Analysis of Search and Replication in Unstructured Peer-to-Peer Networks

Saurabh Tewari, Leonard Kleinrock

Computer Science Department, University of California at Los Angeles

Los Angeles, CA90095

{stewari, lk@cs.ucla.edu}

ABSTRACT

This paper investigates the effect of the number of file replicas on search performance in unstructured peer-to-peer networks. We observe that for a search network with a random graph topology where file replicas are uniformly distributed, the hop distance to a replica of a file is logarithmic in the number of replicas. Using this observation we show that flooding-based search is optimized when the number of replicas is proportional to the file request rates. This replica distribution is also optimal for download time and since flooding has logarithmically better search time than random walk under its optimal replica distribution, we investigate the query-processing load using this distribution.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: *Design studies*

General Terms

Performance, Design.

Keywords

Peer-to-peer, Replication, Search Performance, Random Graphs, Unstructured Networks, Flooding, Optimal File Replication.

1. BACKGROUND

In fully distributed unstructured peer-to-peer networks, the content index of a node is maintained only at that node and hence, a search for a file must probe individual peers to find the location of the file. Since these networks are usually very large and highly dynamic, each node only stores the addresses for a subset of peers and other nodes are reached via these neighbors. Given this search overlay network, in the absence of any information about the location of the queried object, the two main alternatives for search are *flooding* search and *random walk* search. As the name implies, in flooding search, the query is sent to all the neighbors and if they do not have the object they forward the query to all their neighbors and so on. In contrast, in random walk search, the querying node sends the query to one randomly selected neighbor and if that neighbor does not have the desired object, it will forward the query to one of its neighbors (selected randomly).

The performance metrics for any search mechanism are the (average) search time, i.e. how long does it take to find a file, and the (average) query-processing load, i.e. the number of nodes that have to be queried per search. The performance depends on many factors including the search overlay network, the search mechanism and the number of available copies for the file being searched. In this paper, we restrict ourselves to search overlay networks that can be modeled as Erdos-Renyi random graphs, although the results are similar for a variety of other topologies [4]. We focus on the search time in flooding search and derive results on the effect of the number of replicas of a file on the search time for that file assuming that the file replicas are uniformly distributed over the entire network. The search time for a file is approximated by the hop distance to the replica of that file that is nearest to the node requesting the file. After finding the search time for a file under aforementioned assumptions, we explore the performance of controlled flooding search (flooding that terminates as soon as the first replica is found) and provide results on the optimal replica distribution, the average search distance and the query-processing load for this optimal replica distribution. Since this replica distribution also ensures fairness in download load distribution and minimizes the download time [3], we compare the query-processing load at this replica distribution to the optimal query-processing load. Finally, we compare the controlled flooding search and the random-walk search at the replica distributions that minimize their respective search times.

2. MAIN RESULTS

We are interested in the shortest distance from a "querying" node (the node searching for a file) to a replica of the file measured in number of hops (i.e. the number of hops taken by a breadth-first search from the "querying" node). Let $\tau_i(n_i)$ be the expected shortest distance from a querying node to a replica of file i when there are n_i replicas of the file in the network and $n_i \geq 1$. We simulated a number of Erdos-Renyi random graph topologies with a varying number of nodes and varying per-node average degree. The results are shown in Figures 1 and 2. As we can see, $\tau_i(n_i)$, is negative-logarithmically related to n_i/M, the fraction of nodes that have the file, independent of the total number of nodes and increasing the per-node average degree decreases the slope of the relation between $\tau_i(n_i)$ and $-ln(n_i/M)$. Upon further investigation, we found this slope to be inversely proportional to the per-node average degree (see [4]). Thus, we get:

$$\tau_i(n_i) = \alpha \log_d(M/n_i) \qquad (1)$$

This can be analytically shown for finite n_i with $M \to \infty$ [4]. To investigate the search performance optima, we use the peer-to-peer system model described earlier and add the following description. There are N unique files in the system, each with an associated request rate λ_i for file i per node. Each file is assumed to be of equal size. Nodes have finite local storage space to store file replicas. The storage space at each node is assumed to be

SIGMETRICS'05, June 6–10, 2005, Banff, Alberta, Canada.

ACM 1-59593-022-1/05/0006

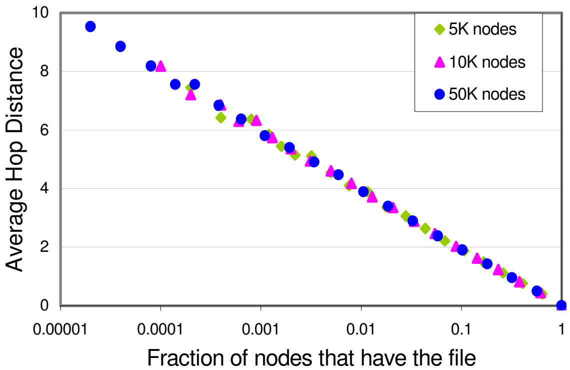

Figure 1. Random Graph: Effect of Number of Nodes (Avg Degree ~3.2)

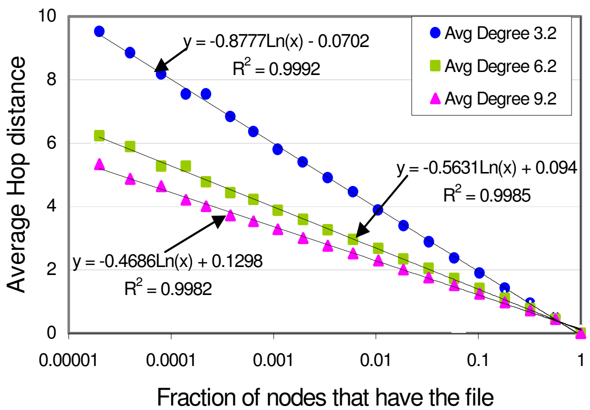

Figure 2. Random Graphs: Effect of Degree (~50K nodes)

equal with the capacity to store K files. The search mechanism is controlled flooding and a node will always satisfy a request for a file present in its local storage. Let us denote $\sum_{i=1}^{N}\lambda_i$ by λ and the average search distance $\sum_{i=1}^{N}\frac{\lambda_i}{\lambda}\tau_i(n_i)$ by τ. The optimization problem then is:

$$\underset{\{n_i\}_{i=1}^{N}}{Min}\left[\tau=\sum_{i=1}^{N}\frac{\lambda_i}{\lambda}\tau_i(n_i)\right] \tag{2}$$

Subject to:

$$\sum_{i=1}^{N} n_i \le KM \tag{3}$$

$$n_i \le M \quad \text{for all } i = 1 \text{ to } N \tag{4}$$

$$n_i \ge 1 \quad \text{for all } i = 1 \text{ to } N \tag{5}$$

Lagrangian optimization yields $n_i = Max(1,Min(\frac{\lambda_i\beta}{\gamma_0}, M))$ where γ_0 is s.t. $\sum_{i=1}^{N} n_i = KM$. Assuming $\frac{1}{KM}\le\frac{\lambda_i}{\lambda}\le\frac{1}{K}$ $\forall i$, we get $n_i = \frac{\lambda_i}{\lambda}KM$ and, the minimum search distance is:

$$\tau_{\text{opt}} = -\alpha\sum_{i=1}^{N}\frac{\lambda_i}{\lambda}\log_d\left(\frac{\lambda_i}{\lambda}K\right) = -\alpha\sum_{i=1}^{N}\frac{\lambda_i}{\lambda}\log_d\frac{\lambda_i}{\lambda} - \alpha\log_d K \tag{6}$$

The query-processing load for a blind search for file i is M/n_i [1]. Therefore, $n_i \propto \sqrt{\lambda_i}$ is the optimum distribution for query-processing load and the optimal query-processing load is:

$$Q_{\text{opt}} = \sum_{i=1}^{N}\frac{M\lambda_i}{\lambda}\frac{\sum_{i=1}^{N}\sqrt{\lambda_i}}{KM\sqrt{\lambda_i}} = \frac{(\sum_{i=1}^{N}\sqrt{\lambda_i})^2}{\lambda K} \tag{7}$$

In contrast, the linear proportionality replica distribution gives:

$$Q_{\text{linear}} = \sum_{i=1}^{N}\frac{M}{n_i} = \frac{N}{K} \tag{8}$$

However, if the file request rate distribution is not extremely skewed, the gain factor in query-processing load with the $n_i \propto \sqrt{\lambda_i}$ replica distribution instead of the $n_i \propto \lambda_i$ replica distribution is not very large [4]. A recent measurement study of the Gnutella peer-to-peer file sharing system estimated the file request rate distribution to be zipf-distributed with zipf-exponent around 0.4 [2]. For such a file request rate distribution, the $n_i \propto \lambda_i$ replica distribution is only 1.07 times worse than the $n_i \propto \sqrt{\lambda_i}$ replica distribution for query-processing load. Thus, given the other benefits of the $n_i \propto \lambda_i$ replica distribution (optimal average search time, fair distribution of the download load and minimum download time), the $n_i \propto \lambda_i$ replica distribution is preferable if the file popularities are not extremely skewed.

While the $n_i \propto \sqrt{\lambda_i}$ replica distribution does optimize the average search time for random walk, the optimum search time for controlled flooding search is easily shown to be logarithmically better than the optimum search time for random walk search (see [4]). Table 1 summarizes the different performance metrics for Controlled Flooding search and Random Walk search.

Table 1. Search Performance Summary at Optimal Replica Distribution (all quantities are per-query)

Metric	Controlled Flooding	Random Walk	Controlled Flooding is
Average Search Distance*	$\alpha \log_d(M/n_i)$	(M/n_i)	Logarithmic-ally better
Average Search Distance	$-\alpha\sum_{i=1}^{N}\frac{\lambda_i}{\lambda}\log_d\frac{\lambda_i}{\lambda}$ $-\alpha\log_d K$	$\frac{(\sum_{i=1}^{N}\sqrt{\lambda_i})^2}{\lambda K}$	Logarithmic-ally better
Query-Processing Load	$\frac{N}{K}$	$\frac{(\sum_{i=1}^{N}\sqrt{\lambda_i})^2}{\lambda K}$	Similar if request rate distribution not skewed

** These expressions are valid for arbitrary replica distributions*

3. REFERENCES

[1] Cohen, E. and Shenker, S. Replication Strategies in Unstructured Peer-to-Peer Networks. *Proc. of ACM SIGCOMM 2002*.

[2] Klemm, A., Lindemann, C., Vernon, M.K., Waldhorst, O.P. Characterizing the query behavior in peer-to-peer file sharing systems. *Proc. of ACM Internet Measurement Conference (IMC) 2004*.

[3] Tewari, S., Kleinrock, L. On Fairness, Optimal Download Performance and Proportional Replication in Peer-to-Peer Networks. to appear in *Proc. of IFIP Networking*, May 2005.

[4] Tewari, S., Kleinrock, L. Search and Replication in Unstructured Peer-to-Peer Networks. UCLA Computer Science Dept Technical Report UCLA-CSD-TR050006, March 2005.