

Random Walk algorithm implementation for unstructured networks

Suraj Singh Chandel
M.Tech (I.T)
LNCT BHOPAL(M.P)

Dr. Manish Shrivastava
H.O.D. Information Technology
LNCT BHOPAL(M.P)

Abstract

The current peer-to-peer (P2P) content distribution systems are constricted by their simple on-demand content discovery mechanism. The utility of these systems can be greatly enhanced by incorporating two capabilities, namely a mechanism through which peers can register their long term interests with the network so that they can be continuously notified of new data items, and a means for the peers to advertise their contents. P2P system we are going to use under the content distribution networks. In one single peer any user registered according to his own interest it can available like long interest specification process. We are going to introduce the unstructured overlay networks. No such user are not select particular item automatically we are going to remove item from the peer. This paper argues that for many P2P applications. This process it can handle with the help of overlay analysis specification. Only one single item uses frequently load it can contains high under the particular peer we are going to use random walk graph specification process. Those load we are going to distribute to another peer with notification messages. This process it can works as a dynamic peer registration process it can occur under the network creation process.

Keywords

Peer-to-peer networks, Continuous queries, Publish-subscribe systems, Random walk.

1. Introduction

Peer-to-peer (P2P) networks are increasingly becoming popular because they offer opportunities for real-time communication, ad-hoc collaboration and information sharing in a large-scale distributed environment. Peer-to-peer computing is defined as the sharing of computer resources and information through direct exchange. The most distinct characteristic of P2P computing is that there is symmetric communication between the peers; each peer has both a client and a server role. The advantages of the P2P systems are multi-dimensional; they improve scalability by enabling direct and real-time sharing of services and information;

enable knowledge sharing by aggregating information and resources from nodes that are located on geographically distributed and potentially heterogeneous platforms; and, provide high availability by eliminating the need for a single centralized component. Different number of unstructured peer-to-peer systems are provides content share distribution process. It can provide popular solutions specification process. Dynamically changes the resources under the peer. Searching process it can starts in different number of peers. According to user demand we are allocate the resources of information and discover the peer where it can available with resources of information. Query it can provides the good response for accessing the content of information or download the content of information. This kind of peers are contains ad-hoc based query processing representation process ^[1].

There has been a growing interest in peer-to-peer networks since the initial success of some very popular file -sharing applications such as Napster and Gnutella ^[15]. A peer-to-peer (P2P) network is a distributed system in which *peers* employ distributed resources to perform a critical function in a decentralized fashion. Nodes in a P2P network normally play *equal roles*, therefore, these nodes are also called peers. A typical P2P network often includes computers in unrelated administrative domains. These P2P participants join or leave the P2P system frequently, hence, P2P networks are *dynamic* in nature. P2P networks are overlay networks, where nodes are end systems in the Internet and maintain information about a set of other nodes (called neighbours) in the P2P layer. These nodes form a virtual overlay network on top of the Internet. Each link in a P2P overlay corresponds to a sequence of physical links in the underlying network. Examples of P2P applications are distributed file -sharing systems, event notification services, and chat services ^{[2] [3] [4] [5]}.

While the ad hoc query model for data discovery is essential for P2P content distribution

networks, it suffers from two serious limitations. First, due to its very nature, an ad hoc query is only capable of retrieving content that exists in the P2P network during the time period when it is actively propagated and processed in the network. Further, an ad hoc query can never reach a peer that joins the network after the query has completed its circulation, and hence cannot discover matching data-items on the new peer. In this scenario, the only way for a peer to discover newly added data-items would be to repeatedly issue the same query, thereby imposing unnecessary overheads on the network. Second, the ad hoc query model provides no support for peers to advertise or announce the data items they own to other interested peers. Such capabilities are important for P2P communities where peers trade content^[6].

As an example, consider a community comprising of amateur musicians and patrons interested in buying the music produced by the musicians. The musicians would need a mechanism to advertise their new music works to prospective buyers. A naive approach for tackling this problem would be to send advertisements to large subsets of peers through flooding. However, this approach is unviable. Besides heavy messaging overheads, this scheme could overwhelm the peers with unwanted advertisements. What is really needed is a *targeted advertisement* service that sends advertisements to peers who would be interested in the content being advertised.

The well-studied paradigm of publish-subscribe (pub/sub) systems [7], [9], [15] is a possible approach to address the above limitations. The pub-sub interaction paradigm is a *guaranteed notification service* that lets the subscribers to register their interests². When an event producer publishes an event, the system checks the registered subscriptions and generates notifications to all the subscribers who have registered a matching subscription.

2. Problem Statement

The difficulties in implementing pub-sub systems on top of unstructured overlays can be attributed to the inherent mismatch between the design requirements of pub-sub systems and the very nature of unstructured P2P systems. In order to provide strict notification guarantee, pub-sub systems have to maintain subscription information in well-organized structures. In contrast, unstructured overlays, by their very nature, are decentralized, loosely coupled and, to certain extent, haphazard. This makes it difficult to build full-fledged pub-sub systems on these platforms.

3. Related Work

A natural question that comes up is whether the continuous query model is amenable to efficient and significantly less complex implementations? In other words, what techniques and mechanisms are necessary to implement this model on top of generic unstructured overlay networks such that the system is not only effective and scalable but is also resilient to the churn of the overlay network? Towards answering this question, we design a lightweight middleware called CoQUOS (continuous queries in unstructured overlays) for supporting continuous queries and advertisements in unstructured P2P networks. One of our goals in designing the CoQUOS system has been to preserve the design simplicity of the underlying unstructured P2P networks, and their flexibility towards network churn. In this regard, the design of the CoQUOS system exhibits two unique features. First, the CoQUOS system does not impose any topological constraints on the underlying P2P network, and it can be implemented as an independent module in any unstructured overlay network. Second, the CoQUOS system does not require complex index structures or routing mechanisms. Instead, our design is based upon very lightweight P2P primitives.

Specifically, this paper makes four technical contributions:

- First, we present the architectural design of the Co-QUOS system for supporting continuous queries in unstructured P2P networks. In our architecture each peer maintains a set of continuous queries and notifies the respective query issuers of any matching data items that it discovers. This architecture includes a completely decentralized mechanism for registering a query at various regions of the P2P network.
- Second, we propose a novel *cluster resilient random walk (CRW)* technique for propagating a query to various regions of the network. While preserving the overall framework of random walks^[8], the CRW scheme favours neighbours that are more likely to send the message deeper into the network.
- Third, we design a *dynamic probability scheme* for ensuring that query registrations are well distributed along the path of the query. In this scheme, a query that has not been registered in

the past several hops has a higher chance of getting registered in its next hop. The proposed schemes are evaluated through series of experiments. The results indicate that the techniques are highly effective, and they impose very little overheads on the P2P network.

- Finally, we propose a local load re-distribution strategy to achieve fair distribution of notification loads among the participating peers.

4. Selecting Beacon Nodes Of A Query

The highly decentralized nature of unstructured overlay network makes the problem of selecting beacon nodes that satisfies the two properties particularly challenging. The CoQUOS system incorporates a completely decentralized technique for beacon node selection. In this scheme, a continuous query is circulated in the network (by neighbor forwarding), and each peer that receives the continuous query decides independently whether to register and store the query. An important feature of our scheme is that although each peer makes a local decision regarding query registration, the resulting beacon node sets manifest the two important characteristics to a very high degree.

First and foremost, the beacon nodes of a query should be distributed in every major region of the overlay network. This implies that most peers in the network should be reachable from at least one node in the beacon set in a very small number of hops. This property is essential for achieving high notification success rates, since the announcements reach only the peers located within a small number of hops from their respective source peers. Second, the beacon nodes of a query should not be located very close to one another.

In this context, we need to address two important problems:

(1) *What mechanisms should be adopted for circulating continuous queries?*

(2) *How should a peer receiving a continuous query decide whether to register the query?*

We answer these questions by describing the two novel components of our beacon node selection scheme, namely *cluster-resilient random walk (CRW)* mechanism for query dissemination and *dynamic probability (DP)* technique for query registration.

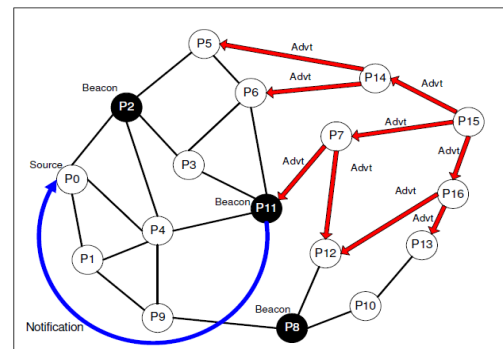


Figure1. Overview of the CoQUOS System

4.1 Cluster Resilient Random Walk

Flooding-based broadcast is an option for circulating continuous queries. However, this would be analogous to a breadth first traversal of the network. As previous studies have reported, in this scheme, messages remain in close vicinity of the source node and do not go deep into the network [10], [11]. Random walk is another message propagation paradigm that has received considerable attention from the P2P research community [10], [12]. In the context of P2P networks, random walk works as follows: When a peer node P_i receives a message whose TTL has not expired, it selects one of its neighbors completely at random and forwards the message to that peer. Since, at each step the message is forwarded to only one neighbor, the message load imposed by random walk is very low. Random walk corresponds to a depth-first traversal of the network, and a message propagated through random walks has a higher probability of reaching remote regions of the network than its flooding-based counterpart. In this paper we use the terms random walk and pure random walk (PRW) interchangeably

4.2 Dynamic Probability Based Query Registration

The CRW scheme provides a mechanism for propagating a continuous query. But, how does a node receiving this message decide whether to register the query? A straightforward solution would be to register a query at every node it visits. However, this would result in large numbers of unnecessary subscriptions, which affects the efficiency of the network. Alternatively, each peer receiving a query message can decide register it with a certain fixed probability, say R_p . We call this scheme the *fixed probability-based query registration scheme (FP scheme, for short)*. Although this strategy

seems intuitive, it cannot guarantee high notification success rates for every query.

The experiments in Section 6 confirm our contention in this regard. The reason is that for some continuous queries a long series of peers in the path of the query message may all decide not to register the query, whereas another sequence of consecutive nodes may all decide to host the query. The announcements originated near the dry patches of a query's path might fail to reach any of

its beacon nodes, thus leading to low success rates. Considering these requirements, we have designed a novel dynamic probability-based technique (DP scheme, for short) for peers to decide whether to register a continuous query. As in the fixed probability scheme, a peer receiving a query-message registers it with certain probability. However, the registration probability of a query varies as the query traverses along its route. The central idea of the dynamic probability scheme can be summarized as follows: The probability of registering a query at a peer node would be high if the query has not been registered at the nodes it visited in the recent past. In contrast, if the query has been registered at a node that visited in the past few hops, the probability of it getting registered at the current peer would be low.

4.3 Overlay Churn

P2P networks are, in general, highly dynamic systems, with nodes entering and exiting the system quite frequently. This *churn* of the overlay network can adversely impact the success of continuous queries and announcements. When a node P_i gracefully leaves the system, it asks one of its neighbors to handle all registered queries at P_i and also notifies all the beacon nodes with queries issued by P_i to remove the queries. However, when P_i exits the system unexpectedly, all the registrations are lost and the notification success rates of the respective queries and the matching announcements drop. Thus, effective mechanisms are needed to alleviate the negative effects of churn in the overlay network.

4.4 Load Balancing

Achieving good load distribution among peers is another important requirement for the performance of the Co- QUOS system. The number of queries and the numbers of notifications sent out per unit time by various nodes represent two key load metrics for CoQUOS system. These load parameters can vary widely among the nodes of the CoQUOS system due to

variety of reasons, including topological characteristics of the network, skewed announcement and query popularities, variation in the resource availabilities at peers or a combination of these factors. Irrespective of the cause, load imbalances not only degrade the performance of the system, but may also cause overloaded peers to exit the network.

Ensuring good load balancing in decentralized, loosely coupled systems such as unstructured P2P overlays is challenging. In fact, achieving optimal load balancing on a global scale may turn out to be prohibitively expensive as it would require collection and maintenance of load information on a global scale. Skip graphs have been used to alleviate some of the problems associated with global load balancing in the context of range data [16]. However, skip graphs require maintenance of circular linked lists consisting of all nodes in the system. Hence, they are not suitable for a large-scale, dynamic environment like unstructured P2P content distribution platforms.

5. Proposed System

Proposed system represents with different peers of we are collect the same information. This particular item cluster results of information. It can contain good scalability inside the popular solution representation process. We are going to perfect overlay network representation. Overlay network creates the information like indexing. It can provide random walk representation process. It can increase the popularity specification process. One peer contains load is high we are going to distribute to another peer allocation. We are going to provide random notification from one peer to another peer. We are going to good and perfect results of information.

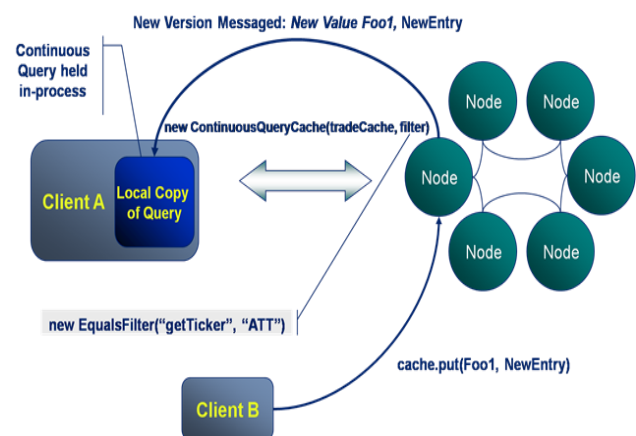


Figure2. SYSTEM ARCHITECTURE:

6. Problem Statement

We are creating complete decentralized system with the help of beacon nodes. Each and every node itself we are creates the calculation of particular node index specification. That kind of peer is register inside the main server representation or specification process. This is one best feature for overcome the previous system problems. We are going try to implement random walk representation. We are calculating the probability specification. This is one of the attractive specification generation processes. Using random walk representation maintains the load balancing procedure representation.

7. Conclusion

Mechanisms that enable individual peers of unstructured P2P content sharing networks to register longstanding queries and receive notification when new matching items appear can significantly improve their utility and effectiveness. While the pub-sub paradigm can provide this capability, implementing pub-sub systems on unstructured overlays is often a very complex end ever. The continuous query paradigm studied in this paper is similar to pub-sub, but it provides best effort notification service. We presented the design and evaluation of a lightweight system, called CoQUOS, which supports continuous queries in unstructured P2P networks. The CoQUOS system incorporates several novel features such as cluster resilient random walk for query propagation, dynamic probability scheme for query registration, and a lazy replication technique for countering network churn. The proposed techniques have been evaluated through theoretical analysis and simulation-based experiments.

8. Reference

[1] Vana Kalogeraki, Dimitrios Gunopulos, D. ZeinalipourYazti *A Local Search Mechanism for PeertoPeer Networks CIKM'02*, November 4–9, 2002, McLean, Virginia, USA. Copyright 2002 ACM 15811349 24/ 02/0011

[2] H. Balakrishnan, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica, "Looking up data in P2P systems", *Communications of ACM*, Vol.46, No.2, 2003.

[3] N. Daswani, H. Garcia-Molina, and B. Yang, "Open problems in data-sharing peer-to-peer systems", *Proc. of the 9th International Conference on Database Theory (ICDT'03)*, 2003.

[4] D. S. Milojevic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rolins, and Z. Xu, "Peer-to-peer computing", *HP Lab technical report*, HPL-2002-57, 2002.

[5] D. Barkai, "Technologies for sharing and collaborating on the net", *Proc. of the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, 2002.

[6] Lakshmish Ramaswamy, Jianxia Chen and Piyush Parate *CoQUOS: Lightweight Support for Continuous Queries in Unstructured Overlays* 1-4244-0910-1/07/\$20.00 c 2007 IEEE

[7] G. Banavar, T. Chandra, B. Mukherjee, J. Nagarajarao, R. E. Strom, and D. C. Sturman. An Ef_icient Multicast Protocol for Content- Based Publish-Subscribe Systems. In *Proceedings of ICDCS 1999*, 1999.

[8] C. Gkantsidis, M. Mihail, and A. Saberi. Random Walks in Peer-to-Peer Networks. In *Proceedings of the IEEE INFOCOM 2004*, 2004.

[9] A. Carzaniga, D. S. Rosenblum, and A. L. Wolf. Design and evaluation of a wide-area event noti_cation service. *ACM Transactions on Computer Systems*, 19(3):332.383, 2001.

[10] C. Gkantsidis, M. Mihail, and A. Saberi. Random Walks in Peerto- Peer Networks. In *Proceedings of the IEEE INFOCOM 2004*, 2004.

[11] C. Gkantsidis, M. Mihail, and A. Saberi. Hybrid search schemes for unstructured peer-to-peer networks. In *INFOCOM*, 2005.

[12] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker. Search and Replication in Unstructured Peer-to-Peer Networks. In *Supercomputing*, 2002

[13] S. Androutsellis-Theotokis and D. Spinellis. A Survey of Peer-to-Peer Content Distribution Technologies. *ACM Comput. Surv.*, 2004.

[14] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker. Making Gnutella-like P2P Systems Scalable. In *Proceedings of ACM SIGCOMM 2003*, 2003.