou S., Clegg R., Latif L., Griffin D., and Rio M., "Scalable Peer-to-Peer Streaming for Live Entertainment Content," *IEEE Communication Magazine*, vol. 46, no. 12, pp. 40-46, 2008.

[18] Ohkawara T., Aikebaier A., Enokido T., and Takizawa M., "Quorums for Replication of Multimedia Objects in P2P Overlay Networks," *in Proceedings of 6th International Conference Complex, Intelligent and Software Intensive Systems*, Palermo, pp. 148-155, 2012.

[19] Pietzuch P., Ledlie J., Mitzenmacher M., and Seltzer M., "Network-Aware Overlays with Network Coordinates," *in Proceedings of the 26th IEEE International ConferenceWorkshops on Distributed Computing Systems*, Portugal, pp.12, 2006.

[20] Rakotoarivelo T., Senac P., Seneviratne A., and Diaz M., "A Peer-to-Peer Scheme to Discover and Select QoS Enhanced Alternate Paths," *in Proceedings of the 2nd International Conference on Communication System Software and Middleware*, Bangalore, pp. 1-9, 2007.

[21] Sellappan A. and Natarajan S., "A Robust Byzantine Fault-Tolerant Replication Technique for Peer-to-Peer Content Distribution," *Computer Science*, vol. 7, no. 2, pp. 159-166, 2011.

[22] Shakkottai S. and Johari R., "Demand-Aware Content Distribution on the Internet," *IEEE/ACM Transactions On Networking*, vol. 18, no. 2, pp. 476-489, 2010.

Sasikumar Kandasamy completed his B.E. (Computer Science and Engineering) in 1999 from VMKV Engineering College under Madras University and M.E. (Computer and Communication) in 2009 from Tamilnadu College of Engineering under Anna University Chennai. Currently he is pursuing PhD degree from Anna University, Chennai. He is working as an Assistant Professor in the Department of Computer Science and Engineering at Tamilnadu College of Engineering, Coimbatore. He is a member of various professional bodies like ISTE, CSI and IAENG. His research areas include P2P networks, Overlay Networks, Load Balancing, Quality of Services and Artificial intelligence. He is having 14 years of teaching experience in various Engineering Colleges.

Sivanandam Natarajan completed his B.E. (Electrical Engineering) in 1964 from Government College of Technology, Coimbatore, and MSc (Engineering) in Power Systems in the year 1966 from PSG College of Techchnology,Coimbatore. He acquired PhD in control systems in 1982 from Madras University. He received best teacher award in the year 2001 and Dhakshina Murthy Award for teaching excellence from PSG College of technology. He received the citation for best teaching and technical contribution in the year 2002, Government College of Technology, Coimbatore. His research areas include Modeling and Simulation, Neural Networks, Fuzzy Systems and Genetic Algorithm, Pattern Recognition, Multidimensional system analysis, Linear and Non linear control system, Signal and Image processing, Control System, Power System, Numerical methods, Parallel Computing, Data Mining and Database Security. He is a member of various professional bodies like IE (India), ISTE, CSI, ACS and SSI. He is a technical advisor for various reputed industries and engineering institutions.

Ayyasamy Sellappan completed his B.E. (Electronics and Communication Engineering) in 1999 from Maharaja Engineering College and M.E. (Computer Science and Engineering) in 2002 from PSG College of Technology, both under Bharathiar University, Coimbatore. He completed his Ph.D on 2012 under Anna University Coimbatore. PhD degree from Anna University, Coimbatore. He is working as a Professor and Head in the Department of Information Technology at Tamilnadu College of Engineering, Coimbatore. He is a member of various professional bodies like ISTE, CSI and IAENG. His research areas include P2P networks, Overlay Networks, Load Balancing and Quality of Services and having 13 years of teaching experience in Engineering Colleges.