CSF641 – P2P Computing

點對點計算

A Measurement Study of Peer-to-Peer File Sharing Systems

Stefan Saroiu, P. Krishna Gummadi, and Steven D. Gribble
Department of Computer Science and Engineering
University of Washington
SPIE/ACM Conf. on Multimedia Computing and Networking (MMCN) 2002

Abstract

- Contribution: performing a detailed measurement study of the two popular P2P file sharing systems, namely Napster and Gnutella
- Uncovered characterization:
  - the bottleneck bandwidths between these hosts and the Internet at large
  - IP-level latencies to send packets to these hosts
  - how often hosts connect and disconnect from the system
  - how many files hosts share and download
  - the degree of cooperation between the hosts, and
  - several correlations between these characteristics
- Show that there is **significant heterogeneity** and **lack of cooperation** across peers participating in these systems

Outline

1. Introduction
2. Methodology
   - 2.1 The Napster and Gnutella Architectures
   - 2.2 Crawling the Peer-to-Peer Systems
   - 2.3 Directly Measured Peer Characteristics
     - 2.3.1 Latency Measurements
     - 2.3.2 Lifetime Measurements
     - 2.3.3 Bottleneck Bandwidth Measurements
     - 2.3.4 A Summary of the Active Measurements
3. Measurement Results
   - 3.1 How many peers fit the high-bandwidth, low-latency profile of a server?
   - 3.2 How many peers fit the high-availability profile of a server?
   - 3.3 How many peers fit the no-share, always-downloading profile of a client?
   - 3.4 The Nature of Shared Files
   - 3.5 How much are peers willing to cooperate in a P2P file-sharing system?
   - 3.6 Resilience of the Gnutella Overlay in the Face of Attacks
4. Recommendations to P2P System Designers
5. Conclusions

1. Introduction

- Challenges of P2P systems
  - how to organizing peers into a cooperative, global index so that all content can be quickly and efficiently located by any peer in the system
  - the system must take into account the suitability of a given peer for a specific task before explicitly or implicitly delegating that task to the peer
- Motivation: little work has been done to evaluate such considerations due to lack of information about the characteristics of P2P hosts
- This measurements consist of detailed traces over long periods of time
  - 4 days for Napster and 8 days for Gnutella

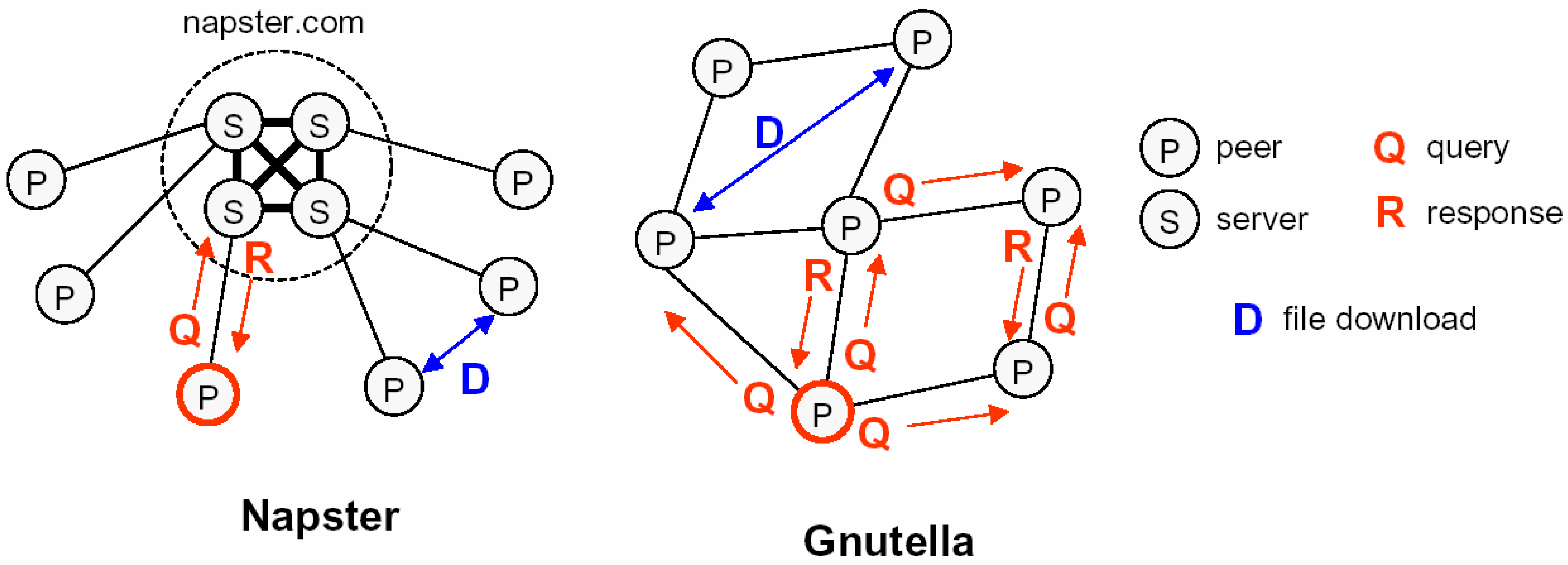
Two Main Lessons Learned

1. A significant amount of heterogeneity in both Gnutella and Napster
   – Bandwidth, latency, availability, and the degree of sharing vary between three and five orders of magnitude across the peers in the system
   – This implies that any similar peer-to-peer system must be very careful about delegating responsibilities across peers
2. Peers tend to deliberately misreport information if there is an incentive to do so
   – Because effective delegation of responsibility depends on accurate information, this implies that future systems must have built-in incentives for peers to tell the truth, or systems must be able to directly measure or verify reported information

2. Methodology

- Two steps
  1. Periodically crawled each system to gather instantaneous snapshots of large subsets of the systems’ user population
     - including: IP address and port number
     - some information about the users as reported by their software
  2. Actively probed the users in the snapshot over a period of several days to directly measure various properties about them

The Napster and Gnutella Architectures



- Gnutella's ping/pong messages: similar to query/response messages
  - peers occasionally forge new neighbor connections with other peers discovered through the ping/pong mechanism
  - controlled by time-to-live (TTL)

The Napster Crawler

- difficulty: no direct access to indexes
  - issuing queries for files with the names of popular song artists and keeping a list of peers referenced in the responses
- difficulty: cluster of approximately 160 servers
  (local users on the same server first, remote users later)
  - establishing a large number of connections to a single server and issuing many queries in parallel
    - Each crawl typically captured between 40-60% of the local peers on the crawled server
    - These peers contributed between 80-95% of the total files reported to the servers
  - gather the following metadata:
    1. the bandwidth of the peer’s connection reported by the peer
    2. the number of files currently being shared by the peer
    3. the current numbers of uploads and downloads in progress
    4. the names and sizes of all the files being shared
    5. the IP address of the peer

The Gnutella Crawler

- Method
  1. the crawler connects to several well-known, popular peers
  2. begins iterative process of sending ping messages with large TTLs to known peers
     - iterating for 2 minutes and waiting for 2 minutes
     - no specific names for queries as Napster
  3. adds newly discovered peers to its list of known peers based on the contents of received pong messages
     - typically gather 8,000 – 10,000 unique peers
     - corresponds to at least 25% to 50% of the total population (according to Clip2's report)
  4. metadata similar to Napster are collected
     - additional: the number and total size of files being shared

Number of Gnutella hosts captured

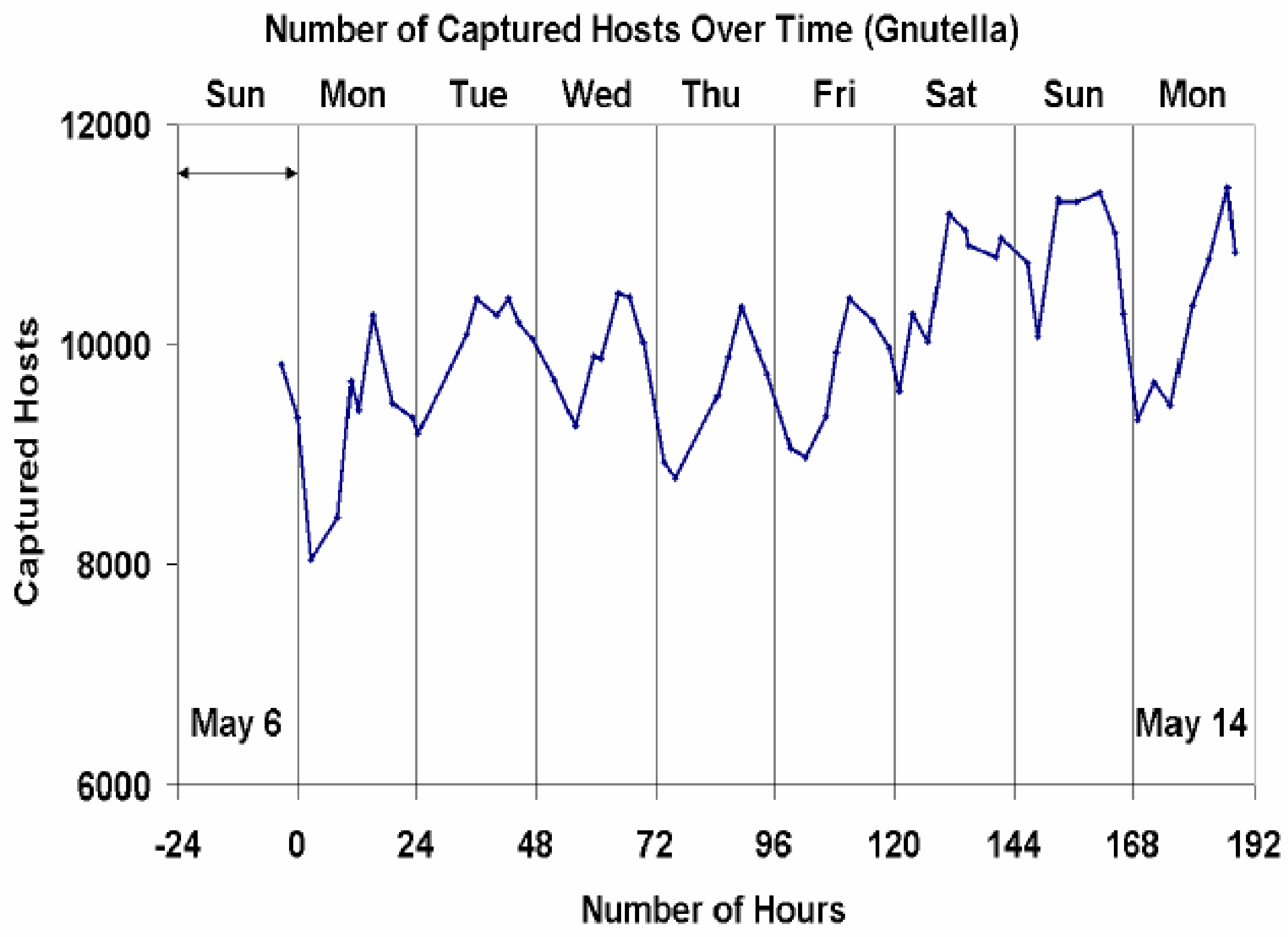


Figure 2: Number of Gnutella hosts captured by our crawler over time

Crawler Statistics

- Both the crawlers were written in Java using the IBM Java 1.18 JRE on Linux 2.2.16
  - on a small number of P-III 700 MHz computers with 2GB RAM and 4 40GB SCSI disks
- Napster trace: 4 days of activity
  - Sunday May 6th 2001 through Wed May 9th, 2001
  - 509,538 Napster peers on 546,401 unique IP addresses
- Gnutella trace: 8 days
  - Sunday May 6th, 2001 through Monday May 14th, 2001
  - 1,239,487 Gnutella peers on 1,180,205 unique IP addresses

Directly Measured Peer Characteristics

- Latency Measurements
  - measuring the RTT to the probed peers with a 40-byte TCP packet exchanged
- Lifetime Measurements (Sting platform)
  - send a TCP SYN-packet and then wait up to 20 seconds
    - May receive (1) none (2) RST (3) SYN/ACK
  - 3 possible states
    - offline: not connected or not responding to TCP SYN’s because it is behind a firewall or NAT proxy
    - inactive: responding to TCP SYN’s and TCP RST’s
    - active: accepting incoming TCP connections
- Bottleneck Bandwidth Measurements (SProbe tool)
  - difficulties
    - fluctuating over short periods of time
    - determined by the loss rate of an open connection

A Summary of the Active Measurements

- 17,125 Gnutella peers over a period of 60 hours
  - Determined its status once every 7 minutes
  - bottleneck bandwidths and latencies to a random set of 595,974 unique peers
    - successful downstream (active/inactive): 223,552 of these peers
    - successful upstream bandwidth: 16,252 of the peers
    - measured latency: 339,502 peers
- 7,000 Napster peers over a period of 25 hours
  - Determined its status every 2 minutes
    - downstream: 4,079 peers
    - upstream: 2,049 peers

Measurement Results

3.1 How many peers fit the high-bandwidth, low-latency profile of a server?
- Downstream and upstream measured bottleneck link bandwidths
- Reported bandwidths for Napster peers
- Measured latencies for Gnutella peers

3.2 How many peers fit the high-availability profile of a server?

3.3 How many peers fit the no-share, always-downloading profile of a client?
- Number of shared files in Napster and Gnutella
- Number of downloads and uploads in Napster
- Correlation between the number of downloads, uploads, and shared files

3.4 The Nature of Shared Files

3.5 How much are peers willing to cooperate in a P2P file-sharing system?

3.6 Resilience of the Gnutella Overlay in the Face of Attacks

3.1.1 Downstream and upstream measured bottleneck link bandwidths

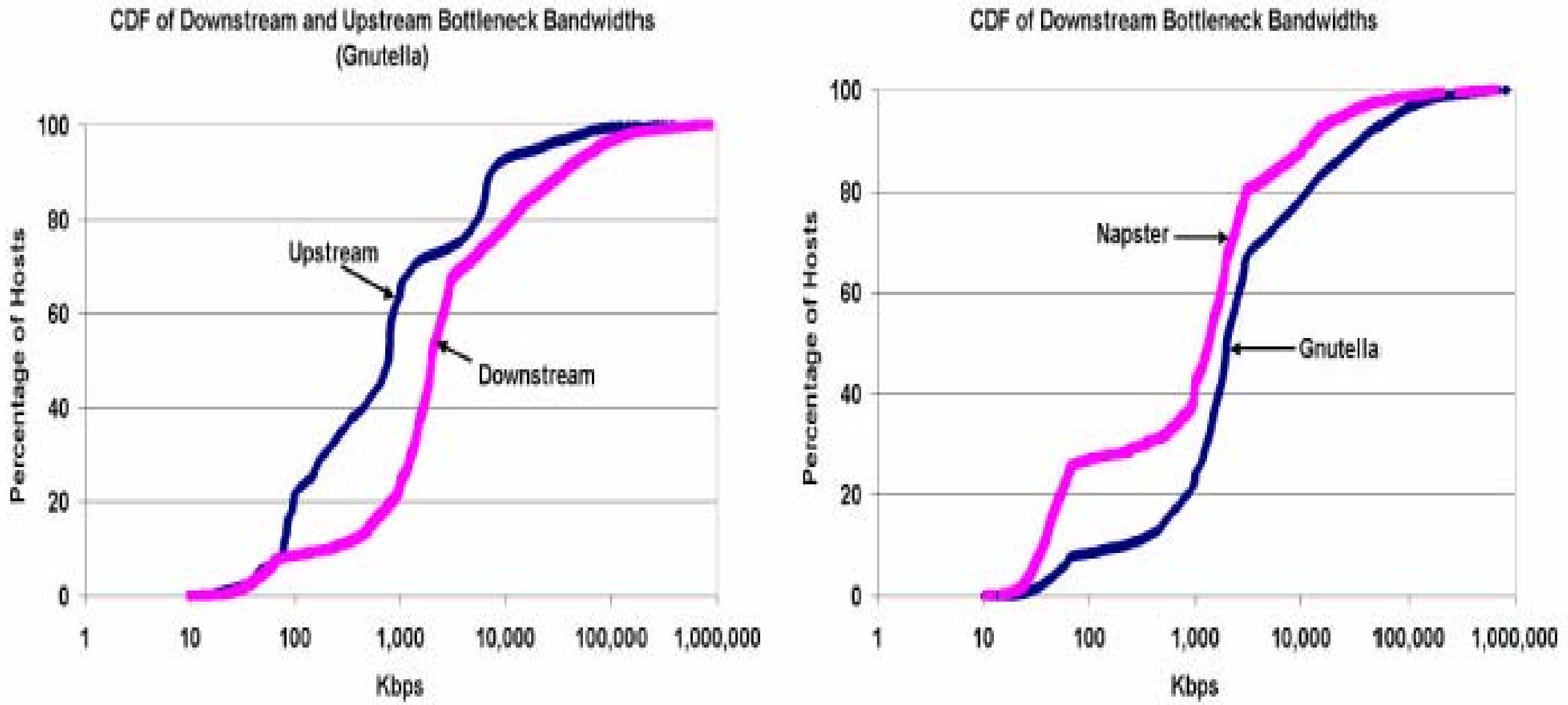


Figure 3. Left: CDFs of upstream and downstream bottleneck bandwidths for Gnutella peers; Right: CDFs of downstream bottleneck bandwidths for Napster and Gnutella peers.

- Left: The asymmetric links such as ADSL will affect the upstream and downstream
- Right: Gnutella protocol is too high of a burden on low bandwidth connections, discouraging them from participating (or more popular to technique-savvy users)

3.1.2 Reported bandwidths for Napster peers

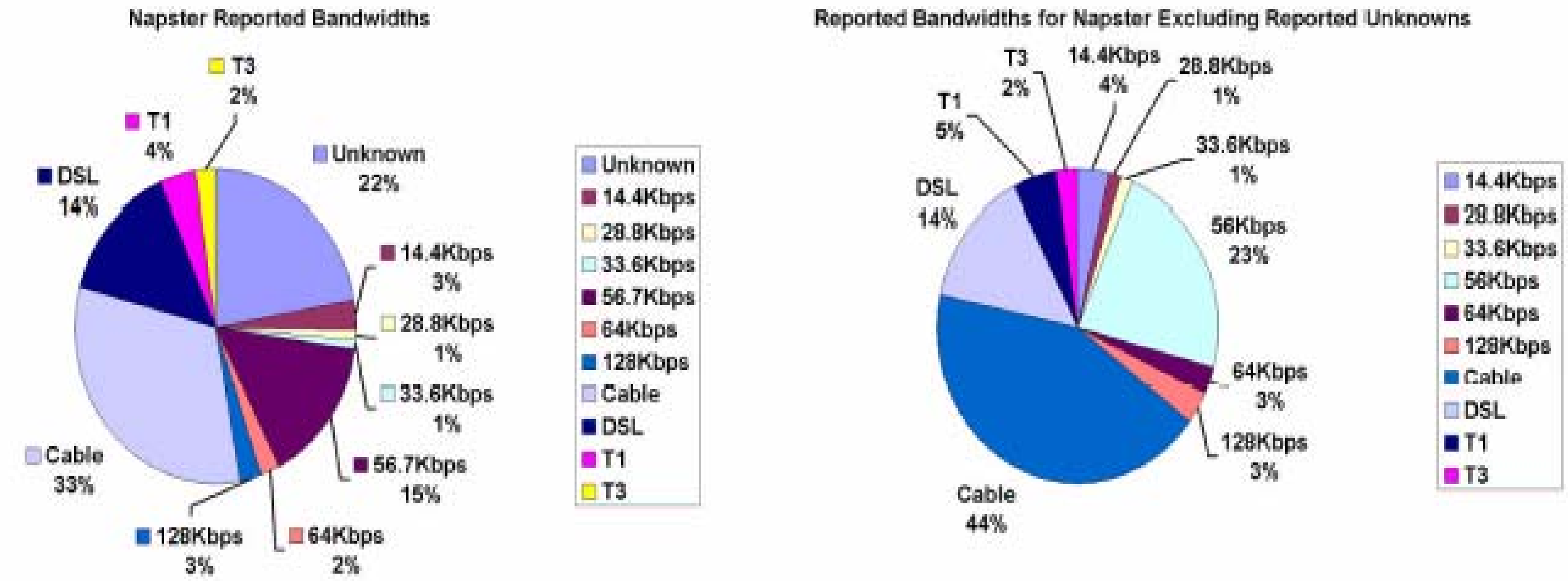


Figure 4. Left: Reported bandwidths For Napster peers; Right: Reported bandwidths for Napster peers, excluding peers that reported "unknown".

- High percent of "Unknown"
- Users have an incentive to misreport their connection speeds
- Figure 4 confirms that the most popular forms of Internet access for Napster and Gnutella peers are cable modems and DSLs (1Mbps and 3.5Mbps)

3.1.3 Measured latencies for Gnutella peers

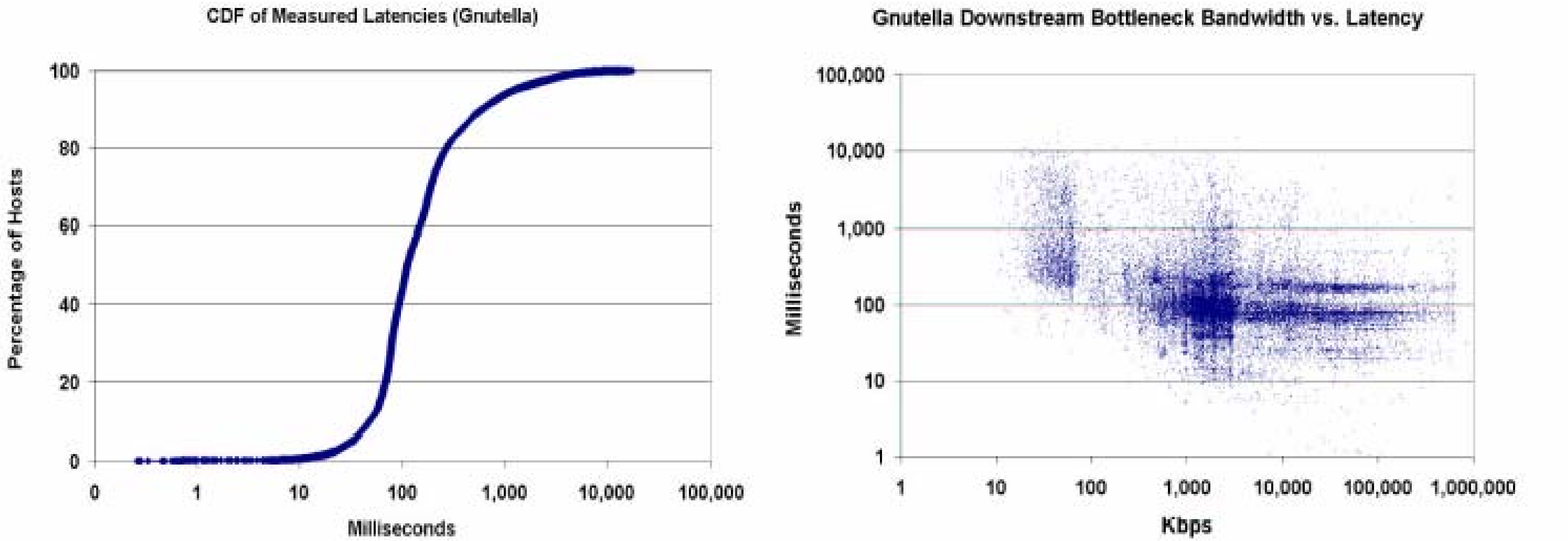


Figure 5. Left: Measured latencies to Gnutella peers; Right: Correlation between Gnutella peers' downstream bottleneck bandwidth and latency.

•Left: From this, we can deduce that in a peer-to-peer system where peers' connections are forged in an unstructured, ad-hoc way, a substantial Fraction of the connections will suffer from high-latency.

•Right: Two clusters, the set of modems and the set of broadband connections.

3.2 How many peers fit the high-availability profile of a server?

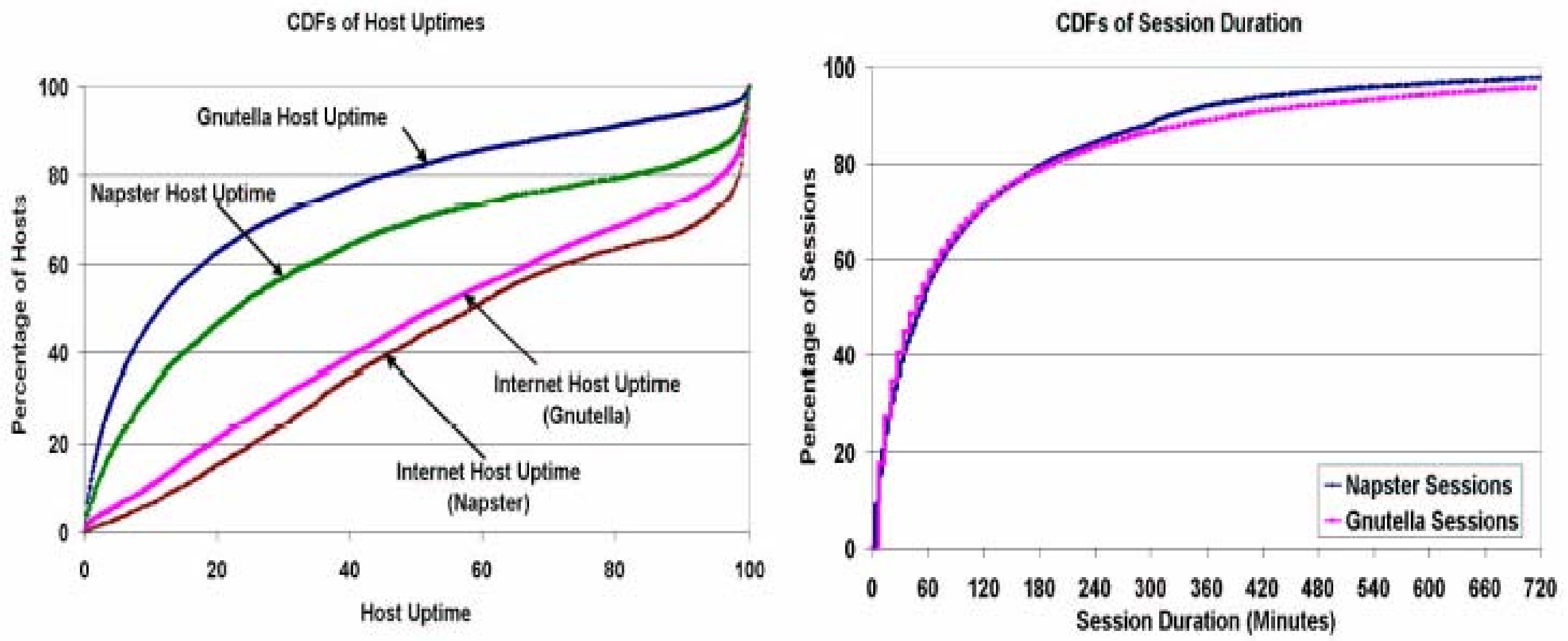


Figure 6. Left: IP-level uptime of peers ("Internet Host Uptime"), and application-level uptime of peers ("Gnutella/Napster Host Uptime") in both Napster and Gnutella, as measured by the percentage of time the peers are reachable; Right: The distribution of Napster/Gnutella session durations.

- Napster's software has several features (such as built-in chat client) that cause users to run it for longer periods of time

3.3.1 Number of shared files in Napster and Gnutella

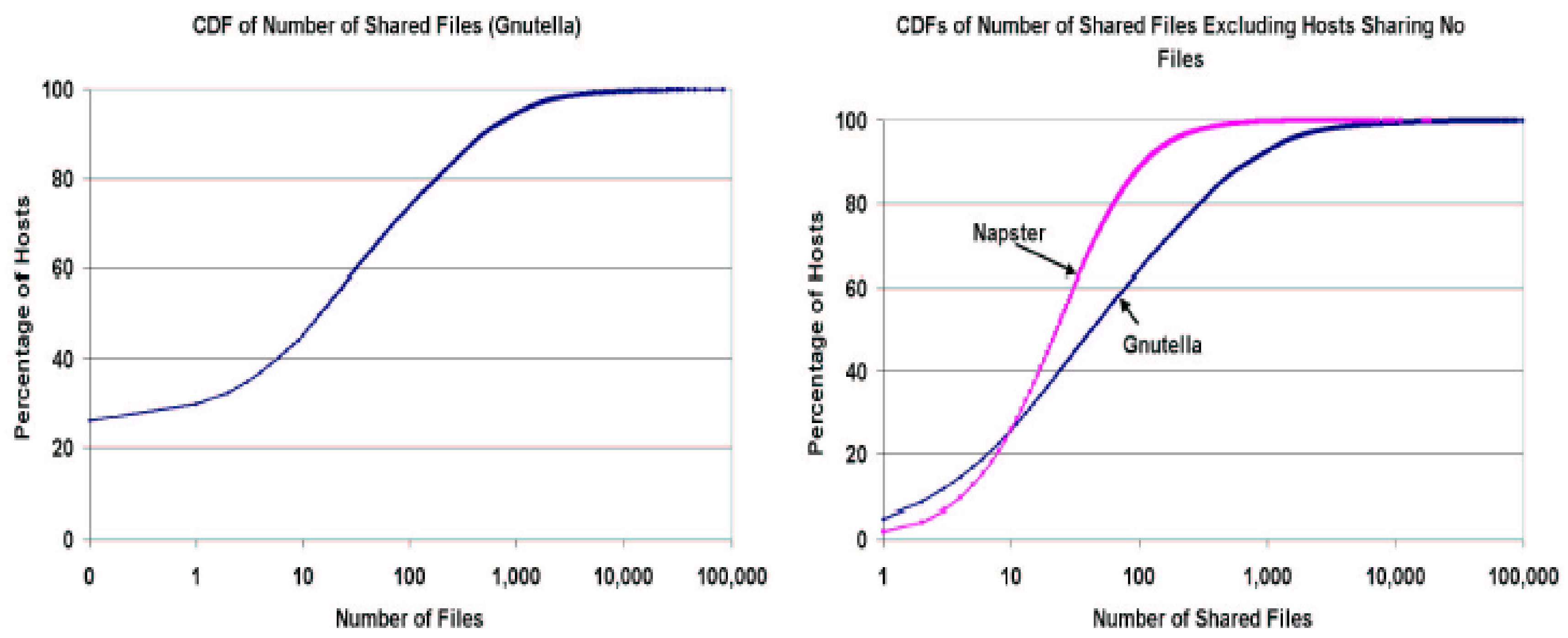


Figure 7. Left: The number of shared files for Gnutella peers; Right: The number of shared files for Napster and Gnutella peers (peers with no files to share are excluded).

- Napster and Gnutella both has a large amount of free-riding
  - 25% of Gnutella peers shared nothing; 75% share 100 files or less; 7% of the clients share more 1000 files and offer more than all the other users
  - 40-60% of Napster peers share only 5-20%

3.3.2 Number of downloads and uploads in Napster

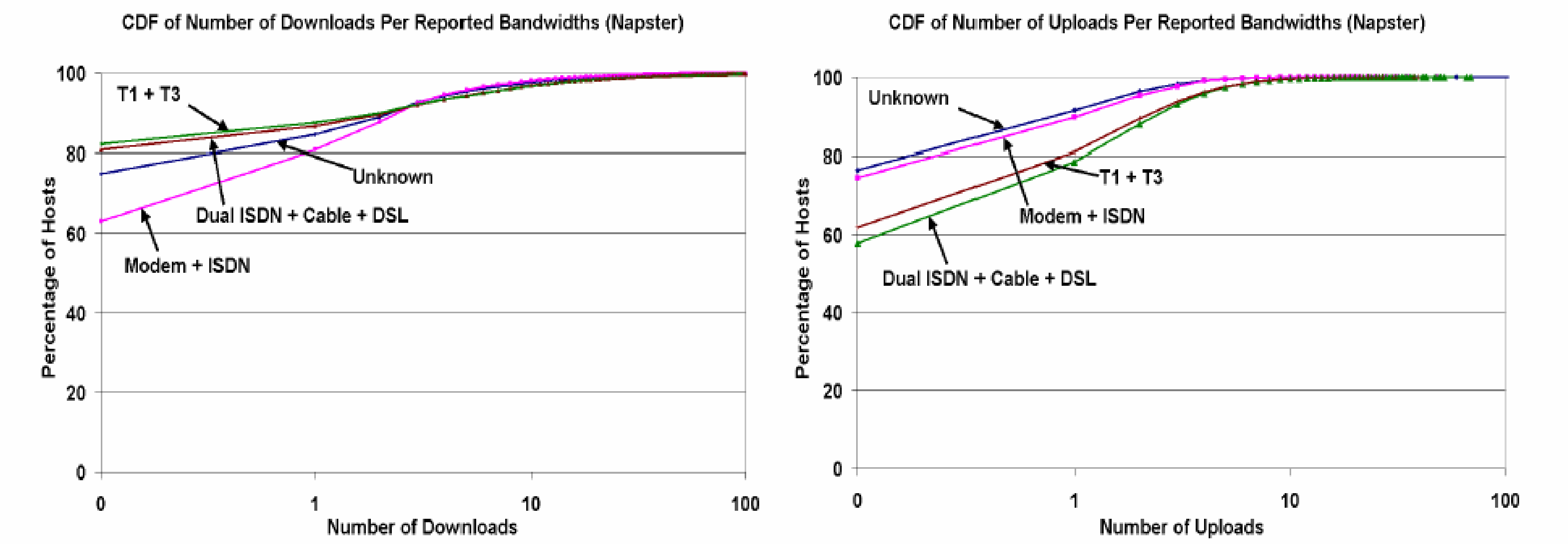


Figure 8. Left: The number of downloads by Napster users, grouped by their reported bandwidths; Right: The number of uploads by Napster users, grouped by their reported bandwidths.

- There are 20% more zero-download high-speed peers than zero-download low-speed peers
  - higher-bandwidth peers tend to download less often; spend less time downloading
  - correlation between bandwidth and download is reversed related to bandwidths and uploads

3.3.3 Correlation between the number of downloads, uploads, and shared files

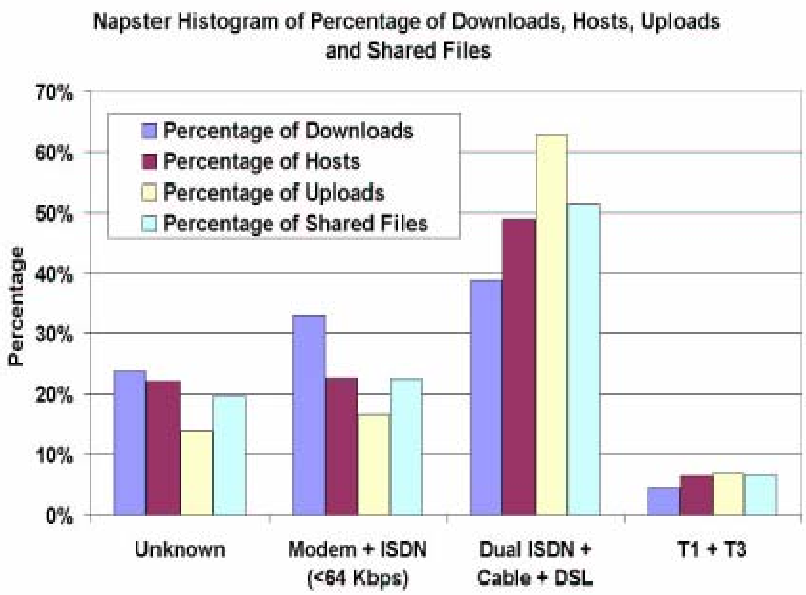


Figure 9: Percentage of downloads, peers, uploads and shared files, grouped by reported bandwidths (in Napster).

- The number of shared files seems to be uniformly distributed across the population
- The skew in the number of uploads is attributed by users selecting high-bandwidth peers from which to download content
- The skew in the number of downloads seems to be more representative of the natural tendency of low-bandwidth peers to be free-riders

3.4 The Nature of Shared Files

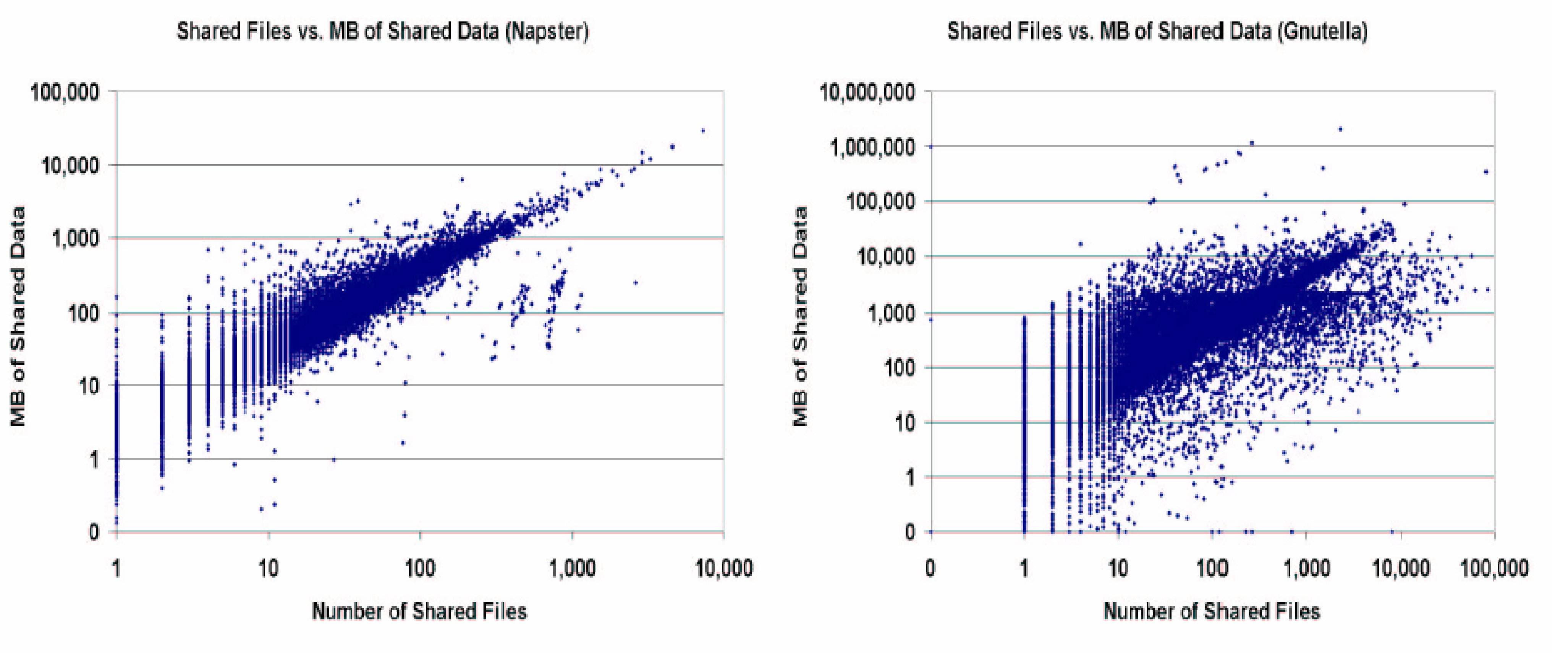


Figure 10: Shared files vs. shared MB of data in Napster and Gnutella.

- In Napster, all shared files must be in MP3 format. In Gnutella, any file type can be exchanged in Gnutella
- Left: The slopes of the lines in both graphs are virtually identical at 3.7MB, corresponding to the size of a shared typical MP3 audio files.

3.5 How much are peers willing to cooperate in a P2P file-sharing system?

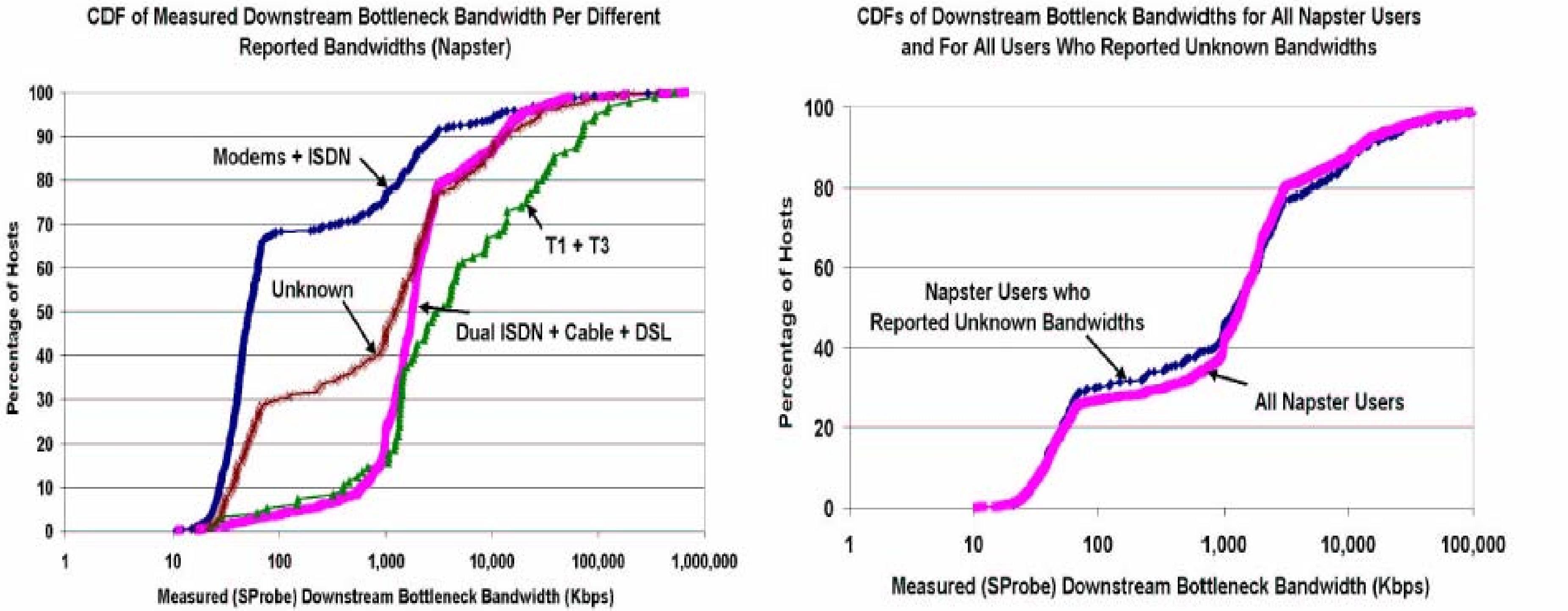


Figure 11. Left: Measured downstream bottleneck bandwidths for peers, grouped by their reported bandwidths; Right: CDFs of measured downstream bottleneck bandwidths for those peers reporting unknown bandwidths along with all Napster users.

- Note that as high of the users that report their bandwidth as 64 Kbps or less actually have a significantly greater bandwidth. (Left)

3.6 Resilience of the Gnutella Overlay in the Face of Attacks

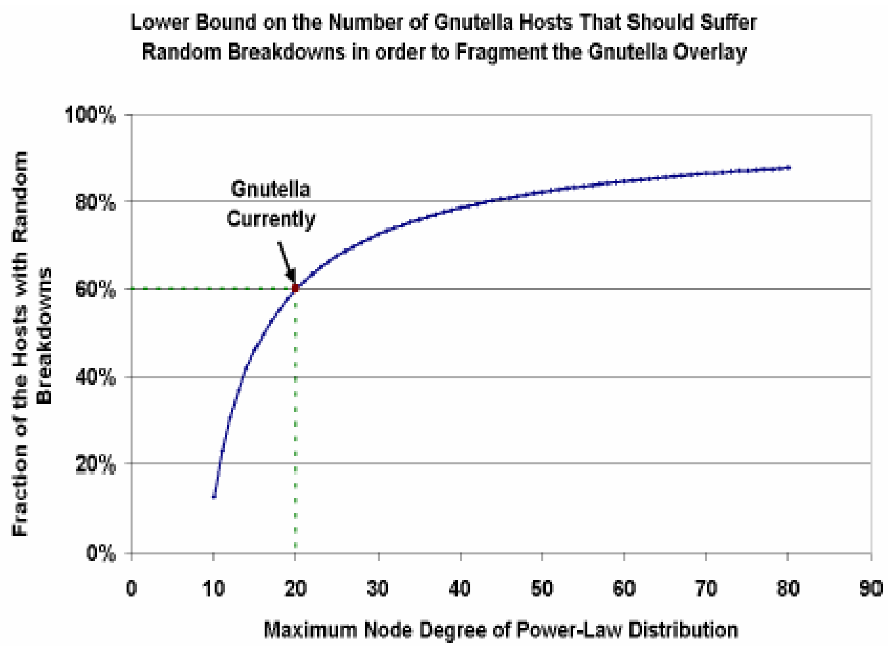


Figure 12. Lower bound on the number of Gnutella peers that must suffer random breakdowns in order to fragment the Gnutella network.

- vertex connectivity power-law distribution for the Gnutella overlay: α=2.3
- For a maximum node degree of 20, the overlay fragments only when more than 60% of the nodes shutdown

Topology of Gnutella Networks

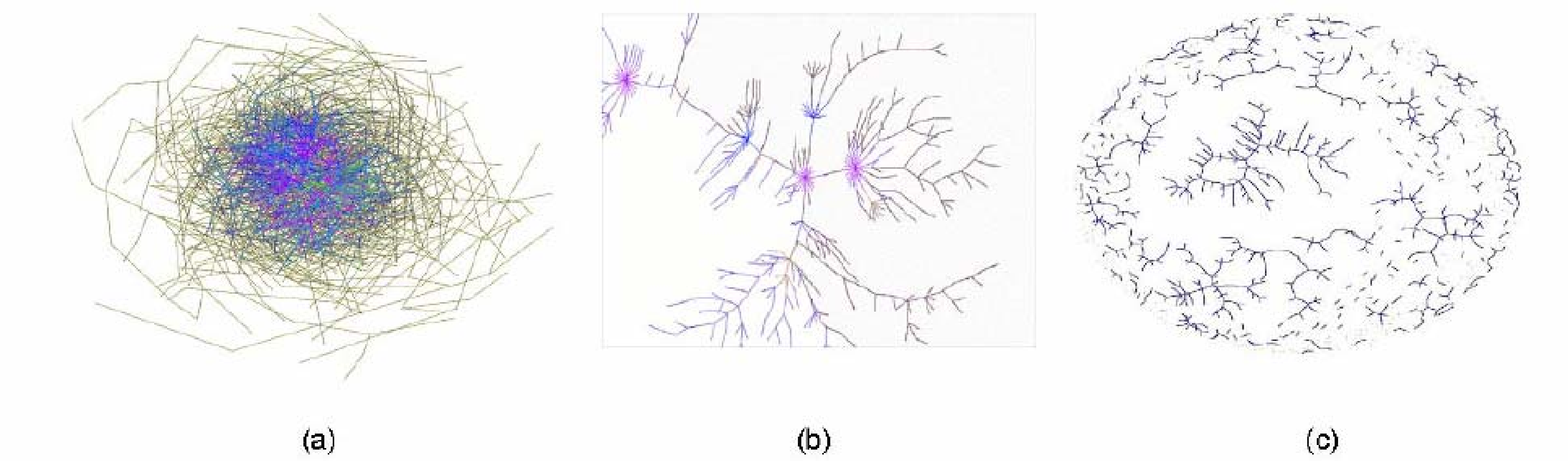


Figure 13. Left: Topology of the Gnutella network as of February 16, 2001 (1771 peers); Middle: Topology of the Gnutella network after a random 30% of the nodes are removed; Right: Topology of the Gnutella network after the highest-degree 4% of the nodes are removed.

(c) highly vulnerable if 63 (less than 4%) best connected Gnutella peers are attacked

Recommendations to P2P System Designers

- Heterogeneous (3 to 5 orders) to connection speeds, latencies, lifetimes, and shared data
  - P2P systems should delegate different degrees of responsibility to different hosts
- A robust system should attempt to directly measure the characteristics of peers in the systems
  - parameters left unspecified or deliberately misreported
- Client-like and server-like behavior clearly identified
  - Myth: behave equally
  - 26% of Gnutella users shared no data
  - 20-40% of Napster users share little or no files

Conclusions

1. There is a significant amount of heterogeneity in both Gnutella and Napster
   - Bandwidth, latency, availability, and the degree of sharing
   - Any similar P2P system must be very deliberate and careful about delegating responsibilities across peers
2. There is clear evidence of client-like or server-like behavior
3. Peers tend to deliberately misreport information if there is an incentive to do so
   - Future systems must either have built-in incentives for peers to tell the truth or systems must be able to directly measure and verify reported information