Delay-Optimized Path in Peer-to-Peer Networks

Francis Lowu and Venansius Baryamureeba

Peer-to-Peer (P2P) networks are popular in sharing multimedia resources such as video, audio and images. Its popularity has attracted very many users thus increasing data streaming along a congested path. The increased sharing of the resources in the network has caused delays making the network resource sharing inefficient. This paper presents a delay-optimized path algorithm that increases data flow within the network among the peers and utilizes the network resources for effective and efficient streaming of data. We consider a distributed search for peer nodes with multi-peer participation in both searching and initiating resource transmission within the network. The resource streaming across the peer network is controlled by the resource function. Single peer initiated searching is also considered where a single peer node initiates the resource search process. Our analysis shows that the nature of growth of the network is linear.

1. Introduction

In a P2P network, resource streaming is based on active nodes in a decentralized and distributed network. Sharing of resources such as images, video, audio and speech among others in the network booms. Peers stream packets to each node as long as the peer nodes in the network request for the resources. Decentralized lookup services that store key value pairs are good in scaling peer-to-peer networks [Stoica et al. 2001]. Other peers may also need to advertise some resources available to them for searching in the network by other peer nodes. Peer-to-Peer network’s resource allocations are not normally distributed. Each peer in the network has its own resources interests and multimedia requirements for each user is different. Enterprises such as businesses, educational institutions and government organizations have collections of computing resources that require P2P architecture in order to use these resources efficiently [Adnan Fida 2004]. The transmission in bit/sec of the multimedia streams to the recipient has a varying quality of service provision for each multimedia packet along a given path. In their study Valduriez and Pacitti [2004] say that, Peer-to-Peer computing offers new opportunities for building highly distributed data systems. Unlike client-server computing, P2P can operate without central coordination and offer important advantages such as a very dynamic environment where peers can join and leave the network at any time; direct and fast communication between peers, and scale up to large
number of peers. However, most deployed P2P systems have severe limitations: file-level sharing, read-only access, simple search and poor scaling. Keong et al. [2004] say that Internet today, computing and communications environments are significantly more complex and chaotic than classical distributed systems, lacking any centralized organization or hierarchical control. Peers need effortless ways to discover, consume and provide services so as to take advantage of new services as they become available [Elenius et al. 2004].

1.1 Overview

The rest of the paper is arranged as follows: in Section 2 we present the problem formulation, the network model and work of other researchers. Section 3 discusses the network maintenance for joining and departing peers. Section 4 presents the design of the algorithm. In Section 5 we present results and discussion of our findings and lastly Section 6 presents the concluding remarks.

2. Problem Formulation and Network Model

Peer-to-peer network involve multiple participating nodes streaming resources to different peer nodes. The scaling nature of the network makes participation in the streaming of resources not normally distributed. More peers may want to join the network at different times and this requires a way of catering for both the existing and joining or leaving members. We consider two cases in the network: a single-peer initiated streaming, where a single peer streams resources to multiple peer nodes and secondly a multi-peer initiated streaming where many peer nodes stream either the same or different resources along the network paths with an optimal delay path and high throughput. The network is modeled as an undirected graph G(V,E), with V being the set of network vertices or peers and E is the link or the streaming path to the network nodes. For the case of the single-peer streaming there is a source S and the receiving node R such that the overall participation in streaming and receiving resources within the network is such that \( V = (S, R) \), where \(|(S, R)| = n\), where n is the number of nodes. While for the case of the multi-peer streaming there is no specified source node since at least two peers can stream network resources along any path to the recipients at the same time. Our objective is having an optimal-delay path for peer-to-peer resource streaming in the network.
a. Single-Peer-Initiated Streaming

We consider that a single peer streams packets to all active nodes in the network. We further consider that for all links \((i, j) \in E\) there exists an optimal delay path \(p_{(i,j)}\) such that, the delay \(d(i,j)\) is minimal. Resources from the peers is directed to the node that request or search for it as shown in Figure 1. Each peer node can act as a source while others are at the receiving end, a centralized resource streaming, however this happens if other nodes do not have resources to stream to the expectation of the receiver nodes. The peer node ready to transmit packets is called the initiator node, it sends messages to the other nodes to inform them that its ready to stream resources via their addresses.

b. Multi-Peer-Initiated Streaming

In this type of packet streaming; peers are distributed and each node has a link in an undirected graph \(G(V,E)\). We formulate the problem by considering that each peer node is in the neighborhood of another peer node in a defined magnitude \(\gamma\), which also determines the rate of resource streaming along a link. A peer \(p_k\) for \(k = 1, 2, \ldots, n\) will have resources to share with the other peers in the network; this peer will identify its neighbor as \(p_i = \gamma(p_j)\), for \(i \in j\) and where \(\gamma\) is the magnitude of \(p_i\) from \(p_j\) on a link \((i, j)\). Then peer \(p_i\) will forward a request for a given resource from peer \(p_j\) and visa-versa. The presence of resource at a peer node will be identified as Boolean \((0,1)\), and a requested resource will be labeled as either at level 0 or level 1 in \((0,1)\). A resource \(r_x\) is at level 0 if it can not be retrieved from a given peer node by its neighbors or any other peer in the network. For example a resource \(r_x0(p_i) = p_j\) indicates that a resource \(r_x\) can not be retrieved from peer node \(p_i\) by a peer \(p_j\). The magnitude \(\gamma\) will vary for each link in the network, and it will be key in determining the delay along that link.

%Each path in the network will be subject to a delay counter \(c\) between peer nodes. A delay counter \(c_{d(i,j)}\) determines how much delay a packet will take along a link \((i,j)\).

We require that the delay be minimized along the path streaming packets to all destination node-receiving resources from nodes with resource \(\{1\}\) level. The
optimal delay path is such that delay \((i,j) = \min \{rx_{1p}(i,j)\}, \forall (i,j) \in E\) also we require that delay\((i,j) = \min \sum rx_{1p}(i,j), \forall (i,j) \in E\). Under the condition that the resource function, holds for all the nodes in the network.

\[
\text{ResourceFunction}(i,j) \begin{cases} 
  rx = 1 & \text{If resources are available} \\
  rx = 0 & \text{Otherwise}
\end{cases}
\]

The Figure 2 below shows how multi-peer streaming of resources in the network is arranged. Every peer node can be an initiator node in this type of streaming.

**Fig. 2: Multi-Peer-Initiated Streaming**

3. Maintaining the Network Model

The topological graph of the network has to be maintained in order to scale, since peers may require leaving or joining the network to share resources such as images, video, audio and speech. Peers that join the network have to identify a peer node already in the network in its nearest neighborhood. The following steps are essential for a peer joining the network. The node that accepts other peers to join the network is termed the integrating peer node.

a. Joining Peer

Initially we assume that there is only one peer node, say, and an arbitrary peer \(p_i\). A peer \(p_j\) interested in the resources of \(p_i\) requests to join the network. Since only \(p_i\) has resources, then only resource level-1 will apply for the arbitrary peer \(p_i\) to join. If the peer joins the network, the magnitude \(\gamma\) of the peer \(p_j\) from peer \(p_i\) is noted and the delay function \(d(i,j)\) is computed subject to \(\gamma\) for the available resource \(rx\) at peer \(p_i\). This implies that \(p_j\) becomes the neighbor of \(p_i\). Any other peer \(p_k\) will contact any of \(p_i\) or \(p_j\) to join the network. Suppose it contacts peer
pj, then it will become the immediate neighbor of that peer node. A new node that has been integrated will have an address that depicts its presence in the network. Peers will need to identify their network requirements immediately they start sending and receiving packets.

b. Departing Peer

The peer node leaving the network sends a departure message to all other peer nodes in the network. With the departure message, it embeds a resource level 0 in the message to inform the other peers that it does not have any resource to share. This makes all its immediate neighbors to terminate the entire request to the departing node, such that $\gamma = 0$ along the links connecting to the departing peer. Nodes can be lost during the process of joining the network. For such cases no departure message is received from that node by either the integrating peer node or any other node in the network.

4. Delay-optimization Algorithm

The delay optimization algorithm is presented in this section. To create the peer-to-peer graph, there is an initiator node that has resources it may share with other peers. A peer gets interest in the resources of the initiator node and sends a request to share the resources. The initiator node realizes that it has no immediate neighbor, on getting the request from the joining peer. If the neighborhood node is found, it’s labeled a peer node and a resource-sharing link is immediately formed between the peers.

For each link formed a link delay requirement is registered and its magnitude is assigned. The magnitude is assigned only if the new peer has resources to share that make the link a busy path for resources streaming. Each link in the network will have a delay bound Delta which depends on the resource requirement along that link.

**Input:** Number of peer nodes n

**Output:** An optimized delay path

begin algorithm

Initiator node = $p_o$

for all $n$ nodes $p_k$, $k = 1,2,\ldots,n$

$p_k = \text{(true or false)}$

if $p_k = \text{true}$, find next node $p_o$

link $(p_o, p_k)$

else $p_k = \text{false}$

Delta = link delay bound

while $(p_o, p_k)$ is linked do

magnitude$(p_o, p_k) = \gamma$
\[(p_k, p_{k+1}) = \text{linkDelReq}\]

if \(\text{linkDelReq} \leq \Delta\) and resource level = 1
Stream resource
\[\text{delay}(i,j) \leq \sum \text{rx1p}(i,j) \leq \Delta\]
\[\leq \text{rx1p}(i,j) = \text{optimal}\]
else
resource level = 0
locate new node = \(p_{k-1}\)
repeat
return optimal delay
end

5. Results

5.1 Discussion

Quality of service (QoS) constraints such as delay is common in peer-to-peer networks and need to be addressed. In our study we talk about the optimal delay path in peer networks and how we can achieve it to increase resource streaming among the peers. We present two types of streaming: the single-peer-initiated streaming where a single peer will be allowed to initiate resource searching in the network. This streaming is considered a broadcast and only the initiator node will be responsible for allowing new peers to join as others in the network search for the resources. This is to help in reducing the congestion encountered if all peers are at the same level of streaming resources hence minimal delay on the network paths. We also present the multi-peer initiated streaming where every node is free to stream and request for resources in the network. However only nodes with resources to share are eligible for search. A peer node that does not have resources to share cannot be an initiator node and therefore no new member can join from that peer node. This increases reliability and reduces congestion due to crowding at the peer node and along the streaming link.

5.2 Correctness

Peers in the entire network can at least share or search for resources, subject to the resource function. Each peer joins the peer network as a sole node, this scales the network additively. Since no link can exist without a peer in the network, this study assumes that the initial state of the algorithm holds. A peer is either in the network or outside, if not joining the network. We denote by true a peer already in the network or false a peer not in sight. The algorithm further expects that for all peers \((p_k, p_{k-1})\) in the network, there is a binding link that scales the network. In line 7, the while loop holds only if \((p_k, p_{k-1})\) holds as a link: whose magnitude can then be computed. It is easy to see that the algorithm is linear, from line 2, the while loop iterates once for every entry of a peer in the network, requires \((n+1)\) number of times to execute. Whereas the for loop requires \(n\) times to read through each input values (joining nodes).
5.3 Contribution

Considering the network model discussed, the paper made the following contribution towards the study.

1. The network model designed gives a good picture of how members join and leave the network without disorganizing packet streaming.
2. We have designed an algorithm that optimizes delay along the peer links and we find that the nature of growth of the network is linear.
3. The paper clearly gives how a peer can easily find a resource at the other peer node subject to the resource function: from the time the peer joins and leaves the network.
4. These searching of resources at the streaming (source) node is found to be proportional to the searching peer. We define the constant of proportionality as the magnitude, $\gamma$, which varies depending on the placement of the nodes within the network.

5.4 Related Work

Rongmei et al. [2004] proposed the improved search in unstructured peer-to-peer (P2P) overlay networks by building a partial index of shared data. The index maintains two types of information: the top interests of peers and globally unpopular data, both characterized by data properties. A variety of P2P systems for sharing digital information are currently available and most of them perform searching by exact key matching. Panos et al. [2003] focused on similarity searching and describe Fuzzy peer, a generic broadcast-based P2P system that supports a wide range of fuzzy queries. As a case they studied and presented an image retrieval application implemented on top of Fuzzy peer where users provide sample images whose sets of features are propagated through the peers. Tzanetakis et al. [2004] describe a robust, scalable P2P system that provides flexible search semantics based on attribute value pairs and supports automatic extraction of musical features and content-based similarity retrieval in P2P networks. Srdjan et al. [2005] present P2P sensor networks overlay on 3G mobile networks. Each sensor network acts as one peer node and is represented by its gateway in the P2P network. Peers communicate and provide collaboration and information on the executions. Zihui et al. [2003] develop a mathematical model to explore and illustrate the fundamental performance issues of peer-to-peer file sharing. Their model showed flexibility in different characteristics of the P2P system. Much work has been published in the peer-to-peer optimization topology, [Aaron et al. 2004] however little work has been done on the resource optimization between neighbor peers. They propose traffic based learning protocol that learns new connections between the neighbors. Nazanin and Reza [2006] examined the key issues and tradeoffs in incorporating swarm-like content delivery into mesh-based peer-to-peer streaming of live content. Ashen and John [2003] propose a rank-based selection mechanism for peer-to-peer media streaming systems. Their mechanism provides a differentiation of services with incentives for cooperation.
6. Conclusion and Future Work

Communication in peer-to-peer networks is scaling very fast with the changing Internet and support of wireless technology. Peer nodes require to stream and search for resources with minimum delay constraint. The paper has presented a delay optimization algorithm that caters for the minimum delay path along a streaming link in the network, for a scaling peer group. Where each link is given a link magnitude that helps in identifying the QoS requirements of any given link in the network. Further the link delay requirements are noted for each link with resource level1, such that its easy to measure the delay on the link that attempts to stream resources. We further looked at the way peers behave to minimize congestion during streaming. Our proof of the correctness of the algorithm supports the objective of the study and it shows that the members scale linearly, with a time complexity of $O(n)$, which is consistent with the precondition and the post condition of the algorithm. In future we intend to consider how an insecure peer node in the network affects P2P multimedia streaming. And how that peer node on attack can increase delay and affect searching. Our interest will be in securing a peer node with resources to share.

References


ADNAN, F. 2004. Scheduling Policies for Enterprise Wide Peer-to-Peer Computing Department of Computer Sciences University of Saskatchewan 57 Campus Drive Saskatoon, SK, Canada.


