Analyzing peer-to-peer traffic across large networks

Subhabrata Sen and Jia Wang
AT&T Labs–Research
180 Park Ave., Florham Park, NJ
{*sen,jiawang*}*@research.att.com*

Abstract—**The use of peer-to-peer (P2P) applications is growing dramatically, particularly for sharing large video/audio files and software. In this paper, we analyze P2P traffic by measuring flow-level information collected at multiple border routers across a large ISP network, and report our investigation of three popular P2P systems – FastTrack, Gnutella, and DirectConnect. We characterize the P2P traffic observed at a single ISP and its impact on the underlying network. We observe very skewed distribution in the traffic across the network at different levels of spatial aggregation (IP, prefix, AS). All three P2P systems exhibit significant dynamics at short times scale and particularly at the IP address level. Still, the fraction of P2P traffic contributed by each prefix is much more stable than the corresponding distribution of either Web traffic or overall traffic. The high volume and good stability properties of P2P traffic indicates that the P2P workload is a good candidate for being managed via application-specific layer-3 traffic engineering in an ISP's network.**

I. Introduction

The use of peer-to-peer (P2P) applications is growing dramatically, particularly for sharing large video/audio files and software. The stunning growth and the bandwidth intensive nature of such applications suggests that P2P traffic can have significant impact on the underlying network. It is therefore important to understand and characterize this traffic in terms of end-system behavior and network impact in order to develop workload models and provide insights into network traffic engineering and capacity planning.

P2P traffic can be broadly classified into two categories: signaling and data transfer. Both type of traffic need to be measured in order to gain a solid understanding of P2P system behavior. The signaling traffic includes TCP connection setup, search queries and query replies. Early P2P systems like Gnutella used controlled flooding to propagate queries to all the P2P hosts – this can lead to bandwidth scaling problems. Newer systems such as FastTrack and DirectConnect and newer versions of Gnutella perform more targeted forwarding to only a subset of hosts (we shall describe this later in the paper), and are much more bandwidth efficient in terms of signaling.

The leading content shared in the P2P systems, such as audio and video files, tend to be large in size, e.g., 4.8 MB for a 5 minutes long 128 Kbps MP3 audio clip, 450 MB for a 2 hour long MPEG-4 video clip encoded at 500 Kbps. In comparison, typical query and response messages are much smaller in size, on the order of several hundred bytes. Hence, the actual data transfer is likely to be the dominant component of the total traffic in such systems, and have significant impact on the underlying network.

Previous research [1], [2], [3], [4] has focused almost exclusively on P2P signaling traffic, and on the public-domain Gnutella and Napster systems. These projects all gather P2P signaling traffic by setting up P2P crawlers on the Internet. The crawler joins the P2P network and maintains active TCP connections with a number of hosts (called *neighbors*). It iteratively builds a list of hosts in the system by communicating with known hosts, and adding newly discovered hosts to its known list. It logs all the messages that it sends to and receives from other hosts. Since the data collection depends on the number of active TCP connections that the crawler maintains, such an approach is not suitable for conducting large scale data gathering. In addition, this inherently active probing approach makes it an expensive proposition from a bandwidth perspective, to map large P2P systems which can have several million hosts.

Following are a number of interesting research questions that have implications for P2P system design and traffic engineering.

- How is the P2P traffic distributed across the Internet? The spatial distribution characteristics, for instance, can influence traffic management decisions, such as identifying potential hot spots for capacity planning.
- What are the characteristics of the application-level P2P network connectivity? The connectivity behavior can yield insights towards developing appropriate protocols for searching and for fetching objects on such a system.
- How dynamic are the P2P systems, both temporally and spatially? This understanding can yield clues for developing systems with good performance properties in terms of scalability, reliability and reachability.

This paper performs a systematic characterization of P2P traffic and its impact on the underlying network, as a first step to answering the questions above. Complementing the earlier techniques, in this paper, we present a novel approach for conducting large scale non-intrusive measurement of P2P traffic covering both signaling traffic and actual data traffic, that can be used for mapping both proprietary and non-proprietary P2P systems. We focus on the P2P traffic observed at a large ISP, and aim at characterizing the workload and understanding its impact on the underlying ISP network. This is of interest to service providers because the P2P traffic has increased dramatically during the last couple of years and accounts for a significant portion of the total traffic observed at large ISPs. The workload characterization will enable service providers to better cope with such traffic through suitable traffic engineering measures such as identifying heavy hitter network prefixes for private route peering arrange-

ments, pricing, rate limiting, and routing.

We extracted and analyzed 800 million flow-level records collected at multiple border routers across the ISP's network over a period of 3 months. Our study focused on three popular P2P systems – FastTrack [5], Gnutella [6], and DirectConnect [7]. We had 4 major observations, (i) All three systems exhibit significant increases in both traffic volume and number of users, even across consecutive months. The traffic volume generated by individual hosts is extremely variable – less than 10% of the IPs contribute around 99% of the total traffic volume. (ii) The P2P traffic distributions for traffic volume, connectivity, on-time (to be defined), and average bandwidths usage are extremely skewed. (iii) All three P2P systems exhibit a high level of system dynamics – only a small fraction of hosts are persistent over long time periods. (iv) The fraction of P2P traffic contributed by each network prefix remains relatively unchanged and much more stable than the corresponding distribution of either Web traffic or overall traffic over the time period of one month. This is good news for ISPs, as the high volume and good stability properties of P2P traffic indicates that application-specific layer-3 traffic engineering may be a promising way to manage the P2P workload in an ISP's network.

The remainder of the paper is organized as follows. Section II presents our methodology for analyzing P2P traffic. Section III describes the metrics we use for traffic characterization. We provide an overview of traffic data for the three P2P systems in Section IV and examine the P2P system dynamics in Section V. Section VI explores two key questions towards modeling P2P workload and Section VII compares P2P traffic with Web traffic. We summarize the main results in Section VIII and finally conclude the paper in Section IX.

II. Methodology

We focus on three popular P2P systems – the open source Gnutella (the network accessed by client interfaces such as Bearshare and Limewire [8], [9]), and the proprietary FastTrack (better known by the popular client names KaZaA and Grokster [5], [10]) and DirectConnect systems. At the time our measurements were conducted, the popular Morpheus [11] file swapping service was using the FastTrack system, and our data includes the Morpheus traffic as well. We first highlight some key features of these systems and then outline our data collection and measurement methodology.

A. Popular P2P applications

FastTrack, Gnutella and DirectConnect are all decentralized, self-organizing file sharing systems with data and index information (metadata for searching) distributed over a set of end hosts or *peers*, each of which can be both a client and a server. Hosts can join and leave frequently, and organize in a distributed fashion into an application-level overlay via point-to-point application-level connections between a host and a set of other hosts (its neighbors). All the communications occur over default, well known ports. The process of obtaining a file can be broadly divided into two phases. First, a host uses the P2P protocol to search the hosts in the P2P system for a particular resource, receives one or more responses, and identifies one or more target hosts from which to download that resource. The search queries as well as the responses are transmitted via the overlay connections using protocol-specific application level routing. The details of how the signaling is propagated through the overlay is protocol-dependent. In Gnutella, all hosts are considered equal and participate in query processing. A host initiates a query by flooding it to all its neighbors in the overlay. The neighboring hosts in turn, flood to their neighbors, using a scoping mechanism to control the query flood. In contrast, for both FastTrack and DirectConnect, queries are forwarded to and handled by only a subset of special hosts (called *SuperNodes* in FastTrack and *Hubs* in DirectConnect). A host transmits an index of its content to the "special host" to which it is connected. The special host then uses the corresponding P2P protocol to forward the query to other such hosts in the system. Newer versions of the Gnutella protocol adopt a similar approach with such special hosts called Reflectors, Defenders or Ultrapeers [12], [13].

In the second phase, the requesting host directly contacts the target host, typically using HTTP (the target host runs has a HTTP server listening by default on a known, protocol-specific port), to get the requested resource. Some newer systems, such as FastTrack and Gnutella, use file swaming – a file is download in chunks. The term *P2P network* has been typically used in existing works to refer to the application-level peer-to-peer connections used for signalling among the hosts, and does not consider the download path followed by the actual data.

B. Measurement approach

Any large scale measurement effort has to be efficient and scalable in terms of network resource usage, should not impact the system being measured, and should be able to capture the behavior and system dynamics in sufficient detail. The highly decentralized, self-organizing nature of a P2P system, the large number of hosts involved, the transient nature of peer membership, and the closed proprietary nature of some of the most popular P2P systems in existence make it a challenging proposition to gather information for mapping and characterizing such systems in terms of network topology, generated traffic, and dynamic behavior.

In this study, we adopt a *passive* measurement approach, involving post-mortem analysis of flow-level data gathered from multiple routers across a large tier-1 ISP's backbone. We measure each P2P system at several levels of granularity: IP address, network prefix, and AS (Autonomous System). IP level information is interesting as it provides a fine-grained view of the load distribution across the network. Note that in general there may not be a one-to-one mapping between each host and an a unique valid IP address, because of the use of dynamically assigned IP addresses, NAT (Network Address Translation) and forward proxies at the edge of the network. However, this has less effect on our analysis results since we are interested more in characterizing the overall traffic pattern and the continuous activity for the hosts during each single day period instead of analyzing the traffic from individual hosts across days. Also, since each IP address maps to a unique interface (subnet) at the edge of the network, IP level analysis is still useful for understanding the overall traffic distribution.

As an intermediate level of granularity, we use the network

routing prefix for characterizing P2P traffic patterns. Prefixes are the unit of routing at the IP layer, so understanding traffic at this level is important for ISP traffic engineering. Also, the prefix level aggregation, by grouping IP addresses that are topologically close together from a network routing viewpoint, enables capturing locality characteristics in the P2P system. At the AS level, we identify an AS by its unique public AS number. The dynamic assignment of IP addresses is less of an issue for the prefix and AS level aggregations since one would expect a host's new IP address to fall in the same prefix and originating AS as the old one.

We collect router level data using Cisco's *NetFlow* services. *NetFlow* [14] enables accumulation of traffic flow statistics. A *flow* is defined to be an unidirectional sequence of packets between a particular source and destination IP address pair. The source and destination IP address pair, transport layer application port numbers, IP Protocol type, Type of Service (ToS) and the input interface identifier are used to uniquely identify a flow. For each flow, *NetFlow* maintains a record in the router cache, containing a number of fields including the source and destination IP addresses, source and destination BGP routing prefixes, source and destination ASes, source and destination port numbers, the protocol type, type of service, flow starting and finishing timestamps, number of bytes and number of packets transmitted. The BGP prefix and AS information are obtained by running longest prefix matching on the IP addresses with prefixes in the router's forwarding table entries. A flow is expired from the cache and a corresponding *flow record* transmitted via UDP to a *NetFlow collector* machine for storage, under one of the following conditions: there is no activity between the corresponding two endpoints for a certain period of time, the router's cache gets filled up, or the flow is active over long periods of time (by default, a flow is expired if it is active for more than 30 minutes).

C. Advantages and limitations

The following are some key advantages of our passive network wide measurement approach over earlier efforts that use active probing. First, our approach does not require knowledge about the P2P protocol, beyond port number information. This is a clear advantage for studying proprietary protocols such as FastTrack. The same would be far more difficult to do using an active measurement approach which would require a P2P crawler to actually join the P2P network, and therefore involve intimate knowledge of the P2P protocol itself and of any encryption being used.

Second, the approach is non-intrusive and all the traffic data can be collected without interfering with or impacting the peers themselves. We can conduct more complete measurements of large systems over long periods of time which would be prohibitively expensive in an active measurement based approach. As a result, we can get a more complete view of the P2P host distribution and their traffic patterns. From the underlying network viewpoint, the flow data collection does involve additional overheads, as the router cards have to gather flow-level records and ship these to a collection server. However, the usefulness of such measurements as inputs to a range of practical applications such performance monitoring, traffic engineering, and capacity provisioning, is driving ISPs to increasingly deploy measurement infrastructures such as flow and packet monitors in their networks. In a way, therefore, the network is engineered to handle any additional overhead of such netflow collection.

Third, our approach gathers information on both the P2P signaling traffic as well as the actual data download traffic. Given that these systems are being used to download large files, it is important to be able to capture and characterize the actual data traffic. This is a key distinction from prior work which was able to profile the signaling traffic only.

Finally, by controlling which routers the data is gathered from, our approach is conducive to determining the impact of P2P traffic on certain regions of the network – e.g., the total internal P2P traffic or the total incoming or outgoing P2P traffic for a single ISP. Such localized analysis capability would be important and desirable, for instance, for local traffic engineering and provisioning at an ISP.

While flow-based analysis provides us with valuable insights into P2P traffic characteristics, it has some limitations. First, the data is aggregated at the flow-level. We are not able to obtain application-level details such as the actual P2P messages exchanged between peers, or the specific files being requested and actually downloaded. Given the recent trend towards secure communication by some P2P systems (e.g., FastTrack encrypts all signaling) to prevent unauthorized clients from accessing the network, this problem will be common to all third-party based evaluations of P2P systems.

Second, we may not capture the complete flow of traffic. In this work,we gathered *netflow* data in the backbone of a tier-1 ISP, from a significant fraction of the border routers that are the conduits for traffic flowing to/from other tier-1 and tier-2 ISPs. We speculate that we observed a significant portion of the traffic from other top-tier ISPs entering or leaving the target ISP for those three P2P systems. We intend to periodically conduct the analysis on data sets gathered from more routers, as netflow deployment across the ISP increases. Such an exercise will also capture an updated view of P2P traffic behavior and help in tracking the evolution of this traffic.

Another potential issue is that, due to asymmetric IP routing, we may see only one direction of the traffic between a given pair of hosts. However, this is not a limitation for our measurements which aims to understand the P2P traffic pattern and its impact at single ISP, and therefore is concerned only with the traffic that is visible to that ISP.

III. Characterization metrics

The goal of our study is to characterize P2P systems behavior with a view to understanding how these systems impact the underlying network, and for gaining insights into developing P2P systems with superior performance. We are interested in: (i) *topology characterization* – the distribution of P2P hosts across the network, the topology of the application-level overlay connecting hosts; (ii) *traffic characterization* – the distribution of traffic volumes transmitted or received by different hosts; (iii) *dynamic behavior characterization* – in theory, the dynamic nature of a P2P system distinguishes it from traditional distributed server systems. We are interested in characterizing the dynamics observed in practice, e.g., how frequently hosts join and leave

the system, how long a host stays in the system, how active the "live" hosts are during a certain period of time, etc.

We use the following metrics to study the P2P behavior. We shall use the terms *upstream* and *downstream* respectively to refer to the direction of traffic emanating from a host, and traffic coming into the host.

A. Host distribution

We compute the number of unique IP addresses, prefixes, and ASes participating in each P2P system in each one-day period across several weeks spread over several months. This will indicate the trends in the size of the three systems. Comparing the measurement results at different levels of topological granularity, we can infer locality characteristics of the P2P hosts distribution which can be used in traffic engineering and P2P architecture design.

B. Traffic volume

Since P2P systems are mainly used for sharing audio/video files and software, the size of the files that are transmitted is much larger compared to the traditional Web content size. To better understand the P2P traffic patterns, we measure the traffic volume transmitted between P2P hosts, and compute the aggregate data transmitted or received by each IP address, prefix and AS per day.

C. Host connectivity

For each aggregation level (IP, prefix, or AS), we compute the total number of unique entities at the same aggregation level that it communicates with (either transmits to or receives data from). The resulting distribution is used to characterize the connectivity in the P2P network.

D. Traffic pattern over time

We measure the aggregate traffic characteristics across time for the different P2P systems. First, we would like to know how many hosts participate in the P2P system and the traffic volume transferred among hosts at a given time. We measure this by dividing the entire data set into small time bins. For each bin, we compute the number of unique entities (IPs, prefixes, or ASes) that are participating in the P2P system and the traffic volume transferred among them. It is possible that the starting time and the finishing time of a single flow fall into two different bins. In such a case, the traffic volume of this flow is divided into multiple segments and the traffic volume of each segment is assigned in proportion to the length of time that the flow exists within a given bin. The traffic volume of a given bin is the aggregate traffic volume of all the flow segments within this bin. Suppose there are n flows f_1, f_2, ... , f_n, that last over m bins b_1, b_2, ..., b_m. A given flow f_i that spans from bin j to bin k ($k \leq j$) is divided into $k - j + 1$ segments s_{ij}, ..., s_{ik}. Then, the traffic volume of bin b_l ($0 \leq l \leq m$) is computed as below.

$$Volume(b_l) = \sum_{i=1}^{n} Volume(s_{il})$$

Figure 1 illustrates an example of how to compute the traffic volume of each bin. There are 5 flows f_1, ..., f_5 that last in 8 bins b_1, ..., b_8. Flows are divided into segments. For example, flow f_1 lasts in bins b_1 and b_2, and is divided into segments s_{11} and s_{12}. The traffic volume of bin b_1 is the summation of the traffic volume of segments s_{11}, s_{21}, and s_{31}.

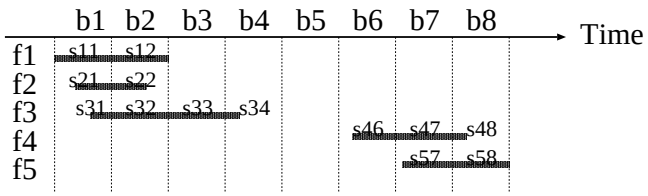


Fig. 1. The binning of *netflow* records.

E. Connection duration and on-time

We measure how long a host stays in the P2P system. We define a *connection* to correspond to the duration between a host joining and leaving the P2P system. Because we cannot obtain the information of a host connection from *netflow* records, we approximate the starting time and finishing time of a host connection using the time when the host starts to send or receive data and the time when the host finishes sending or receiving data. At a given time, the data transfer by a host may be spread over multiple flows. Let f_i and f_j ($i \neq j$) be two flows associated with the given host. Without loss of generality, we assume that StartTime(f_i) $\leq$ StartTime(f_j). We say f_i and f_j is *concurrent* iff StartTime(f_j) $\leq$ FinishTime(f_i) + δ, where δ is a threshold factor. Later, we shall show how to select the appropriate value for δ.

The host connection duration is computed as the longest consecutive period that the host is transferring (either sending or receiving) data. A connection c of a host can be represented by a set of concurrent flows associated with it. A host may have multiple connections over time, but there is no overlap in time between any two connections of a host.

$$StartTime(c) = \min_{f \in c} StartTime(f)$$

$$FinishTime(c) = \max_{f \in c} FinishTime(f)$$

$$Volume(c) = \sum_{f \in c} Volume(f)$$

The length of the host connection duration tells us how long a host stays in the P2P system once it joins. The *on-time* of a host characterizes how long the host stay in the P2P system during a certain period of time. It is computed as the sum of all the connection durations over a given time period.

F. Mean bandwidth usage

We would like to characterize the transmission bandwidth usage (for P2P traffic) by individual hosts in a P2P system. Such a characterization would enable network administrators to understand the bandwidth demand that hosts running P2P applications might impose on the network. In this study, we measure the average bandwidth a host consumes once it joins the P2P

system. Note that this is an estimate of the average bandwidth, as we may not capture all the traffic to and from a host. We separately measure the upstream and downstream bandwidths for each host. The upstream (or downstream) bandwidth is the aggregate average bandwidth at which the host transmits (or receives) data to (or from) other hosts in the P2P system. For a given host h,

$$Bandwidth_R(h) = \frac{Volume_R(h)}{OnTime_R(h)}$$

where R is either upstream or downstream and $OnTime_R(h)$ is the total time that host h is transmitting (or receiving) data (i.e., $\delta = 0$) during a certain period of time, respectively. This measure also gives us a lower bound on the bottleneck bandwidth of the hosts. Note that another relevant bandwidth measure here would be the distribution of maximum bandwidth usage for hosts in the P2P system. This would be useful, for example, in estimating the burstable bandwidth demand that may be imposed by the P2P system on the underlying network. However flow-level data only records the aggregate data transmitted in the flow and the total flow duration, and does not provide any information regarding potential short-time-scale burstiness in the transmission bandwidths at the application level. Using this coarse-grain information to estimate the burstable bandwidth usage of a host across multiple overlapping flows, can result in either under- or over-estimations of the actual peak. Determining accurate estimates of the peak bandwidth usage is part of future work.

IV. Overview of P2P traffic

The flow records from multiple border routers interfaces across the ISP backbone form the basis of our analysis. For each P2P system, we extracted records that matched the corresponding default application ports (source or destination), involving TCP traffic (no P2P system uses UDP): 6346/6347 (Gnutella), 1214 (FastTrack), and 411/412 (DirectConnect). The data was collected for one week each month between September and December, 2001. The collected data is then processed to handle corruption and loss effects as follows. We considered all the IP addresses that are in the following ranges as invalid IP addresses: 10.0.0.0 - 10.255.255.255, 172.16.0.0 - 172.31.255.255, and 192.168.0.0 - 192.168.255.255. We eliminate records (i) for which either the source or the destination is an invalid IP address, (ii) for which either the source or the destination IP address does not match with any entries in the router forwarding table. (iii) which either have a AS number in the range 64513 $\sim$ 65535 (valid public AS number ranges from 1 to 64512 [15]). We thereby eliminated 4% of the total flow records we captured. During our study, we also noticed that the timestamps of the *netflow* records are sometimes not consistent. This is usually due to brief time periods where the clock on a given linecard has not yet been slaved to the main router clock. We corrected the timestamps by slaving the linecard clock to the closest netflow records from interfaces on the main route processor. Additionally, we eliminated a few *netflow* records that have invalid timestamps. The final dataset consists of around 800 million flow records.

Table I provides summary statistics for the P2P data set obtained above. Among the three P2P systems, FastTrack is the most popular in terms of both the number of hosts participating in the P2P system and the average traffic volume (per day) that is transferred among hosts. We collected 110 million, 184 million, and 341 million *netflow* records for FastTrack in September, October, and December 2001, respectively. There were a total of 3.4 million unique IP addresses participating in the FastTrack system during a 6-day period in September with an average of 1 million unique IP addresses participating in the system each day. The average number of unique IP addresses participating in the FastTrack system per day grows to 1.5 million in October (50% growth) and 1.9 million in December (90% growth). In September, the average total data traffic is 773 GBytes/day and the average data traffic contributed by each individual IP address is 1.6 MBytes/day. While the average total data traffic grows rapidly to 1.15 TBytes/day (50% growth) in October and 1.78 TBytes/day (130% growth) in December, the average traffic volume contributed by an individual IP address remains in the range of 1.6$\sim$1.8 MBytes/day across the months. This indicates that the rapid growth of the P2P traffic is mainly caused by the increasing number of hosts participating in the system. Similar trends hold for Gnutella and DirectConnect.

Gnutella is the second most popular P2P system. Although the number of IP addresses participating in Gnutella and the total traffic volume transferred using Gnutella are smaller than that of FastTrack, the average traffic volume contributed by an individual IP address is similar ($\sim$ 2MBytes/day) to that of FastTrack across the three months.

Compared to FastTrack and Gnutella, DirectConnect is a more recent system and has a smaller user base. We collected 0.5 $\sim$ 0.7 million flow records during the data gathering week of each month, in which a total of 20 $\sim$ 30 thousand IP addresses participate. However, the average traffic volume contributed by an individual IP address ranges from 15.4 MBytes/day to 19.6 MBytes/day, which is much higher than the corresponding values for FastTrack and Gnutella. As we shall see later, the DirectConnect hosts tend to stay active longer than FastTrack and Gnutella hosts. They also have higher average bandwidths (both upstream and downstream) than the FastTrack and Gnutella hosts.

A. Host distribution

For each P2P system, we compute the number of unique (i) IPs, (ii) prefixes, and (iii) ASes observed every day across three months (Figure 2). The number of IP addresses participating in FastTrack each day ranges from 0.5 million to 2 million. The average daily figure increases slightly from September to December as the total number of IPs participating in FastTrack increases. This figure is 5 $\sim$ 7 times that of Gnutella and 150 $\sim$ 300 times that of DirectConnect. The number of unique prefixes participating in FastTrack ranges from 17 $\sim$ 26 thousand, and the number of unique ASes ranges from 4 $\sim$ 5.5 thousand.

To measure the spatial locality of the P2P hosts, we define the *density* of a prefix as the number of unique active IP addresses belonging to it. Similarly, the density of an AS is defined to be the number of unique prefixes belonging to it. From Figure 2, we observe that the FastTrack hosts are distributed

Date	Protocol	Number of records	Total number of unique IPs	Number of unique IPs per day	Total traffic volume (GBytes/day)	traffic volume per IP (MBytes/day)
9/10/2001 - 9/15/2001	Gnutella	37,853,281	718,464	197,445	211	2.2
	FastTrack	110,533,024	3,403,900	998,669	773	1.6
	DirectConnect	595,606	22,852	6,244	48	15.4
10/9/2001 - 10/13/2001	Gnutella	49,649,348	823,532	247,114	272	2.2
	FastTrack	184,113,038	4,450,149	1,485,370	1,153	1.6
	DirectConnect	566,740	23,211	7,193	56	15.6
12/10/2001 - 12/16/2001	Gnutella	69,578,723	887,520	236,954	242	2.0
	FastTrack	340,690,074	5,924,072	1,934,460	1,776	1.8
	DirectConnect	701,712	29,925	7,213	71	19.6

TABLE I
Netflow DATA SET OF P2P TRAFFIC OVER TCP.

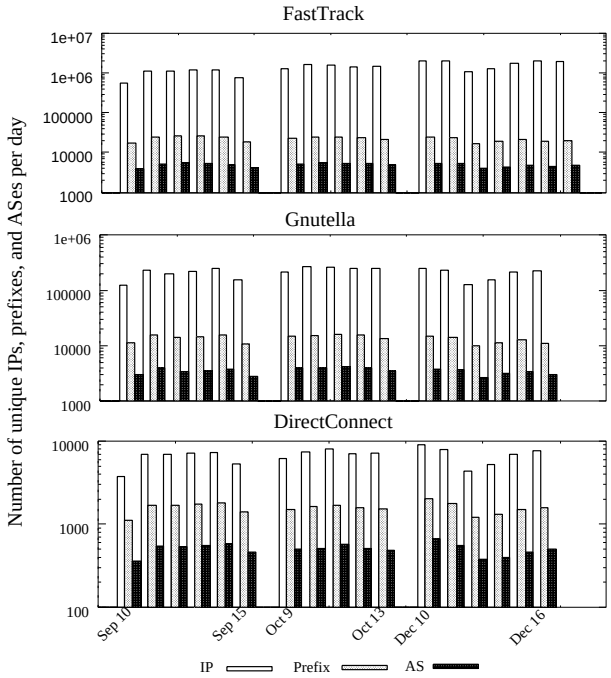


Fig. 2. Host density: the distribution of the hosts participating in three P2P systems per day (y-axis is in logscale).

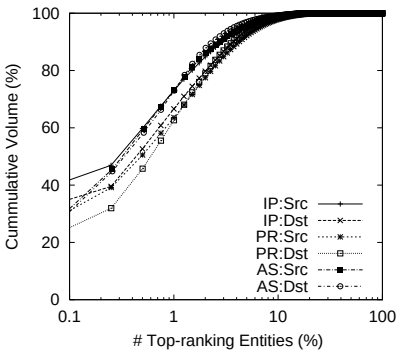


Fig. 3. The cumulative distribution of traffic volume associated with IP addresses ranked in decreasing order of volume, for September 14, 2001 (x-axis is in logscale). Aggregate traffic observed for for FastTrack on this day was 960 GB.

much more densely than the Gnutella and DirectConnect hosts. This is likely due to the different sizes of the three P2P systems. The average density of prefixes in FastTrack is 64, while that of Gnutella and DirectConnect are 16 and 4, respectively. The average AS density in FastTrack, Gnutella, and DirectConnect are similar (i.e., 3∼4). This implies that the FastTrack hosts have better potential to find nearby peers, and that most of the queries can be resolved locally (e.g., within a network prefix). One potential improvement to the Fast Track protocol is to take advantage of the hosts location.

B. Traffic volume distribution

Figures 3 plots the ranked CDF for the FastTrack network (Gnutella and DirectConnect show similar trends), for the aggregate upstream (denoted as "Src") and downstream (denoted as "Dst") traffic volumes at different network aggregation levels. The ranked CDF is obtained by first ordering the IPs (or prefixes or ASes) in order of decreasing volume (separate ranks for outgoing and incoming volumes), and plotting the cumulative volume for the ranked list.

We observe extreme skews in the distributions of upstream and downstream volumes at the three aggregation grains – a few heavy hitters account for much of the traffic. For instance, the top 0.1% of IPs, prefixes, and ASes transmit 33%, 27%, and 26% of the total traffic in FastTrack, respectively. The top 1% of the IPs, prefixes, and ASes transmit 73%, 64%, 73% of the total traffic, respectively. An individual IP address may transmit over 10GB data during a single day.

The skewed traffic patterns are observed across the three months. Figure 4 shows that distribution of the upstream traffic volume from each individual IP address for the three P2P networks. We present results for two days (one weekday and one weekend day) for each of the three months. We observe that the top $1-2\%$ of the IP addresses account for more than 50%, and the top 10% of the IPs account for more than 90%, of the total traffic volume. Similar patterns are observed for both weekday and weekend in the same month. We do observe (Figure 4) that the top 10% of the IPs account for a slightly smaller percentage of the total traffic volume in December than in September.

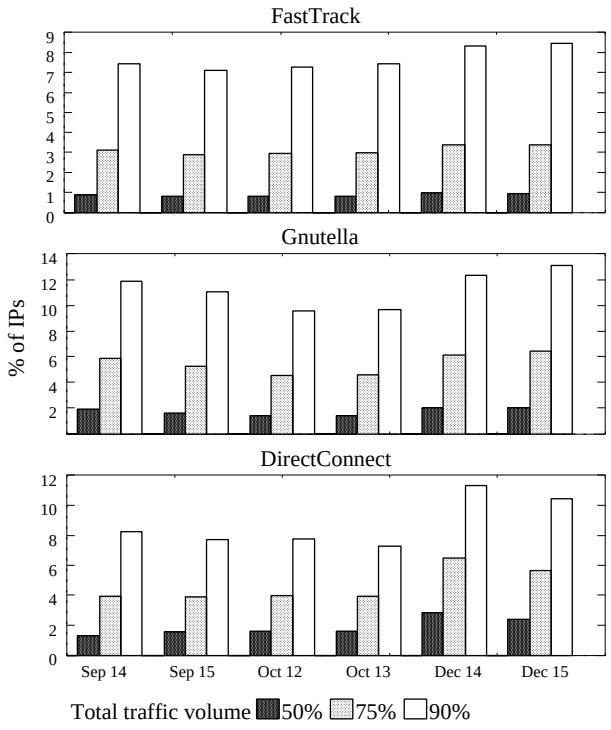


Fig. 4. The distribution of P2P upstream traffic volume across three months.

The above behavior is more reminiscent of a client-server environment where a few popular servers with popular content are responsible for originating most of the traffic. This might suggest that the P2P search protocol should first query the contents of the relatively small number of heavy hitters, so that queries for popular objects (which are likely to be most of the queries) will be served by searching only a small number of hosts. Only for relatively uncommon queries, it may be still necessary to search a large number of hosts.

Similar skewed behavior is observed for the downstream traffic, across all three P2P systems. In FastTrack, the top 100 IP addresses receive a substantial 57 GB (6% of the total traffic), while the top 0.1% of the IPs ($1,190$ IP addresses) together receive 245 GB (25% of the total traffic). Video file downloads could easily explain the high levels of incoming traffic for the heavy hitters. Hosts with different bandwidth connectivities accessing the system may also contribute to the high variability observed. The skewed distribution for FastTrack can also be attributed to the fact that all the query-response traffic is only circulated between a small subset of hosts – the SuperNodes. However, in Gnutella, all the hosts take part in the signaling, yet we still observe the extreme variation in downstream traffic volumes. The skewed traffic distribution (upstream and downstream) at the prefix, and AS level suggests that coarse-grained traffic management and policing mechanisms such as rate limiting and pricing targeted at the heavy hitter entities would be useful for network traffic engineering and provisioning purposes.

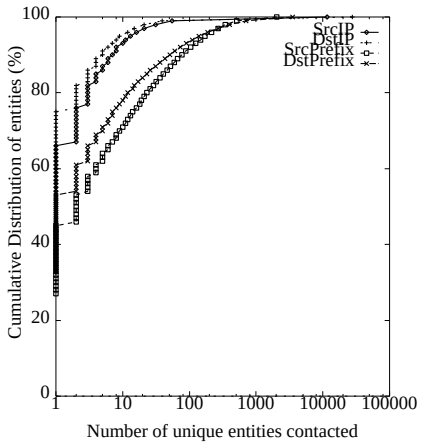


Fig. 5. The cumulative distribution of network connectivity at the IP and network prefix (PR) levels, for hosts participating in FastTrack on September 14, 2001.

C. Host connectivity

To study application-level connectivity between the hosts, we next consider the distribution of the total number of unique entities (IP, network prefix, AS) that a single entity communicates with. Figure 5 plots the CDF of the network connectivity at the IP and network prefix aggregation levels for FastTrack. The distribution at the AS aggregation level shows similar behavior as the prefix level distribution and is not shown in the figure. We find that about 48% of the individual IPs communicate with (send data to or receive data from) at most one IP, and 89% with at most 10 other IP addresses. This may be due to the fact that each FastTrack client connect to only one SuperNode at a time. Only the top 1% of the IP addresses communicate with more than 80 other IP addresses. The distribution is less skewed for the network prefix and AS level connectivity. 75% of the prefixes communicate with at least 2 prefixes, and the top 1% of the prefixes talk with at least $1,000$ prefixes. 80% of the ASes communicate with multiple ASes, and the top 1% of the ASes communicate with at least 476 other ASes.

Because few hosts have very high connectivity and most hosts have very small connectivity, the above data suggests that routing via the heavy-hitter (in terms of IP connectivity) nodes could make it possible to reach any host within a relatively small number of hops (IP, network prefix, or AS), i.e., the P2P networks have small diameters. The IP level statistics also suggests that such P2P networks could be highly vulnerable to failures of the tiny percentage of hosts with high degrees of connectivity. This is consistent with findings from recent studies based on Gnutella signalling traffic [3], [2]. We also find that the prefix and AS level connectivity distributions are less skewed than at the IP address level. This suggests that at these coarser grained levels of aggregation, a node is less vulnerable to disconnection from the P2P network as a significant percentage of prefixes (and ASes) communicate with more than one prefix (or AS).

V. P2P System Dynamics

Hosts joining and leaving can potentially make the P2P system very dynamic. In the following, we quantitatively measure the dynamics in terms of (i) how many hosts are active during

a certain period of time; (ii) how long a host stays in the P2P system; (iii) how active the hosts are during a certain period of time.

A. Traffic pattern over time

We first measure how many hosts are active in the P2P systems during a certain time period. This is of interest to an ISP, as it helps to understand the overall pattern of the P2P traffic across its network. We characterize the traffic pattern using discrete *time bins*. We vary the bin size from 15 minutes to 1 hour and compute the number of unique IP addresses, network prefixes, and ASes observed with each bin and the corresponding traffic volume transferred. Figure 6 shows the binning results for the FastTrack system. The results are similar for Gnutella and DirectConnect and are not reported here. Figure 6(a) shows the traffic volume (for hourly bins) transferred among FastTrack hosts across 24 hours on September 14, 2001 (GMT). We observe the "time of the day" effect on the traffic volume per hour. The traffic volume is heavy between evening and midnight and tapers down gradually early in the morning. This suggests many users are likely to join the P2P network after work and keep downloading data at night. We also observe two peaks 3 hours apart during the heavy traffic period. This is likely due to the time difference between east and west coasts of the continental US.

Figure 6(b) shows the "time of the day" effect for the number of IP addresses, network prefixes, and ASes that are either sending or receiving data. We notice that FastTrack hosts have little activity in the earlier morning but start to be more active at noon. However, the increasing number of active IP addresses in the late morning do not have major impact on the traffic volume that transferred among them. This seems to indicate that the hosts are mostly involved in signalling communications around this time, and that the heavier data downloads start to happen later in the day. Similar observation holds on the number of active network prefixes and ASes.

Figure 6(c) shows the number of active IP addresses (normalized by the size of the bin) for various sizes of time bins. We observe larger variance in the number of active IP addresses for smaller bins. However, the number does not decrease proportionally as the bin size decreases. This indicates that the P2P system is very dynamic with hosts joining and leaving the system very frequently. If the majority of the hosts stayed in the system for longer than 1 hour each time they join the system, we would see the normalized number of active IP addresses during each 1 hour bin is approximately 1/2 of that of the 30 minute bin and 1/4 of that of the 15 minute bin. If the hosts are very dynamic and only stay in the system for less than 1 minute each time they join the system for example, then we would expect that the normalized number of the active IP addresses during each 1 hour bin roughly equals to that of the 30 minute bin and the 15 minute bin. Figure 6(c) shows the normalized number of the active IP addresses in each 1 hour bin is much larger than half of that of the 30 minutes. This suggests that most IP addresses are active for less than 30 minutes each time they join the system. Similar observations on the results of 30 minute bin and 15 minute bin implies that most of the IP addresses are active for less than 15 minutes each time they join the system.

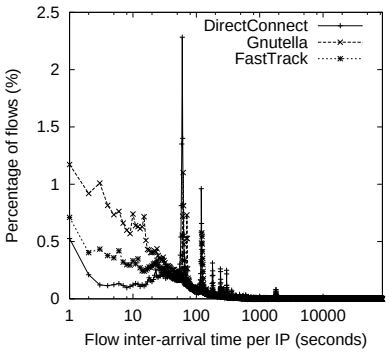


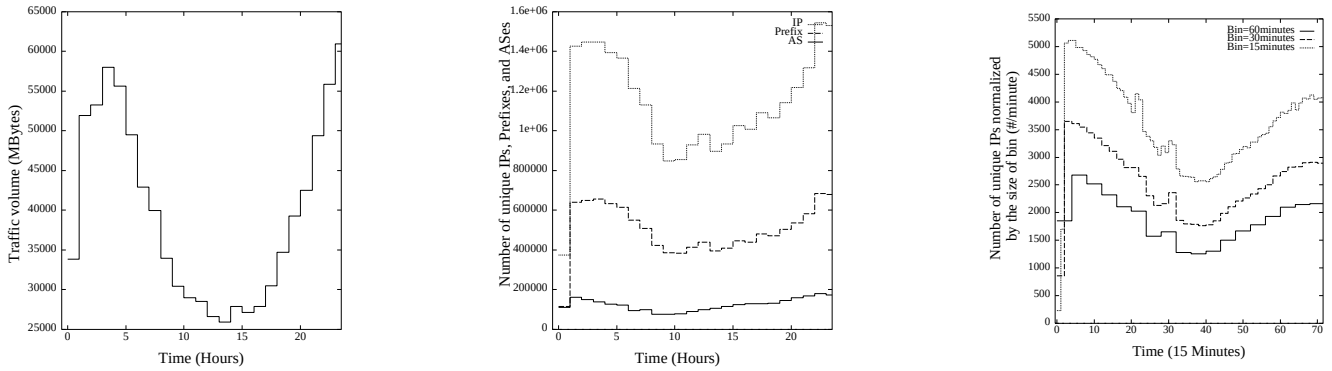
Fig. 7. Histogram of flow inter-arrival time for IP addresses.

B. Host connection duration and on-time

We next study how long an IP address stays in the P2P system. We examine the distribution of connection duration, on-time per day, and number of connections per day for P2P hosts. Recall (Section III-E) that the computation of these metrics involves a threshold δ. We chose the value of δ by examining the distribution of the flow inter-arrival time for IP addresses. The *flow inter-arrival time* of an IP address is the time interval between the termination of one flow and and the initiation of the next flow with the same IP as one end-point. Figure 7 shows that the flow inter-arrival time distributions for all three P2P systems exhibit similar trends. Note that there are a large number of very short (few seconds) flow inter-arrival times. Many of the short flow inter-arrival times (particularly the one second intervals) could be a result of programmed sequential downloading of multiple files. It may also be due to a single session being split into multiple flows by Cisco *netflow* – this can happen, e.g., for long sessions (see Section II). Beyond the few second range, there are some prominent spikes between 60 seconds and 500 seconds. These spikes could be due to the time spent by users in composing and submitting new queries to the P2P network. As part of our ongoing work, we are further investigating the distribution of flow inter-arrival times.

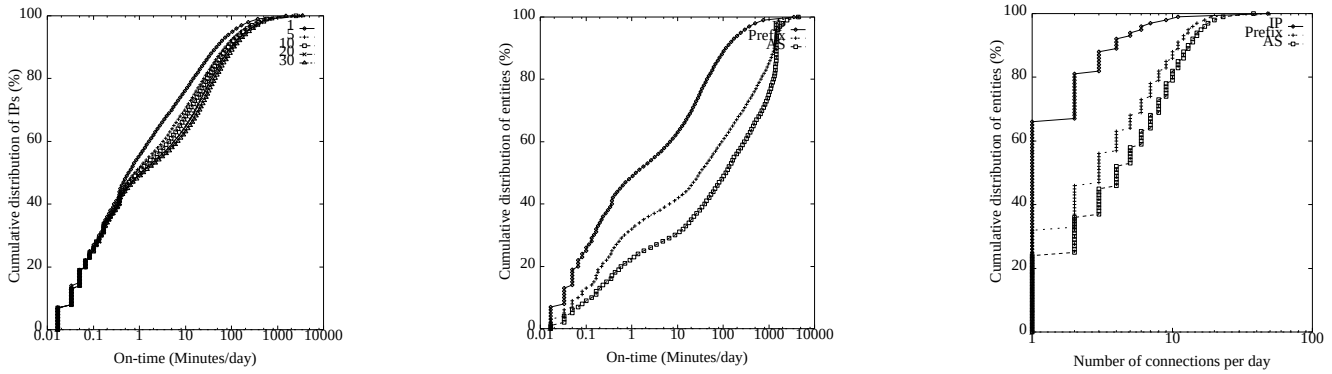
Based on our observation on the distribution of flow inter-arrival time, we vary the threshold δ from 1 minute to 30 minutes to evaluate the sensitivity of the choice of δ and identify suitable value of δ. Figure 8(a) shows the distribution of total on-time for an IP address per day. We observe that the results vary significantly for small values of δ in the $1-5$ minutes range. However, the results become progressively less sensitive for large values of δ, and there is very little difference between the curves for $\delta = 20$ or 30 minutes. We therefore use $\delta = 30$ minutes to compute the on-time, number of connections and average connection duration.

Figure 8(b) shows that cumulative distribution of the on-time of the FastTrack hosts on September 14, 2001. We observe that 60% of the IP addresses, 40% of network prefixes and $30\%$of the ASes stay in FastTrack for 10 minutes or less per day. The graphs show that the P2P system is much less transient at the prefix and AS aggregation levels. This suggests that inserting indexing/caching nodes locally (e.g., within a network cluster or AS) could be a promising way to reduce the effect of the dynamism in the P2P system. This is because the node joining and



(a) (b) (c)

Fig. 6. The distribution of number of IP addresses and traffic volume across hours in FastTrack on September 14, 2001 (GMT). (a) The traffic volume transferred in each bin. (b) The number of unique IP addresses, network prefixes, and ASes that are active in each bin. (c) The number of unique IP addresses that are active (normalized by the size of bin) in each bin of various sizes.



(a) (b) (c)

Fig. 8. The distribution of FastTrack hosts' connections on September 14, 2001. (a) Choice of δ : the cumulative distribution of the total on-time, for $\delta = 1, 5, 10, 20, 30$ minutes. (b) The cumulative distribution of the total on-time at IP, prefix and AS levels ($\delta = 30$ minutes), (c) The cumulative distribution of the number of connections at IP, prefix, and AS levels ($\delta = 30$ minutes). The x-axis is in logscale.

leaving only affect their local indexing/caching nodes without propogating the effect to the rest of the network. This is consistent with the findings from a previous study of the Gnutella system [1].

Figure 8(c) shows the cumulative distribution of the number of connections per day. The number of connections of a host measures how frequently it joins the P2P system. We observe that 65% of the IP addresses join FastTrack only once. The distribution of number of connections is less skewed at the network prefix and AS levels. This is consistent with our observations on the distribution of the connection duration. We found that most of the connections are very short. Over 20% of the connections last 1 minute or less. This observation holds at the IP, network prefix, and AS levels. The large number of short connections may be because the vast majority of the data transfer events are queries and responses.

We also compared the distributions of the on-time, the number of connections, and the average connection duration of hosts in the three P2P systems. Around 60% of the IPs keep active in FastTrack for no longer than 10 minutes each time they join the system. This justifies our speculation in the previous section that most of the IPs are active for less than 15 minutes in FastTrack. Hosts tend to stay longer in DirectConnect than in the other two P2P systems. However, the distribution of the number of connections of the hosts in all three P2P systems are similar. Unlike FastTrack and Gnutella hosts, DirectConnect hosts tend to stay in the P2P system longer each time they join the system. This may be one factor contributing to our earlier observation that individual DirectConnect hosts usually contributes more traffic than FastTrack or Gnutella hosts.

C. Mean bandwidth usage for hosts

Figure 9 shows the cumulative distribution of the mean upstream (denoted as "Src") and downstream bandwidth (denoted as "Dst") usage at the IP address level in FastTrack, Gnutella, and DirectConnect. In FastTrack (Figure 9(a)), for around 1/3 of the IP addresses, we observe mean downstream bandwidths of 56Kbits/second or less. They are probably dial-up Internet service users. Overall $2/3$ of the IP addresses exhibit downstream bandwidths in excess of 56 Kbps, suggesting that these correspond to users with broadband network connectivity. The average upstream bandwidth is usually smaller than the average downstream bandwidth. This may be partly due to the presence of nodes with asymetric bandwidth connectivity as is the case

for DSL and cable modem users. Another potential contributing factor behind this behavior could be that individual users have the ability (in many P2P systems) to rate-limit the upstream data transfers from their machines. Half of the IP addresses have average upstream bandwidth of 56Kbits/second or less. The distribution of the average bandwidth of Gnutella hosts is similar to that of FastTrack hosts and is not shown here. However, we observe higher bandwidths (both upstream and downstream) for the DirectConnect hosts in Figure 9(b). 20% of the IP addresses have mean downstream bandwidth of 56Kbits/second or less, while another 40% of the IP addresses have mean downstream bandwidth of 56 $\sim$256 Kbits/second. Correspondingly, 1/3 of the hosts have average upstream bandwidth of 56 Kbits/second. This is the second factor contributing to our earlier observation that individual DirectConnect hosts contributes much more traffic volume than FastTrack and Gnutella hosts.

VI. Traffic Characterization

In order to develop a model for P2P workload, it is important to understand the distribution of the individual metrics of interest, as well as the relationships between the different metrics. As a first step towards developing a workload model for P2P traffic, we explore two questions in this section. First, Zipf's law [16] is widely used to model skewed distributions because of its simple form and has been recently applied to research on Web caching [17] and Internet topology models [18]. Earlier work [19] suggests that the Zipf's law also applies to the signaling traffic for Gnutella. An interesting question is whether Zipf's law is suitable for modeling the overall P2P traffic including both the signaling and the data traffic. We explore this question first. Second, we explore the relationships between the different metrics of interest.

A. The power law

Zipf's law is best described with an example, such as words in a book. Let V be the vocabulary size, and let f_1 be the occurrence frequency of the most frequent vocabulary word, f_2 for the second most frequent, and so on. The *rank-frequency* plot is the plot of the occurrence frequency f_r of each vocabulary word versus its rank r, in log-log scales. The rank-frequency version of Zipf's law states that $f_r \propto 1/r$. This is typically referred to as the *Zipf's law* or the *Zipf distribution*. In log-log scales, the Zipf distribution gives a straight line with slope -1. The *generalized Zipf distribution* (or "Zipf-like" distribution) is defined as $f_r \propto 1/r^{\theta}$, where the slope $-\theta$ in log-log scales can be different than -1. The generalized Zipf distribution is also referred to as the "power law".

We studied four metrics: host connectivity, traffic volume, on-time, and average bandwidth of the hosts. Figure 10(a) shows the rank-frequency plot of the number of unique IP addresses that an IP address contacted. We observe that the distribution of the number of unique IP addresses is heavy-tailed. However, the distribution is not a straight line in log-log scales as we might expect from Zipf (or generalized-Zipf) distribution. As we observe a clear tilting in the rank-frequency plot, it is not clear that Zipf's law is the suitable model for the host connectivity. Similar observation holds on the distribution of the traffic volume contributed by an IP address, on-time of an IP address, and average upstream and downstream bandwidths except that they are even more skewed.

Metric pair (x, y)	Correlation	
	(x, y)	(log x, log y)
Volume, On-time	0.180	0.410
Volume, # Unique IPs	0.167	0.219
Volume, BW	0.138	0.269
# Unique IPs, On-time	0.430	0.617
# Unique IPs, BW	-0.086	-0.519
BW, On-time	-0.344	-0.754

TABLE II
Correlations of traffic volume, on-time, mean bandwidth usage (BW), number of unique IP addresses that each host connects to, all measured for the upstream direction.

We next focus on the top 10% of the hosts that source about 90% of the total traffic. Figure 10(b) shows the rank-frequency plot of the number of unique IP addresses that an IP address contacted for the top 10% of the IP addresses. We observe that the distribution of the number of unique IP addresses that each IP address contacts is heavy-tailed. However, it is not a straight line in log-log scales, either. Similar observation holds on the distribution of the traffic volume (Figures 10(c)), on-time (Figures 10(d)), and average upstream and downstream bandwidth (not shown here) of an IP address for the top 10% of the IP addresses. The above observations also hold for Gnutella and DirectConnect. In summary, we conclude that (in general) both the overall P2P traffic and the traffic from the top 10% heavy hitters seem to be heavy tailed for the data we examined, but they might not be precisely Zipf's distributions. Further analysis is required for developing accurate models for these distributions. We are addressing this as part of ongoing work.

B. Relationships between measures

We next explore the relationships between the different metrics of interest. We consider FastTrack and focus on the top 1% of the IP addresses that source about 73% of the total traffic volume for September 14 and compute the correlation coefficients for six pairs of metrics (see Table II). Note that the values for a metric can span a large range and be unevenly distributed over the range. Therefore, in addition to the correlation coefficient for the original data, we also consider the correlation coefficient for the logarithmic transformation of the data values, to limit the impact of outliers. The coefficient values thus obtained suggest weak positive correlations between (i) traffic volume and on-time, (ii) traffic volume sourced by an IP and the number of unique IPs that it connects to, and (iii)volume and bandwidth, moderate positive correlation between the number of unique IPs and on-time, and moderate negative correlation between (i) mean bandwidth usage and number of unique IPs, and (ii) mean bandwidth usage and on-time. The last of the above correlations can be explained by recalling the relationship between volume, bandwidth and on-time. Overall, this correlation data does not indicate towards the existence of a strong linear relationship for any of the metric pairs examined.

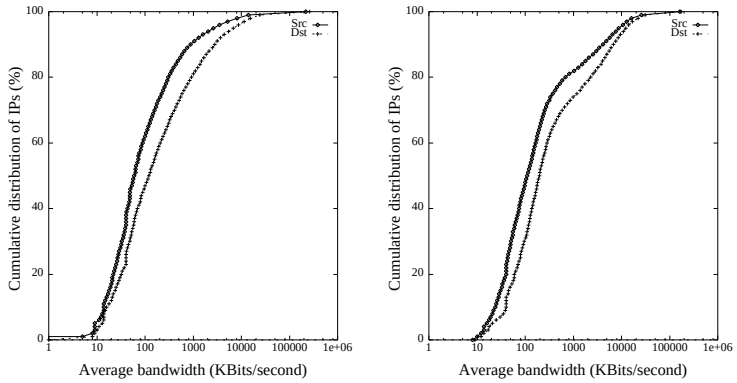


Fig. 9. The cumulative distribution of the mean upstream and downstream bandwidth usage of hosts participating in FastTrack, and DirectConnect on September 14, 2001 (x-axis is in logscale). (a) FastTrack, (b) DirectConnect.

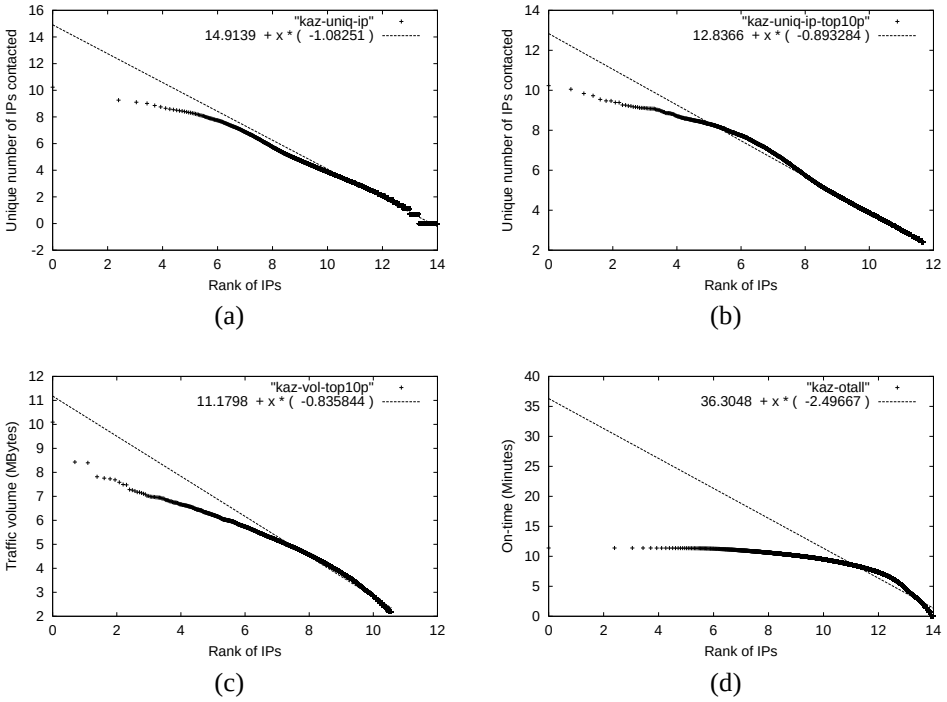


Fig. 10. The rank-frequency plots of the P2P metrics: (a) overall host connectivity; (b) host connectivity the top 10% IP addresses; (c) traffic volume of the top 10% Ip addresses; (d) On-time the top 10% IP addresses (both x-axis and y-axis are labeled in logscale).

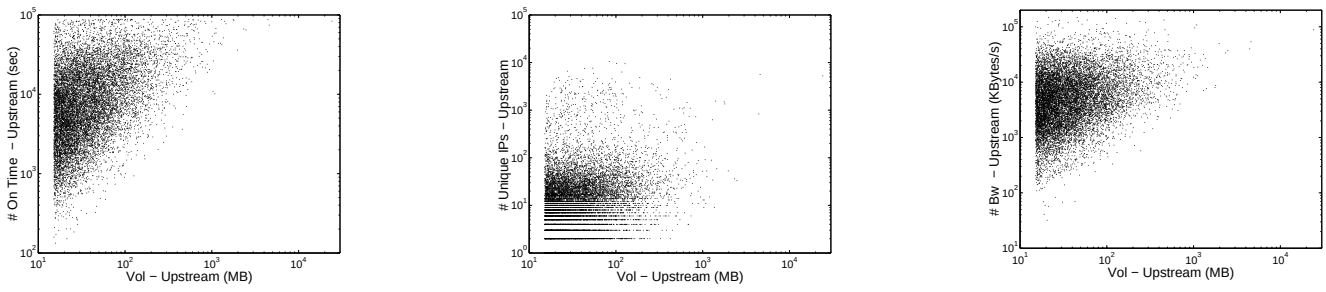
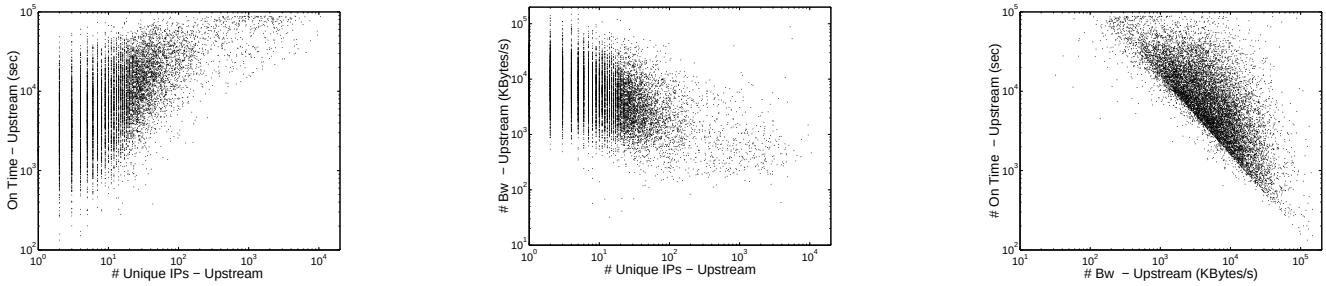


Fig. 11. FastTrack data set for September 14, 2001 – top 1% IP addresses ranked by volume of data sent out. Scatter plots (log-log scale): (a) upstream volume vs. upstream on-time, (b) upstream volume vs. number of unique upstream IP addresses that an IP address connects to, (c) upstream volume vs. average upstream bandwidth of an IP address.



(a) (b) (c)

Fig. 12. FastTrack data set for September 14, 2001 – top 1% IP addresses ranked by volume of data sent out. Scatter plots (log-log scale): (a) number of unique upstream IP addresses that a host connects to vs. total upstream on-time of the IP address, (b) number of unique upstream IP addresses vs. average upstream bandwidth, (c) average upstream bandwidth vs. total upstream on-time.

To provide more insight into the nature of the pairwise relationships, we next consider the corresponding scatter plots (Figures. 11-12). Figure 11(a) indicates that the top-ranking volume heavy hitters are likely to have long on-times, and that IP addresses with short on-times are likely to contribute small traffic volumes. However, IP addresses with very long on-times span the range of traffic volumes. Also, IP addresses with small upstream volumes are distributed over the range of upstream on-time values. Long on-time coupled with small traffic volume would be consistent with long-lived Supernodes handling query communications, if the IP connectivity were high. Figure 11(b) shows that an IP address communicating with large number of other IP addresses can transmit a small amount of traffic. Supernodes are likely to exhibit such behavior. The figure also shows that an IP address communicating with a handful of IP addresses can source significant traffic – this would be consistent with actual file transfers. Figure 11(c) shows that the top-ranking volume heavy hitters are likely to have large bandwidths, and that IP addresses with very small bandwidths are likely to contribute small traffic volumes. However, IP addresses with very long bandwidths are distributed across the range of traffic volumes. Similarly, IP addresses sourcing small traffic volumes are distributed over the range of bandwidth values.

Figure 12(a) shows that the IP addresses that have large IP connectivity counts tend to have very long on-times, and that IP addresses with short on-times are likely to communicate with a small number of IPs. However, IP addresses with very long on-times are distributed across a range of IP connectivity counts. Small IP connectivity and long on-times would be consistent with hosts transferring large data files. Figure 12(b) shows that there are IP addresses with high upstream bandwidths that have low IP connectivity counts. IP addresses that send traffic to a large number of IPs tend to span a range of upstream bandwidths, but are not among the IP addresses with the highest bandwidths, suggesting that these might be Supernodes handling query communications. Figure 12(c) shows that IP addresses with low upstream bandwidths have very long on-times. This may be due to either the long time taken to download large files, or because the corresponding node is a Supernode with a large IP connectivity. The figure also shows that IP addresses with very long on-times tend to span the range of upstream bandwidths. Similarly, IP addresses with high upstream bandwidths span a range of upstream on-times. The scatter plots for the top 10% of the IP addresses are similar and we don't show them here.

VII. P2P TRAFFIC VS WEB TRAFFIC

As part of our ongoing work, we are comparing P2P and Web traffic. We present some initial results here. We examine flow level traffic from a large ISP's peering links for March 2002. The Web traffic is extracted by considering TCP flows which use port numbers 80 and 8080 as the source or destination ports. To reduce the processing overhead, the stratified sampling technique [20] is applied. Considering per-prefix daily traffic volumes, we find that over 97% of the prefixes contributing to P2P traffic also contribute Web traffic. In addition, we observe that the heavy hitter P2P prefixes all tend to be heavy hitters in terms of Web traffic.

We say that the traffic from a prefix is *stable* if the percentages of the daily aggregated traffic volume from the prefix do not change over days. There are two factors that contribute to traffic instability. First, the traffic from a prefix may fluctuate over time. Second, there might be a trend of growing traffic volume over time. Either case would be of great interest to service providers to be able to capture and predict. In our initial analysis, we do not separate the affect of two type of changes in traffic volume. Instead, we character them as a whole and analyze traffic volume changes from day to day.

To characterize the traffic stability of a prefix, we compute the range of the traffic volume changes for each prefix over 31 days and normalize it by the mean daily traffic volume from the prefix. We examine the top 0.01%, 0.1%, 1%, and 10% heavy hitter prefixes which are responsible for 10%, 30%, 50%, and 90% of the corresponding monthly aggregated traffic volume. We show the results for the top 0.01% and 1% heavy hitter prefixes, other results are similar. Figure 13 compares the P2P traffic with the Web traffic and total traffic. Figure 13 (a) shows the results for the top 0.01% prefixes. For P2P traffic, each prefix has traffic volume changes that are within 10% of the mean daily traffic for that prefix. In the cases of Web and total traffic, less than 40% and 20% of the prefixes have traffic volume changes that are within 10% of the mean daily traffic for that prefix, respectively. The P2P traffic contributed by the top heavy hitter prefixes is more stable than that of the Web traffic

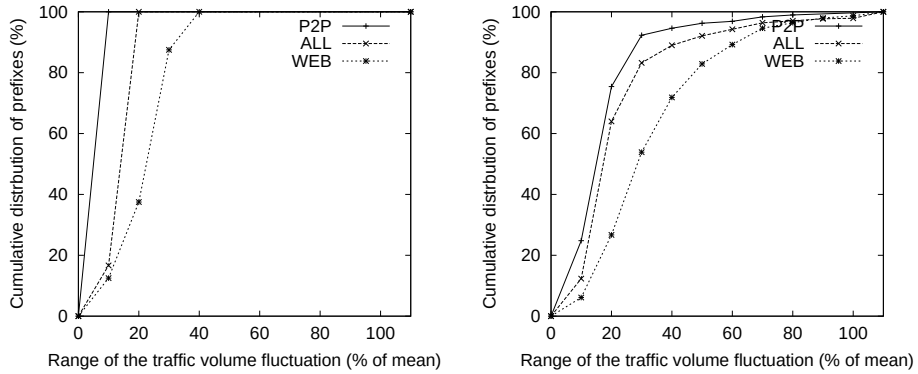


Fig. 13. The cumulative distribution of the traffic volume changes for top heavy hitter prefixes. (a) Top 0.01% prefixes; (b) Top 1% prefixes.

and the total traffic. This is somewhat counterintuitive, given the dynamism of P2P systems we observed at the IP addresses level, as discussed in previous sections. Some of the stability may be caused by long lasting data transfer flows that contribute bulk of the data. Figure 13 (b) shows the stability results for the top 1% prefixes. Note that a small fraction ($\leq 3\%$) of the prefixes have traffic volume changes $> 100\%$. The figure indicates that the P2P traffic contributed by the top heavy hitter prefixes is more stable than that of the Web traffic. The total traffic contributed by the top heavy hitter prefixes is also more stable than that of the Web traffic. This might be because the traffic from the top heavy hitter prefixes is dominated by P2P traffic which tends to be more stable. We are currently exploring if the above behavior holds across larger time periods. A stable traffic load at the prefix level makes it easier to model and predict workload for P2P traffic. The high volume and good stability of P2P traffic indicates that application-specific layer-3 traffic engineering may be a promising way to manage the P2P workload in an ISP's network.

VIII. Implications

In this paper, we presented a novel approach to measure and characterize the P2P traffic by analyzing flow-level data collected from multiple routers in a large ISP. We studied three popular P2P systems - FastTrack, Gnutella, and DirectConnect. We analyzed flow-level records across 3 months and observed that all three systems exhibit significant increases in both traffic volume and number of users, even across consecutive months. Our analysis covers both signaling traffic and actual data traffic. This complements previous work which only considered signaling traffic for Gnutella. The following are our key findings.

The traffic volume of individual hosts is extremely variable at IP, prefix and AS levels. The traffic volume generated by individual hosts is extremely variable - less than 10% of the IP addresses contribute around 99% of the total traffic volume. Individual heavy-hitter hosts can generate significant traffic volumes. Connectivity between different hosts is highly skewed - a very large fraction of IPs communicate with less than 10 other IPs, and a very tiny fraction communicates with a large number of hosts. The IP level statistics suggests that such P2P networks could be highly vulnerable to failures of the tiny percentage of hosts with high degrees of connectivity. Though host connectivity, traffic volume, host on-time, average bandwidth are highly skewed and exhibit heavy tails, they cannot be well modeled by the Zipf's distribution. The skewed traffic distribution (upstream and downstream) at the prefix, and AS level suggests that coarse-grained traffic management and policing mechanisms such as rate limiting and pricing targeted at the heavy hitter entities would be useful for network traffic engineering and provisioning.

All three P2P systems exhibit a high level of system dynamics- only a small fraction of hosts are persistent over long time periods. This behavior makes it a challenging proposition to realize a large P2P system based on a well-defined structured overlay architecture [21], [22], [23]. However, we also find that the P2P systems exhibit much more stability and persistence at the prefix and AS aggregation levels. Thus, inserting local indexing/caching nodes may help to reduce the effect of the dynamism in the system. We also find that the fraction of P2P traffic contributed by each prefix remains relatively unchanged and much more stable than the corresponding distribution for either Web traffic or even for overall traffic, over time periods of several days. This is somewhat counter-intuitive given the conventional notion that P2P architectures are very dynamic. This is good news for ISPs, as the high volume and good stability properties of P2P traffic at the coarser aggregations indicate that application-specific layer-3 traffic engineering may be a promising way to manage the P2P workload in an ISP's network.

IX. Conclusion

We presented a novel approach to measure and characterize the P2P traffic by analyzing flow-level data collected from multiple routers in a large ISP. We presented analysis of three popular P2P systems - FastTrack, Gnutella, and DirectConnect, across 3 months. As part of ongoing work, we are developing practical workload models for P2P traffic that can be used to evaluate traffic engineering and provisioning policies for P2P systems. We are also in the process of enhancing our passive measurement approach with selective active probing techniques.

X. Acknowledgments

We thank the many people whose efforts helped make this study possible, including Matt Grossglauser and Jennifer Rexford for their many helpful comments on an earlier version of the paper, and Matt Roughan for discussions on appropriate statistical methods. This work also benefited from the expert insight of Carsten Lund and Fred True in handling netflow data. Finally we thank the anonymous reviews and Steve Gribble, whose suggestions benefited the final version of the paper.

References

[1] B. Krishnamurthy, J. Wang, and Y. Xie, "Early Measurements of a Cluster-based Architecture for P2P Systems," in *Proceedings of ACM Sigcomm Internet Measurement Workshop*, November 2001.

[2] M. Ripeanu and I. Foster, "Mapping the Gnutella Network," in *IEEE Internet Computing*, January 2002.

[3] S. Saroiu, P. K. Gummadi, and S. D. Gribble, "A measurement study of peer-to-peer file sharing systems," in *Proceedings of Multimedia Computing and Networking (MMCN)*, January 2002.

[4] E. Adar and B. Huberman, "Free Riding on Gnutella," *First Monday*, vol. 5, no. 10, Oct 2000.

[5] "KaZaA," http://www.kazaa.com.

[6] "Gnutella hosts," http://www.gnutellahosts.com.

[7] "Direct Connect," http://www.neo-modus.com.

[8] "Bearshare," http://www.bearshare.com.

[9] "Limewire," http://www.limewire.com.

[10] "Grokster," http://www.grokster.com.

[11] "Morpheus," http://www.musiccity.com.

[12] K. Truelove and A. Chasin, "Morpheus out of the underworld," in *The O'Reilly Network*, July 2001.

[13] A. Singla and C. Rohrs, "Ultrapeers: Another Step Towards Gnutella Scalability," http://RFC-Gnutella.sourceforge.net/Proposals/Ultrapeer, Dec 2001.

[14] "White paper-netflow services and applications," http://www.cisco.com/warp/public/cc/pd/iosw/ioft/neflct/tech/napps_wp.htm.

[15] J. Hawkinson and T. Bates, "RFC 1930: Guidelines for creation, selection, and registration of an autonomous system (AS)," http://ftp.apnic.net/ietf/rfc/rfc1000/rfc1930.txt, March 1996.

[16] L. Adamic, "Zipf, power-laws and pareto - a ranking toturial," http://www.parc.xerox.com/istl/groups/iea/papers/ ranking/ranking.html.

[17] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications," in *Proceedings of IEEE Infocom*, March 1999.

[18] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the Internet topology," in *Proceedings of ACM Sigcomm*, August/September 1999, pp. 251–262.

[19] M. Jovanovic, F. S. Annexstein, and K. A. Berman, "Scalability Issues in Large Peer-to-Peer Networks - A Case Study of Gnutella," in *Technical Report, University of Cincinnati*, 2001.

[20] N. Duffield, C. Lund, and M. Thorup, "Charging From Sampled Network Usage," in *Proceedings of ACM Sigcomm Internet Measurement Workshop*, November 2001.

[21] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network," in *Proceedings of ACM Sigcomm*, August 2001.

[22] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for Internet applications," in *Proceedings of ACM Sigcomm*, August 2001.

[23] Y. Zhao, John D. Kubiatowicz, and Anthony Joseph, "Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing," Tech. Rep. UCB//CSD-01-1141, U. C. Berkeley, April 2000.