Hierarchical Geometric Overlay Multicast Network

Eng Keong Lua* Xiaoming Zhou† Jon Crowcroft* Piet Van Mieghem†
* University of Cambridge, Computer Laboratory
Email: {eng.keong-lua,jon.crowcroft}@cl.cam.ac.uk
†Delft University of Technology, Faculty of Electrical Engineering, Mathematics, and Computer Science
Email: {x.zhou,p.vanmieghem}@ewi.tudelft.nl

***Abstract*— In this poster proposal, we present *Bos*: a hierarchical geometric overlay multicast network that is built on a 2-tier hierarchical architecture called Lightweight SuperPeer Topologies (LST) [1], [2]. *Bos* makes use of the geometric connectivity and Minimum Spanning Trees (MST) properties of the LST overlay network to provide efficient overlay multicast. We evaluate *Bos*'s performance using the large-scale network models that were used in Scribe [3] and SplitStream [4]. The results show that *Bos* performs reasonably well in large size networks with reasonable link and node stress.**

I. The LST Overlay Network

The construction of 2-tier hierarchical architecture of LST overlay network consists of *three* key steps: **Step 1: SuperPeer election.** When a new overlay node joins the LST overlay network, the following criteria are evaluated to determine whether this overlay node will be elected as a SuperPeer or normal Peer: A) The SuperPeer should have *enough* resources to serve other SuperPeers and peers. B) The SuperPeer should be *reliable or stable* and it is not joining and leaving the LST overlay network very frequently. With the above criteria imposed, the SuperPeers layer will consist of SuperPeers acting as backbone high-speed gateway for the peers in the peers layer. The list of SuperPeers are being classified as the list of landmark nodes (lighthouses) for the procedure in Step 2. **Step 2: *Highways* [5].** *Highways* is an overlay network control plane service [5] that performs scalable network embedding [6] to map overlay nodes in metric space onto geometric points in geometric space and assign geometric coordinates to the overlay nodes to represent their geometric position for the construction of the geometric overlay network. If accurate, such techniques would allow us to predict Internet distances without extensive measurements. We use landmark-based and Singular Value Decomposition (SVD) embedding techniques for low-dimensional network embedding. Firstly, Round-Trip-Time's (RTT's) measurements of each overlay node to at least $d+1$ landmark nodes (SuperPeers) are performed for embedding into d-dimensional geometric space. Network superspace embedding embeds the whole set of overlay nodes in the system as one large set into *Global* geometric space while subspace embedding embeds all small partitioned clusters of overlay nodes into *Local* geometric space. The rationale for performing network subspace embedding arises from the scalability (meta-) metric observations in [6], subspace embeddings into Euclidean space of various partitioned clusters of overlay nodes achieve *better* accuracy in geometric distance estimation. Using the overlay nodes' *Global* geometric position information, all overlay nodes in the overlay network are partitioned into clusters by adopting a simplistic approach of the K-means method (we use $K=3$). Network subspace embedding is done to overlay nodes within these clusters. Therefore, all overlay nodes will have both *Global* and *Local* geometric position information. The local geometric position information helps to provide a more accurate geometric distance estimation among overlay nodes in the cluster while the global geometric position estimates the geometric distances between overlay nodes in different clusters. We recognize the fact that slim possibility of inaccuracy in overlay nodes' rank ordering through geometric distance estimation may happen. In order to mitigate this, from the perspective of each node, a sanitary check is done for the list of *closest* $g=10$ nodes derived from its geometric distances. That is, a comparison is performed with its measured RTTs and re-ordering of the list of *closest* nodes is done if distance ordering errors were found. This sanitary check helps in ensuring the list of *closest* nodes is identified. **Step 3: SuperPeers and Peers Topology Construction.** In the SuperPeers layer, we use *Yao-Graphs* [7] to construct the overlay network connectivity among the SuperPeers by using their geometric position information and estimated geometric distances with other SuperPeers as computed from Step 2. Since the geometric space around the SuperPeer is cut into *six* sectors of equal angle $\theta < \pi/3$), every SuperPeer choose the *six closest* SuperPeers in terms of their geometric distances to connect to. These **SuperPeer-SuperPeer Yao-Graphs routes** serve as the reliable high-bandwidth backbone network connectivity. In the Peers layer, Peers are directly connected to the first *closest* SuperPeers that are capable of serving an additional Peer and this connectivity is called the **Peer-SuperPeer 1-Hop route**. Among the Peers being served by their closest SuperPeer, direct connectivity between these Peers can be established if there exists a shortcut route between the Peers. That is, a **Peer-Peer Shortcut route** is established between two Peers belonging to a SuperPeer, if the direct connectivity between these two Peers is the shortest route compared to their Peer-SuperPeer 1-Hop routes.

II. *Bos*

Bos is an overlay multicast network that is built on LST overlay network and uses tree topology for multicasting. The key performance of *Bos* is the system scalability, service availability and multicast efficiency with the available service bandwidth. The multicast tree is formed by the union of

source-destination routes in the geometric space. The overlay route from the source to any other overlay node is derived from the geometric overlay routing algorithm (i.e. an adapted version of the combination of greedy and face routing) implemented in LST overlay network. *Bos* builds geometric shortest-route multicast distribution trees among the SuperPeers in the SuperPeers layer of the LST overlay network using an adapted version of Reverse Path Forwarding (RPF) algorithm. This multicast tree construction method ensures that each SuperPeer group member of the multicast group receives a multicast message on the correct incoming connection interface. When a SuperPeer source (which is the tree's root) sends a multicast message to its direct neighnoring SuperPeers, those neighboring SuperPeers further in turn forward the message to their neighboring SuperPeers that belong to the multicast group. If a neighboring SuperPeer does not belong to the multicast group and there is no other group members, it returns a **Prune** message to the SuperPeer root. The SuperPeer root does not forward subsequent messages to neighboring SuperPeers who respond with the **Prune** messages. The adapted RPF algorithm allows a SuperPeer group member to accept a multicast message only on the connection interface from which the SuperPeer group member would send a unicast message to the SuperPeer root. From the SuperPeer root to all other SuperPeer group members, we build a multicast distribution tree rooted at the SuperPeer root such that each SuperPeer group member has shortest geometric overlay route back to the SuperPeer root. As the geometric overlay route between a SuperPeer root and another SuperPeer group member is symmetrical, the adpated RPF algorithm constructs the shortest geometric route tree on the *Yao-Graphs* topologies at the SuperPeers layer which could exhibit MST characteristics. Any SuperPeer creates a multicast group with a random *groupID* may become a SuperPeer root, and the multicast tree is constructed by the union of the LST geometric routes from each group member to the SuperPeer root. During multicasting, messages are flooded down all the branches of the multicast distribution tree. Therefore, *Bos* builds a multicast tree per application group. Multiple multicast groups can be formed at the SuperPeers layer of the LST overlay network. The multicast tree constructed at the upper SuperPeers layer provides the reliable high-bandwidth backbone multicast connectivity for the lower layer's Peers who basically connect to their SuperPeers using their direct Peer-SuperPeer 1-Hop route for overlay multicasting. If the Peer wishes to be the member of the multicast tree, the SuperPeer serving this Peer will be responsible to become the member of the multicast tree. All multicast communications are done using standard TCP for the reliability and rely on the flow maintenance control in LST overlay network to repair the overlay topology when overlay node fails and multicast group membership management operations are invoked.

III. Implementation and Experiments

The simulation experiments are implemented in the large global-scale network testbed that were used by the Scribe [3] and SplitStream [4]. The large-scale network testbed are generated by Georgia Tech (GT) random graph generator. The hierarchical transit-stub model containing 5050 routers: There are 10 transit domains at the top level with an average of 5 routers in each. Each transit router has an average of 10 stub domains attached, and each stub has an average of 10 routers. There are $100,000$ end-system nodes that were randomly assigned to routers in the core with uniform probability. Each end-system node was directly attached by a LAN link to its assigned router. There are 10 different network models using the same parameters but different random seeds. For a large-scale network of this size, the only feasible way is to develop a customized network simulator for our experiments and well-known network simulator such as *ns*-2, would not be able to handle the large size of the network testbed and the self-organization features of the overlay networks. Our network simulator is developed to model the propagation delay on the physical links. The delay of each LAN link was set to 1 ms and the average delay of core links was 40.7 ms. Similar to the work of Scribe/SplitStream, the simulator does not model queuing delay, packet losses, or any cross network traffic because modeling of such parameters would prevent the simulation of **large** networks. Cross network traffic are also not modeled in our experiments. To examine whether *Bos* is an efficient infrastructure supporting multiple concurrent applications with varying requirements, we use the same environment as Scribe/SpliStream and perform experiments using a large number of groups with a wide range of group sizes. A Zipf-like distribution for the group sizes is adopted. Groups are ranked by size. The size of the group with group rank r is given by $gsize(r) = \lfloor Nr^{-1.25} + 0.5 \rfloor$, where N is the total number of overlay nodes. In each network model, the total number of group ranks was fixed at 150 and the number of overlay nodes (N) was fixed at $100,000$, which were the numbers being simulated (as in Scribe/SplitStream). In each group, we choose 10% of the total number of overlay nodes to be the SuperPeers based on the election criteria described in section I. The reason for the choice was derived from the recent study in [8] which states that there are approximately 10% of the overlay nodes have high capacity, and they exhibit stability and reliable connectivity in the overlay network. We run our simulation system on these 10 large-scale network models. We generate 2-dimensional geometric coordinates for all the nodes in the system. The performance results shown in all the figures in this paper are the **average values** over the 10 large-scale network models. Experiments show the following preliminary performance results. **Link Stress:** Link stress is the number of duplicate packets carried by each network physical link incurred in overlay multicast. A node with high link stress can be easily exhausted. Figure 1 shows the distribution of the mean link stress for the *Bos* overlay multicast network, when a message is multicast in each of the 1500 groups. The results show that most links have low stress in the *Bos* overlay multicast network. The average link stress is 4.4, with standard deviation 5.6. This means that the average link stress induced by the LST overlay network is approximately 4.4 times that for an IP multicast on the same experiment. Our result is acceptable but moderately higher than the result of Scribe, which generates 3.4 times link stress than that of IP multicast. The average link stress in SplitStream is close to the average stress in links used by IP multicast

(28% worse). The significance lies in the tail of the result plot. Only one link has a stress of 100 and a relatively small number of links have a stress above 20. Our result indicates that we have maximum link stress of 100 which is much smaller than the maximum link stress published in Scribe which is 4031. However, the Scribe results are generated for 100,000 nodes in the overlay network whereas our results are derived from 10,000 SuperPeers at the SuperPeers layer and the rest of 90,000 peers are connected directly to the SuperPeers. In general, our results show a low average link stress in the *Bos* overlay multicast network. **Node Stress:** Node stress quantifies the load on nodes which is equivalent to the number of messages that it receives. We measured the node stress by counting the number of nodes in each node's routing table and the number of messages received by each node when members join the groups. Figure 2 shows the distribution of mean node stress for each group. For all 1500 groups, the mean node stress are in the range between 5 to 8. Our experimental results show that the maximum node stress is 110. The results suggest that in *Bos*, end nodes just need to forward multicast messages only to a small number of other nodes: this is helpful to achieve **scalability**. It has a comparable average node stress with that published in Scribe (6.2), which may suggest that *Bos* overlay multicast network is efficient in spreading data over all nodes. The node stress of each SplitStream node is published to be equal to its desired Indegree, and this enables nodes in SplitStream with enough bandwidth to participate in the system. Depending on the configuration of SplitStream, the desired Indegree is typically set to 16 and all nodes will have Indegree of 16. To evaluate the underlay routing, we also calculate the **average underlay physical node degree** that measure the average number of underlay multicast connectivity of nodes in each group, and show the distribution of mean underlay physical node degree for each group in Figure 3. It shows that for all 1500 groups, the mean underlay node degrees are relatively small (between 3 to 5). These results suggest that in the *Bos* overlay multicast network, end underlay nodes just need to forward multicast messages only to a small number of other nodes.

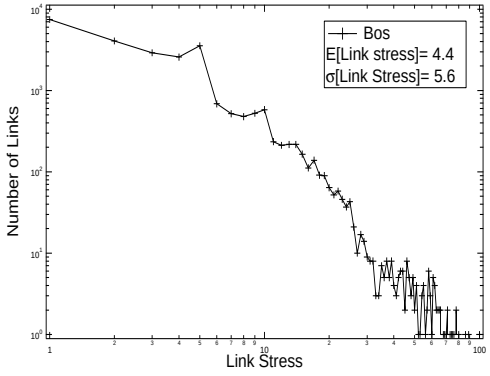

Fig. 1. Distribution of Mean Link Stress in *Bos* Hierarchical Geometric Overlay Multicast Network, Group Rank 1 to 150

IV. Acknowledgements

The authors would like to thank both Miguel Castro and Manuel Costa (Microsoft Research, Cambridge) for their discussion on the large-scale network models provided, Michael Kleis (Fraunhofer Institute FOKUS) for his partial contribution on the initial part of the work and Michael Smirnov (Fraunhofer Institute FOKUS) for his valuable comments. This work has been partly supported by the European Union under the E-NEXT NoE FP6-506869. Eng Keong Lua is sponsored by Microsoft Research and Xiaoming Zhou is supported by the NWO SAID Project.

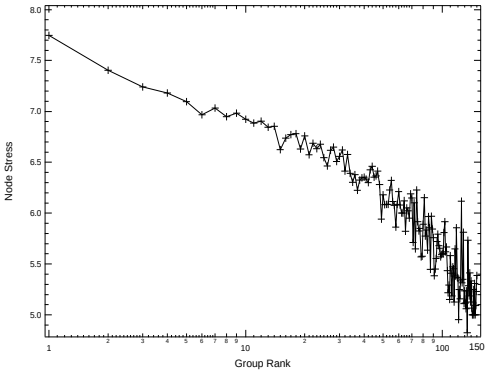

Fig. 2. Distribution of Mean Node Stress in *Bos* Hierarchical Geometric Overlay Multicast Network, Group Rank 1 to 150

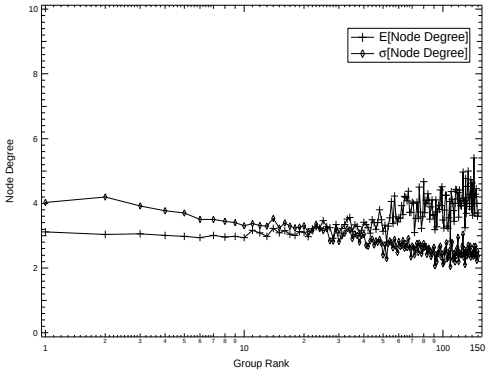

Fig. 3. Distribution of Mean Underlay Node Degree in *Bos* Hierarchical Geometric Overlay Multicast Network, Group Rank 1 to 150

References

[1] M. Kleis, E. K. Lua, and X. Zhou, "A case for lightweight superpeer topologies." in *KiVS Kurzbeiträge und Workshop*, 2005, pp. 185–188.

[2] ——, "Hierarchical peer-to-peer networks using lightweight superpeer topologies," in *Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC 2005)*, La Manga del Mar Menor, Cartagena, Spain, June 27-30 2005.

[3] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron, "Scribe: A large-scale and decentralized application-level multicast infrastructure," *IEEE Journal on Selected Areas in Communication (JSAC)*, vol. 20, no. 8, Oct. 2002.

[4] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "Splitstream: High-bandwidth multicast in a cooperative environment," in *19th ACM Symposium on Operating Systems Principles (SOSP'03)*, Oct. 2003.

[5] 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)*, 2004, pp. 266–267.

[6] E. K. Lua, T. Griffin, M. Pias, H. Zheng, and J. Crowcroft, "On the accuracy of embeddings for internet coordinate systems," in *Proceedings of ACM SIGCOMM-USENIX IMC 2005*, October 19-21 2005.

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

[8] L. Plissonneau, J.-L. Costeux, and P. Brown, "Analysis of peer-to-peer traffic on adsl," in *Passive and Active Measurement Workshop 2005 (PAM 2005)*, March 31 to April 1 2005.