ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Path Aggregation for QOS Provisioning for Peer-to-Peer Streaming

M.Prathyusha
M.Tech,
Department of CSE,
AVNIET, JNTUH,Hyderabad.

G.Anitha
Assistant Professor,
Department of CSE,
AVNIET, JNTUH, Hyderabad.

Abstract:
Peer-to-peer (P2P) streaming has been widely deployed over the Internet. A streaming system usually has multiple channels, and peers may form multiple groups for content distribution. In this paper, we propose a distributed overlay framework (called SMesh) for dynamic groups where users may frequently hop from one group to another while the total pool of users remain stable. P2P streaming has been proposed and developed to overcome the limitations of traditional server-based streaming. P2P streaming has some problem like high delay and high link stress, since all the nodes are connected as a single network and the packets are travelling to the unnecessary nodes. To overcome this problem, this system is going to use SMesh concept, with the help of this concept we can able to improve the performance of the P2P Streaming with low delay and low link stress.

Keywords:
Peer-to-peer (P2P) streaming, dynamic group, SMesh.

INTRODUCTION:
WITH the penetration of broadband Internet access, there has been an increasing interest in media streaming services. Recently, P2P streaming has been proposed and developed to overcome the limitations of traditional server-based streaming. In a P2P streaming system, cooperative peers self-organize themselves into an overlay network via uni cast connections. They cache and relay data for each other, thereby eliminating the need for resourceful servers from the system. Today, several practical P2P streaming software implementations have been shown to be able to serve up to thousands of peers with acceptable quality of service In fact, there are many other similar applications over the Internet. In the application, the system contains multiple groups with different sources and contents. A user may join a specific group according to its interest.

While the lifetime of users in the system is relatively long and the user pool is rather stable, users may hop from one group to another quite frequently. Examples include stock quotes, news-on-demand, and multisession conferencing. A more typical example is group chat of Skype . Skype allows up to around 100 users to chat together. While millions of Skype users stay online and relay data for each other, the users may form multiple small groups for group chat. According to Rossi et al. , except for very short sessions, most Skype peers are alive for about one third of a day. Generally, such life time of a Skype peer is longer than the duration of a group chat.

In above applications, as peers may dynamically hop from one group to another, it becomes an important issue to efficiently deliver specific contents to peers. One obvious approach is to broadcast all contents to all hosts and let them select the contents. Clearly, this is not efficient in terms of bandwidth and end-to-end delay, especially for unpopular channels. Maintaining a separate and distinct delivery overlay for each channel appears to be another solution.

However, this approach introduces high control overhead to maintain multiple dynamic overlays. When users frequently hop from one channel to another, overlay reformation becomes costly and may lead to high packet loss. of joining and leaving events in any group. This relatively stable mesh is used for control messaging and guiding the construction of overlay trees. With the help of the mesh, trees can be efficiently constructed with no need of loop detection and elimination. Since an overlay tree serves only a subset of peers in the network, we term this framework Subset-Mesh, or SMesh.We can see that there is a large pool of peers in the streaming network. Peers are divided into multiple small groups, each corresponding to a channel. Peers in the same group share and relay the same streaming content for each other.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The mesh is formed by all peers in the system and is, hence, independent of joining and leaving events in any group. This relatively stable mesh is used for control messaging and guiding the construction of overlay trees. With the help of the mesh, trees can be efficiently constructed with no need of loop detection and elimination. Since an overlay tree serves only a subset of peers in the network, we term this framework Subset-Mesh, or SMesh. Our framework may use any existing mesh based overlay network. In this application, we use Delaunay Triangulation (DT) [2].

The traditional DT protocol has the following limitations: Inaccuracy in estimating host locations, Single point of failure, Message looping. We propose several techniques to improve the DT mesh, e.g., for accurately estimating host locations and distributed partition detection. The two important issues in construction SMesh: Mesh formation and maintenance, Construction of data delivery trees. SMesh does not rely on a static mesh. In the case of host joining or leaving, the underlying DTmesh can automatically adjust itself to form a new mesh. The trees on top of it will then accordingly adjust tree nodes and tree edges. Also note that in SMesh a host may join as many groups as its local resource allows. If a host joins multiple groups, its operations in different groups are independent of each other.

Internet access, there has been an increasing interest in media streaming services. Recently, P2P streaming has been proposed and developed to overcome the limitations of traditional server-based streaming. In a P2P streaming system, cooperative peers self-organize themselves into an overlay network via unicast connections. They cache and relay data for each other, thereby eliminating the need for resourceful servers from the system. In a P2P streaming system, the server (or a set of servers) usually provides multiple channels. A peer can freely switch from one channel to another for example, one of the most popular P2P streaming systems.

In this paper, we consider building a data delivery tree for each group. To reduce tree construction and maintenance costs, we build a single shared overlay mesh. The mesh is formed by all peers in the system and is, hence, independent of joining and leaving events in any group.

This relatively stable mesh is used for control messaging and guiding the construction of overlay trees. With the help of the mesh, trees can be efficiently constructed with no need of loop detection and elimination. Since an overlay tree serves only a subset of peers in the network it is called SMesh.

Existing System:

P2P overlay network, hosts are responsible for packets replication and forwarding. A P2P network only uses unicast and does not need multicast capable routers. It is, hence, more deployable and flexible. Currently, there are two types of overlays for P2P streaming: tree structure and gossip mesh. The first one builds one or multiple overlay tree(s) to distribute data among hosts. Examples include application-layer multicast protocols. The existing networking infrastructure are multicast capable. Emerging commercial video transport and distribution networks heavily make use of IP multicasting. However, there are many operational issues that limit the use of IP multicasting into individual autonomous networks. Furthermore, only trusted hosts are allowed to be multicast sources. Thus, while it is highly efficient, IP multicasting is still not an option for P2P streaming at the user level.

Proposed System:

In the Proposed applications, as peers may dynamically hop from one group to another, it becomes an important issue to efficiently deliver specific contents to peers. One obvious approach is to broadcast all contents to all hosts and let them select the contents. Clearly, this is not efficient in terms of bandwidth and end-to-end delay, especially for unpopular channels. Maintaining a separate and distinct delivery overlay for each channel appears to be another solution. However, this approach introduces high control overhead to maintain multiple dynamic overlays. When users frequently hop from one channel to another, overlay reformation becomes costly and may lead to high packet loss. In this paper, we consider building a data delivery tree for each group. To reduce tree construction and maintenance costs, we build a single shared overlay mesh. The mesh is formed by all peers in the system and is, hence, independent of joining and leaving events in any group.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

This relatively stable mesh is used for control messaging and guiding the construction of overlay trees. With the help of the mesh, trees can be efficiently constructed with no need of loop detection and elimination. Since an overlay tree serves only a subset of peers in the network, we term this framework Subset-Mesh, or SMesh.

IMPLEMENTATION:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

Peer to Peer Network:

P2P streaming system, the server (or a set of servers) usually provides multiple channels. A peer can freely switch from one channel to another, Peers in the same group share and relay the same streaming content for each other

Distributed partition detection:

We use Delaunay Triangulation (DT) as an example. We propose several techniques to improve the DT mesh, e.g., for accurately estimating host locations and distributed partition detection. Based on the mesh, we study several tree construction mechanisms to trade off delay and network resource consumption automatically Adjust itself to form anew mesh.

The trees on top of it will then accordingly adjust tree nodes and tree edges. Also note that in SMesh a host may join as many groups as its local resource allows. If a host joins multiple groups, its operations in different groups are independent of each other.

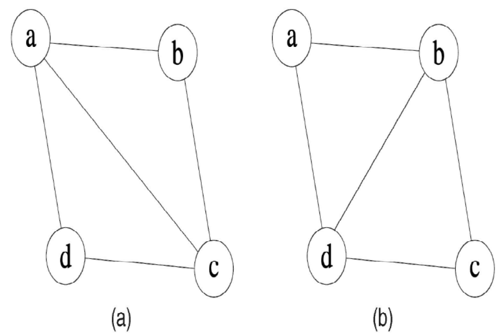
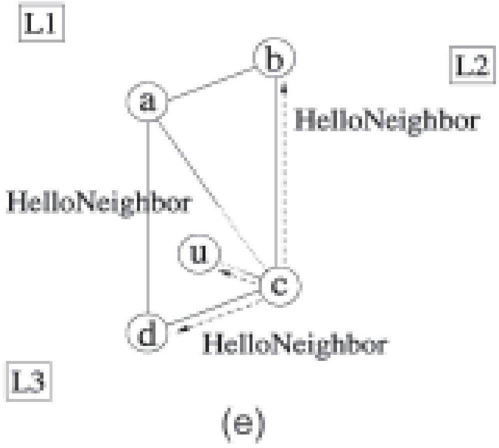


Fig : Distributed detection peer to peer two adjacent triangles nodes

Dynamic Joining Host:

A joining host, after obtaining its coordinates, sends a MeshJoin message with its coordinates to any host in the system. MeshJoin is then sent back to the joining host along the DT mesh based on compass routing. Since the joining host is not a member of the mesh yet, it can be considered as a partitioned mesh consisting of a single host. The MeshJoin message finally triggers the partition recovery mechanism at a particular host in the mesh, which helps the new host join the mesh.

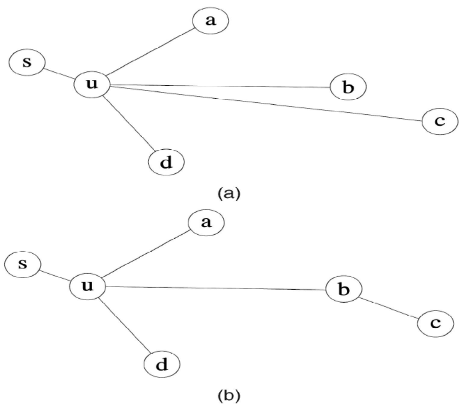


QoS Provisioning:

Two independent connections across domains A and B are set up, which leads to high usage of long paths and hence high network resource consumption. Furthermore, in the traditional DT protocol, a host may have many children. However, a host often has a node stress threshold K for each group depending on its resource. To address these problems, we require that the minimum adjacent angle between two children of a host should exceed a certain threshold T

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Consider a source s and a host u in the network. Once u accepts a child, u checks whether its node stress exceeds K or whether the minimum adjacent angle between its children is less than T It selects a pair of children with the minimum adjacent angle and delegates the child farther from the source to the other. Note that after aggregation, the overlay tree is still loop free because hosts are still topologically sorted according to their distances from the source.



RELATED WORK:

we discuss related work on P2P streaming and overlay construction. In one-to-many multimedia streaming and communications applications, an effecient approach is to use IP multicasting . Today, many of the existing networking infrastructure are multicast capable.

Emerging commercial video transport and distribution networks heavily make use of IP multicasting. However, there are many operational issues that limit the use of IP multicasting into individual autonomous networks. Furthermore, only trusted hosts are allowed to be multicast sources.

Thus, while it is highly efficient, IP multicasting is still not an option for P2P streaming at the user level. As a comparison, in a P2P overlay network, hosts are responsible for packets replication and forwarding. A P2P network only uses unicast and does not need multicastcapable routers. It is, hence, more deployable and flexible. Currently, there are two types of overlays for P2P streaming: tree structure and gossip mesh.

The first one builds one or multiple overlay tree(s) to distribute data among hosts.Examples include application-layer multicast protocols.

Our work in this paper falls into the first category. Although we build a mesh spanning hosts, we do not directlyuse mesh edges for data exchange. Instead, we build overlay trees on top of the mesh for data exchange. In order toimprove tree resilience, we may incorporate additional loss recovery schemes into our system. We now compare our work with other P2P tree construction protocols. Most previously proposed tree-based protocols build a single overlay tree rather than dealing with multiple dynamic trees as we investigate here. To support multiple groups or channels, these protocols have to build multiple independent trees. As a result, host joining or leaving of a group leads to reconstruction of the tree. Such cost may be high when hosts frequently change their groups. SMesh addresses this problem by using a relatively stable mesh consisting of all hosts, even though the membership of each group can be quite dynamic.

Bandwidth Distribution:

We design a scheduler to determine the order of packets to be transmitted from the queues according to the bandwidth ratio “br” for each type of traffic. The bandwidth ratio “br” represents the amount of bandwidth dedicated to urgent downloading and prefetching. Moreover, both classes can borrow bandwidth from each other when one of the two types of traffic is nonexistent or under the limit. This br value is also used to calculate the service rate forboth types of traffic on that particular peer with bri and µi- bri being respectively the service rate for urgent downloading and prefetching for peer i. µi is the total available bandwidth of peer i. In order to calculate the value of br we monitor the first queue (urgent downloading) in periodic interval. We calculate the total size of data chunks requested and their corresponding deadlines. Let CSi represents the chunk size requested by peer i with deadline ti then,

Conclusion:

In P2P streaming networks, users may frequently hop from one group to another. The proposed novel framework called SMesh serves dynamic groups for Internet streaming. SMesh supports multiple groups and can efficiently distribute data to these dynamic groups. It first builds a shared overlay mesh for all hosts in the system.

This stable mesh is then used to guide the construction of data delivery trees for each group. We construct three types of data delivery trees, i.e., embedded, bypass, and intermediate trees. We also propose and study an aggregation anddelegation algorithm to balance the load among hosts, which trades off end-to-end delay with lower network resource usage.To provide statistically guaranteed quality we proposed algorithms at two stages. First one at channel level i.e., admission control algorithms and second one at peer level i.e., algorithm for differentiated queuing and bandwidth allocation.

REFERENCES:

[1] X. Zhang, J. Liu, B. Li, and T.-S.P. Yum, "CoolStreaming/DONet:A Data-Driven Overlay Network for Peer-to-Peer Live Media Streaming," Proc. IEEE INFOCOM '05, pp. 2102-2111, Mar. 2005.

[2] X. Liao, H. Jin, Y. Liu, L.M. Ni, and D. Deng, "Anysee: Peer-to- Peer Live Streaming," Proc. IEEE INFOCOM '06, Apr. 2006.

[3] Y. Tang, J.-G. Luo, Q. Zhang, M. Zhang, and S.-Q. Yang, "Deploying P2P Networks for Large-Scale Live Video-Streaming Service," IEEE Comm. Magazine, vol. 45, no. 6, pp. 100-106, June 2007.[4] PPLive, http://www.pplive.com, 2009.

[5] X. Hei, Y. Liu, and K.W. Ross, "Inferring Network-Wide Quality in P2P Live Streaming Systems," IEEE J. Selected Areas in Comm., vol. 25, no. 9, pp. 1640-1654, Dec. 2007.

[6] X. Hei, C. Liang, J. Liang, Y. Liu, and K.W. Ross, "A Measurement Study of a Large-Scale P2P IPTV System," IEEE Trans. Multimedia, vol. 9, no. 8, pp. 1672-1687, Dec. 2007.

[7] M. Cha, P. Rodriguez, J. Crowcroft, S. Moon, and X. Amatriain, "Watching Television over an IP Network," Proc. ACM Internet Measurement Conf. (IMC '08), pp. 71-83, Oct. 2008.

[8] Skype, http://www.skype.com/, 2009.

[9] D. Rossi, M. Mellia, and M. Meo, "A Detailed Measurement of Skype Network Traffic," Proc. Int'l Workshop Peer-To-Peer Systems (IPTPS '08), Feb. 2008.

[10] J. Liebeherr, M. Nahas, and W. Si, "Application-Layer Multicasting with Delaunay Triangulation Overlays," IEEE J. Selected Areas in Comm., vol. 20, no. 8, pp. 1472-1488, Oct. 2002.

[11] T.S.E. Ng and H. Zhang, "Predicting Internet Network Distance with Coordinates-Based Approaches," Proc. IEEE INFOCOM '02, pp. 170-179, June 2002.

[12] L. Tang and M. Crovella, "Virtual Landmarks for the Internet," Proc. ACM Internet Measurement Conf. (IMC '03), pp. 143-152, Oct. 2003.

[13] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: A Decentralized Network Coordinate System," Proc. ACM SIGCOMM '04, pp. 15-26, Aug. 2004.

[14] H. Lim, J.C. Hou, and C.-H. Choi, "Constructing Internet Coordinate System Based on Delay Measurement," IEEE/ACM Trans. Networking, vol. 13, no. 3, pp. 513-525, June 2005.

[15] E. Kranakis, H. Singh, and J. Urrutia, "Compass Routing on Geometric Networks," Proc. Canadian Conf. Computational Geometry (CCCG '99), pp. 51-54, Aug. 1999.

[16] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, Computational Geometry: Algorithms and Applications. Springer- Verlag, 2000.

[17] R. Sibson, "Locally Equiangular Triangulations," Computer J., vol. 3, no. 21, pp. 243-245, 1978.

[18] E. Zegura, K. Calvert, and S. Bhattacharjee, "How to Model an Internetwork," Proc. IEEE INFOCOM '96, pp. 594-602, Mar. 1996.

[19] Y.H. Chu, S. Rao, S. Seshan, and H. Zhang, "A Case for End System Multicast," IEEE J. Selected Areas in Comm., vol. 20, no. 8, pp. 1456-1471, Oct. 2002.

[20] S.E. Deering, "Multicast Routing in Internetworks and Extended LANs," ACM SIGCOMM Computer Comm. Rev., vol. 18, no. 4, pp. 55-64, Aug. 1988.

[21] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable Application Layer Multicast," Proc. ACM SIGCOMM '02, pp. 205- 217, Aug. 2002.

[22] Y. Guo, K. Suh, J. Kurose, and D. Towsley, "P2Cast: Peer-to-Peer Patching Scheme for VoD Service," Proc. Int'l World Wide Web Conf. (WWW '03), pp. 301-309, May 2003.

[23] T. Do, K.A. Hua, and M. Tantaoui, "P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment," Proc. IEEE Int'l Conf. Comm. (ICC '04), pp. 1467-1472, June 2004.

[24] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "SplitStream: High-Bandwidth Multicast in Cooperative Environments," Proc. ACM Symp. Operating Systems Principles (SOSP '03), pp. 298-313, Oct. 2003.