

**EE 5371- Computer Systems Performance
Measurement and Evaluation**

Spring - 2004

Benchmarking P2P Applications

**Varun Chandola - 2960111
Devdatta Kulkarni - 2544849**

**Department of Computer Science
University of Minnesota, MN 55455 USA
{chandola, dkulk}@cs.umn.edu**

Benchmarking P2P Applications

Varun Chandola, Devdatta Kulkarni

Department of Computer Science
University of Minnesota, MN 55455 USA
{chandola, dkulk}@cs.umn.edu

We intend to propose a set of performance metrics to evaluate a peer to peer (P2P) file sharing system. P2P file sharing systems are increasingly becoming popular in the internet community for efficient and easy sharing of files among users. A group of people have to choose from several commercially available P2P clients to share their files. The available P2P networks cater to a specific type of files like multimedia or research papers etc. We propose a set of benchmarks which can be used by a group of people to evaluate a P2P system based on their specific requirements. Using these benchmarks one can characterize the performance of any P2P based system and compare different available alternatives. We present a methodology to use these benchmarks to evaluate and compare the alternatives. Based on the user requirements, we also propose a methodology to create an input suite which can be used to measure these proposed metrics. As an example we use our benchmarks to evaluate and compare two popular P2P based file sharing systems.

1 Introduction

The popularity of peer-to-peer file sharing applications [8] such as Gnutella [1], Imesh [3], Kazaa [2] etc. has necessitated the performance analysis of such systems. A typical peer to peer system can be defined as a set of clients which share files with each other across the Internet. In such system multiple computers can hold a copy of one file.

Whenever a client requires a file, it obtains the location of all such peers (individually or a central server holding indices) and then requests the file from these computers. The main issue with such systems is that there is a considerable lack of cooperation between the nodes. So any node can go down even during the transfer. In that case the client has to redistribute load among the other nodes.

The main criteria on which a P2P system needs to be tested are *latency*, *availability* and *reliability*. We provide the exact definition of these metrics and the rationale behind them in the subsequent sections.

The basic idea behind our proposal is that for any user, intending to use a system as a node in a P2P based file-sharing system; these metrics can be used as benchmarking targets to evaluate the system. Besides proposing these set of metrics we also propose a set of test suite which can be used to evaluate any system for these metrics.

Different users might have different requirements as far as the type of files being shared is concerned. We provide a methodology to generate an appropriate input suite for those particular types of files. Our metrics will act as a black-box which will generate certain numbers based on the performance of a P2P system which is fed to the black-box along with the input suite. We also describe how the results can be used to characterize a system and how two alternatives can be compared based on these results to find which one would suit the needs of the users better. A typical scenario would be if a group or an individual wishes to share multimedia files. The alternatives could be Kazaa, Napster or Imesh. If the group wants to share research papers it might have to choose between E-donkey and Overnet. By choosing an input suite and then running the benchmarks, the user can compare the alternatives.

The rest of this reported is organized as follows - Section 2 discusses the related works in this field of research. Section 3 gives a detailed description of our proposed metrics. Section 4 describes the input suite used to test the proposed metrics. Section 5 describes the methodology to evaluate a P2P network and how to interpret the results and compare two systems based on these results. Section 6 contains the results of our experiments on

two popular multimedia based P2P networks viz. Kazaa and Imesh and a comparative analysis of both based on the obtained results. We present our conclusions in the last section.

2 Related Works

In this section we discuss the related work that has been done to measure the characteristics of peer-to-peer file sharing systems. In that regard [8] is one of the earliest studies about different performance aspects of a peer-to-peer system. The authors evaluate the performance of Napster and Gnutella systems. The objective of the evaluation is to find overall characteristics of the peer-to-peer file systems such as number of peers present, latency of file download, lifetime of the peers, bottle bandwidth measurements. The paper presents detail analysis of the Napster and Gnutella and provides comparison between the two. Our objective is orthogonal to their research. Our objective is to provide a methodology for the user to evaluate different peer-to-peer file sharing systems treating them as black box. Hence we do not concentrate on the parameters such as the available network bandwidth. Instead we stress the steps a user should take so that he/she is able to answer user level questions such as: Is the particular peer-to-peer system suitable for the kind of files in which the user is interested.

In [6] quantitative measures for searching performance are proposed. Specifically they provide measures of query efficiency, search responsiveness and search efficiency. These are mainly network related performance parameters and could be considered to be an addition to the performance measurements suggested in [8]. The search related parameters can be used to characterize different search algorithms used in the peer-to-

peer systems. We think that the search algorithm is important but not as much important from the end user perspective.

In [7] authors give two benchmarks for evaluating the performance of two overlay networks namely Chord and Tapestry. These benchmarks are from the perspective of the application developer/p2p system builder and would be orthogonal to our set of benchmarks.

We used availability as one of the metric for evaluation. In [4] it is empirically shown that the availability of any p2p system changes over time in a single day. Hence they propose that any measurements for the availability should be done multiple times in a single day. In our experiments we took the measurements at an interval of 6 hours for 2 days. This allowed us to include the varying effects of host joining/leaving the p2p system. It is also shown that finding availability based on the IP addresses is not suitable because technologies such as DHCP and Network Address Translation (NAT) boxes tend to skew the distribution of the IP addresses present in the network at any given time.

3 Performance Metrics

The main functionality of a P2P based file-sharing system is to share files. Different users have potentially overlapping set of files. The users access files from other members of the group across this P2P network. To utilize the existence of one file on different machines, a user tries to download the file from many sources in parallel. The main requirements of a good P2P based system would be - existence of the desired file and lesser time to

download. The constraints on a P2P system are - Number of peers holding a particular file and Number of peers active at a particular time. Almost all P2P systems currently existing do not impose any condition of active participation by the peers at all times. By active participation we mean that a peer is active or alive at that time. Based on these requirements and constraints, we propose following three metrics which characterize a typical P2P based file-sharing system.

- **Latency** The time elapsed between the issue of a request for a file from a peer to the receiving of the complete file. We argue that this is an important metric because it signifies how fast a user can download a file from its peers. A network which has lower latency for a particular file than another network is better. A lower latency means that the file is available with multiple peers and a considerable number of them are active at a given time. Similarly higher latency for a particular file could either mean that it is held by a few peers or only a few peers holding the file are active at that time.
- **Availability** Number of peers holding a file. Most of the current P2P clients are such that they first identify all nodes holding a file. They then access parts of the file from different clients in parallel. So if multiple computers have the file, the access time would be lower. Moreover it also leads to reliable communication because even if one node is down the client still can obtain the file from other nodes.
- **Reliability** This means that if any one or more peers hold a file then it can be downloaded by other peers without any error incurred during the transfer of the file. This tests the reliability of the file transfer protocol in the whole system.

Moreover if the system is robust it can check for the integrity of the file (sometimes faults during transmission can lead to errors in the received file) obtained and in case of an error try to download the file again from the same or different sources.

4 Input Suite

As stated above, the main aim of this project is to evaluate performance of a P2P system from an end-user perspective. Thus it is very important to choose an input suite which represents the actual input domain. We propose three characteristics of a test suite:

- **Availability** The most important characteristic of an input suite must be that it should test the availability of the files which the user is interested in. For example, if a user is interested in sharing research literature in Computer Science the input suite must comprise of related files such as research papers from Computer Science journals, research papers from Computer Science conferences and so on.
- **Size** One of the benchmarks for a P2P system is latency. To get an accurate measure of the latency it is necessary to test the systems performance for different sizes of the input data. Thus the input suite should contain files of varying sizes from small to large and very large as the case maybe.
- **Popularity** Intuitively one can say that any P2P system would perform better for popular files and would not perform as good for rare files. From an end-user's perspective, it is important to know how a P2P system performs according to the popularity of the input file.

5 Methodology

This section describes how we can use the above proposed metrics to evaluate the performance of a particular P2P application.

5.1 Choosing an Input Suite

As mentioned before, choosing an input suite is governed solely by the user requirements i.e. what kind of files would the user want to share using the P2P applications. Some possible classes can be - multimedia, research papers, software etc. The input suite would contain files of the type the user wants to share using the P2P application.

The next step is to take files belonging to different popularity levels. Thus if the user is interested in movies, he/she can take ratings available on the internet. For research papers one might consider the number of references as the measure of availability. A good input suite would contain file names of popular, moderately popular and rare files.

The final step is to take files of varying sizes from small to large in each category of popularity level.

5.2 Running tests

After creating the input suite the above mentioned metrics are obtained as follows:

- **Latency** For each file in the input suite, we run a query in the P2P system. For any of the results obtained the file is downloaded and time required to download the file is noted. Since a P2P system comprises of peers who might go offline at different times of the day, the same experiment needs to be repeated four times a day at intervals of 6 hours to account for such behavior. The mean latency for each file is recorded.

- **Availability** For each of the file in the input suite we note down the number of results for the search query. The average value of the availability for the entire input suite is the mean availability of the system.
- **Reliability** For each file in the input suite, we run a query in the P2P system. From the results of the search query, we randomly select one of the links and try downloading the file. If the download completes successfully the file is checked for any errors. If there are no errors the reliability for that file is 1. If there is an error or the file does not get downloaded within mean latency for that file then another link is selected randomly from the results. If a correct copy is obtained at the second instance the reliability for the file is recorded 0.75. Similarly the latency would be 0.50 if the correct file is obtained in two tries, 0.25 for three tries and 0 after that. The average value of the reliability for the entire input suite is the mean reliability of the system.

5.3 Comparing alternatives

To compare two P2P systems for a test suite we first record data as described above for each of the values. For each of the metric we apply the comparing two alternatives scheme [5]. If the confidence interval for the mean of differences of the results for each of the alternative contains a zero we can say that there is no statistical difference. Otherwise the difference would be significant. Looking at the plots one can find which alternative is better than the other for each of the benchmark metric.

6 Experiment and Results

We applied the above proposed methodology to compare two different P2P systems for a particular type of data. We as users want to find out which is the best available P2P system for sharing classic English rock music. We identified two main alternatives - *Kazaa* and *Imesh*. We collected data four times a day for two days.

6.1 Input Suite

Following the guidelines stated above we came up with the input suite as shown in figure 1. The suite is divided into three categories based on the sizes viz. play lists, songs and videos. In each category we have three files, two from the most popular category, two from moderately popular category and two from the rare category. The popularity rankings are obtained from the internet for each of the categories.

6.2 Results

The results obtained by running the experiments for above test suite are represented in the following graphs.

Figure 2(a) and 2(b) show how latency changes as we increase the size of the file being downloaded. Both graphs show that latency with the size of the file.

Figures 3(a) and 3(b) plot the availability of the files based on their popularity level. The results show that popular files are more available while rare files have low availability.

In figures 4(a) and 4(b) we try to establish relation between reliability and availability of a file. The graphs suggest that there might not be any relation between reliability and availability. But we feel that running the tests on more data might be able to shed more light on this aspect.

Play list	Band Name	Size (KB)
P1	Beatles	2
P2	Eagles	3
P3	Pink Floyd	3
P4	Metallica	3
P5	Tom Petty	3
P6	Bob Dylan	3

<http://www.top10links.com/cat.php/Arts:Music:Artists:Classic+Rock>

Song	Song Name	Size (KB)
S1	Hotel California	6090
S2	Stairway to Heaven	5527
S3	Rainy Day Women	3478
S4	Rebel Rebel	4988
S5	Reeling in the years	5234
S6	Space Oddity	3384

<http://classicrock.about.com/library/misc/bltop500a.htm>

Video	Song Name	Size (KB)
V1	Teen Spirit – Nirvana	58493
V2	All The Small Things - Blink 182	30276
V3	Freak on a leash – Korn	48411
V4	Dont tell me - Avril Lavigne	59862
V5	Alive - Pearl Jam	38789
V6	November Rain - Guns & Roses	92389

<http://in.news.yahoo.com/040409/139/2ch5l.html>

Figure 1: Input Suite for the experiment. The link below each is the source where we obtained the rankings

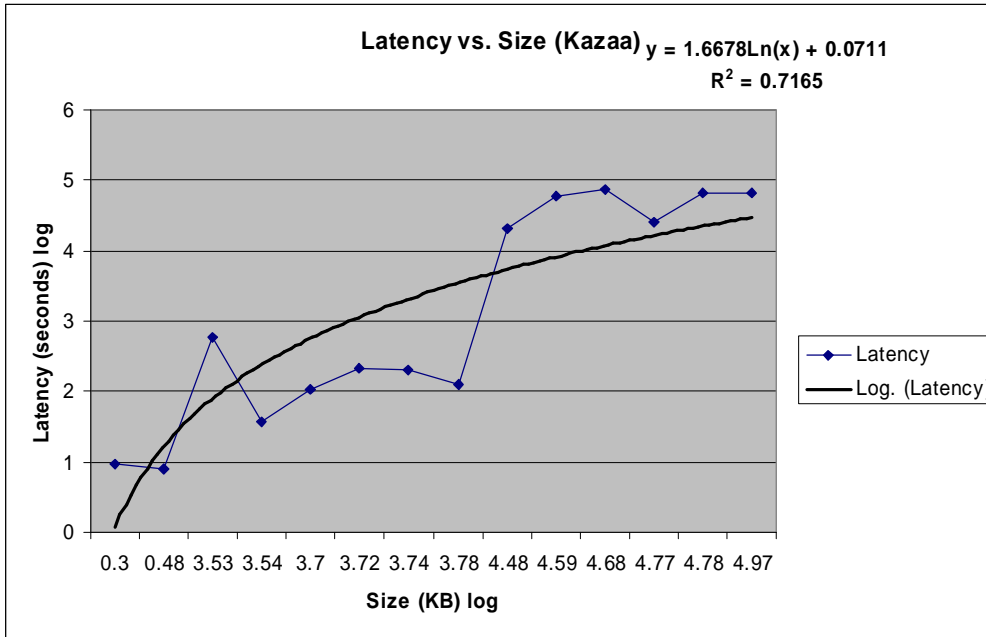


Figure 2(a): Latency vs. Size for Kazaa

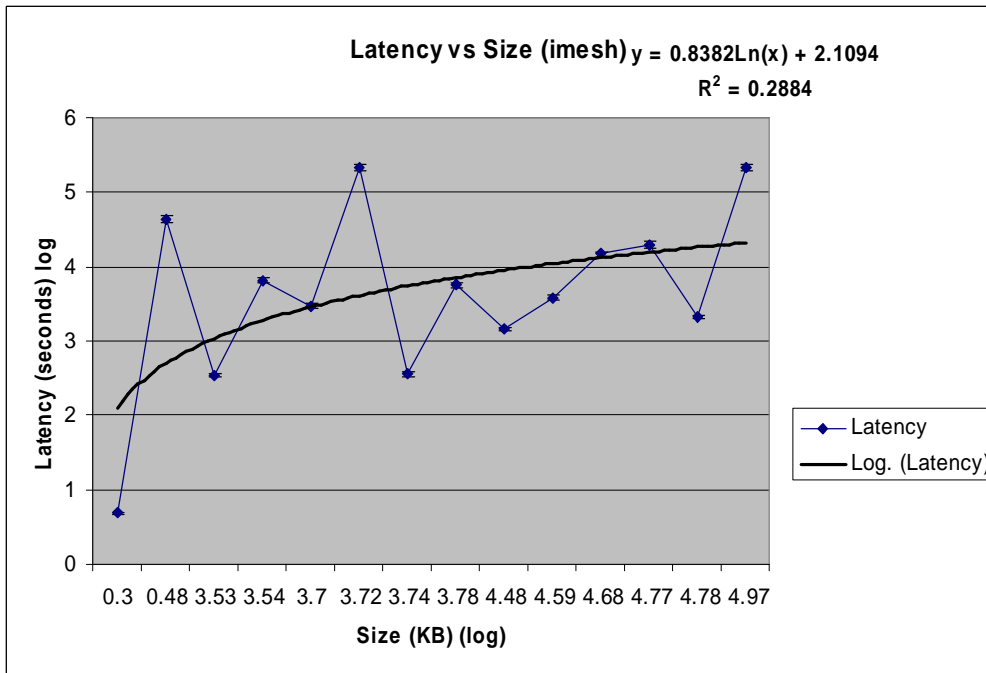


Figure 2(b): Latency vs. Size for Iimesh

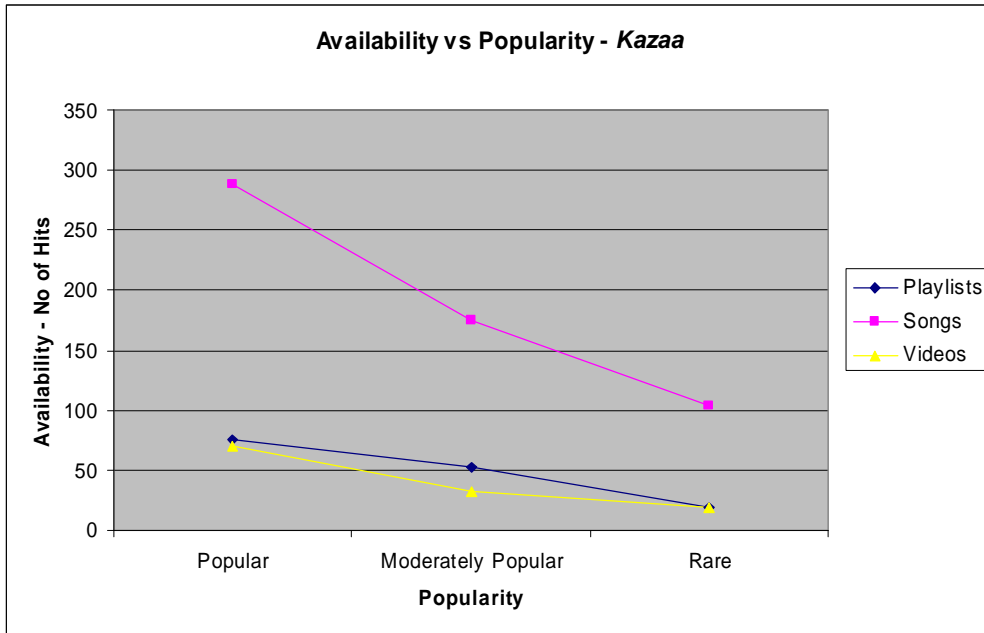


Figure 3(a): Availability vs. Popularity for Kazaa

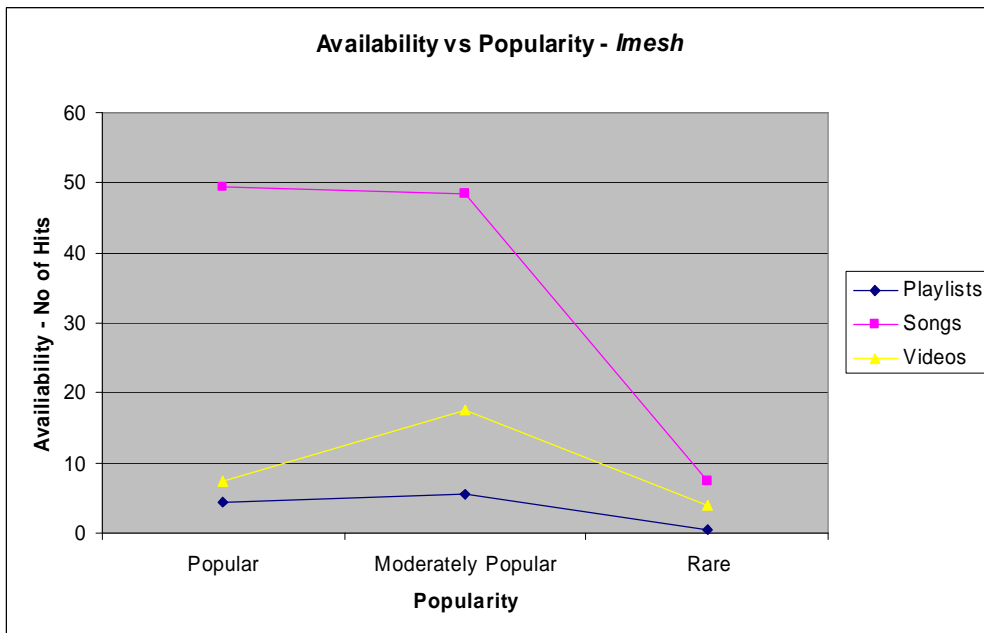


Figure 3(b): Availability vs. Popularity for Imesh

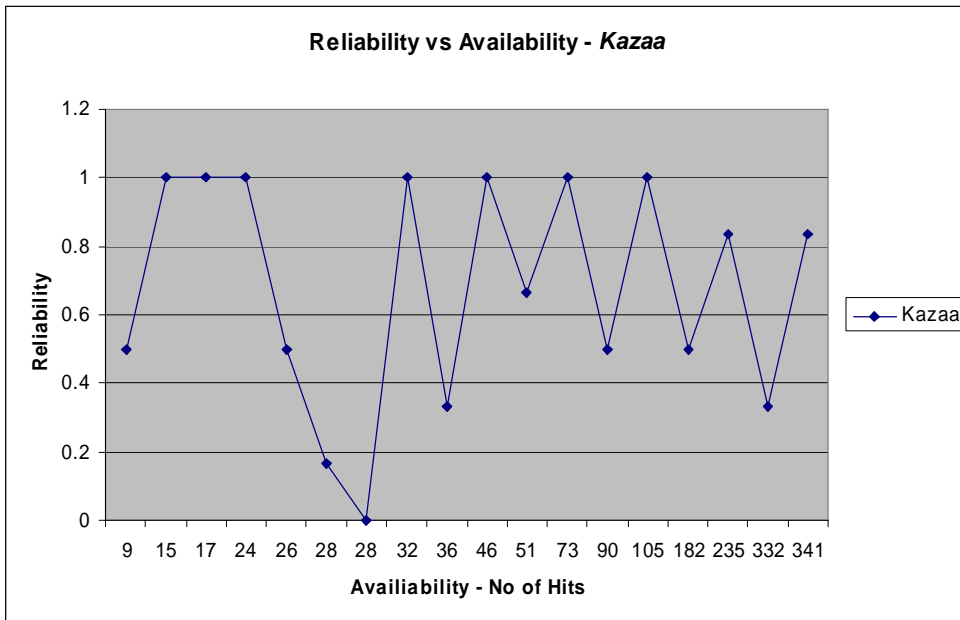


Figure 4(a): Reliability vs. Availability for Kazaa

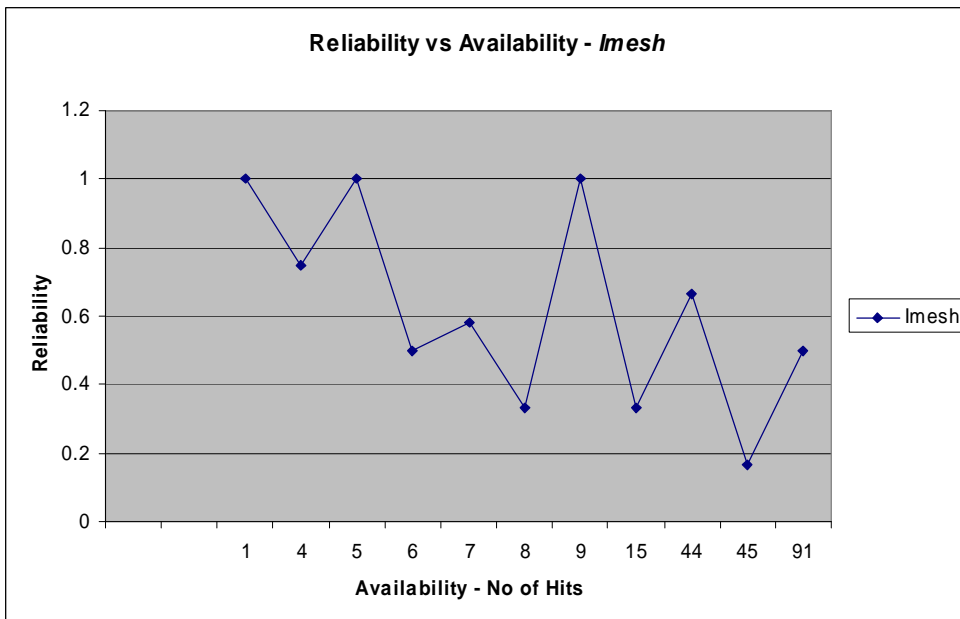


Figure 4(b): Reliability vs. Availability for Imesh

6.3 Comparing alternatives

We employ the “before and after comparing alternatives” scheme to find out if there is any statistically significant difference between the two alternatives for any of the proposed metric.

latency		
t_{kazaa}	t_{imesh}	$t_{kazaa} - t_{imesh}$
9.333333	4.833333	-4.5
2.666667	4	1.333333
3.5	95.83333	92.33333
13	201.3333	188.3333
6.5	215999	215992.5
14.33333	581.3333	567
124	5648	5524
199	356.5	157.5
38	6592.333	6554.333
108.8333	2968.333	2859.5
218.3333	215999	215780.7
580.5	342.1667	-238.333
26052.67	19432.67	-6620
20156.5	1453.167	-18703.3
73250.5	14683.17	-58567.3
68029.83	2103.167	-65926.7
61007.83	3849.5	-57158.3
66266.33	215999	149732.7
	mean	21679.54
	variance	83558.07
	Std dev	289.0641
	c	133.5383
	c1	21546
	c2	21813.08

Table 1: Before and After Comparison of Latency Values for Kazaa and Imesh

As can be seen from Table1, the confidence interval of the difference of latencies does not contain zero. Hence, we can conclude that there is *statistically significant* difference

between the two. From Table 1 and Figure 2 we can also conclude that latency for Kazaa is less than latency for Imesh.

Availability		
a_{kazaa}	a_{imesh}	$a_{kazaa} - a_{imesh}$
105	5	-100
46	4	-42
32	7	-25
73	4	-69
15	0	-15
24	1	-23
341	45	-296
235	44	-191
17	6	-11
332	91	-241
26	0	-26
182	15	-167
90	6	-84
51	9	-42
36	7	-29
28	8	-20
9	6	-3
28	4	-24
	mean	-78.2222
	variance	87.20399
	Std dev	9.338308
	c	4.313996
	c1	-82.5362
	c2	-73.9082

Table 2: Before and After Comparison of Availability Values for Kazaa and Imesh

As can be seen from Table 2, the confidence interval of the difference of availabilities does not contain zero. Hence, we can conclude that there is *statistically significant* difference between the two. From Table 2 and Figure 3 we can also conclude that availability for Kazaa is more than availability for Imesh.

reliability		
r_{kazaa}	r_{imesh}	$r_{kazaa} - r_{imesh}$
1	1	0
1	1	0
1	1	0
1	1	0
1	0	-1
1	1	0
0.833333	0.166667	-0.66667
0.833333	0.666667	-0.16667
1	0.666667	-0.33333
0.333333	0.5	0.166667
0.5	0	-0.5
0.5	0.333333	-0.16667
0.5	0.833333	0.333333
0.666667	1	0.333333
0.333333	0.166667	-0.16667
0.166667	0.333333	0.166667
0.166667	0	-0.16667
0	0	0
	mean	-0.12037
	Variance	0.337259
	Std Dev	0.58074
	c	0.268283
	c1	-0.38865
	c2	0.147913

Table 3: Before and After Comparison of Reliability Values for Kazaa and Imesh

As can be seen from Table 3, the confidence interval of the difference of reliabilities does contain zero. Hence, we can conclude that there is no *statistically significant* difference between the two.

7 Conclusions

From the above tests we can conclude that Kazaa performs *statistically better* than Imesh in terms of latency and availability while there is *no statistically significant difference* between the two in terms of reliability.

In this project we developed a benchmark test suite for peer-to-peer applications based on the end user's requirements. We evaluated and compared the performance of two different peer-to-peer systems using the proposed benchmarks.

References

1. <http://www.gnutella.com>.
2. <http://www.kazaa.com>.
3. <http://www.imesh.com>.
4. Ranjita Bhagwan, Stefan Savage, and Geofiregy M. Voelker. *Understanding availability*. In *Peer-to-Peer Systems II*, Second International Workshop, IPTPS 2003, Berkeley, CA, USA, pages 256{267, 2003}.
5. David Lilja. Comparing alternatives. In *Measuring Computer System Performance*, pages 61{81, 2000}.
6. Tsungnan Lin and Hsinping Wang. Search performance analysis in peer-to-peer networks. In *Third International Conference on Peer-to-Peer Computing (P2P'03)*, 2003.
7. Sean Rhea, Timothy Roscoe, and John Kubiawicz. Structured peer-to-peer overlays need application-driven benchmarks. In *Peer-to-Peer Systems II*, Second International Workshop, IPTPS 2003, Berkeley, CA, USA, pages 56-67.
8. Stefan Saroiu, P. Krishna Gummadi, and Steven D. Gribble. A measurement study of peer-to-peer file sharing systems. In *Proceedings of Multimedia Computing and Networking 2002 (MMCN '02)*, San Jose, CA, USA, January 2002.