

Effects of Network Dynamics on Routing Efficiency in P2P Networks

Mojca Ciglarič, Andrej Krevl, Matjaz Pančur, and Tone Vidmar

Abstract—P2P Networks are highly dynamic structures since their nodes – peer users keep joining and leaving continuously. In the paper, we study the effects of network change rates on query routing efficiency. First we describe some background and an abstract system model. The chosen routing technique makes use of cached metadata from previous answer messages and also employs a mechanism for broken path detection and metadata maintenance. Several metrics are used to show that the protocol behaves quite well even with high rate of node departures, but above a certain threshold it literally breaks down and exhibits considerable efficiency degradation.

Keywords—Network dynamics, overlay network, P2P system, routing efficiency.

I. INTRODUCTION

ONE of inherent properties observed in P2P network topologies is their dynamics – they change all the time. While new members are joining, the others are leaving; some stay connected only for a few minutes while others only leave after weeks or even months of activity. In our recent research we have been proposing new query routing strategies to reduce the cumulative query traffic without degradation of end user's experience, particularly average system response times. The present paper studies the behaviour of one of the suggested routing protocols under different circumstances regarding network dynamics. In our simulations we will vary the degree of network dynamics from 1 departure per 10 generated queries to 5 departures per query and present its effects on system metrics.

II. BACKGROUND AND RELATED WORK

In unstructured peer-to-peer networks, all the nodes have equal functions, while in two-layered systems, some nodes are superior and act as proxies among their clients or subordinate nodes. Ordinary nodes only connect to one or a small number of the supernodes, and their supernodes communicate with the rest of the overlay on their behalf. The subnetwork of super nodes can also be viewed as an unstructured peer-to-

peer network, since the subordinate nodes can only send queries to “their” supernodes. Basic mechanism for query message routing is usually based on flooding, which is robust and reliable but also exhibits high redundancy and creates very high network load (as first observed in the Gnutella file-sharing network [1]). Better routing strategies are based on locally saved routing metadata from previous queries and answers.

While P2P network overlay changes, the metadata becomes obsolete and should be removed or replaced. However the processes of obsolete metadata detection and establishing new routes take time and the purpose of this paper is to imply the directions for its estimation.

Detailed description of the evaluated protocol can be found in our previous papers [2], [3] and [9], together with the discussion of related routing techniques. P2P overlay network topology and its properties (power law and small world) were researched by several authors, among others in [4], while other P2P-related issues can be found in [6-8]. In [5] Bu and Towsley suggest a topology generator which produces suitable topologies and which we have also used in our simulations.

III. MODEL AND SIMULATIONS

Every peer node shares a set of files. File F_i is described by its metadata m_i , a set of metadata elements: $m_i = \{k_1, k_2, \dots\}$: name, type, size, keywords, hash etc. Query Q_i is a message, identified by a globally unique identifier and containing a subset of available metadata elements: $Q_i = \{k_m, k_n, \dots\}$. The available files are not equally popular. A measure of file popularity q_i [9] is defined as percent of queries looking for file F_i (i.e. matching its metadata according to some matching function). With the term repetitive query we refer to subsequent queries with a positive match to the same file, which does not necessary mean that the repetitive queries contain same keywords and/or other metadata elements.

Each node is able to generate query messages about the files, receive and forward query messages from other nodes, generate answers, and receive and forward answer messages from other nodes. Each node is able to generate and store metadata on received messages. Messages can only be passed on to one or a subset of neighbours, chosen by a routing mechanism. Answers return to the query originator over the same path. When a matching file is found, the node generates answer message containing complete file metadata.

A. Query Routing with Metadata Exchange

Each node that passes on an answer message also caches the

metadata and the neighbour ID in order to use it for routing later when a similar query is issued elsewhere in the overlay. However the cached answers are never passed back to another node again - they are merely used to route the queries. In appropriate time intervals, the nodes send all their metadata to their neighbour nodes, thus helping to build efficient routes in the overlay faster. When no route exists, the query still needs to be flooded.

When storing answer metadata as routing info, the nodes also keep track of the time needed from forwarding a query until the answer message came back. When routing next query over the same route, a node should estimate when an answer should come back, allowing some extra time for unexpected delays. If the answer message does not arrive within that time, the path is considered broken. Therefore, a query flood should be triggered. Since the nodes still keep track of query GUIDs, the flood is not multiplied at each node - each node forwards each query only once (in each direction). This way, the query achieves the best response time possible and also reaches the same set of nodes as if it was flooded from the beginning.

B. Metrics

The simplest metric from the system's point of view is the total number of message hops (HT) in the whole simulation period (including cold start), however more relevant is the average value in the stable system state (HS), when the routes are already configured. R is the average number of nodes reached by a query and M is the average node load (the number of forwarded query messages).

A somewhat modified definition of a query price from [9] is

$$C = \frac{\text{Total query hops}}{\text{Nodes reached}},$$

while in [6] the percent of redundant hops P and query efficiency D are defined as

$$P = \frac{HT - R}{HT} \quad \text{and} \quad D = \frac{\text{All query hops}}{\text{Effective hops}},$$

where a hop is effective when it reaches a node with the matching file.

Another group of metrics are user-related: the number of time intervals (AT) before an answer is received, the number of hops from answer node to the source node (AH), and a share of answered queries (QA) for the queries where an answer can be found within the TTL radius (our routing with metadata exchange should always find such an answer).

We want to minimise C, P, M, R, HS, HT, D, AT and AH, and have QA as close to 100% as possible.

IV. ROUTING IN A HIGHLY DYNAMIC ENVIRONMENT

Yang [7, 8] implies that in common P2P file-sharing networks, the rate of query generation is roughly 10 times higher than the rate of network changes. For observations of degradation in routing efficiency, node departures are most important. However to keep the overlay connected and stable over longer simulation runs, as well as to capture real system

properties, we also introduce new nodes into the topology. New nodes are joining at the same rate as the old ones are leaving. Let us define network change rate NR as the average number of node joins or departures per query. For example, NR = 0.1 means that one node joins and one node leaves the overlay after ten queries are issued.

In our simulation environment we generate one query per time step and to evaluate different routing techniques we have used NR = 0.1, however here we want to compare system metrics with higher NR values.

Our expectations are as follows. As long as the dynamics is within certain limits, its effect on the network behaviour will be negligible. But when the NR will go over the threshold value, the average query traffic (i.e. the number of hops per query) will grow rapidly. Many paths will be broken and the subsequent queries will have to be partially flooded after the original path will not return the expected answer. So we can be pretty sure that the traffic (HS) will go up and average response times (AT) will grow to some extent, while we cannot be sure about the answer distance (AH) - it may as well stay within the boundaries of previous average values. With more query floods, node load (M) will grow together with redundancy (P), while query efficiency will get worse.

V. RESULTS AND DISCUSSION

Our first observation was that there were no significant changes when we varied NR step by step from 0.1 (one node departure / join per ten queries) to 1 (one node departure / join per query). Only when we used NR of 5 (5 node departures / joins per query) we observed the expected degradation of routing efficiency. Figure 1 represents the most obvious degradation - the increase in average number of hops per query. Figure 2 shows the average response times, which are slightly elevated due to the delayed floods, while in Figure 3 we can see that even the percent of answered queries decreases with very high values of NR, possibly because the overlay can sometimes fall apart into two or more disconnected components and some files are not accessible any more.

Figure 4 shows the average node load (M) - the number of query messages to be passed on to the neighbor nodes per time unit (i.e. per one query generated in the overlay). As soon as a few floods are necessary, the average load quickly increases. Sometimes even more important is the maximum node load, because it can cause congestion and drastic decrease of response times. In our case, the maximum node load is over 1200 for NR = 5, while for other values of NR it hovers around 800.

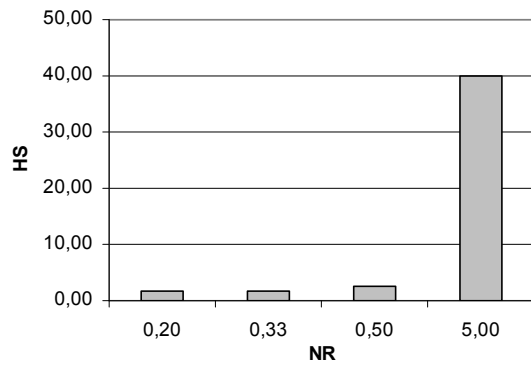


Fig. 1 Average number of hops per query in a stable system state (HS), for different values of network change rate (NR)

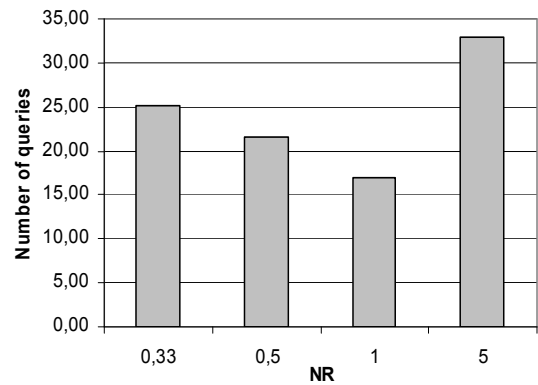


Fig. 4 Average node load (M) for different values of NR

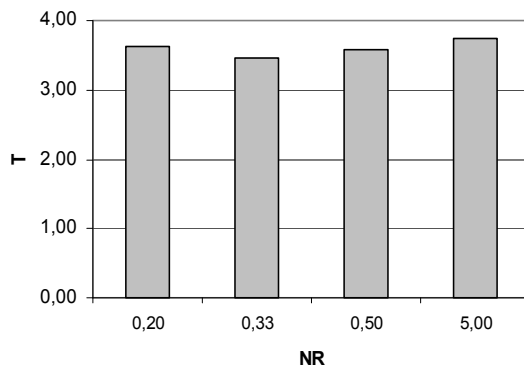


Fig. 2 Average response times for different values of NR

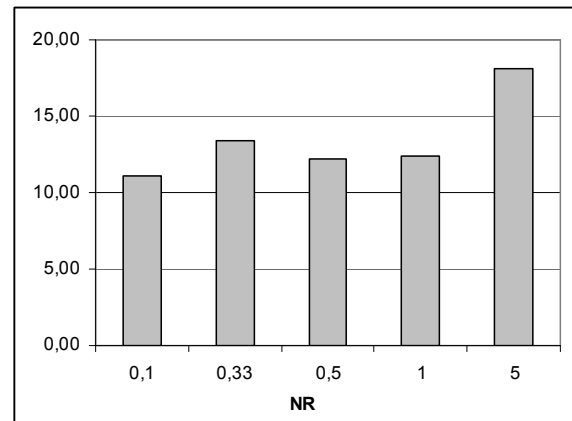


Fig. 5 Query efficiency (D) for different values of NR. Lower values mean more efficiency

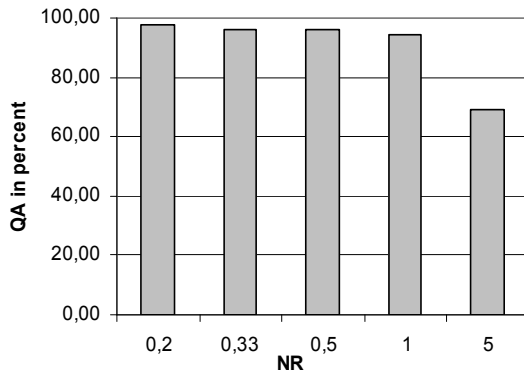


Fig. 3 Percent of the answered queries for different values of NR

Figure 5 shows the values of query efficiency - D. Since D is defined as the ration between the number of all query hops and the number of efficient query hops (those that reach the node with matching file), higher value of D shows that there was more query hops for needed for reaching one node that could generate an answer. We do not show graphs for other metrics since they are quite similar to the ones presented above.

The results obtained from simulations are expected and confirm our assumptions about system behaviour. What surprised us a bit is the threshold value. In a real P2P system, users usually join when they want to get some new files. There are also the altruistic users, who stay longer and share files also when they do not need anything from the system. But in the worst case, when every user is selfish and just joins to make one query and then disconnects (in this case NR is exactly 1), the network behaviour stays below the threshold value. This is very good news since it tells us that the routing protocol is robust enough and can be used even in the most dynamic environment.

VI. CONCLUSIONS AND FURTHER WORK

In the paper, we presented the problem of dynamic nature in

P2P network overlays. We briefly described the abstract system model, the chosen routing mechanism, the mechanism for keeping metadata up-to-date and a set of metrics, describing the system behaviour under certain routing protocol. Further we explained the issues related to the network dynamicity and illustrated our hypothesis on system behaviour in a highly dynamic environment.

By means of simulation we confirmed our assumptions. The main finding, based on the simulation results, is that the routing mechanism with metadata exchange is robust and predictable within the boundaries of NR that can be expected in the common P2P file sharing networks.

Since the research is still in progress, more simulations will be performed: for other values of NR, for other routing mechanisms, and on different network topologies.

REFERENCES

- [1] M. Ripeanu, I. Foster, A. Iamnitchi, "Mapping the Gnutella Network: Properties of Large-Scale Peer-to-Peer Systems and Implications for System Design". *IEEE Internet Computing*, vol. 6(1), 2002.
- [2] M. Ciglarič, "Content networks : distributed routing decisions in presence of repeated queries". *Int. j. found. comput. sci.*, 2004, vol. 15, no. 3, pp. 555-566.
- [3] M. Ciglarič, Towards More Effective Message Routing in Unstructured Peer-to-Peer Overlays, to be published in *IEE Proc. Communications*, 2005.
- [4] M. A. Jovanovic, F. S. Anexstein, K. A. Berman, Modeling Peer-to-Peer Network Topologies through "Small-World" Models and Power Laws, *Proc. IX. Telecommunications Forum TELFOR*, 2001.
- [5] T. Bu, D. Towsley, On Distinguishing between Internet Power Law Topology Generators, *Proc. INFOCOM 2002*.
- [6] Q. Lv et al.: Search and Replication in Unstructured Peer-to-peer Networks, *Proc. 16th ACM Intl. Conf. Supercomputing ICS'02*.
- [7] B. Yang, H. Garcia-Molina, Comparing Hybrid Peer-to-Peer Systems, *Proc. Very Large Databases VLDB*, 2001.
- [8] B. Yang and H. Garcia-Molina, Efficient Search in Peer-to-Peer Networks, *ICDCS 2002*. <http://dbpubs.stanford.edu/pub/2001-47>.
- [9] M. Ciglarič, Problems in Unstructured P2P Systems, to be published in *Electrotechnical Review*, 2005.