

BCOM: Best Constraints for Overlay network Maintainance

T.VenkataRathnam, S.Vikramphaneendra.

Dept. of Computer Science Engineering Madanapalle Institute of Technology and Sciences (MITS) ,
Madanapalle, India

Abstract—In a overlay network for routing or peer-to-peer file sharing, each node must select a fixed number of overlay neighbors for routing traffic. A selfish node entering such network would select adjacent nodes so as to reduce the weighted sum of expected access costs to all its destinations. Previous work on selfish neighbour selection has built intuition with simple models where edges are undirected, access costs are modeled by hop-counts, and nodes have potentially unbounded degrees. However , in practice, important constraints not captured by these models lead to richer games with substantively and fundamentally different outcomes.

To improve we impose no bounds on peer degrees so that creation of overlay structures that require $O(n)$ instead of $O(n^2)$ link monitoring overhead. in that regard, the central aspects of our model are bounded degree, directed edges, on-uniform preference vectors, and representative distance functions. Our first technical contribution within this model is to express a node's "best response" wiring strategy as a k -median problem on asymmetric distance, and use this observation to obtain pure Nash equilibria through iterative best response walks via local search we design, implement, and deploy EGOIST, a prototype overlay routing network built around best response wiring strategies. EGOIST serves as a building block for the construction of efficient and scalable overlay applications consisting of (potentially) selfish nodes. we demonstrate that the performance of EGOIST approaches that of a (theoretically-optimal) full-mesh topology, while achieving superior scalability, requiring link announcements proportional to nk compared to n^2 for a full mesh topology. Our experimental results show that EGOIST remains highly effective under significant churn and incurs minimal overhead.

Index Terms—Overlay networks, overlay routing, selfish neighbor selection, network formation

I. INTRODUCTION

As the age of information has dawned upon us, it has become imperative that efficient information exchange methodologies be studied. While traditional network models certainly broadened the knowledge and understanding of information exchange, new and emerging paradigms require a different approach. Overlay networks, which are logical networks over an existing network, are becoming more common. Overlay networks are used for a variety of popular applications including routing , content distribution, peer-to-peer (P2P) file sharing and streaming data-center applications and online multi-player games .

Let us consider the overlay network in figure 1 .Here four overlay nodes are placed randomly respectively for the LAN1,LAN2,LAN3,LAN4 respectively. The packet travels through the nodes R1,..R6.R1 is the source and the R4 is the destination and if suppose the route of the packet is

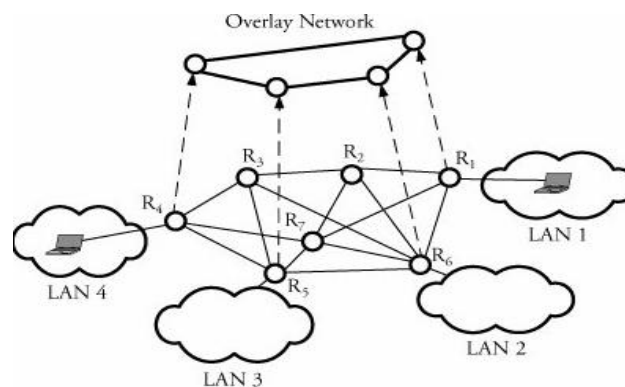


Figure 1:Simple Overlay Network

R1->R7->R6->R5->R4 and if a link failure is happened to occur between R7->R6 then overlay nodes are to deployed dynamically to route the same packet.

II. MODELS OF COMPUTING

A. Desktop Computing

During the early days of personal computers (PCs), the desktop was seen as the central computing tool. All the applications required by the user are provided in the desktop and when new applications are needed they have to be installed on his/her computer. Clearly, this model of computing becomes expensive and infeasible as the number of applications needed by the user grows. More importantly, this model does not allow any resource sharing between the users. These disadvantages meant that new models had to be designed.

B. Client-Server Computing

Client-server computing is a distributed model where two entities, the client and the server, communicate with each other according to some established protocol to perform certain tasks. Examples include (browser, web-server) where using the HTTP protocol the browser sends requests to a web-server and later displays the results, the X Window System (commonly known as X11) where typically a user's local display acts a server, and the like. Figure .2 shows an example of a client-server computing system. While this model has better resource utilization compared to desktop computing, the clients are not left with too much of freedom. In most cases, these systems do not allow any interactions between the clients. Moreover, in this model the server might be overburdened as it has to serve multiple clients. Though there exist solutions to deal with such problems, these require providing special purpose costly hardware. Other problems such as a single point-of-failure at the server also exist. What is needed is a model which allows resource sharing and also cost sharing.

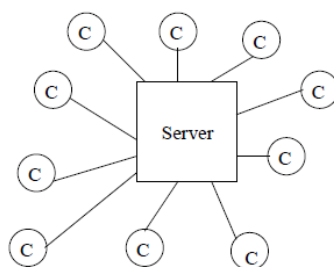


Figure 2: Client-Server model of computing where the server handles all the requests of the clients

C. Peer-to-peer Computing

The recent trend has been towards a model of computing which allows efficient sharing of resources. Also, there is a need to move away from client-server based computing and allow the clients to make some application-level decisions which they are best capable of. This is where the peer-to-peer model of computing enters the picture.

A peer-to-peer system is a self-organizing system of equal, autonomous entities (peers) which aims for the shared usage of distributed resources in a networked environment avoiding central services.

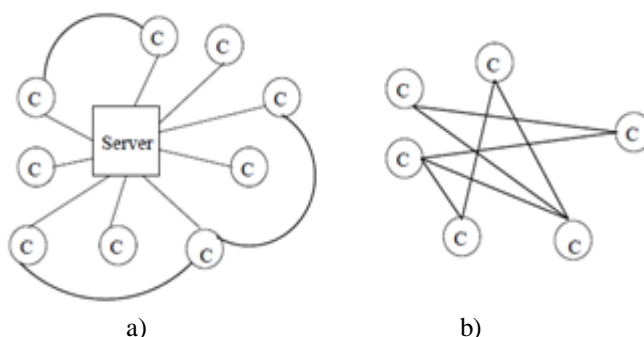


Figure 3. peer-to-peer system

Figure 3.a) shows a supervised peer-to-peer system where the server has certain limited functionality and clients (peers) are allowed to communicate with each other. The bold lines indicate the client-client communication links. Figure 3.b) shows a pure peer-to-peer system where there is no central server.

III. WHY LOGICAL NETWORKS?

In this section, we state the reasons that make overlay networks suitable for many application scenarios. Some of the benefits of using logical networks are that they provide flexibility, ease of implementation, easy customizability and adaptability, and incremental deployability. These advantages make logical networks a good choice for a lot of applications. To provide further justification, we look at examples such as provisioning special features, Virtual Private Networks (VPN's), and grid computing, that benefit from the above features. In the following discussion we view the Internet as the underlying network unless explicitly mentioned.

A. Provisioning Special Features

For many applications, designing logical networks has several advantages compared to relying on the underlying network. A logical network provides a certain degree of flexibility and ease of implementation that is not achievable relying on the underlying network. Consider providing Quality-of-Service (QoS) guarantees to Internet traffic which may be demanded by certain applications such as multimedia, or real time industrial applications. In the current Internet, there is no standard way to pass QoS information across routers. Also, intrinsically any solution to guaranteeing service quality would be a case of weak-link phenomenon where the quality guaranteed will be as weak as the guarantee of the worst link in a path. Moreover, various applications have different QoS requirements which make it difficult to capture in any single solution. Thus, there are serious obstacles to providing end-to-end QoS guarantees. Whether to let the underlying network, the Internet, to allow applications to demand QoS guarantees or to have the end-hosts deal with QoS guarantees is a hotly debated topic in Internet research forums such as the IETF (Internet Engineering Task Force). In this scenario, logical networks offer a solution as proposed. For example, sites requiring certain guarantees can form a logical network to sustain those guarantees without requiring any changes in the underlying network which might be prohibitively difficult for technical or economic reasons. IPv6 is a classic example of the difficulties involved in changing the underlying network operation. IPv6 (IP Version 6) is the new generation Internet protocol that is designed to address the limitations such as a small address space, lack of uniform QoS capabilities, and to increase efficiency and flexibility in the current version of the protocol IPv4. The deployment of IPv6 has encountered huge delays as it involves development and deployment of new software on devices that are connected to the network, and upgrading millions of routers on the Internet to use IPv6 instead of IPv4. When using logical networks such special protocols, or protocols implementing special features that depend on application specific knowledge, can be implemented without in any way burdening the underlying network. This approach also gives an additional ease of maintenance as updates or fixes to the protocols can be carried over with less effort.

Previous work on overlay network creation has focused on physical telecommunication networks and primarily the Internet. Overlay networks are substantially different which prompts us to consider the following overlay network model.

B. Overlay Network Model

We start by relaxing and modifying some of the central modeling assumptions of previous work[4],[7],[5]. In that regard, the central aspects of our model are:

Bounded Degree: Most protocols used for implementing overlay routing or content sharing impose hard constraints on the maximum number of overlay neighbors. For example, in popular versions of Bit Torrent a client may select up to 50 nodes from a neighbors' list provided by the *Tracker* of a particular torrent file[12]. In overlay routing systems [13], the number of immediate nodes has to be kept small so as to reduce the monitoring and reporting overhead imposed by the link-state routing protocol implemented at the overlay layer. Hard constraints on the number of first hop neighbors are also imposed in most P2P systems to address scalability issues, up-link fragmentation, and CPU consumption due to contention[14]. Motivated by these systems, we explicitly model such hard constraints on node degrees. Notice that in the prior studies cited above, node degrees were *implicitly bounded* (as opposed to *explicitly constrained*) by virtue of the trade-off between the additional cost of setting up more links and the decreased communication distance achieved through the addition of new links. We also note that some of these earlier network creation games were proposed in the context of physical communication networks[4],[7]. In such networks, the cost of acquiring a link is instrumental to the design and operation of a critical infrastructure. Such concerns do not apply in the case of overlay networks such as those we consider in this paper.

Directed Edges: Another important consideration in the settings we envision for our work relates to link directionality. Prior models have generally assumed bi-directional (undirected) links[4],[5],[6],[7],[8]. This is an acceptable assumption that fits naturally with the unbounded node degree assumption for models that target physical telecommunication networks because actual wire-line communication links are almost exclusively

bidirectional. In overlay settings we consider, this assumption needs to be relaxed since the fact that node v forwards traffic or requests to node u does not mean that node u may also forward traffic or requests to v . Undirected links are created by the establishment of two directed links.

Non-uniform preference vectors: In our model, we supply each node with a vector that captures its local preference for all other destinations. In overlay routing such preference may capture the percentage of locally generated traffic that a node routes to each destination, and then the aggregation of all preference vectors would amount to a origin/destination traffic matrix. In P2P overlays such preference may amount to speculations from the local node about the quality of, or interest in, the content held by other nodes. Other considerations may also include subjective criteria such as the perceived capacity of the node, its geographic location, or its availability profile.

C. Definitions

Let $V = \{v_1, v_2, \dots, v_n\}$ denote a set of nodes. Associated with node v_i is a preference vector $p_i = \{p_{i1}, p_{i2}, \dots, p_{ii-1}, p_{ii+1}, \dots, p_{in}\}$, where $p_{ij} \in [0, 1]$ denotes the preference of v_i for v_j , $i \neq j$; $\sum_{j=1, j \neq i}^n p_{ij} = 1$. Node v_i establishes a *wiring* $s_i = \{v_{i1}, v_{i2}, \dots, v_{iki}\}$ by creating links to k_i other nodes (we will use the terms link, wire, and edge interchangeably). Edges are *directed* and *weighted*, thus $e = (v_i, v_j)$ can only be crossed in the direction from v_i to v_j , and has cost d_{ij} ($d_{ji} \neq d_{ij}$ in the general case). Let $S = \{s_1, s_2, \dots, s_n\}$ denote a *global wiring* between the nodes of V and let $d_S(v_i, v_j)$ denote the cost of a shortest directed path between v_i and v_j over this global wiring; $d_S(v_i, v_j) = M \gg n$ if there's no directed path connecting the two nodes. If the links are also annotated, then $M \gg \max_{i,j} d_{ij}$. For the overlay networks discussed here, the above definition of cost amounts to the incurred end-to-end delay when performing shortest-path routing along the overlay topology S , whose direct links have weights that capture the delay of crossing the underlying IP layer path that goes from the one end of the overlay link to the other. Let $C_i(S)$ denote the cost of v_i under the global wiring S , defined as the weighted (by preference) summation of its distances to all other nodes, i.e., $C_i(S) = \sum_{j=1, j \neq i}^n p_{ij} \cdot d_S(v_i, v_j)$.

Definition 1: (The SNS Game) The selfish neighbor selection game is defined by the tuple $\langle V, \{S_i\}, \{C_i\} \rangle$, where:

- V is the set of n players, which in this case are the nodes.
- $\{S_i\}$ is the set of strategies available to the individual players. S_i is the set of strategies available to v_i . Strategies correspond to wirings and, thus, player v_i has $\sum_{k_i=1}^n k_i$ possible strategies $s_i \in S_i$.
- $\{C_i\}$ is the set of cost functions for the individual players. The cost of player v_i under an outcome S , which in this case is a global wiring, is $C_i(S)$. The above definition amounts to a local connection [9], non-cooperative, non-zero sum, n -player game [15]. Let $S_{-i} = S - \{s_i\}$ denote the *residual wiring* obtained from S by taking away v_i 's outgoing links.

Definition 2: (Best Response) Given a residual wiring S_{-i} , a best response for node v_i is a wiring $s_i \in S_i$ such that $C_i(S_{-i} + \{s_i\}) \leq C_i(S_{-i} + \{s'_i\})$, $\forall s'_i \in S_i$.

Definition 3: (Stable Wiring) A global wiring S is stable iff it is composed of individual wirings that are best responses. Therefore stable wirings are pure Nash equilibria of the SNS game, i.e., they have the property that no node can rewire unilaterally and reduce its cost. Fundamentally different is the work on Selfish Routing in which the network topology is part of the input to the game, and selfish source routing is the outcome. In a way, this is the inverse of our work, in which network-based (shortest-path) routing is an input of the game, and topology is the outcome.

IV. THE EGOIST OVERLAY ROUTING SYSTEM

The previous results have not shown :

- How to incorporate additional metrics other than delay, e.g., bandwidth?
- It is also unclear what is the average performance gain when SNS wiring strategies are used in highly dynamic environments, whether such overlays are robust against churn, and whether they scale?

We address the questions mentioned above by describing the design and implementation of EGOIST:

An SNS-inspired prototype overlay routing network. EGOIST serves as a building block for the distributed construction of efficient and resilient overlays where both individual and social performance is close to optimal. EGOIST is a distributed system that allows the creation and maintenance of an overlay network, in which every node selects and continuously updates its k overlay neighbors in a selfish manner. To minimize its (weighted) sum of distances to all destinations under shortest-path routing.

Basic Design

- In EGOIST, a *newcomer* overlay node v_i connects to the system by querying a *bootstrap* node, from which it receives a list of *potential* overlay neighbors.
- The newcomer connects to at least one of these nodes, enabling it to participate in the link state routing protocol running at the overlay layer. As a result, after some time, v_i obtains the full residual graph $G-i$ of the overlay.
- By running all-pairs shortest path algorithm on $G-i$, the newcomer is able to obtain the pair-wise distance (delay) function d_{G-i} .
- In addition to this information, the newcomer estimates d_{ij} , the weight of a potential direct overlay link from itself to node v_j , for all $v_j \in V-i$. Using the values of d_{ij} and d_{G-i} , the newcomer connects to $G-i$ using one of the wiring strategies discussed in Section V.
- In our implementation, each node acts as a server that listens to all the messages of the link state protocol and propagates them only to its immediate neighbors.
- In order to reduce the traffic in the system, each node propagates only unique messages by dropping messages that have been received more than once or have been superseded.
- There are also two threads, one for estimating d_{ij} , and one responsible for estimating the new wiring and propagating the wiring to the immediate neighbors.
- In order to minimize the load in the system, a node propagates its wiring to its immediate neighbors only if this changes.

V. CONCLUSION

In a network some system act as router with out there knowledge and some will know and come out of the network to improve the performance of the system, they keep interpreting the network by this the performance of the network will degenerate this known as selfish behavior of the node in network. To overcome that we have developed the EGOIST prototype in this when new node enter the network that will be assigned to the node in the network which has fewer children based on node degree so that bandwidth of the network will not decrease and also selects the neighbor by distance from it to its neighbor instated of number of hopes finally EGOIST prototype will impose best constraints on overlay network then the previous one.

REFERENCES

- [1] Selfish Overlay Network Creation and Maintenance GEORGIOS SMARAGDAKIS, NIKOLAOS LAOUTARIS AZER BESTAVROS, JOHN W. BYERS§
- [2] N. Laoutaris, G. Smaragdakis, A. Bestavros, and J. W. Byers, "Implications of Selfish Neighbor Selection in Overlay Networks," in *Proc. IEEE INFOCOM'07*.
- [3] J. Ledlie, P. Pietzuch, and M. Seltzer, "Network Coordinates in the Wild," in *Proc. USENIX/ACM NSDI'07*. IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 19, NO. 1, APR 2011.
- [4] A. Fabrikant, A. Luthra, E. Maneva, C. H. Papadimitriou, and S. Shenker, "On a Network Creation Game," in *Proc. ACM PODC'03*.
- [5] J. Corbo and D. C. Parkes, "The Price of Selfish Behavior in Bilateral Network Formation," in *Proc. ACM PODC'05*.
- [6] E. D. Demaine, M. Hajiaghayi, H. Mahini, and M. Zadimoghaddam, "The Price of Anarchy in Network Creation Games," in *Proc. ACM PODC'07*.
- [7] B.-G. Chun, R. Fonseca, I. Stoica, and J. Kubiawicz, "Characterizing Selfishly Constructed Overlay Routing Networks," in *Proc. IEEE INFOCOM'04*.
- [8] B. G. Rocha, V. Almeida, and D. Guedes, "Improving Reliability of Selfish Overlay Networks," in *Proc. WWW'06*.
- [9] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press, 2007.
- [10] X. Yang and G. de Veciana, "Performance of Peer-to-peer Networks: Service Capacity and Role of Resource Sharing Policies," *Perform. Eval.*, vol. 63, no. 3, pp. 175–194, 2006.
- [11] Z. Yao, D. Leonard, X. Wang, and D. Loguinov, "Modeling Heterogeneous User Churn and Local Resilience of Unstructured P2P Networks," in *Proc. IEEE ICNP'06*.
- [12] R. Bindal, P. Cao, W. Chan, J. Medval, G. Suwala, T. Bates, and A. Zhang, "Improving Traffic Locality in BitTorrent via Biased Neighbor Selection," in *Proc. IEEE ICDCS'06*.
- [13] Y. Liu, H. Zhang, W. Gong, and D. F. Towsley, "On the Interaction between Overlay Routing and Underlay Routing," in *Proc. IEEE INFOCOM'05*.
- [14] Y. Tian, D. Wu, and K.-W. Ng, "Analyzing Multiple File Downloading in BitTorrent," in *Proc. ICPP'06*.
- [15] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*. MIT Press, 1994.
- [16] Z. Li and P. Mohapatra. Qron: Qos-aware routing in overlay networks. *IEEE Journal On Selected Areas In Communications*, 22(1), 2004.
- [17] Overlay Networks for Peer-to-Peer Networks Andrzej W. Richa Christian Scheidelery November 17, 2005
- [18] EGOIST: Overlay Routing using Selfish Neighbor Selection Georgios Smaragdakis Vassilis Lekakis Nikolaos Laoutaris Deutsche Telekom Labs/TU Berlin FORTH & U. of Crete Telefónica Research, Barcelona Azer Bestavros John W. Byers Mema Roussopoulos Boston University Boston University FORTH &