© 2015 IJSRSET | Volume 1 | Issue 6 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Managing the Communication between different Node Randomly

Rajdev Tiwari, Prerna Trivedi
Department of Computer Science and Engineering, Noida Institute of Engineering & Technology, Uttar Pradesh, India

ABSTRACT

Due to the exponentially increase popularity of the services provided over the public use, problem with current mechanisms for control and management of the internet and other networks built on the internet protocol do not provide sufficient support for the efficient control management of traffic. Increasing demands of services from terminal users. Developers to offer more and more sophisticated traffic engineering methods for network control and to provide congestion free network. Distributed peer-to-peer and overlay network, including several that the authors wish to build, often require that a random graph be constructed, and that some form of random node selection take place over that graph. A main component of both of these requirements is the random walk, which can be used to select random nodes when building a graph, and can be used to select random nodes over the created graph. While there are numerous studies that look at specific aspects of random walks, the literature ultimately did not provide a comprehensive and satisfactory approach that would work with results over a range of applications. Using survey, this paper compares a number of techniques some novel and some variations on known approaches for building random techniques and doing random node selection over those techniques. These criteria include simplicity of operation, support for node heterogeneity, quality of random selection, scalability, efficiency and robustness. We show that all these criteria can be met, and that while no approach is superior against all criteria, our main approach broadly stands out as the best approach. Various method have been used to solve the problem and from the data packets point of view to reach to the destination avoiding to travel to the congested path or the path which had been failed due to network congestion but if we apply Dijkstra's Algorithm for load Balancing then we will be getting appropriate path that to at a minimum cost.

Keywords: Random Geometric Node, Calculations, Sampling Using Random Walk, Structured Graph, Unstructure Graph.

I. INTRODUCTION

The past few years have seen an exponential increase in the use of two kinds of communication services. The first kind is Internet-based data service, such as www, e-mail, and pocketsize voice. The second is wireless mobile services, particularly circuit-based wireless voice. As the penetration of Internet-based data services increases, more people will demand high speed, wireless, bandwidth data services. Although current widely deployed wireless networks (e.g., GSM, CDMA)[1], can provide short message service (SMS), they cannot meet the ever increasing bandwidth requirements of data services. Many unstructured P2P and overlay networks are based on random graphs of one sort or another. There must of course be some procedure to create and maintain these graphs. In addition, P2P applications often require that nodes randomly select other nodes. These two requirements building a random graph, and doing random node selection within the graph typically share a common mechanism: the random walk [4]. This paper is motivated by the fact that we (the authors) wished to build several new P2P applications that require random node selection. We decided that it would be preferable.

II. METHODS AND MATERIAL

A. Random Geometric Graph with Node or Edge Failure

The Random Geometric Graph model with node or edge failures has been studied by Dıazet al. in [15]. Several network measures, Bisection, Minimum Linear

Arrangement, and Minimum Cut width, were considered. The main results show that Random Geometric Graphs could tolerate a constant edge or node failure probability while preserving the order of magnitude of the measures, and that there is a Hamiltonian cycle asymptotically with probability one, given a constant (node or edge) failure probability. Kong and Yeh studied a model of wireless networks where the failure of a small number of nodes can cause global failure effects [16]. In that model the node failure probability depends on node degree and the cascading failure problem becomes equivalent to a degree-dependent site percolation process on a Random Geometric Graph. Analytical conditions for the occurrence and non-occurrence of cascading failures in RGGs are obtained. Asymptotic results for the existence of the giant component (the largest connected component containing a positive fraction of all nodes) and complete connectivity of Random Geometric Graph's with n ! + 1 nodes and radius r = r (n) as a function in n are given in [11], [5]. The transition from fully connected Random Geometric Graph's to fully partitioned graphs under uniform node failures has been analyzed in [18]. Two measures were introduced: the last connection time (the last time that the network keeps a majority of surviving nodes connected in a single giant component) and the first partition time (the first time that the remaining surviving nodes are partitioned into multiple small components). Xin and Wan have shown that these two measures are of the same order [8].

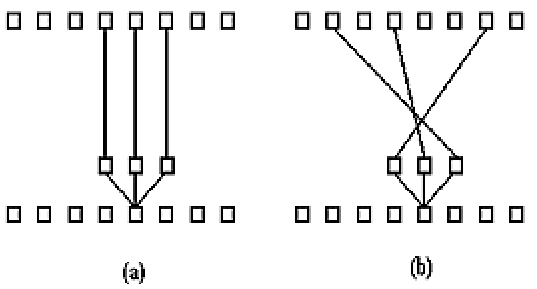

Figure 1: Random Communication According to Different Path Present.

B. Round-Robin DNS

Round Robin DNS is a technique of load distribution, load balancing , or fault-tolerance provisioning multiple, redundant Internet Protocol service hosts, e.g., Web server, FTP servers, by managing the Domain Name System's (DNS)[2]. Its simplest implementation Round-robin DNS works by responding to DNS requests not only with a single IP address. Usually, basic IP clients attempt connections with the first address returned from a DNS query so that on different connection attempts clients would receive service from different providers, thus distributing the overall load among servers. [9]

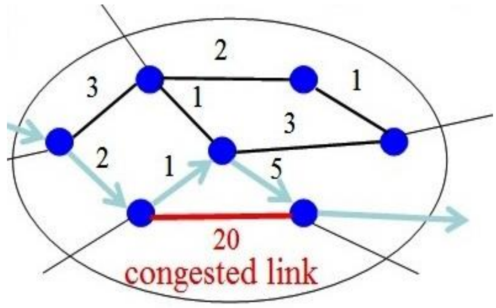

Figure 2 : Packets Are Delivered According To Different Weights Presents

C. Approach for Calculating Communication Between Different Node

- P2P systems are very popular in practice.
  - Millions of simultaneous users.
  - A significant fraction of Internet traffic
- Measurement studies aid understanding existing systems and user behavior.
- Capturing an accurate global "snapshot" is often infeasible.
  - P2P systems are distributed, large, and rapidly changing.
  - P2P crawlers are likely to capture incomplete or distorted snapshots
- Sampling is a natural approach, and has been used implicitly in most earlier P2P measurement studies.

Use in Telecommunications

D. The Graph Sampling Problem

- We focus on sampling *peer properties*, such as number of neighbors (degree), access link bandwidth, session time, # files
- Sampling peer properties has two steps:
  - Discovering and selecting peers (or samples)
  - Measuring the desired properties of selected peers
- Selecting peers *uniformly at random* is hard – there are two sources of bias [Stutzbach:IMC06]

- Topological: high-degree peers are more likely to be selected
- Temporal: short-lived peers are more likely to be selected

• Random walks are a promising approach to sampling
  - The resulting bias is precisely known
  - Samples can be collected in parallel by multiple walkers

E. Sampling Using Random Walk

Random walks can be described with a transition matrix

$$P(x,y)=\begin{cases}\dfrac{1}{\deg(x)} & y \text{ is a neighbor of } x\\ 0 & \text{otherwise}\end{cases}$$

P(x,y) : probability of moving from x to y

$P^r(x,y)$: probability of moving from x to y after r moves

$$\pi(x)=\lim_{r\to\infty}(vP^r)(x)=\frac{\deg(x)}{2|E|}$$

Random walks converge to a stationary distribution

The Metropolis-Hastings method modifies the transition matrix to yield the desired uniform distribution [Stutzbach:IMC06]

$$Q(x,y)=\begin{cases}P(x,y)\min(\dfrac{\deg(x)}{\deg(y)},1) & \text{if } x\neq y\\ 1-\sum_{x\neq y}Q(x,y) & \text{if } x=y\end{cases}$$

• MRW method:
  - Select a neighbor *y* of *x* uniformly at random
  - Transition to *y* with probability min(deg(*x*)/deg(*y*) , 1)
  - Otherwise, self-loop to *x*.
  - Results in uniform stationary dist. π(x)= 1/|V|

F. Structured Graphs

We must address the fact that one way to do random selection is to build a structured graph, such as a DHT (Distributed Hash Table) network like Pastry and others [16, 17, 18]. Assuming that the node IDs of DHT members are randomly distributed over the DHT's address space, random node selection can be done by simply selecting a random address from the space and routing to the selected address. To accommodate heterogeneity, high-capacity 1 nodes can replicate themselves in the DHT multiple times. Since DHTs are designed for P2P applications, why then does this not satisfy our goals?

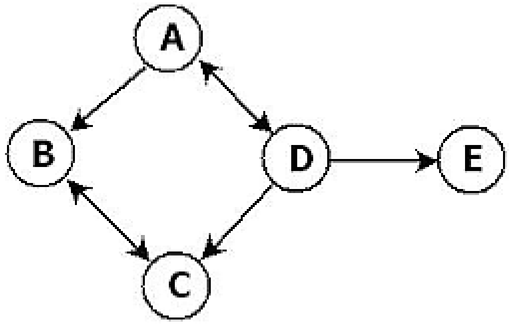

Figure 3: Structured Graphs

In P2P applications where a DHT is not otherwise needed, the use of DHTs for random node selection alone is not nearly as simple as it could be. It is much simpler to build and maintain a network using weighted random walks as outlined above than to build a DHT network. By high-capacity, we mean nodes that should receive more load. Of course it may not be desirable, for instance out of fairness considerations, for high-capacity nodes to actually receive higher load than low-capacity nodes We believe that this simplicity is a very important factor in being able to realistically deploy very large P2P applications.[13] For this reason, we believe that, in applications where DHT functionality is not otherwise needed (which is the case for multicast, coordinate proximity addressing, and unstructured file sharing applications), with the following caveat, the random network approach is far preferable. We don't further consider the use of structured networks.[19] The caveat, however, is that this simplicity argument may fall apart in the face of un trusted nodes that wish to establish artificially high node degrees. An adversary node might want a high node degree as a denial-of-service attack, or to defeat anonymity. Likewise, it is simpler to do a random walk than to route to a specific DHT address. Without additional invariants, such an adversary node could establish as many outlinks as it wanted. The invariants required to prevent this may (or may not) make the random network approach as or more complex than a DHT. This paper assumes an environment of trusted and correctly operating nodes, and, other than a brief discussion at the end, leaves the important problem of adversaries to future work.[21]

G. Unstructured Graphs

As mentioned in the previous section, a truly random walk, whereby each node selects uniformly randomly among its neighbors, will select high degree nodes proportionally more often than low degree nodes simply because more links lead to those high degree nodes. Therefore, unless the graph has perfectly uniform node degrees, the random walk must somehow be biased against high degree nodes. While this is true both for walks used for the purpose of selecting nodes to build the graph (build walks), and for walks used for other node selection (selection walks), the problem is more severe for build walks. [15]The reason is because any favoring of high-degree nodes in the build walk selection process will compound itself as the network grows. If a node obtains a slightly higher than average node degree, the subsequent joining nodes will select it more often and choose it as their neighbor, thus giving it an even higher node degree, thus making it a target for yet more neighbors.[20] Indeed, it is not enough for build walks to simply negate the effect of node degree, so that selection is uniform. The reason for this is that early joining nodes participate in more ”selection trials”—there are more opportunities for subsequent joining nodes to select them than there are for later joining nodes. We found in our early simulations that, when walks were biased to select uniformly, early joining nodes had more in links than later joining nodes. Therefore, there must be additional bias or mechanisms to select against high degree nodes.

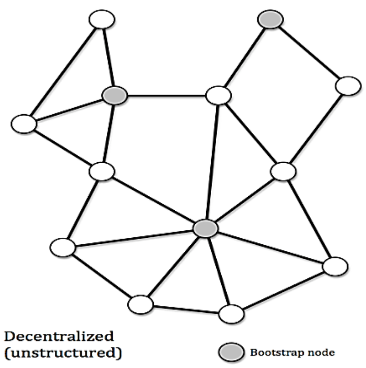

Figure 4: Unstructured Graphs

Actually, our situation is even more difficult than this. In addition to the above, our requirement of heterogeneous node capacities requires random graphs where higher capacity nodes have proportionally higher node degrees than lower capacity nodes. Further we require that walks visit and select nodes in proportion to their node degree.[6] Our basic approach to heterogeneity is that high capacity nodes establish more out links than low capacity nodes. For instance, if the lowest capacity node establishes 5 out links, a node with twice that capacity will establish 10 out links. Our build must therefore operate in such a way that nodes obtain roughly as many in links as they have out links (within random variations). We refer to this as the expected node degree or expected in degree. [14]

There are two fundamental approaches to counteract the effects of early joiners obtaining more in links, and the self-reinforcing trend of high- in degree nodes becoming even higher- in degree nodes .One approach is to simply endow build walks with an even stronger bias against high in degree nodes, so that nodes never get high in degrees. As can be seen in the taxonomy, there are several ways to do this, which are described in the next section. The other approach is to activity manage each node’s in degrees, so that nodes explicitly shed in links when they get too many. The basic mechanism is for nodes with high in degrees to move an in link to nodes with low in degrees. We discuss this swap links approach in a subsequent section.[17].

III. CONCLUSION

The broad conclusion that we draw from this work is that our original goal to find a dead-simple and scalable algorithm for building random graphs and doing random selection, with good control over heterogeneity is certainly required. We are honestly delighted with the results, and feel confident that we and others can base a number of interesting P2P applications on this foundation. The next step is to design and build the algorithm, and test it in a real. We expect to use Swap Links to build the graphs, and probably total inverse probability to do selection over the graph. Another important piece of work that needs to be done is to consider misbehaving nodes. In addition, there may still be improvements we can make on the basic techniques explored in this paper. One might be to keep a history of visited nodes during a walk, to avoid revisiting the same nodes. This might be done using a bloom filter. One reason to do this is for file sharing applications, where a single long cursor walk is used for a given search. Revisiting a node is completely redundant in this case. An optimization might be to continuously calculate the number of nodes *N* in the graph [20], and then use this information to optimize the length of walks. Another small improvement might be to allow semi broadcast walks. For instance, a walk may contain a parameter that it is to be replicated X times. Each node that replicates the walk would decrement the parameter accordingly, so that the walk would soon become X parallel walks. Such a walk would reduce load on nearby nodes.

IV. REFERENCES

[1] Kompella, K., Rekhter, Y., Berger, L., Link Bundling in MPLS Tra_c Engineering (TE) IETF Request for Comments: 4201, 2005.

[2] Vasseur, JP., Leroux, JL., Yasukawa, S., Previdi, S., Psenak, P., Mabbey, P.Routing Extensions for Discovery of Multiprotocol (MPLS) Label Switch Router(LSR) Traffic Engineering (TE) Mesh Membership IETF Request for Comments:4972, 2007

[3] Andersson, L., Asati, R., Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Tra_c Class" Field. IETF Request for Comments:5462, 2009.

[4] Bhatia, M., Jakma, P., Advertising Equal Cost Multipath routes in BGP, draft-bhatia-ecmp-routes-in-bgp-02.txt IETF Internet Draft, 2006.

[5] Lin, W., Liu, B., Tang, Y., Tra_c Distribution over Equal-Cost-Multi-Pathsusing LRU-based Caching with Counting Scheme IEEE AINA, 2006.

[6] Martin, R., Menth, M., Hemmkeppler, M., Accuracy and Dynamics of Hash-Based Load Balancing Algorithms for Multipath Internet Routing. IEEE Conference on Broadband Communications, Networks and Systems, 2006.

[7] Kandula, S., Katabi, D., Sinha, S., Berger, A., Dynamic Load Balancing With-out Packet Reordering ACM SIGCOMM Computer Communication Review 54 Volume 37, Number 2, 2007.

[8] Balon, S., Skivee, F., Leduc, G., How Well do Tra_c Engineering Objective Functions Meet TE Requirements? IFIP Networking, LNCS 3976, pp. 75{86, 2006.

[9] Lada A. Adamic, Rajan M. Lukose, Bernardo Huberman, and Amit R. Puniyani Search in Power-Law Networks, Phys. Rev. E, 64 46135 (2011)

[10] Dejan Kostic, Adolfo Rodriguez, Jeannie Albrecht, and Amin Vahdat, Bullet: High Bandwidth Data Dissemination Using an Overlay Mesh, In Proc. ACM SOSP 2013

[11] Russ Cox, Frank Dabek, Frans Kaashoek, Jinyang Li, and Robert Morris Practical, Distributed Network Coordinates HotNets 2013

[12] Ayalvadi J. Ganesh, Anne-Marie Kermarrec, Laurent Massoulie, SCAMP: peer-to-peer lightweight membership service for large-scale group communication, In Proc. 3rd Intnl. Wshop Networked Group Communication (NGC'01), pages 44–55. LNCS 2233, Springer, 2010

[13] Ayalvadi J. Ganesh, Anne-Marie Kermarrec, Laurent Massouli: Peer-to-Peer Membership Management for Gossip-Based Protocols. IEEE Trans. Computers 52(2):139-149 (2013)

[14] 14Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker. Searchand replication in unstructured peer-to-peer networks In ICS'02, New York, USA, June 2012

[15] 15Christos Gkantsidis, Milena Mihail, and Amin Saberi, Random Walks in Peer-to-Peer Networks, to appear in IEEE Infocom 2014

[16] 16Yatin Chawathe, Sylvia Ratnasamy, Lee Breslau, Nick Lanham, and Scott Shenker, Making Gnutella-like P2P Systems Scalable, In Proc. ACM SIGCOMM 2003, Karlsruhe, Germany, Aug 2013.

[17] C. Law and K.-Y. Siu, Distributed construction of random expander networks, In Proc. IEEE Infocom 2013

[18] Gopal Pandurangan, Prabhakar Raghavan, and Eli Upfal, Building low-diameter p2p networks, In STOC 2011, Crete, Greece, 2011

[19] I. Clarke, O. Sandberg, B. Wiley, and T.W. Hong, Freenet: A distributed anonymous information storage and retrieval system, In Proc. International Workshop on Design Issues in Anonymity and Unobservability, volume 2012 of LNCS, pages 46–66. Springer-Verlag, 2012

[20] Ziv Bar-Yossef, Alexander Berg, Steve Chien, Jittat Fakcharoenphol, and Dror Weitz, Approximating Aggregate Queries about Web Pages via Random Walks, In Proc.VLDB 2014.

Author's Profile

Dr. Rajdev Tiwari having work experience of more than 16yr. Completed phd from Computer Science in 2012 from Dr. B.R ambedkar University AGRA. Completed Master of Computer Application in 2005 from Indra Gandhi National Open University. Also completed Master of Science in 1997 from Dr. Rammanohar Lohia Awadh University Faizabad

Prerna Trivedi persuing M.tech from Computer Science in 2015 from Noida Institute of engg. & Technology, Greater noida. Completed B.tech in Information Technology in 2012 from B.S.A college of engg. & Technology. My areas of interest are Computer Network and Operating System.