Hierarchical Peer-to-Peer Networks using Lightweight SuperPeer Topologies

Michael Kleis
Fraunhofer Institute FOKUS
kleis@fokus.fraunhofer.de

Eng Keong Lua
University of Cambridge
eng.keong-lua@cl.cam.ac.uk

Xiaoming Zhou
Delft University of Technology
X.Zhou@ewi.tudelft.nl

Abstract

The use of SuperPeers has been proposed to improve the performance of both Structured and Unstructured Peer-to-Peer (P2P) Networks. In this paper, we study the performance of Yao-Graph based SuperPeer Topologies for Hierarchical P2P networks. Since a Yao-Graph is defined as a geometric structure, we are using the "Highways" proximity clustering and placement scheme to assign geometric co-ordinates to SuperPeers with respect to the underlying network conditions. Because of the lightweight structure of Yao-Graphs, the resulting hierarchical P2P networks have promising properties with regard to scalability and performance, while still offering the benefits of the P2P approach with regard to resiliency.

1 Introduction

Recent work has been dedicated to use P2P Networks as the platform for application layer multicast, content distribution, file sharing and so forth [12]. In general a P2P Network is formed by interconnecting end-systems (i.e. the peers), where each link in the resulting P2P topology is corresponding to an IP-layer path. In current P2P research two main trends for organising the nodes into an overlay network have been established, namely using *Structured* or *Unstructured* overlay topologies. While in a *Structured* P2P network the topological properties of the overlay in combination with an addressing scheme are used to establish a platform with provable communication characteristics, *Unstructured* P2P networks rely on statistical properties and have proven to have a global scale phenomena. In a hierarchical P2P network the overlay topology is divided into two tiers, the lower representing *normal* peers and the higher representing SuperPeers. The SuperPeers are selected based on metrics like connectivity, CPU capacity, reliability as well as other issues like security, privacy and trust.

This paper, which is an extension of [7], discusses *Lightweight SuperPeer Topologies* (LST) for hierarchical P2P Networks. The main target is to explore a class of P2P Networks between *Structured* and *Unstructured*. LST are designed to have a low management complexity as well as overhead. In this paper we evaluate *Yao-Graphs* [16] as the first candidate for LST. While this type of graphs has already been studied in the area of MANETs [15], to the best of our knowledge there is no study in the area of P2P-Networks. To use *Yao-Graphs*, a *mapping function* is necessary to accurately embed nodes from the underlying network into a geometric space [8]. In this paper we are using the *Highways* [11] proximity clustering scheme, introduced by one of the authors, to assign accurate geometric co-ordinates to nodes.

Yao-Graphs are interesting from the perspective of P2P Networks, since these graphs can be efficiently computed and maintained in a distributed manner, allowing fast recovery from node failures. Moreover, this graphs contain the *Euclidean Minimum Spanning Tree* (EMST), while their structure is relatively lightweight compared to other geometric structures containing the EMST like *Delaunay-Triangulations* [10]. As one expected result, the application-layer multicast of multimedia data or search requests between SuperPeers is about to perform well because of the EMST property and the network-aware mapping of SuperPeers to a geometric space. To evaluate the proposed geometric model, we perform experiments based on data derived from the planetary-scale PlanetLab testbed [1].

The remainder of the paper is organized as follows: Section 2 provides background information about EMST's and *Yao-Graphs*; Section 3 describes the used LST principle and its network-aware construction using *Highways*; Section 4 provides an evaluation of the model based on PlanetLab measurements; and Section 5 concludes our results.

2 Theoretical Background

During this paper, the model for a computer network is a weighted graph $G(V, E)$ with nonzero positive edge costs, where V denotes the set of vertices and E the set

---

Parts of this research have been supported by the E-NEXT project and Microsoft Research.

of edges between the vertices. A minimum (cost) spanning tree (MST) of a connected graph G is defined as the tree connecting all the vertices of G by the cheapest subset of edges (with regard to the sum of edge costs). While in *Unstructed* P2P networks flooding is a common method for the distribution of search request, the main target of LST is to optimise the SuperPeer Topology for Multicast distribution. More specifically, we target in providing an efficient algorithm for a distributed approximation of the MST using the principle of Euclidean Minimum Spanning Trees.

2.1 Euclidean Minimum Spanning Trees

In general an EMST can be interpreted as the geometric counterpart of the MST of a fully meshed graph. Given a fully meshed graph $G(V, E)$, where V corresponds to a set of points in the euclidean space R^2, and E to the set of edges with weight corresponding to the Euclidean length of an edge, the EMST and the MST of G are identical.

This is of interest, since calculating the MST of a weighted graph containing n nodes, connected by m edges requires $O(m \log n)$ time, using Kruskal's algorithm [5]. In case a two-dimensional geometric representation of the graph is available, the calculation of the EMST can be done in $O(n \log n)$ time [14].

2.2 Yao-Graphs

A graph structure having interesting properties with regard to the EMST are *Yao-Graphs* [16]. Given a set of points in R^2, the basic principle of a *Yao-Graph* is *cutting the space around each point into sectors of equal angle θ (e.g. $\theta < \pi/3$) and connecting the point to its closest neighbour (with regard to euclidean distance) in each of this sectors*. Figure 1 shows an example of an *undirected Yao-Graph*. The following result presented in [16], together with the above observations about EMST's motivates our usage of these graphs.

Lemma 1 (*Yao-Graphs*) *Let P be a point set in $\mathbb{R}^2$. Let G be the undirected* Yao-Graphs *for P with $\theta < \pi/3$. Then, the Euclidean minimum spanning tree of P is a subgraph of the* Yao-Graphs *G.*

Moreover these graphs have been the first solution to break the $O(n^2)$ time complexity barrier for calculating the EMST in a connected graph with n nodes [16]. With the aim to minimise management overhead, the above described construction principle results in a directed *Yao-Graph*, which still includes an approximation of the EMST for a set of points following the observations provided in [6].

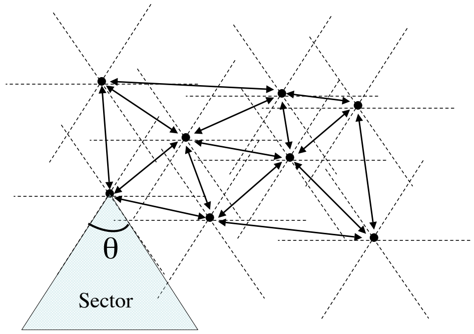


Figure 1. *Undirected Yao-Graph*

3 Lightweight SuperPeer Topologies (LST)

The LST scheme, as illustrated in Figure 2 is based on three main steps:

1. **SuperPeer Estimation:** Estimate if a peer willing to join a LST based P2P-Network is a SuperPeer candidate. The estimation should include the following two metrics
   - The peer should have *enough* resources to serve other peers.
   - The peer should be reliable in the sense that it is not joining and leaving the P2P Network frequently.

   Further trust and security incentives are to be considered as central.
2. **Embedding and Clustering:** Assign a target cluster and a geometric co-ordinate to the peer using *Highways*.
3. **Integration into P2P overlay:** Integrate the new peer by updating the SuperPeer *Yao-Graph* or assigning it to a SuperPeer capable to serve an additional client.

Our main focus in this paper are step two and three of the LST construction. We will therefor continue with an overview description of the used *Highways* principle. For a deeper discussion of *Highways* we refer to [11].

3.1 The *Highways* principle

The network-aware LST construction is based on ideas extend from *Highways*, a landmark-based distance estimation and proximity clustering scheme. The principle of

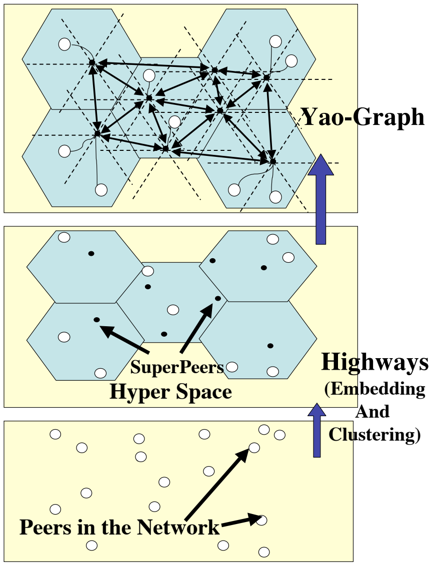


Figure 2. *LST construction*

landmark-based distance estimation is to estimate the distance between a set of nodes in a network by just measuring the distance of each node to $d + 1$ landmark nodes. By conceiving the results as the components of a vector, each node is embedded into a d-dimensional Euclidean space. The distances between two embedded network nodes within each cluster are now estimated by computing the Euclidean distance between their respective co-ordinate vectors in the Euclidean Space.

The *Highways* scheme achieves a high estimation precision by combining classical *landmark based* distance estimation with *Principle Component Analysis* (PCA) and clustering. The clustering method used, adopts a simplistic approach of K-means clustering developed by MacQueen [13]. The algorithm clusters nodes in the network by assigning each node to the cluster having the nearest centroid (mean) based on RTT distance. Depending on the target dimension d for the embedding, the total number of Landmarks in each cluster has to be **at least** $d + 1$.

Since *Highways* is using a PCA technique to minimise the error of the distance estimation, it is usually required to recalculate the co-ordinates of all peers in a cluster in the case a landmark left. Because of this fact, we select SuperPeers as Landmarks. This result is in $d + 1$ SuperPeers per cluster, to be chosen according to the criteria mentioned in the previous section. The steps that are necessary to maintain the SuperPeer topology in the case of a leaving SuperPeer are described in section 3.3. After the calculation of geometric co-ordinates for the SuperPeers, it is possible to exploit *Yao-Graphs* to archive a global characteristic of the SuperPeer topology (i.e. the EMST property) by applying a comparable simple local construction algorithm.

3.2 Topology Construction and Routing

Using a Yao-Graph construction as described in section 2, every node has a bounded Out-degree but possibly a high In-degree, if for instance a special node is the nearest node of many other node's. To overcome this problem eventually resulting in exhausting a node, *directed* or *Sparsified Yao-Graphs* can be considered. A sparsified *Yao-Graph* is a *Yao-Graph* where in case the In-Degree of a sector exceeds one, only the shortest incoming edge is accepted. For the construction of a LST topology based on a *Yao-Graph* we implemented the following algorithm:

1. To be able to join the P2P network, a new peer has to know at least one node which is already a member, and can be used for a standard overlay join procedure [10].

2. As a new part of the join procedure, an overlay network address in the form of a geometric co-ordinate is calculated, and is assigned to the joining node.

3. The decision is made if the new node is about to become a SuperPeer, based on metrics like connectivity, reliability etc.

    (a) In the case the node is a SuperPeer candidate, the co-ordinate of the node is used to guide the new SuperPeer through the SuperPeer topology using a geometric routing principle (e.g. compass routing [9]). As soon as the new SuperPeer has reached its destination, the SuperPeer topology is locally updated, by inserting the new node and updating the local *Yao-Graph* neighbour relations.

    (b) In the case the new node is not a SuperPeer candidate, it is guided through the SuperPeer topology until it has reached the SuperPeer with the geometrically closest co-ordinate that is capable of accepting a further client.

As already mentioned, one possible routing principle to be used for LST is compass routing [9], and the broadcasting of search request between the SuperPeers can be realised using compass routing in combination with *Reverse Path Forwarding*.

3.3 LST Maintenance

Due to the local construction principle of LST and the clustering principle used by *Highways*, the impact of churn to the SuperPeer topology can be kept small. To be able to detect Peer and SuperPeer failures in LST e.g. a *heartbeat* principle can be used, where direct neighbours in the P2P Network are sending periodically *alive* messages to each other. With regard to LST maintenance we have to distinguish two main cases:

1. **A *normal* peer is leaving the network:** The corresponding SuperPeer frees the resources corresponding to its connection to the peer after observing a missing heartbeat.

2. **A SuperPeer is leaving the network:** In the case a SuperPeer leaves the network, an adjacent SuperPeer will notice this failure (e.g. through a missing heartbeat) and triggers a *local* repair procedure, which updates the local neighbour relations to re-establish the *Yao-Graph*. To avoid that all the peers of a SuperPeer have to rejoin the P2P-Network, a *normal* peer should maintain spare connections to more than one SuperPeer in the network. In the case the SuperPeer was used as a Landmark, the co-ordinates of all nodes in the cluster need to be recomputed. Since the *Yao-Graph* structure can be repaired locally before this recomputation occurs, it is possible to co-ordinate the required steps using the SuperPeer topology for communication.

4 Testbed Experiments and Evaluations

To evaluate LST, we conducted simulation experiments based on measurement data obtained from the global-scale network testbed PlanetLab [1]. Since in many cases multiple nodes reside within each site of the PlanetLab testbed, we have chosen one peer per site as an representative SuperPeer. We performed experiments simulating four different SuperPeer layer sizes using *Yao-Graph* topologies, which contain 20, 40, 60 and 81 SuperPeers. A resultant SuperPeer topology with 81 PlanetLab sites is shown in Figure 3. Based on the variation in the size of the SuperPeer layer, we will examine the trends of the measurement results and discuss them in the next sub-section. In the experiments conducted, we first simulated a join procedure using the *Highways* to assign two dimensional geometric co-ordinates. The variant of the K-means clustering algorithm used for the experiments consist of three steps:

1. Partition the selected set of nodes into K initial clusters. We determine K initial centroids (seed points) first by randomly choosing K nodes' locations to act as the K cluster centers. For our experiment, we use $K = 3$.

2. Proceed through the list of overlay nodes in the network, assigning an overlay node to the specific cluster whose centroid (mean) is the shortest in terms of RTT distance. Recomputation of the centroid is done for the cluster having gained a new node and for the cluster losing the node.

3. Repeat Step 2 until no more assignments take place.

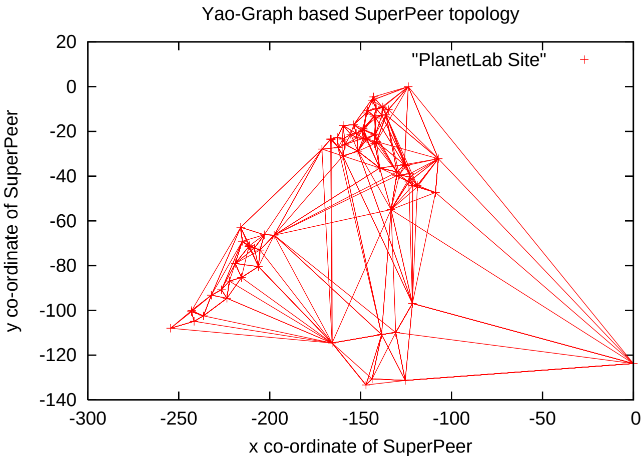
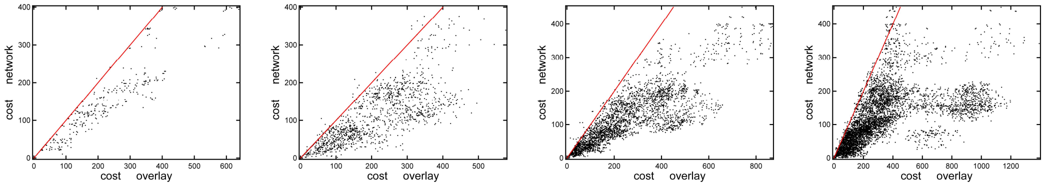


Figure 3. *Yao-Graph containing 81 SuperPeers (PlanetLab sites)*

The method attempts to minimize the sum of the within-cluster variances. The strength of the simplistic K-means clustering is its relatively efficiency of $O(tKn)$, where n is the total number of nodes, K is the number of clusters, and t is the number of iterations. Normally, $K, t \ll n$. The K-means clustering algorithm often terminates at a local optimum. The global optimum maybe found using techniques such as deterministic annealing and generic algorithms. However, one of the weakness of this straightforward algorithm requires the K, the number of clusters, to be specified in advance.

Once the SuperPeers in each corresponding experimental sets have been integrated into the topologies, we used the following metrics to estimate the quality of the P2P structure:

1. **Diameter:** The *Diameter* of the topology is the longest shortest-path length (in terms of hops) between any pair of nodes in the system. For each of the different SuperPeer set sizes 20, 40, 60 and 81, we calculated the *Diameter* of the resulting *Yao-Graph*, as well as its average and standard deviation. The results of the performed measurements are presented in Figure 4. On the X-axis the *Diameter* is measured, the Y-axis is used to display the probability density function (pdf) of the *Diameter*.



(a) 20 SuperPeers (b) 40 SuperPeers (c) 60 SuperPeers (d) 81 SuperPeers

Figure 5. *Overlay Performance*

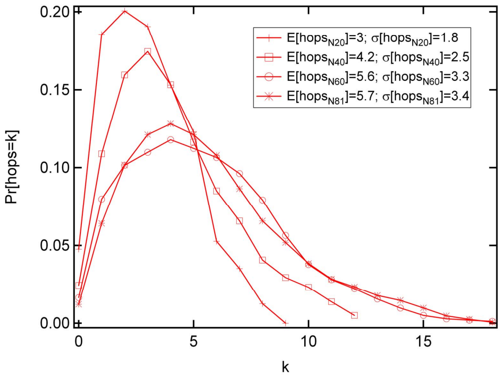


Figure 4. *Diameter and Average Number of Hops*

2. **Overlay Performance:** The Overlay Performance is measured by a comparison of the network cost of direct IP communication between two SuperPeers, utilizing the underlying network, and the cost of using the LST topology, as the platform for SuperPeer to SuperPeer communication. The used LST routing algorithm in the experiment was compass routing [9]. The results are presented in Figure 5. The cost of LST communication is displayed on the X-axis while Network cost are displayed using the Y-axis.

3. **In/Out Degree of a Node:** In/Out degree of a node demotes the number of In/Out SuperPeer connections or neighbours that must be maintained by a single SuperPeer. The average numbers for SuperPeer In/Out Degree are also computed for each of the test topologies. Since the used *Yao-Graphs* in our experiments are directed *Yao-Graphs*, the distribution of the different In-Degree and Out-Degree of different sets of SuperPeer are shown in Figure 6. The X-axis shows the node degree d, and the Y-axis the probability density function (pdf) of the node degrees in our constructed topologies.

4.1 Evaluation of Results

The experiment results for *Diameter* are reasonable small for two-dimensional LST. *Diameter* represents an upper bound on the search path length which is the primary factor for the scalability. In our measurements, the average Diameter ranges from 3 to 5.7 and its standard deviation ranges from 1.8 to 3.4. An important part of future work will be to study the impact of a higher dimensional geometric target space to the connectivity of the LST connectivity, and to examine the diameter.

The positive impact of the network-aware construction of LST with regard to cost of overlay communication is observable from Figure 5. Confirming the observation presented in [2] in a very few of the cases, using the LST for communication out-performs the direct IP-based communications in the underlying network.

The measurement results for the In-Degree and Out-Degree in each of the experimental sets of the PlanetLab sites are balanced. A small average degree is an indicator for a low *link stress* (as defined in [4]) in the case of the Multicasting of search queries using the LST.

A low degree variance results in better load balancing and an almost regular topology. The average In-Degree for all the experiments was equal to the average Out-Degree, but In-Degree shows larger variance. The average Degree ranges from 3.7 to 5.2. All the SuperPeer have a bounded Out-Degree of about 6 in our case, and a high percentage does not have a high In-Degree. In general, the In-Degree and Out-Degree of a SuperPeer can be bounded through the dimension of the geometric target space and/or by using the principle of the *Sparsified Yao-Graph* as described in section 3.

5 Conclusion and Future Work

In this paper, we have described an alternative proposal for Lightweight Structured SuperPeer Topologies (LST) for hierarchical P2P networks. The LST scheme presented is based on the geometric principle of *Yao-Graphs* [16]

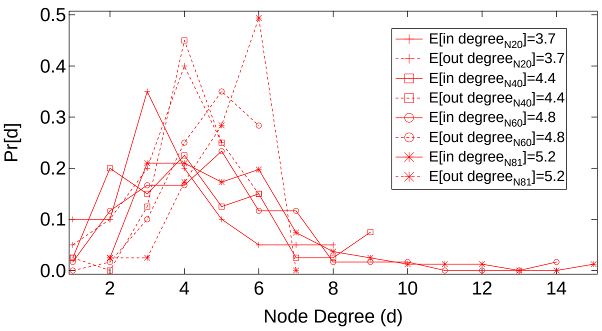


Figure 6. *In-Degree and Out-Degree for 20, 40, 60 and 81 SuperPeers Sites*

in combination with the *Highways* [11] proximity clustering scheme for the assignment of accurate geometric co-ordinates. We attempt to show an initial evaluation of LST based on PlanetLab measurements. The key intent for our geometric approach is that a geometric representation of a communication network, once established, offers a new perspective on a number of problems. For example, computing a minimum spanning tree of a weighted undirected graph of n nodes requires $O(n^2)$ time, in general, but only $O(n \log n)$ for the points in a two-dimensional geometric space [14]. One of the advantages using a *Yao-Graphs* in this context, is the possibility to archive a global characteristic of the SuperPeer topology by applying a comparable simple local construction algorithm. The diameter and average number of hops of the *Yao Graph* based topology are reasonable, and the In-Degree and Out-Degree of a SuperPeer can be bounded through the dimension of the geometric target space and/or by using the principle of the *Sparsified Yao-Graph*. An important part of future work will be to study the impact of higher dimensions to the proposed scheme and evaluating different schemes for the network-aware assignment of geometric co-ordinates to node in a communication network. Further a comparison with *Random-Graph* based topologies and a comparison of the application layer multicast performance of our scheme with Structured P2P overlay multicasting such as in Scribe [3].

6 Acknowledgements

The authors would like to thank the anonymous reviewers for their insightful comments which helped to improve the paper.

References

[1] PlanetLab home page. http://www.planet-lab.org.

[2] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proceedings of Symposium on Operating Systems Principles*, pages 131–145, 2001.

[3] M. Castro, P. Druschel, A. Kermarrec, and A. Rowstron. Scalable application-level anycast for highly dynamic groups. In *5th International Workshop on Networked Group Communications*, 2003.

[4] Y.-H. Chu, S. G. Rao, and H. Zhang. A case for end system multicast. In *Measurement and Modeling of Computer Systems*, pages 1–12, 2000.

[5] T. Cormen, C. Leiserson, R. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, London, UK, 2 edition, 2001.

[6] A. Czumaj, F. Ergun, L. Fortnow, A. Magen, I. Newman, R. Rubinfeld, and C. Sohler. Sublinear-time approximation of euclidean minimum spanning tree. In *14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 813–822, 2003.

[7] M. Kleis, E. K. Lua, and X. Zhou. A case for lightweight superpeer topologies. In *KiVS Kurzbeiträge und Workshop*, pages 185–188, 2005.

[8] M. Kleis and X. Zhou. A placement scheme for peer-to-peer networks based on principles from geometry. In *Proceedings of the IEEE Fourth International Conference on Peer-to-Peer Computing (P2P'04)*, pages 134–141, 2004.

[9] E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *In Proceedings of the 11th Canadian Conference on Computational Geometry*, pages 51–54, Vancouver, August 1999.

[10] J. Liebeherr, M. Nahas, and W. Si. Application-layer multicast with delaunay triangulations. In *Proceedings of IEEE Globecom 2001, Global Internet Symposium*, Nov. 2001.

[11] E. K. Lua, J. Crowcroft, and M. Pias. Highways: Proximity clustering for scalable peer-to-peer network. In *Proceedings of the IEEE Fourth International Conference on Peer-to-Peer Computing (P2P'04)*, pages 266–267, 2004.

[12] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer network schemes. *In Submission to IEEE Communications Tutorials and Surveys*, March 2004.

[13] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. *5th Berkeley Symposium on Mathematical Statistics and Probability*, 1:281–297, 1967.

[14] M. I. Shamos and D. Hoey. Closest-point problems. In *Proceedings of 16th IEEE Symposium on Foundations of Computer Science*, pages 151–162, 1975.

[15] Y. Wang and X.-Y. Li. Distributed spanner with bounded degree for wireless ad hoc networks. In *IPDPS '02: Proceedings of the 16th International Parallel and Distributed Processing Symposium*, page 120, Washington, DC, USA, 2002.

[16] A. C.-C. Yao. On constructing minimum spanning trees in k-dimensional space and related problems. *SIAM Journal on Computing*, 11:721–736, 1982.