An Efficient Search to Improve Neighbour Selection Mechanism in P2P Network

C.R. Totekar and P. Santhi Thilagam

Department of Computer Engineering,
National Institute of Technology Karnataka - Surathkal,
India - 575025
{chinmaytoekar@gmail.com, santhi_soci@yahoo.co.in}

Abstract. One of the key challenging aspects of peer-to-peer systems has been efficient search for objects. For this, we need to minimize the number of nodes that have to be searched, by using minimum number of messages during the search process. This can be done by selectively sending requests to nodes having higher probability of a hit for queried object. In this paper, we present an enhanced selective walk searching algorithm along with low cost replication schemes. Our algorithm is based on the fact that most users in peer-to-peer network share various types of data in different proportions. This knowledge of amount of different kinds of data shared by each node is used to selectively forward the query to a node having higher hit-ratio for the data of requested type, based on history of recently succeeded queries. Replication scheme replicates frequently accessed data objects on the nodes which get high number of similar queries or closer to the peers from where most of the queries are being issued. Two simple replication schemes have been discussed and their performances are compared. Experimental results prove that our searching algorithm performs better than the selective walk searching algorithm.

Keywords: Peer-Peer systems, selective walk, query table, replication schemes.

1 Introduction

Peer-to-Peer systems are increasingly being used nowadays with increasing speed of communication links and high computational power of nodes. Decentralized P2P systems have gained enormous popularity particularly Gnutella [11], Freenet [12] etc. Compared to Centralized P2P systems, which have high probability of single or multiple point of failure. On the other hand purely decentralized system achieves higher fault tolerance by dynamically re-configuring the network as the nodes join and leave.

Structured P2P systems such as CAN [14], Pastry [13], and Chord [15] guarantee to find existing data and provide bounded data lookup efficiency. However, it suffers from high overhead to handle node churn, which is a frequent occurrence of node joining/leaving. Unstructured P2P systems such as gnutella[11] are more flexible in that there is no need to maintain special network structure, and they can easily support complex queries like keyword/full text search. The drawback is that their routing

S.K. Prasad et al. (Eds.): ICISTM 2009, CCIS 31, pp. 119–127, 2009.

efficiency is low because a large number of peer nodes have to be visited during the search process. The flooding mechanism used by gnutella[11] like systems, results in increased amount of traffic as exponential number of redundant messages are generated during the search process, as the value of TTL is increased.

To address the problems of the original flooding several alternative schemes have been proposed. These algorithms include iterative deepening[4], directed BFS[4], local indices based search[4], random walk[7], Probabilistic search[9], popularity-biased random walk [6], adaptive probabilistic search[5], and dynamic index allocation scheme[2], Selective walk searching[8]. All these techniques try to reduce the number of redundant messages generated because of flooding during search and make search more efficient. These methods either try to maintain the indices of neighboring nodes by each node, or by selecting only those nodes having high probability of having hits, based on recent past.

Selective search scheme uses hints from neighboring peers such as number of shared files, most recent succeeded queries etc to select the most promising candidate for forwarding the queries. However it does not consider the type of data objects that are shared by each node in the algorithm. Most users share files which they themselves are interested in, some might exclusively share movies and some may share only music, and may not share even a single file of picture or document, or archive type. Thus forwarding the query for music file type, to a neighbor having higher share of data as done in Selective Walk searching, might not give proper results if the selected neighbor is sharing no files of type audio, but is having higher share of overall data. Similarly each peer in Selective Walk technique gets rating based on information about most recent succeeded queries maintained in its query table. We can extend this by having a query counter at each node which stores the most recent succeeded queries of each type, thus node having higher value of counter for requested file type gets higher chance of getting selected.

There are very few replication schemes in literature, such as owner replication as used by gnutella and path replication as used by freenet. Gnutella creates copy on the requester node whereas path replication replicates on the path between requester and provider. The third replication algorithm is random replication which is harder to implement. There is need for a simpler approach to find replication site and how many copies to be created.

In this paper, we propose an enhanced selective search which rates each peer based on information about the type of data files it provided recently and number of hits it had. It also uses other details of most recent successful queries at each peer such as list of frequently accessed files and details of total number of different types of files shared by each of the neighboring peers, while selecting the neighbor.

We would like to draw your attention to the fact that it is not possible to modify a paper in any way, once it has been published. This applies to both the printed book and the online version of the publication. Every detail, including the order of the names of the authors, should be checked before the paper is sent to the Volume Editors.