Volume 4, Issue 11, November 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

NSP: Price Variation for Network Users

A. Malla Reddy[1], Rukmaji Rao[2], R. Yadagiri Rao[3]
[1] Research Scholar (JNTUH), Department of Computer Science & Engineering, India
Professor & HOD, Sri Indu Institute of Engineering & Technology, Hyderabad, India
[2] M.Tech(CSE), Sri Indu Institute of Engineering & Technology, Hyderabad, India
[3]Associate Professor, Sri Indu Institute of Engineering & Technology, Hyderabad, India

Abstract- Recent studies have shown that the majority of today's internet traffic is related to Peer to Peer (P2P) traffic. The study of bandwidth in P2P networks is very important. Because it helps us in more efficient capacity planning and QoS provisioning when we would like to design a large scale computer networks. In this paper motivated by the behavior of peers (sources or seeds) that is modeled by Ornstein Ohlin beck (OU) process, we propose a model for bandwidth in P2P networks. This model is represented with a stochastic integral. We also model the bandwidth when we have multiple downloads or uploads. The auto covariance structure of bandwidth in either case is studied and the statistical parameters such as mean, variance and auto covariance are obtained. We then study the queue length behavior of the bandwidth model. The methods for generating synthetic bandwidth process and estimation of the bandwidth parameters using maximum like hood estimation are presented.

Key Words: Bandwidth Modeling, Peer to Peer networks, Ito calculus, Long Range Dependence, OU process, Maximum Like hood Estimation.

1. INTRODUCTION

The term "peer-to-peer" refers to a class of systems and applications that employ distributed resources to perform a function in a decentralized manner. Benefits of peer to peer systems are cost sharing/reduction, resource aggregation, increased autonomy, anonymity/privacy of the users and finally enabling ad hoc communication and collaboration [1]. Napster [2] was the first popular peer to peer service. This service has allowed hundreds of thousands of users to efficiently share MP3 formatted files. The success of Napster was a big motivation and several other peer to peer file sharing systems were introduced. These include KaZaA [3], Gnutella [4], and eDonkey [5] and BitTorrent [6]. According to Cache Logic [7] by the end of 2004, BitTorrent accounted for as much as 30% of all Internet traffic. Peer to peer represented 60% of Internet traffic at the end of 2004. Most dominant P2P systems are BitTorrent, eDonkey and Gnutella. The number of people that use peer to peer file sharing is growing. In January 2005, 2,975,477 online eDonkeyy2k users were reported. In January 2006, the number was increased to 3,351,754 [8].

The number of P2P users, the average file size transported in P2P files sharing systems and percentage of overall network traffic include P2P network traffic are growing, so bandwidth management plays an inevitable role in designing efficient computer networks nowadays. We still do not have accurate models for P2P bandwidth. In this paper, we propose a novel model for modeling P2P bandwidth. To achieve this goal, we should consider both the customers and the share of these customers. The customers are peers and the shares are their traffics in a given network. The second part was accomplished in the authors' previous work. So our starting points deeming the behavior of peers and pondering it. We use OU type process to describe the peer behavior. We then proceed to model the P2P bandwidth. We use the stochastic calculus approach in our proposed model. Some statistical parameters of the bandwidth model are derived. We also present a model for the total bandwidth .It is shown that the total bandwidth asymptotically has a LRD property. We also derive the length of the buffer fed by the bandwidth. A method for generating synthetic bandwidth is presented. Estimation of bandwidth parameters is another contribution of this paper.

II. SERVICE DIFFERENTIATION IN P2PNETWORKS

In this paper, we assume that the peer reputation scores are used to map them into various LoSs. The parameters that can be used to define each LoS are discussed in section 2.2. Assuming that a service differentiation scheme consisting of 3levels of service (LoS) is used, gure 1shows an example of how the peer reputation scores (RSs) can be mapped to various LoSs using parameters a and b. In this scheme, peers for whom RS < a are eligible for basic LoS. a ≤ RS ≤ b provides enhanced LoS to the peers. Peers with RS>b receive premium LoS. The parameter a and b is expected to be known to each peer in the P2Pnetwork.

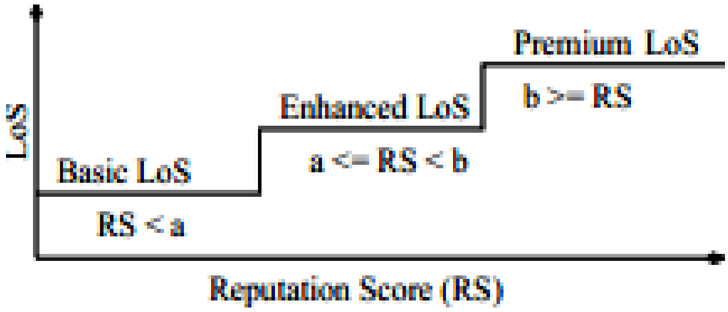

Figure1: Anexampleof3levelsofservicein aP2Pnetwork.

2.1 MainP2PFunctions

Peers carry out three main functions in a P2Pnetwork. Cooperation among the peers is required for all these functions. The overall experience of a peer in aP2Pnetworkdepends on the network conditions and the services and resources provided by the other peers during each of these functions. The network conditions depend on many factors that may not be controllable with in aP2P over lay topology and as a result are not considered in this paper. Focusing on decentralized unstructured P2P networks, this section describes the typical functions peerscarryoutinaP2Pnetwork.

Bootstrapping: Bootstrapping is required to allow peers to join the network. Most unstructured P2P networks use something similar to a GWeb Cache. GWeb Cache servers (typically several hundred in number) are web servers running a special module that allows the Gnutella servents[2] to query them to find the addresses of other servants. These servers comprise a distributed system to solve the boot strapping problem.

Content search: The content retrieval process in all the existing avers of P2P networks involves a content search phase before it can be downloaded. In unstructuredP2Pnetworkslike Gnutella, to search for the desired content, querying peer generates a query with appropriate key words and sends it to all the peer's that it is directly connected to in the overlay topology. The peers who process this query reply back if they have the content in their shared directory and forward their quest to the peers they are directly connected to depending on the hop-count (or the TTL) of the query. This forwarding continues until the TTL specie by the querying peer is exhausted.

Content download: The second phase in content retrieval involves selecting peer to download the content from. The querying peer selects a peer after receiving all the replies from the content search phase. At that point, the content download typically uses a H TTP or a TCP connection with the selected peer.

2.2 Parameters for Service Differentiation

These of parameters that can be mapped to each LoS are guided by the factors that provide service differentiation during the boot strapping, content search, and content download functions in aP2Pnetwork;andhencethe peer' perception of service quality. These factors are based on the salient features of the widely deployed unstructuredP2Pnetworksandthe results of the current researchonunstructuredP2Pnetworks.

2.2.1 Factors Affecting Bootstrapping

During the bootstrapping process, the type of peers a peer directly connects to in the over lay topology play an important role in its overall satisfaction from content search and download functions later on. For example, apart from how cooperative the connecting peers are, their actual network distance, processing power, memory, bandwidth, and storage capacity are important factors.

2.2.2Factors Affecting Content Search

The following factors can be used as a basis for service differentiation because they impact the perception of service.

Number of hops: To search content, the querying peer sets the maximum number of hops in the overlay topology its query would take, by denoting a hop-count. While the success of the content search phase depends on many other factors as well, the number of hops plays an important role. Hence, setting a hop-limit could act as a component of the service differentiation scheme. For a peer during the content search.

Premium content: Peers can choose to classify some of the content they share as premium content, which they can make available only to peers eligible for certain minimum LoS. This classification can be done through k some system wide guide lines.

Hard to find content: AspecialutilityoftheP2Pnetworksfor many peers comes from being able to access hard-to-find content. Although classifying content as hard-to-find may be based on subjective criteria, peers can potentially reserve the hard-to-find content only for peers with a certain minimum LoS.

Query caching: Sripanidkulchai [11] found that the popularity of search strings in Gnutella follows a Zipf – like distribution and that caching a small number of queries results in a significant decrease in the trace in the Internet. In order to distinguish among the peers with various LoSs, the outcome of caching queries may be made available only to peers with a certain minimum LoS eligibility.

Cached content: Kazaa distinguishes between the functionality of super nodes and the rest of the peers in its P2Pnetwork. Peers with higher band widths can choose to becomesupernodesinaKazaaP2Pnetwork.Duringidle periods,

these upper nodes actively query other peers in the network and cache the content so retrieved. This gives the super nodes access to additional content and when queries for the cached content arrive at the super nodes, they can get served faster. For faster retrieval of content in unstructuredP2Pnetworks, Cohenet.al. [12] Have also proposed caching strategies. Due to its Potential to improve the peer experience, caching could be used to distinguish among the service provided to peers with various LoSs.
Interest-based locality: By exploiting interest-based locality, Sripanidkulchaiet.al.[13] have propose ancient content search solution for unstructuredP2Pnetworks. The basic idea is for peers that share similar interests to create shortcuts to each other. These shortcuts can then be used to locate content faster. The basic Gnutella content search paradigm remains as a backup mechanism. In creating such shortcuts, peer LoS eligibility may be used as an additional deciding factor.
Load balancing: An enhancement to maintaining a shortcut to peers that share common interests (as described above) would be to maintain the most recent load for those peers as well. Such an information can help in avoiding the already over whelmed peers and potentially get content faster. The availability of such information however would require a periodic protocol to assess the load for the peers with similar interests. But if this information is available, it can be provided to peers eligible for certain minimum LoS during content search phase.

2.2.3 Factors Affecting Content Download
During the content download phase, following factors can be used as a basis for service differentiation because they impact a peer's overall experience:
Rate of transfer: During content download from the chosen peer, the rate of transfer may be dependent on the LoS the downloading peer is eligible for. The basic idea is to restrict the portion of capacity used to serve the peers with less than a certain LoS. This restriction may either be in effect all the time or may be used only during periods of heavy loads.
Scheduling policy: During periods of heavy load or even otherwise, the peers may use various scheduling policies in order o give priority in serving content to the peers with premium LoS over the peers with enhanced LoS and to enhanced LoS over basic LoS.

III. SERVICE DIFFERENTIATION PROTOCOL (SDP)
This section describes SDP, a protocol to accomplish service differentiation in P2Pnetworks.

3.1 Basic Assumptions:
SDP enhances the basic functionality of Gnutella-likeunstructuredP2Pprotocolstoinclude the service differentiation functionality. It assumes that the peers have immediate access to their own reputation scores. Assuming that the network stores reputations in a decentralized manner, one way to accomplish this is through local storage of reputations. However, security issues in such storage need to be carefully addressed. Another alternative is to compute reputations just-in-time from a distributed storage. These issues are described in detail in section 4.SDP is flexible about the structure of peer reputations. The reputation score could be a scalar or a vector. There is no requirement on how the freshness of the reputation score is guaranteed. If the underlying reputation system wishes to guarantee freshness, it can assign a time-stamp to the reputation scores. SDP only requires that the structure of the reputation scores be known to all peers. Several examples of peer reputations exist in the literature today [5, 6, 7, 8, 9, and 10].Further, SDP assumes that the aping of the reputation scores to the Loss is known to all the peers in theP2P network. Such a mapping could be statically conjured by being a part of the software itself or could be downloaded from the GWeb Cache servers during the boot strapping process. Also, the peers are expected to know what values of the parameters described in section 2.2aremappedto each LoS.The anonymity issues in SDP are dealt with in a manner similar to those in the popular unstructuredP2P protocols. The details of this and other security issues are discussed in section 5.

3.2 SDP Details
During the boot strapping process, most popular unstructured P2P protocols provide an option to connect only to high-capacity (in terms of band width, processing power, memory, and storage) peers. Such high-capacity peers are referred to as super nodes in Kazaa and ultra peers in Lime wire (www.limewire.com). Additionally, binning scheme of the type proposed in [14] can be used to allow peers to connect to peers close-by in the Internet topology. These factors impact the QoS perceived by the peers and can being corporate in a service differentiation scheme. In this paper, we focus only on the service differentiation during the search and download process of the content retrieval.
For the subsequent SDP description, we assume that an unstructured P2P protocol like the one described in the Gnutella speciation [15] is being used. The terminology used is the same as used in that speciation and all enhancements proposed by SDP are on top of such a protocol.

3.2.1 Content Search
This section describes show the content search part of unstructured P2P protocols can be modified to incorporate the service differentiation functionality. The parameters used by SDP to provide service differentiation during this phase are the same as those described in section 2.2.2.
Search phase1: The peer who initiates the search request sends its reputation score along with the Query message. We refer to this enhanced query as Query SDP. This requires enhancing the Gnutella query to include the peer's reputation score in addition to the standard fields like TTL, hops, search criteria etc. The basic idea behind Query SDP is to allow each that processes the query to have access to the querying peer's reputation score.

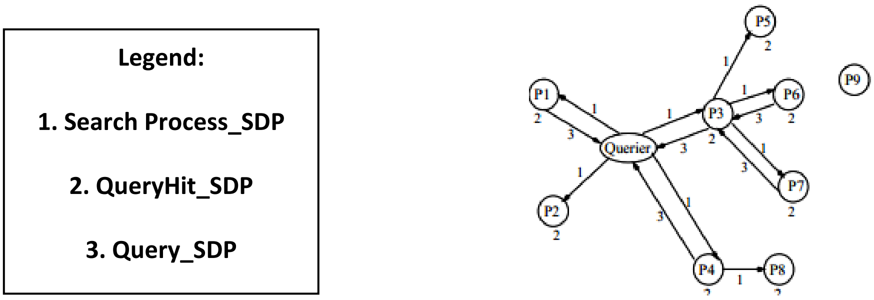

Figure2: SDP during content search for a peer eligible for basic LoS.

Searchphase2: Each peer who receives Query SDP extracts the reputation score. This score is used to map the peer to the LoS it is eligible for and process the query accordingly. The LoS specie processing is referred to as the search Process SDP. Since the processing is dependent only on the reputation score, SDP does not require any identification for the querying peer. Also, the peers who process the query do not have to cache the reputation scores for any other peer in this scheme. This is because the reputation scores may change overtime.

In order to map the reputation score to a LoS, parameters a and b of the type described in section 2. As described before, if the system supports 3LoSs, peer with reputation score<a would be eligible for a basic LoS. a ≦ reputation score <b would be used to provide enhanced LoS to the peers. Peers with reputation score > b would receive premium LoS.

Examples of functions that would be a part of search Process SDP are shown in gure3. These functions would provide appropriate LoS using the parameters described in section 2.2.2.Assuming3LoSsimplies thatthereare3separatefunctions, one for each LoS. The functions in gure3assumethatthe number of allow able hops for basic, enhanced, and premium LoS are given by hops basic, hops enhanced, and hops premium respectively. The basic idea behind these functions is the following. If a peer's query has already traversed more hops than it was eligible for, it is dropped immediately. However, if it has not traversed extra hops but would go farther than should (based on the TTL + Hops), then appropriate value for the TTL needs to be set. Further, for enhanced and premium LoS peers, additional look ups are needed for service differentiation.

After the LoS septic processing, the query continues to be processed per Gnutella guidelines. This is denoted by process Query at the end of each of the functions in 3. Since interest based locality and load balancing parameters from section 2.2.2requireadditionalprotocolsto be run, we eliminate them from these functions. However, if such information is available, it cans bein corporate easily. As a result, the functions currently use query caching results, cached content, premium content and hard-to-find content for differentiating among the peers.

Search phase 3: Notice from the functions in figure3 that the LoS specie query processing may amount to dropping the query. But if a query is not dropped duringsearchphase2, the peer forwards it on its outgoing interfaces according to Gnutella speciations. In case a response is to be sent back to the querying peer after processing the query, Query Hit SDP is sent. Query Hit SDP is an enhancement to the Gnutella Query Hit message. Query Hit SDP allows peers who reply to optionally put their reputation scores in the response. This is to help the querying peer make a decision about who to download the content from based on the reputation of the responders.

3.2.2 Content Download:

This section describe show SDP enhances the content download process in unstructured P2P protocol s to in corporate support for service differentiation. SDP uses the parameters described in section 2.2.3for this phase.

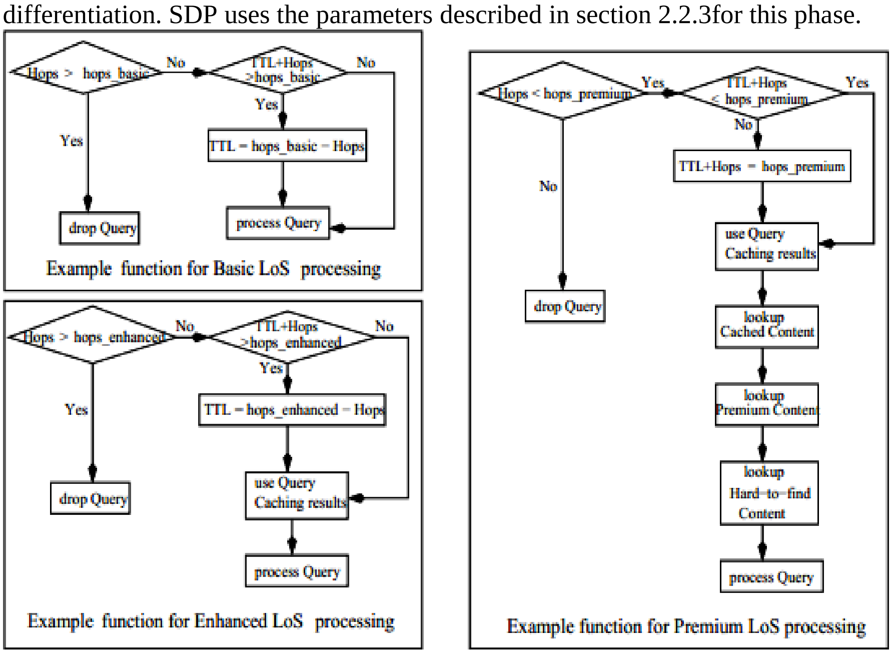

Figure3: Functions for service differentiation during content search.

Download phase1: After selecting a peer to download from, the querying peer connects to the selected peer using a TCP or HTTP connection. Just as in searchphase1, this phase also requires the querying peer to send its reputation score while establishing the connection for downloading.

Downloadphase2: Before serving the content, the sender peer maps the reputation score to the LoS the requester peer is eligible for. Once the LoS is decided, the sender peer picks the appropriate rate of transfer and scheduling policy for the LoS.

The topic of what transfer rates to use and the particular scheduling policies employed needs more research and is beyond the current scope of this paper.

IV. REPUTATION SCORES

This section describes show the SDP requirements translate into guide lines for a reputation system. It also compares the existing reputation systems for their suitability to accomplish service differentiation using SDP.

Tobeabletoprovideservicedifferentiation, SDPrequiresreliablereputationscoresto be available to the peers who process the content search and download requests. Instead of having to deduce the reputation scores by each peer who needs it, SDP assumes that they a resent by the requester peers along with the request. As described in section 3.2, SDP is flexible about the structure of the reputation scores.

Various reputation systems that store reputations in a distributed form have been proposed in the recent research literature [5, 6, 7, 8and 9] for the purposes of helping peers a void transactions with malicious peers. Although these proposals differ from each other in many ways, the following common features make them unsuitable for SDP:

A Partially Distributed Reputation System:

The reputation system proposed in [10] satisfies the requirements of SDP. This partially distributed solution uses a trusted reputation computation agent (RCA) to compute scalar and objective reputations. The basic idea behind ensuring reliable reputation computations is for the peers to collect the credit for their work in the system and periodically contact the RCA to have that credit converted into a reputation equivalent. The RCA encrypts the reputation score by its public key. Since RCA's key is assumed to be known to all the peers, any peer can decrypt the reputation score. Peers store their reputations locally for fast retrieval in this proposal.

To respect the privacy concerns of peers, the solution requires the peers to explicitly enroll with the RCA to participate in the reputation computations. This is a function that needs to be carried out only when a peer joins the P2P network for the first time. The new peers and the peers who choose not to enroll in the reputation computations always have a default reputation score of0in the system. By earning a good reputation score, they can upgrade their reputation score. The reputations in this system either expire or a reused-up by the peers. Hence, they do not always increase. Thus, in order to maintain a good reputation, peers need to continually participate in the system. The reputations a resaved across sessions, so peers with good reputations can continue to be net from them.

V. EVALUATION

SDP can be evaluated along the following 4 dimensions:

Effectiveness: The actual differentiation in services provided to peers belonging to different LoSs during the content search and download phases is an important measure of effectiveness of SDP.

Sensitivity to participation: Expecting that all peers in the system run SDP for it to be effective is not possible due to various reasons. Peers could be running different versions of the underlying Gnutella protocol and also could be malicious. As a result, it is important to gauge the sensitivity of SDP to the extent of participation required from peers in implementing the SDP functionality.

Overheads: SDP enhances the Gnutella Query and Query Hit messages to include the reputation scores. It also involves extra processing on the part of peers who process the content search and download requests. Though the processing as well as the bandwidth over heads of SDP are expected to be minimal, an estimation of these over heads is another dimension of evaluation.

Impact of parameter values: The exact values of parameters used while mapping them to various LoSs is another important consideration. For example, the percentages of premium and hard-to find content, the number of cached les, and the number of cached queries are useful factors in the deployment of SDP.

VI. CONCLUSION

We presented SDP, a protocol for service differentiation in decentralized unstructuredP2Pnetworks. SDP uses parameters that affect the content search and download functions to provide different LoSs. It accomplishes this by enhancing the Gnutella specification. Though SDP is independent of the underlying reputation system used to decide on the LoS a peer is eligible for, a system where peers store their reputations locally is the most efficient. A preliminary evaluation of the service differentiation achieved during the search phase shows the promise of the approach.

REFERENCES

[1] S.Ratnasamy, P. Francis,M.Handley, R.Karp,andS.Shenker,\Ascalable content addressablenetwork," in ACMSIGCOMM, Aug.2001.

[2] I. Stoica,R.Morris,D.Karger, F.Kaashoek, and H.Balakrishnan,\Chord: A scalable Peer-To-Peerlookup servicefor internetapplications," in ACMSIGCOMM, Aug.2001,pp.149{160.

[3] E.Adarand B.A.Huberman, \Free ridingonGnutella,"Tech.Rep.XeroxPARC,2000.

[4] S.Saroiu,P. K.Gummadi, and S.D.Gribble, Ameasurement study of peer-to-peer lesharingsystems,"in SPIE Conference on MultimediaComputingandNetworking(MMCN), Jan.2002.

[5] K.AbererandZ.Despotovic,\Managingtrustin a peer-2-peer information system," in Ninth International Conference on Information and Knowledge Management (CIKM), Nov.200.

[6] E.Damiani,S.DeCapitanidiVimercati, S.Paraboschi,P. Samarati, and F.Violante, \Areputation-based approach for choosingreliable resources in peer-to-peernetworks,"in 9th ACM Conference on Computerand Communications Security, Nov.2002.

[7] S.D.Kamvar, M.Schlosser, and H.Garcia-Molina,\Eigenrep: Reputation management in p2pnetworks," Unpublishedwork,2003.

[8] S.Lee, R.Sherwood, and B.Bhattacharjee, \Cooperative peer groups in nice,"in IEEEINFOCOM, Apr.2003.

[9] L.Xiong and L.Liu, \Buildingtrustin decentralized peer-to-peer communities," in International Conference on Electronic Commerce Research (ICECR-5), Oct.2002.

[10] M.Gupta,P. Judge, and M.H.Ammar, \Areputationsystemfor peer-to-peernetworks,"in NOSSDAV, June 2003.

[11] K.Sripanidkulchai, \The popularity of gnutella queries and its implications on scalability," White Paper FeaturedonO'Reilly'swebsitehttp://www.openp2p.com/,Feb.2001.

[12] E.CohenandS.Shenker,\Replication strategies in unstructured peer-to-peer networks," in ACMSIGCOMM,Aug.2002.

[13] K.Sripanidkulchai,B.Maggs,andH.Zhang,\Ecient content location using interest based locality in peer-to-peer systems," in IEEEINFOCOM, Apr.2003.

[14] S.Ratnasamy, M.Handley, R.Karp, and S.Shenker, \Topologically-aware overlay construction and server selection," in IEEEINFOCOM, Apr.2002.

[15] Gnutellaprotocolspecication,"http://www9.limewire.com/developer/gnutellaprotocol0.4.pdf/.

