Impact of Caching on Peer-to-Peer Traffic in Mesh Networks

Case Study Report

Investing in the future by working together for a sustainable and competitive region

Impact of Caching on Peer-to-Peer Traffic in Mesh Networks

Marcel Castro *Andreas Kassler*
marccava@kau.se *andreas.kassler@kau.se*
Computer Science Department
Karlstad University

Executive summary

In this case study report, we evaluate the impact of caching schemes on P2P lookup efficiency in multihop networks such as Mobile Ad-Hoc Networks (MANET) or mesh networks. We also propose a cross-layer scheme that reduces the routing stretch and leads to more efficient performance of P2P systems over multihop networks. Having more and diverse possibilities to select physical neighbors can greatly improve the lookup efficiency in the overlay. Thus we propose one extension of the default approach where the visibility of a node can be increased by neighbor-of-neighbor (”NoN”) information transmitted via hello (beacon) message, ameliorating then the lookup efficiency while routing it across the network.

1. Introduction and Problem Statement

Since the appearance of Napster in early 1999, peer-to-peer (P2P) networks have experienced tremendous growth. The P2P architectures can be categorized into two main classes: unstructured P2P overlays and structured P2P overlays. Unstructured overlays do not impose a rigid relation between the overlay topology and the indices/resources placement, as flooding is used to locate resources. In contrast to that, structured P2P networks control tightly the overlay topology by arranging the nodes in a logical structure and by placing content at specified locations that will make subsequent lookups more efficient. Very popular representatives of structured P2P networks are realized through so called Distributed Hash Tables (DHTs), such as Chord [1].

Mobile Ad-hoc Networks (MANETs) usually don’t have a dedicated routing infrastructure and rely on multi-hop communication. Nodes in a MANET cooperatively forward other nodes’ data. These networks have a distributed communication architecture, where nodes make individual decisions on routing and medium access. P2P overlay networks in the Internet and MANETs share many key characteristics such as selforganization and decentralization. However, current P2P overlay architectures can not be directly used as is in MANETs, as it abstract the underlying physical topology during the overlay construction and resource lookup. For example, in Chord, each node has to maintain a set of logical neighbors (successor, predecessor and long-range neighbors) in the identifier space, thus issuing control traffic. Long-range neighbors, also called fingers, are used to quickly route messages to remote locations in the identifier space. Given the limited bandwidth in MANETs, the maintenance of logical neighbors can be prohibitively heavy-weight, as the logical neighbors could be located several hops away in the physical wireless topology. While this might be tolerable on the wired Internet with its high bandwidth, it is obviously not feasible for MANETs. Here, the delivery probability of a packet quickly decreases with each physical hop due to factors such as low bandwidth, limited energy, low computation power (of a node), packet collisions, or transmission errors.

2. Alternative Solutions, Selected Approach

Therefore, our envisioned DHT implementation combines a minimalist Chord-like overlay structure which replaces the long-range logical neighbors by physical neighbors. In this (”default”) approach, each node only needs to maintain its successor and predecessor in order to guarantee correctness, which impose then limited management overhead. Thus, the lookup requests rely basically on the physical neighbors. When receiving a lookup request, the node checks among its physical neighbors for the node closest to the key requested. If there is a node closest to the key in its physical neighborhood, such node will become the next destination. If not, the request is forwarded to the next node in the logical path. Therefore, the lookup request traversed in the network decreases, either because we reach the node responsible for the key, or the request is sent to a node closer to the key.

Having more and diverse possibilities to select physical neighbors can greatly improve the lookup efficiency in the overlay. Thus we propose one extension of the default approach where the visibility of a node can be increased by neighbor-of-neighbor (”NoN”) information transmitted via hello (beacon) message, ameliorating then the lookup efficiency while routing it across the MANET. Therefore, as it occurs in the default approach, when routing the request to a destination via the overlay, the node resorts to simple greedy routing by selecting the physical neighbor that makes the most progress in the ID space, and then forwarding the packet along the hop-by-hop route. Another possibility to increase the lookup efficiency is to exploit the lookup request history of the nodes through the use of a cache scheme. As the nodes may participate in the forwarding process of some lookup requests, it should be possible to use such information to augment the ability to find better next logical hops for a specific key lookup request, despite its physical neighbors and neighbor-of-neighbor information. Therefore, each node maintains a ”standard” cache that stores, for each observed lookup request, the searched key and the next logical hop neighbor of the lookup request. Thereafter a cache entry can act as long-range logical neighbor, if it makes more progress to the destination compared to the physical neighbors and neighbors-of-neighbors information. As it is motivated latter on through the results presented in Figure 1, by just caching the lookup requests that pass through a node may not be sufficient to guarantee lookup efficiency, therefore we propose a ”cross-layer” cache scheme that uses the advantage of 1-hop broadcast communication. Then, by also overhearing a key lookup request at the MAC layer, a node can enrich its cache entries. This solution is particularly adapted to MANETs as no additional transmission is required, and energy cost is highly limited.

3. Preliminary Results

In our preliminary results, we consider a mobile ad-hoc network of 500 and 1000 nodes, uniformly distributed at random in a rectangular space. Different node densities are used in order to control the number of physical neighbors a node has in the system. Two nodes are considered neighbors if their distance is smaller than the transmission range limit of 100 meters. We remark that our current evaluation does not target mobility scenarios, thus nodes are stationary along the simulation. The simulation executes a warm-up phase to reach a steady state prior to issuing the key lookups. In the warmup phase, each node generates 20 lookups to random selected keys. We use standard AODV as the routing protocol, which also routes key lookup requests between logical neighbors inside the MANET. For the cross-layer cache scheme, we consider two cases; nodes with limited cache size of 256 entries (Cache256) and nodes with infinite cache size (Cache- 1). The limited cache size is implemented based on the Least Recent Used (LRU) scheme. Here the performance metric under investigation is the DHT routing stretch of key lookups, defined as the ratio between the cost of selected route using the logical neighbors compared to the optimal shortest path routing [2].

Figure 1 motivates the use of ”cross-layer” information to enrich the caching scheme. In this figure we plot the average cache size per node (log scale) versus network density, for 500 and 1000 nodes topology. It is important to observe that we used infinite cache size per node in order to evaluate the maximum cache population that a node can acquire in such scenarios. It can be noted that by just caching lookups which pass through the node (standard cache) does not guarantee a high number of cache entries. Moreover, the number of entries for the standard cache decreases for higher network densities (e.g. 22 and 40 node density) as the number of logical nodes traversed by a key lookup reduces in denser networks. In contrast to that, by taking the advantage of 1-hop broadcast communication, the cross-layer caching enables the MANET’s nodes with the possibility to overhear key lookup packets and therefore acquire a higher (average) cache size per node. In Figure 2, we compare the DHT routing stretch of our DHT implementation versus different network densities. The four different approaches (default, NoN, cache256 and cache-1) are compared against each other. Each result was obtained by averaging over 100 random key lookups performed after the warm-up period. As expected, the use of neighbor-of-neighbor (NoN) information compared with one-hop information (default) ameliorates the DHT routing stretch, specially in higher density topologies. This is because via neighbor-ofneighbor information each node can acquire a greater view of the network topology and therefore select better next logical neighbors. By using caching size of 256 entries allied to neighbor-of-neighbor information (NoN+Cache256), the DHT routing stretch can be reduced even further. In this case, the knowledge acquired by the cache during warm-up phase and the overheard key lookups help in the selection of physical nodes closest to the destination while routing through the logical neighbors. We also verified that by using an infinite cache size leads to approximately a DHT routing stretch of one, which means that the selected route via the overlay is as efficient as the shortest path route.

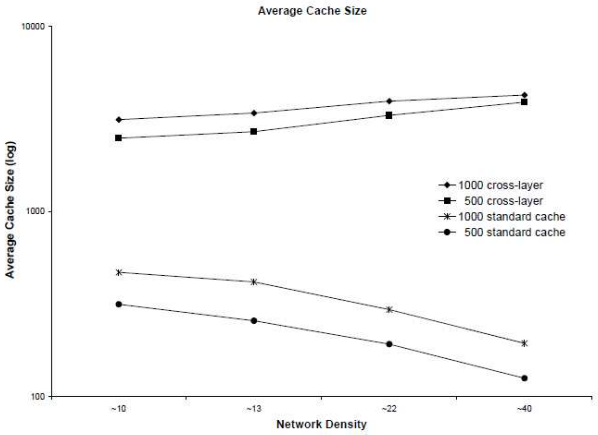

Fig. 1. Average cache size versus network density

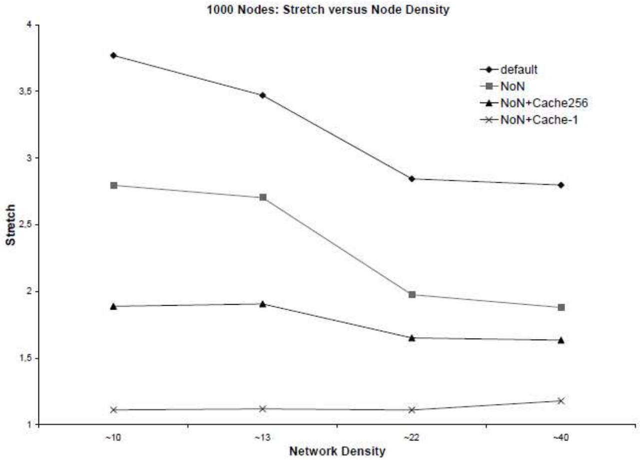

Fig. 2. Routing stretch versus network density

One problem is that the cache may contain many entries which are not necessarily significant, as the cache entries may not be evenly mapped along the logical space. Based on our results, one important point that should be analyzed further is how to better select and manage the cache entries to improve the lookup and consequently reduce its average path cost, while maintaining viable number of cache entries. Possible caching strategies include (but are not limited to) caching according to harmonic distribution (inversely proportional to the logical distance on the ring), trade off between the logical quality of the shortcut and its physical cost, attaching confidence probabilities to the cache entries based on the last validation of the link. Therefore, by carefully selecting the cache entries that must be kept by a node we can 1) have a good coverage of the logical space, 2) cheap shortcut relying on physical nodes and 3) limited memory consumption while keeping only the best entries.

4. Conclusion

In this case study, we have analysed lookup efficiency for Structured Peer-to-Peer Networks in a multihop environment. By introducing caching and overhearing based on a cross-layer approach, we can increase efficiency significantly. With increased network density, a sufficiently dimensioned cache size can reduce routing stretch. We have implemented the proposed approach in a simulator and shown the performance increase that our method can achieve. In order to implement it in a real system, extensions to the P2P protocol and cross-layer message exchange needs to be implemented.

References

[1] Stoica and R. Morris and D. Karger and M. Frans Kaashoek and H. Balakrishnan. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications. *In Proc. of ACM SIGCOMM*, San Diego, USA, August 2001.

[2] C. Cramer and T. Fuhrmann. Proximity neighbor selection for a dht in wireless multi-hop networks. *In P2P '05: Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing*, pages 3–10, Washington, DC, USA, 2005. IEEE Computer Society