Shrack: Description and Performance Evaluation of a Peer-to-Peer System for Document Sharing and Tracking using Pull-Only Information Dissemination

Hathai Tanta-ngai   Vlado Kešelj   Evangelos E. Milios
Faculty of Computer Science, Dalhousie University
Halifax, Nova Scotia, Canada
http://www.cs.dal.ca/~{hathai,vlado,eem}

Abstract

Shrack is a peer-to-peer framework for document sharing and tracking. Shrack peers provide support to researchers in forming direct collaboration in autonomous sharing and keeping track of newly published documents based on their interests. We propose a pull-only information dissemination protocol for peers to distribute information about new documents among peers with similar interests. Each peer can use the disseminated information to build a local view of semantic overlay of peer interests in the network. Each peer can later use the semantic overlay to find new contact information about other peers with a particular interests, as well as search for documents archived by other peers. After presenting an overview architecture of the system and the dissemination protocol, we present the evaluation results of the system performance, based on a simulated environment. The results indicate that the Shrack protocol is scalable and reliable as the network size increases.

1 Introduction

Researchers with a common interest normally form collaborations to advance their knowledge or achieve the common goal. They may collaborate by sharing resources or ideas through discussion, or by working together to integrate their results. The collaborations are not limited to researchers in the same organization. Several Internet applications have been used to help researchers form collaborations around the world. Email, the Web, and chat are examples of such applications.

The Internet also serves as a medium for researchers to share documents of interest. Tools have been developed to help researchers find documents over the Internet. These tools include search engines, mailing lists, and online archives. However, existing tools pose a number of challenges for researchers in getting informed about newly published documents, which from now will be referred to as *new documents*. For instance, to keep track of new documents of interest, researchers need to revisit search engines or on-line archives multiple times and issue the same query. Mailing lists can also be used for keeping track of new documents; however, researchers need to find where to post and subscribe for information. Moreover, these tools usually lack autonomous mechanisms to select documents based on individual interests.

We propose Shrack, a peer-to-peer (P2P) framework for document sharing and tracking, as an infrastructure for researchers to share and keep track of new documents using pull-only communication. The main motivations for researchers to join Shrack are to learn about new documents from a group of researchers that have similar research interests and to make their work visible to the research community. The notion of collaboration in Shrack consists of the promotion of awareness of newly published documents to researchers having similar interests. Shrack is not intended to be used in collaboratively editing and updating documents.

Why a P2P System? P2P systems are robust against single points of failure and have the potential of being scalable as the system size increases. In our proposed P2P framework, each peer is associated with an individual researcher or a research organization. Between the two P2P system architectures, unstructured and structured P2P systems, we choose to design Shrack as an unstructured P2P system.

An unstructured P2P system, such as Gnutella [4] and KaZaa [8], is a P2P system that does not have control over network topology and data placement. Each peer maintains a list of addresses of other chosen peers randomly or based on mutual interest. A structured P2P system [1] has a predefined network topology that defines which peer addresses and data items each peer has to maintain. Although structured P2P systems provide guarantee and efficient algorithms to locate data items in the system, each peer needs to know the unique keys of the data items in advance in order to acquire the items.

Why Pull-Only Communication? Pull communication gives control to peers who seek information to select from which sources and when to pull the information.

The system stimulates cooperation by learning and sharing new documents within a group of researchers having common interest, which is determined and controlled by peers that seek information. Peers that provide good sources of information can build their reputation and make themselves and their work visible. In addition it is easier for researchers to publish documents at one place than push or inform potential readers in many different places or mailing lists.

With pull-only communication, peers that want to provide information would make themselves available for other peers to access the information. Peers that want to receive information, when they are on-line will seek and contact peers that provide information. However, designated peers, called super peers, act as information aggregators and they are expected to be always on-line. Many pull-based applications are widely used in the Internet such as Web browsers, and RSS readers.

Our goal is to build a P2P network, Shrack, to autonomously help researchers keep track and retrieve new documents injected into the system based on their interests. To achieve the goal, we need to design and develop (1) an architecture of Shrack, that supports peer functionality, (2) a dissemination protocol, which defines how information is disseminated among peers, (3) peer profile learning and filtering mechanisms, which describe how a peer learns about the user's interest and define how to filter received information, and (4) a peer neighbour selection mechanism, which enables a formation of peers having similar interest. The focus of this paper is on the first-two design issues.

The remainder of the paper is organized as follows. Section 2 describes the Shrack architecture. Section 3 describes the Shrack network and operations. Section 4 presents our proposed information dissemination protocol. Section 5 defines experimental parameters and performance metrics. Section 6 presents results of experiments. Section 7 discusses related work. Finally, Section 8 provides conclusion with directions for future work.

2 Shrack Architecture

Shrack is designed following the real world research collaborations, assuming that researchers who are interested in similar research areas are willing to share their resources and knowledge to keep track of new documents in the area. Document metadata are used for peers to learn and keep track of new documents. The architecture from a peer's perspective is shown in Figure 1.

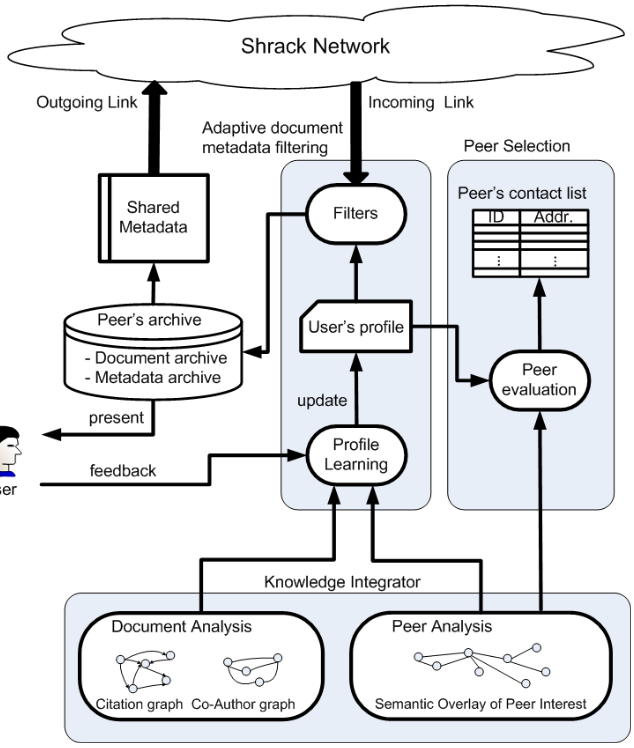

Figure 1: Shrack Peer Architecture

Shrack network is a peer-to-peer network that supports collaboration to keep track of new documents injected by peers into the network. The details of the Shrack network are described in Section 3. *Outgoing links* of peer are accessible through a server port for other peers to contact for pulling shared metadata from. *Incoming links* are connections to other peers that the peer pulls shared metadata. *Shared metadata* are document metadata that are made accessible to other peers. *Peer's archive* is a local repository containing documents and the associated document metadata of interest to the peer. *User's profile* represents the research interests of the peer user. *Peer's contact list* maintains the addresses of peer neighbours.

Three main modules are included in the Shrack architecture to customize disseminated information according to the user's interests, namely, *adaptive document metadata filtering*, *peer selection*, and *knowledge integrator*.

To reduce information overload, the adaptive document metadata filtering module filters received document metadata that are relevant to the user's interest and stores them in the peer's archive. As the system filters metadata, it also refines its knowledge about the user's interests based on the user feedback, using a *profile learning* sub-module. We are exploring techniques used in adaptive document filtering and semi-structured document classification to develop our profile learning and filters.

The peer selection module evaluates and selects peer neighbours providing document metadata that are of interest to the user. A peer selects its neighbours from among other peers with similar interests. The interests of a peer can be evaluated by the information that the peer shared.

The knowledge integrator module extracts and integrates information received from the peer's neighbours into the local knowledge base, which is composed of two main submodules, called *document analysis* and *peer analysis* modules. The document analysis module analyzes the citation graph and the co-authorship graph of documents, and sends this information to the profile learning to update the user's profile. Similarly, the peer analysis module analyzes the interests of peers and creates a local view of a semantic overlay of peer interests, which can be used by the peer analysis module.

3 Shrack Network

Shrack is an unstructured peer-to-peer system. Each peer is associated with an individual researcher or a research organization. Shrack is a purely decentralized network. There is no central directory that maintains global knowledge. Shrack topology is organically formed according to peer interests and collaboration. Peers having common interests and who are willing to collaborate establish pull connections to one another, which are represented as directed edges from information provider peers to peers that receive information. Receiver peers maintain contacts to their information provider peers, which are their neighbours. Receiver peers initiate pull communication with their neighbours. The Shrack network is dynamic, unbounded and may contain cycles. Peers voluntarily join and leave the system. It is unbounded in the sense that no peer can contain information about the complete global network topology, which may be considered infinite from our point of view.

3.1 Joining or Leaving Shrack

To join Shrack, each peer first obtains a Shrack contact address from potential neighbours with whom it has real world collaboration. Researchers can exchange Shrack addresses similarly to exchanging their email addresses or telephone numbers. After joining Shrack, peers learn about new neighbours from disseminated information or from their current neighbours. Peers can leave the network any time without notice. If a peer gets no response from its neighbour after multiple attempts, the neighbour will be removed from the peer's list of neighbours.

3.2 Publishing or Retrieving Documents

To publish a document, a publisher peer pushes the document metadata into its shared metadata directory. When a receiver peer pulls the shared metadata of the peer publisher, then the document metadata are distributed. With successive pulls, the document metadata will be distributed to connected peers. When a peer receives new metadata items, filters will automatically classify document metadata as relevant to the peer or not, based on the peer's profile. The relevant document metadata are stored in the peer's archive, and then presented to the user. The peer can automatically download the relevant documents, or the user can later retrieve documents of interest using information about the document's location contained in the document metadata. An example of when a peer p_1 publishes a document d_1 and its metadata item is disseminated to peer p_2 is shown in Figure 2.

4 Pull-Only Information Dissemination

With pull-only communication, peers in the network cannot push information to other peers. The information will be transferred only by requests. In this way, peers can avoid receiving some non-relevant information or receive it when they are not ready. Moreover, publisher peers need to maintain their reputations, so that other peers will keep pulling information from them. Peers that do not provide good information will be automatically disconnected as nobody wants to pull their shared information.

We propose a pull-only information dissemination protocol for peers to keep track of new documents among peers having similar interest. In our model, peers in the network are keeping track of the new documents for the receiving peer user in advance. In most existing peer-to-peer systems a peer searches for a document when the user requests. We claim that pull-only information dissemination is suitable for the Shrack because a researcher's interests usually persist over a long period of time and change slowly with time.

The pull-only information dissemination is very simple. Each peer periodically pulls information of new document metadata from their neighbours and also makes them available for other peers to pull. However, we need to have a mechanism to ensure that the disseminated information will be terminated and it will not circulate in the system forever. At the same time, peers should receive information of new documents of interest. We also need to enable the receiver peers to learn about the original source of information, so that they can use this information to retrieve the whole documents. We define the document metadata fields described in Table 1.

Each peer has a predefined *pull interval*, specifying how frequently it wishes to automatically pull its neighbours for new documents. The document metadata items will be disseminated to other peers by successive pulls by them. Since the Shrack network is unbounded and may contain cycles, a system-wide predefined $maximum\ hop\ count$ is used to define the maximum number of hops a metadata item can travel from the originating peer. To allow peers to learn and update their pull connections to be closer to the originating peers providing information of interest, each document metadata item contains a $hop\ count$ which shows the number of hops the document metadata item has travelled from the originating peer. Each time a peer pulls a metadata

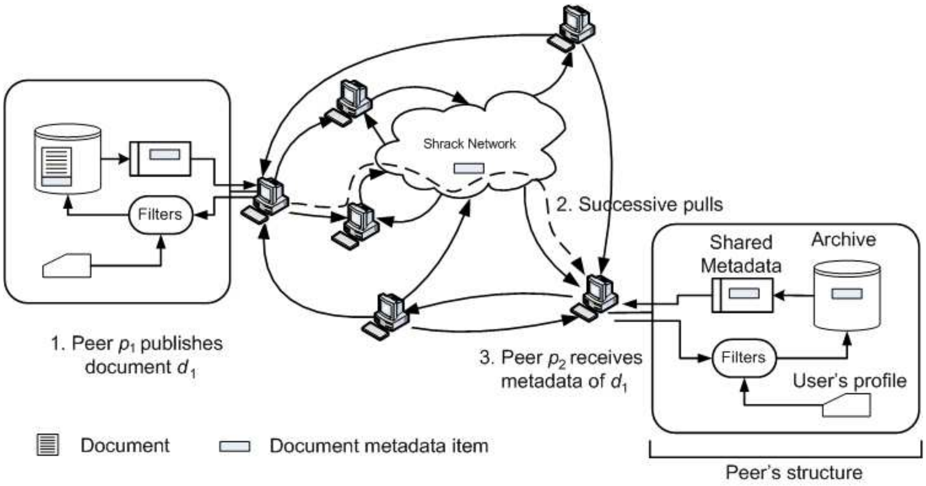

Figure 2: Peer p_1 publishes a document d_1 and the document metadata is disseminated to peer p_2

Table 1: Document Metadata Fields

Fields	Descriptions
$metadata\_id$	unique metadata identifier
$peer\_publisher\_contact$	contact of the peer that publishes the data item such as a URL or an IP address
URI	Uniform Resource Identifier of the document
$document\_description$	Document Content Representation
max_hop_count	the maximum number of hops the metadata item can travel from the peer publisher
hop_count	the number of hops the metadata item has travelled from the peer publisher to the current peer

item, it will increase the metadata item's hop count by one. Peers should update their neighbours if they receive metadata items of documents of interest that have a *hop count* closer to the *maximum hop count*, which is done by the peer selection module as mentioned in Section 2.

A pseudocode of the pull procedure is defined in algorithm 1. In this work, we assume that peers pull metadata items from all of their neighbours using the same pull interval. In the future, pull intervals might be different for each neighbour. Each peer keeps metadata identifiers of the metadata items already seen as a history list. At each pull interval, each peer pulls metadata items from all of its neighbours that arrived after the last time the peer pulled.

Algorithm 1 Pull Procedure of peer p

```
for each neighbour np[i] of peer p do
  peer p pulls a set of document metadata items, m,
  from np[i]'s shared metadata that arrived at np[i] af-
  ter the last time p pulled metadata items from np[i]
  for each metadata item m[j] that is new to p do
    if m[j] is relevant to the user's interest then
      keep m[j] in the local archive
    end if
    increase the m[j].hop_count by one
    if m[j].hop_count is less than the predefined
    max_hop_count then
      add m[j] to the shared metadata of p.
    end if
  end for
end for
```

When receiving each metadata item, the peer checks if the item is new by comparing the metadata item's identifier with its history list. If the identifier of the metadata item is not in the history list, it will be detected as a new metadata item. The new metadata items' hop count will be increased by one. The new metadata item will be compared with the peer's profile. If the metadata item matches the peer profile, the peer will keep that metadata item in the peer local archive. After that, all received metadata items that are new, not only the ones that are selected by the filter, will be buffered in the receiver peer's shared metadata directory for further dissemination to other pulling peers, if they have hop count less than the predefined maximum hop count.

Security and Access Control: Currently many authors keep their documents available on the web anyway, in which case the purpose of Shrack is to improve visibility of the documents. According to the nature of Shrack, which is that peers share only document metadata and use pull-only communication, access is fully controlled by the publishing peers. In practice, such controls can take different forms. Authenticity control of shared document metadata aims to protect the reputation of the publisher. Access to the full documents can be restricted to a group each peer may select, aiming to protect intellectual property rights.

5 Performance Evaluation

In this section, we define parameters and performance metrics to evaluate performance of the pull-only information dissemination.

Parameter Definitions: There are two types of parameters, input parameters (set by each peer) and output parameters (measured during the operation of Shrack).

Input parameters include (1) *pull interval*—time interval between the successive pulls by a peer, (2) *publishing rate*—average number of documents that a peer publishes per time unit, (3) *size of neighbourhood*—number of neighbours that a peer pulls document metadata from, and (4) *maximum hop count*—the maximum number of hops a document metadata item can travel from the peer publisher.

Output parameters include (1) *pull delay*—average time from when a metadata item is published to when a peer first sees the metadata item and (2) *Metadata path length*—average hop count of metadata items when a peer first sees them.

We introduce pull delay to represent average time it takes for peers to learn about new documents of interest that are available in the system. The pull delay is affected by the pull interval and the number of peers in the system. In our experiment, the publishing rate does not affect the pull delay but it is part of our experimental environment. We are interested in assessing the behaviour of the pull delay as the number of peers in the system increases. With a large number of peers the delay could be prohibitively large, e.g., it could be months or years, and this is why we believe that the pull delay metric is meaningful.

Performance Metrics: Since the proposed dissemination protocol is based on pull-only communication in a purely decentralized network, we initially study how well information is disseminated among peers in the network, when the number of peers increases. We assume that every peer is potentially interested in being informed about all documents in the system. The scalability and reliability of the system will be used to measure the performance and suitability of the proposed dissemination protocol when the network size increases. Definitions of the performance metrics are defined as following.

- *Scalability of Information Dissemination:* The dissemination protocol is scalable if an average pull delay over every peer follows a logarithmic function of network size.
- *Reliability of Information Dissemination:* The dissemination protocol is reliable if metadata items are disseminated to every peer in the network. We define the *dissemination coverage* to be the ratio between the average number of peers that receive the metadata items to the network size. Therefore, the dissemination protocol is reliable when the system has 100% dissemination coverage.

To further investigate the behaviour of the dissemination protocol, we also measured the metadata path length to observe the average pathlength associated with pulled items.

6 Experimental Results

We built a simulation of Shrack on PeerSim [7]. "Cycle" is a time measurement unit which could be of any fixed duration. We test the system under the assumption that the network is static, i.e. every peer stays continuously in the network. Pull delay depends on pull intervals and network communication delays. We assume that the point-to-point network communication delay between two peers is negligible because it is several orders of magnitude smaller than the pull interval. For instance, the point-to-point network delay could be a few milliseconds, while the pull interval could be several hours or days. Hence, in our experiments, pull delays depend only on the pull intervals.

We analyze the performance of the information dissemination protocol in two scenarios. First, we test its performance on flat models, where there are no super peers in the system. Second, we test the performance in super-peer models, where peers form collaboration through their super peers, and compare the results with the flat model.

6.1 Experimental Setup

In this section, we describe the experimental setup for flat and super-peer models.

Flat Model: Two peer-to-peer network models are used for testing the performance of the dissemination protocol including a small-world network and a random network. Social network is more likely to give rise to a small-world network as opposed to a random network, which is a simple model to set up a network topology without any preference. In the first case, we model peer neighours based on the small-world model of Watts and Strogatz (WS) [11]. In the second case, peer neighbours are assigned randomly.

In both models, the power-law distribution is used to model the number of neighbours of each peer, where the exponent of the power law is 2.7 (based on [2]), and the minimum and the maximum size of the neighbour lists are

3 and 20, respectively. Each peer publishes documents according to a Poisson distribution with an average publishing rate of one document per 30 cycles, and periodically pulls metadata every 2 cycles (but at a different (random) starting time for each peer). The maximum hop count is 20. A summary of the parameter setup for the experiment is shown in Table 2. We vary the network size and measure the dissemination characteristics observed by each peer and the dissemination coverage.

Table 2: Experimental Setup

Property	Value
Pull Interval	2 cycles (periodically)
Publishing Rate	1 documents per 30 cycles (Poisson distribution)
Size of Neighbours	3 - 20 (power-law distribution; with exponent equal to 2.7)
Max. Hop Count	20

Super-Peer Model: Super peers are peers that have high resources and availability. Super peers can be thought of as research organizations which keep a collection of documents of interest to researchers in their organizations. We study the effect of super peers on local group collaborations in the small-world network model. Each peer is associated with only one super peer. Each super peer periodically pulls metadata only from the peers in its local organization, with the same pull interval as normal (not super) peers. Each normal peer also periodically pulls document metadata from its super peer. In practice, peers may connect to many super peers and super peers can connect to peers outside their groups if they wish. An example of a super-peer model network is shown in Figure 3.

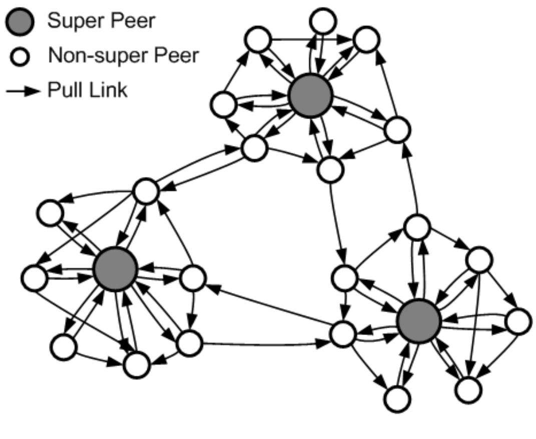

Figure 3: Example of A Super-Peer Model

In the experiments, each normal peer has the same properties defined in Table 2. Only normal peers publish documents. Super peers only support the collaboration among normal peers and build the organization document collections. We test the performance of the system with collaboration of super peers in two configurations. In *Super Peer Model I*, the number of super peers is fixed, at 10, regardless of network size. In *Super Peer Model II*, we fix the number of normal peers associated with each super peer, at 500 regradless of all network size. We vary the number of normal peers, and measure the dissemination characteristics observed by each normal peer.

6.2 Results and Evaluation

The experimental results in flat model show that the proposed dissemination protocol is scalable and reliable because, in both network models, the average pull delay follows a logarithmic function of network size as shown in Figure 4, and the dissemination coverage is 100% as show in Figure 5. This behaviour of the average pull delay is a direct consequence of the way document metadata items propagate through the network. In general, more than one peer pulls shared metadata of a given peer. Hence, document metadata items propagate down in a tree-like topology through the network at an exponential rate. This argument is supported by the observation that the average metadata path length also follows a logarithmic function of network size as shown in Figure 6.

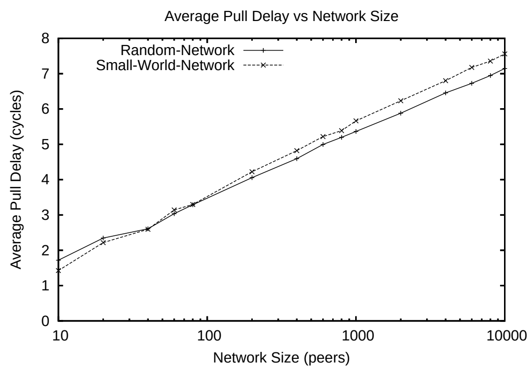

Figure 4: Average Pull Delay

We compare the performance of the super-peer models with the flat model in the small-world network. The results show that, in both configurations, super-peer models help improve the reliability and the scalability of the system. The network achieves 100% dissemination coverage. The average pull delay observed by normal peers highly depends on the number of super peers, as shown in Figure 7. In Super Peer Model I, normal peers observe an almost constant average pull delay in all network size. In Super Peer Model II, the average pull delay is about the same when the network size equals to 200 and 400 peers, as both of them

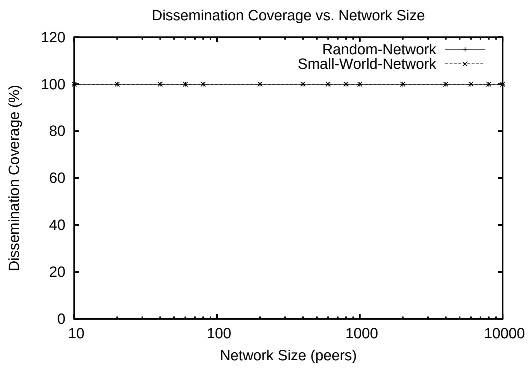

Figure 5: Dissemination Coverage

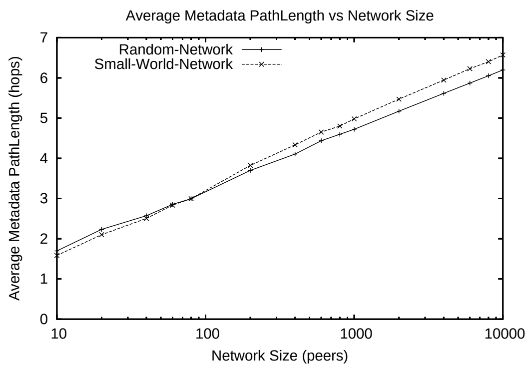

Figure 6: Average Matadata Path Length

have only one super-peer group. The same behaviour repeats when the network size equals to 600 - 1000 peers, as they all have two super-peer groups. Subsequently, when the network size increases to 2,000 - 10,000 peers, the average pull delay increases following a logarithmic function of the number of super-peer in the system.

Our experiments were conducted in the ideal situation where peers can process every pull request. Hence, the experimental results show that the system is highly scalable. Although the propagation path of the metadata items is similar to message routing in flooding-based networks, the behaviour is rather different, as explained futher.

Flooding-based message routing is a push based protocol, which requires peers to forward messages immediately when the message arrives. As a result, when a peer receives the message and it is overloading, the peer will drop the received message, which causes the performance of the flooding protocol to deteriorate. In our protocol, peers will pull information from the shared buffer of other peers, which will be replicated in several peers having similar interests. Peers can choose to pull information when they are ready and can select which peers they want to pull (time is not critical for our application). Moreover, in general P2P

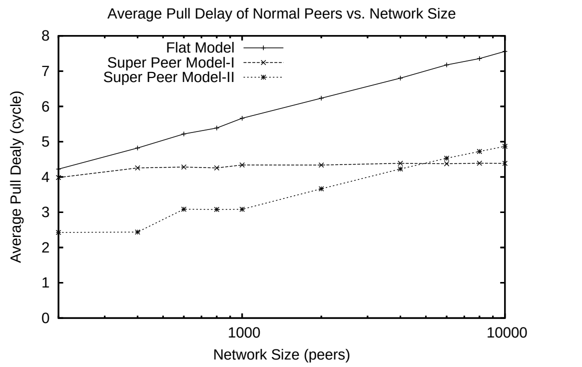

Figure 7: Average Pull Delay of Normal Peers in Super-Peer Models

file-sharing systems, the number of original root messages equals to the number of requests generated by each peer. In Shrack, the number of original root messages equals to the number of publications. Requests in general file-sharing systems might be for the same document but they are created in multiple times or different requests, which result in creating multiple root messages. In Shrack, only new publications will be propagated among peers.

Our pull-only protocol is more similar to the gossip protocol [9] than the flooding-based message routing. The gossip protocol is known to be scalable and reliable for message dissemination in a large-scale network for group communication. Unlike gossip protocol, where peers push messages to peers in their group at random, in our proposed protocol, peers self organize into groups of peers having similar interests and pull information from within the group.

7 Related Work

Several P2P networks have developed to improve data and document sharing over the Internet. Two P2P approaches for data-sharing among researchers based on small-world networks were proposed [6] and [10]. The networks are comprised of several clusters, where each cluster is defined as a community with overlapping data interests. The systems are built to support search and locate documents in the same cluster. In the first approach [6], each peer periodically updates global information of all peers and document locations in the same cluster through a "gossip" protocol [9], a push-based information dissemination mechanism. Hence, each peer can immediately locate documents in the same cluster. In the second approach [10], populated files are replicated among peers in the same cluster. Peers use limited flooding of requests to search for files within their clusters.

Shrack has a similar design as the first approach. However, instead of defining a rigid cluster and requiring peers to maintain global knowledge of every peer in the same cluster, peers maintain knowledge of only the peers from which they wish to pull information.

PlanetP [3], was proposed to improve distributed search in P2P communities. Each peer in the system maintains global addresses of peers in the communities and has global knowledge of an inverted term-to-peer index, which maps terms to peers having documents containing these terms. To search for documents of interest, peers forward a request to a set of peers that contain documents with the requested term based on the inverted index and ranking algorithm.

To improve search in P2P document-sharing networks, selective forwarding of requests to peers based on their semantic topology was proposed in [5]. Peers advertise their expertise to the network. Peers forward a request based on matching the subject of a query and the expertise according to their semantic similarity. Simulation experiments showed that expertise based peer selection with ontology matching and semantic topologies outperform random peer selection.

While there are many initiatives to build P2P systems for research collaborations, the existing systems focus on document searching. In Shrack, we focus on document tracking and filtering based on users' interests.

8 Conclusion and Future Work

We propose a P2P framework for document sharing and tracking, named Shrack. The goal of Shrack is to provide an infrastructure for researchers to autonomously help each other to share and keep track of new documents based on their interests. In this paper, we present the architecture of Shrack including its components and functionality. We also propose a pull-only information dissemination protocol in the system.

We evaluate the proposed protocol based on scalability and reliability criteria. A Shrack simulation is built for the performance analysis. The proposed protocol is tested in two scenarios, flat and super-peer models. The experimental results show that the proposed protocol is scalable and reliable, as the average pull delay of the system follows a logarithmic function of the network size and the network has 100% dissemination coverage. Moreover, the super-peer model shows improvement over direct flat collaboration.

The experiments shown in this paper are initial tests to observe the performance of the proposed pull-only information dissemination protocol in static environments. We will further analyze the performance of the protocol in dynamic environments. Moreover, we currently assume that peers are interested in being informed about all documents in the system, and we use mathematical models to create peer topology. The next challenging steps are to develop the three main modules, including adaptive document metadata filtering, peer selection, and knowledge integrator, for peers to efficiently disseminate and filter shared metadata, form into group of peers having similar interests and build their local knowledge base, respectively.

9 Acknowledgement

We would like to thank Dr. Nick Cercone for his valuable comments and support. This research was funded by grants from the Natural Sciences and Engineering Research Council of Canada, the MITACS Network of Centres of Excellence, and IT Interactive Services Inc. We also would like to thank the anonymous referees for their constructive feedback.

References

[1] H. Balakrishnan, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica. Looking up data in P2P systems. *Commun. ACM*, 46(2):43–48, 2003.

[2] A. Broder, R. Kumar, F. Maghoul1, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph Structure in the Web: Experiments and Models. In *9th World Wide Web Conference*, May 2000.

[3] F. M. Cuenca-Acuna, C. Peery, R. P. Martin, and T. D. Nguyen. PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities. In *Twelfth IEEE International Symposium on High Performance Distributed Computing (HPDC-12)*. IEEE Press, June 2003.

[4] Gnutella.com. `http://www.gnutella.com/`, Accessed May 2006. OSMB, LLC.

[5] P. Haase, J. Broekstra, M. Ehrig, M. Menken, P. Mika, M. Plechawski, P. Pyszlak, B. Schnizler, R. Siebes, S. Staab, and C. Tempich. Bibster — A Semantics-Based Bibliographic Peer-to-Peer System. In S. A. McIlraith, D. Plexousakis, and F. van Harmelen, editors, *Proceedings of the Third International Semantic Web Conference (ISWC 2004)*, pages 122–136, Hiroshima, Japan, Nov. 2004. Springer.

[6] A. Iamnitchi, M. Ripeanu, and I. Foster. Locating Data in (Small-World?) Peer-to-Peer Scientific Collaborations. In *1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, 2002.

[7] G. P. Jesi. PeerSim P2P Simulator. `http://peersim.sourceforge.net/`, Accessed May 2006. Faculty of Computer Science, University of Bologna.

[8] KaZaa.com . `http://www.kazaa.com`, Accessed May 2006. Sharman Networks.

[9] A.-M. Kermarrec, L. Massoulie, and A. J. Ganesh. Probabilistic Reliable Dissemination in Large-Scale Systems. *IEEE Transactions on Parallel and Distributed Systems*, 14(3):248–258, Mar. 2003.

[10] J. Mitre and L. Navarro-Moldes. P2P Architecture for Scientific Collaboration. In *the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE'04)*, 2004.

[11] D. J. Watts and S. H. Strogatz. Collective Dynamics of "Small-World" Networks. *Nature*, 393(4):440–442, June 1998.