ISSN 2347 - 3983
International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 151- 156 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp31.pdf

Optimizing Congestion in Peer-to-Peer File Sharing Based on Network Coding

E.ShyamSundhar[1] B.Lalitha[2]
[1]M.Tech II Year Department Of Computer Science And Engineering, Jawaharlal Nehru Technological University,Anantapur,Andhra Pradesh , India, shyamvijay.goud@gmail.com.
[2] Assistant Professor Department Of Computer Science And Engineering,Jawaharlal Nehru Technological University,Anantapur, Andhra Pradesh,India, lalitha_balla@yahoo.com.

ABSTRACT

In peer-to-peer (P2P) file sharing systems, many autonomous peers without Pre-existing information share files with each other. Due to their open environment and distributed structure, these systems are vulnerable to the significant impact from selfish and un-trusted nodes. This paper focuses on implementing a congestion control method on P2P systems for efficient file sharing. The system uses a congestion detection algorithm to detect and eliminate the congested node in the P2P network. Further this system alerts all the other peers about the detected congestion node for reducing the chances of further transmission performance degradation.

Keywords:peer-to-peer, network control,congestion control, file-sharing.

1. INTRODUCTION

The Internet can be looked upon as a medium to disseminate information. Distribution of large size multimedia content has become commonplace today. The original stereotype of content distribution was the server-client architecture, wherein a dedicated server furnished the required content to all requesting clients. However such a system faces a bottleneck from the constraints of available upload bandwidth and resources at the server and is thus incapable of scaling up to serve a large number of clients. This problem of sharing content with a large number of clients was solved through two main techniques[1]. One was through the use of large-scale commercial content distribution services depicted in figure 1, which mirror customers' content to be published on various servers and employ sophisticated algorithms to locate the mirroring server from which content is transferred to the requesting client.

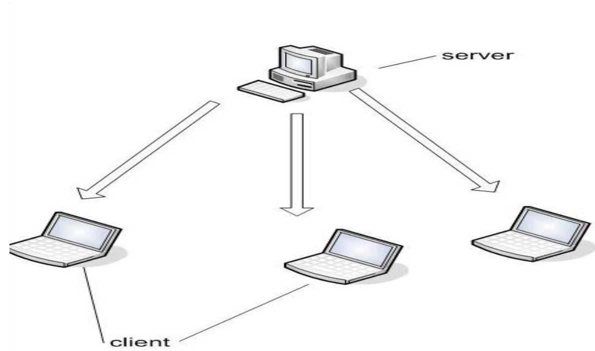

Figure 1: Traditional Client Server System

The second method, peer to peer content distribution, entails the formation of a network topology in which there is no defined server-client architecture and each node (or peer) in the topology has the capability of acting as the source, transferring agent or requester of content.

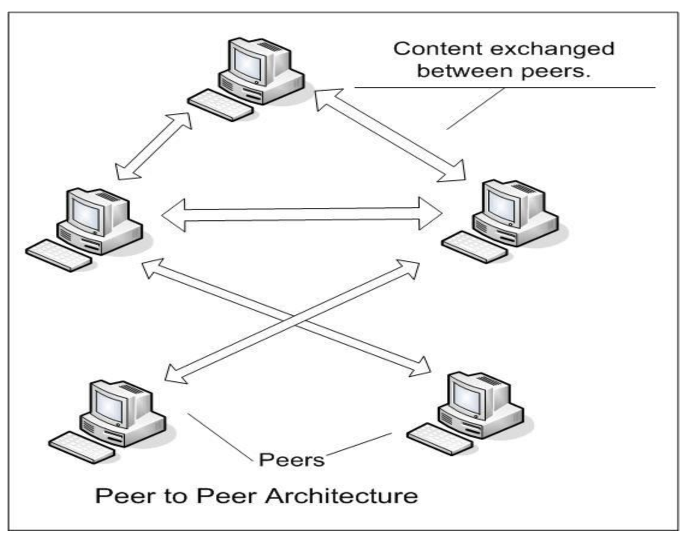

Figure2: Peer-to-Peer Architecture

A centralized network has server(s) that control[2] the whole network; all file requests are sent to the server(s) and are replied to by the server(s) only. In

ISSN 2347 - 3983
International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 151- 156 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp31.pdf

this case, file resources are limited due to the restricted processing ability and storage space of the central server(s) in a large network. Nonetheless, a centralized network provides a manageable environment under central control.

The fundamental premise of P2P networks depicted in figure 2 is the concept of sharing of content and resources amongst all participants of the network. Hence, as the number of peers increases, the resources available to the system increase too and thus such a system is fundamentally scalable. Though some variations exist, P2P networks are characterized by the absence of a central server and the peers are capable of joining or leaving the network at any time without any centralized control.

The P2P systems have the following properties:

a) Authenticity and privacy: It must be ascertained that unauthorized peers are not able to make changes to the content shared or substitute forged content in place of the correct content to be shared. The system must have mechanisms in place to ensure that only authorized peers are entitled to access available content. Several systems impose restrictions on the type of content being shared.
b) System stability: The topology of a P2P system changes continuously as peers join and leave. The system should remain stable and continue to provide services at all times. Though the topology of the system is subject to change, the peers which are a part of the system at any given time should remain unperturbed by the dynamics and their functioning should continue as normal.
c) Performance: This is defined in terms of the duration between the request and the final receipt of the desired content. Similarly, the time to complete other requests like searching, uploading, browsing for content on the system are also taken into account.
d) Scalability: The system's performance is not expected to degrade even in the wake of an increase in the number of peers or the volume of the content shared. The increased number of peers in the network should imply an increase in the number of resources available to the system.
e) Fairness: The performance of the system as experienced by a peer should be contingent with the utility of the peer to the system. Neither should a peer get any undeserved advantage (e.g. disproportionately large share of the bandwidth) nor should a peer be unfairly penalized (due to scarcity of resources or hogging of the bandwidth by another peer).

2. RELATED WORK

There are wide variety of file sharing systems are implemented under P2P architecture the most basic and famous is the Bit-Torrent.

2.1. Bit-Torrent:

The Bit-Torrent protocol [2] is the widely used P2P content sharing protocol. The content to be shared is broken into blocks of a fixed size and these chunks of data are then downloaded from different peers which have the desired file. These chunks are assembled in their correct order by the requesting peer. The requested file thus reaches the peer who had initiated the request for the same. The main elements of Bit-Torrent are:

a)**Torrents:**

For every file that is to be shared using Bit-Torrent, a corresponding torrent file is required to be made. Also referred to as torrent files, torrents are very small sized files which contain hashed information about the file's content, and name, length of the file to be shared, number of blocks of data and the size of each block in which the file's contents are to broken and shared. These files also contain the URL of the tracker (see below).

b)**Tracker:**

Every Bit-Torrent network has a unique machine called the tracker. It is the centralized entity which keeps a record of the peers sharing a particular content and is responsible for coordinating the download of a desired file from several peers which are sharing that content. There are several web-based tracker software available, e.g. Torrent Bits [5] which can be run at the tracker. The tracker software allows peers to register with the tracker, login into their account, browse for shared content, download content and upload torrent files. Some tracker software provide additional functionalities like RSS feeds, discussion forums, personal message boards.

c)**Bit-Torrent Client:**

This is software which runs at each peer in the P2P system and it helps share and download content from other peers. Bitspirit, U-torrent are some popular Bit-Torrent client software.

d) **Uploading content:**

When a peer wishes to share content with other peers, it creates a torrent file [3] containing the URL of the unique tracker in the system. It then uploads this torrent file on the tracker and shares this file for the P2P network using its Bit-Torrent client software. This sharing is called seeding of content. Once a file is seeded, it becomes available to other peers for download.

e)**Downloading content:**

When a peer wishes to download content from the P2P network, it searches for it on the tracker. If the search is successful, the peer downloads the torrent file for the same.

This torrent file is then opened with theBit-Torrent client software. Since the torrent file contains the URL of the tracker, the Bit-Torrent client software contacts the tracker using this URL. The tracker has a list of peers who are sharing the requested content. Since the requested content is broken into blocks of fixed size for sharing, different peers send different blocks of data to the requesting peer. The tracker coordinates and decides the blocks to be obtained from different peers. These blocks when assembled by the Bit-Torrent client complete the file download.

2.2 Network Coding

None of the peers are completely aware of which peers have which pieces of the file, or even that there is a bottleneck. If two different peers on one side of the slow link send the same block to two different peers on the other side of the link, that link has carried redundant information. Even if only one copy of a particular block is sent across the slow link, peers on the other side will be able to quickly swap that block among them; it would be a more efficient use of resources to send only distinct blocks across the slow link.

Network coding [GR05] allows peers to participate in peer-to-peer file distribution without concern for which blocks of the file other peers have. As with Bit-Torrent, the file is divided into blocks. However, instead of transmitting a block of the file, a seed transmits a coded block that contains an amalgamation of some information from each block of the file. Peers aid in file distribution by sending coded blocks that contain an amalgamation of all of the information they have received so far. If there are n blocks in the original file, peers need to receive n of these amalgamated blocks before being able to decode the original file. This is done using linear algebra. Each block transmitted is a linear combination of all of the information that the peer has, and recovering the original file amounts to solving a linear algebra problem. With each coded block, a list of coefficients used to generate it is transmitted. Because random coefficients are used in the computation of each of these linear combinations, there is a low probability that any of the coded blocks transmitted will be identical. Therefore, there is very little risk that the same data is ever transmitted more than once over any particular network link.

2.3 Congestion Control

Communication among peers is basically implemented with more than one concurrent TCP associations in file-sharing[3] scenario as well as in most recent streaming systems. As a result, the ratio of TCP connections for applications such as VoIP and HTTP over the complete number of TCP connections are minor, so the part of the bandwidth these latter applications will become weaker. In other words, the reasonable bandwidth distribution mechanism of TCP fails in guarantying a fair sharing between applications.

This study may be neglected if the network congestion was far from the end users. Various studies have unfortunately revealed that most network congestions in the transmissions are either on the links between ISPs or in the access network [5]. When congestion take place on the mutual access network which connect an access router to the Internet, P2P applications with more than one TCP connections unfairly degrade the bandwidth from other conventional Internet applications. Moreover, the requirement for pervasive Internet makes the number of devices associated to one access point, and accordingly the number of applications served by this first router, enhances significantly.

3. PROPOSED CONGESTION CONTROL TECHNIQUE

Our approach to solve the congestion problem at a particular node is to employ a congestion control strategy based on deleting and transferring the message, as the congested node calculate the storage value of each packet according to the forwarding possibility and Time to Live (TTL) value of packet, then deletes the packet with the smallest amount of storage value and transfers the packet with the lesser forwarding possibility to neighboring nodes. The nearest nodes calculate the receiving value of the packet according to the forwarding possibility and free buffer capacity of this packet. The congested node transfers the packet to the adjacent node with the highest receiving value. The packet will be deleted when there is no adjacent node to receive it.

3.1Network Model:

- When congestion occurs, the congested node determines a set of messages for deletion and another set of messages for migration.
- Meanwhile, the neighboring nodes determine whether to receive the transferred message and give a response to the congested node.
- If the neighboring nodes refuse to receive the transferred message, it will be deleted

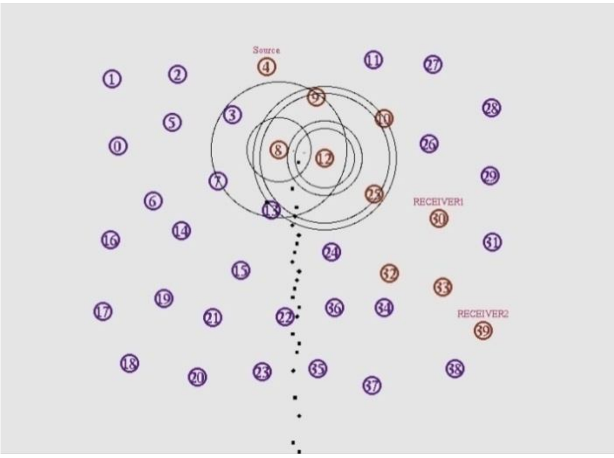

Figure 3: Packets Dropping.

The figure 3 depicts the packets continuously dropping between the node 8 and node 12 leading that the 12 is incapable of handling the incoming traffic and the figure 4 depicts that the node 12 is detected as congested node.

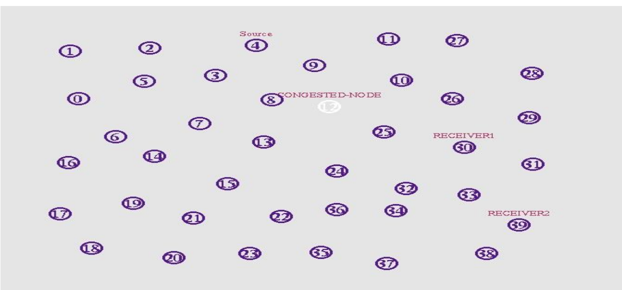

Figure 4: Detected Congested Node.

3.2 Congested Node Analyzing and Processing Algorithm

Step 1: The congested node measures the storage value of each packet and deletes the packet with the smallest amount of storage value.

Step 2: The congested node transfers the packet with the smallest amount of forwarding probability to its adjacent nodes.

Step 3: The congested node broadcasts transfer request message to its neighboring nodes and gets their receiving values of the message. If there is no response from any neighboring node, go to Step5

Step 4: The congested node selects the adjacent node with the maximum receiving value according to the responses and transfers the chosen message to this neighboring node.

Step 5: The congested node deletes this packet.

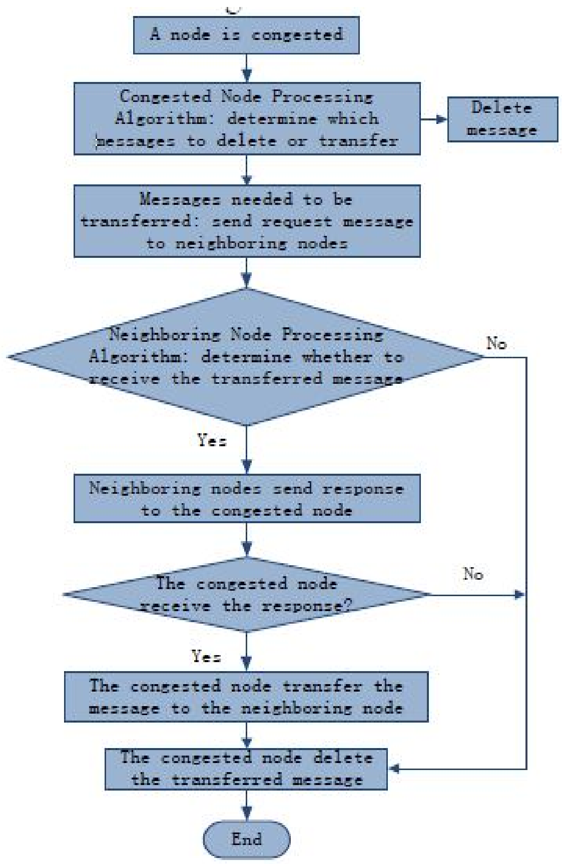

Figure 5: Flow of the Node Analysis Process

ISSN 2347 - 3983

International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 151- 156 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp31.pdf

3.3 Neighbor Node Analysis Algorithm

- The neighboring node judges whether its own buffer space can meet the requirements and calculates the storage value of the transferred message.
- If the storage value of the transferred message is smaller than the minimum storage value in the neighboring node, the process is over.
- Otherwise, the neighboring node will calculate the receiving value of the transferred message and return this value to the congested node.

4. EXPERIMENTAL EVALUATION

The proposed congestion control technique is tested under various general metrics including

- Packet Drops
- Error Rate
- Delay Ratio and
- Throughput

The results of the proposed scheme (red)are compared with the traditional Bit-Torrent (green) System and the Network coding (blue) scheme and are represented.

Figure 6 depicts the packet drops for all the three schemes and the result clearly show that the proposed scheme has less number of packet drops compared to the existing two schemes.

Figure 6: Packet Drops

Figure 7: Error Rate

The next metric to measure is the Error Rate and is depicted in the figure 7 where the rate of error is low for the proposed scheme.

The delay ratio plays an important in any network transmission where the delay indicates the time taken to deliver the packets. The proposed system out performs both the traditional systems and the results can be observed in Figure 8.

Figure 8: Delay Ratio

The final and most crucial metric is the throughput. This defines the efficiency of the complete network the higher the throughput then the performance will be greater.

Observing all the results depicted above the proposed scheme clearly out performs the traditional file sharing systems.

Figure 9: Throughput Rate

5. CONCLUSION

The area of congestion in a P2P file sharing is the trending research field, where the efficient congestion control methods are not defined properly. This paper proposed a congestion control method for P2P file sharing system. This scheme is also tested for all possible performance measures and the results are promising compared to the traditional file sharing schemes like Bit-Torrent and Network Coding Systems.

REFERENCES:

[1] Min Yang and Yuanyuan Yang, "**Applying Network Coding to Peer-to-Peer File Sharing**", *IEEE TRANSACTIONS ON COMPUTERS*, VOL. 63, NO. 8, AUGUST 2014.

[2] Yaning Liu, Hongbo Wang, "**Friendly P2P: Application-level Congestion Control for Peer-to-Peer Applications**," *Global Telecommunications Conference*, 2008. IEEE GLOBECOM 2008. IEEE, Nov. 30 2008-Dec. 4 2008.

[3] Deng-yin Zhang , Min Yang , and Lei Cui, "**Congestion control strategy for opportunistic network based on Message values**" *JOURNAL OF NETWORKS*, VOL. 9, NO. 5, MAY 2014.

[4] Y. Zhu, B. C. Li, and J. Guo, "**Multicast with network coding in application layer overlay networks**"*IEEE J. Sel. Areas Commun.*, vol. 22, no. 1, Sep. 2004.

[5] D . S. Lun, M.Medard, T. Ho, and R. Koetter, "**Network coding with a cost criterion**," in Proc. Int. Symp. Inf. Theory Appl. (ISITA'04), Oct. 2004.

[6] T.Ho, M. Medard, R. Koetter, D. Karger, M. Effros, J. Shi et al., "**A random linear network coding approach to multicast,**" *IEEE Trans.Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.

[7] Y. Yang and J. Wang, "**A new self-routing multicast network**," *IEEE Trans. Parallel Distrib.Syst.*, vol. 10, no. 12, pp. 1299–1316, Dec. 1999.

[8] C. Gkantsidis and P. R. Rodriguez, "**Network coding for large scale content distribution**," *IEEE INFOCOM 2005*, Miami, FL, USA, Mar. 2005.

[9] M. Yang and Y. Yang, "**An efficient hybrid peer-to-peer system for distributed data sharing**," *IEEE Trans. Comput.*, vol. 59, no. 9, pp. 1158–1171, Sep. 2010.

[10] S.Ratnasamy, M.Handley, R.M.Karp, and S.Shenker, "**Topologically aware overlay construction and server selection**," *IEEEINFOCOM'*02, New York, NY, USA, Jun 2002.