WindChimer: The Partially Centralized and Controlled P2P System For Better Discovery

Jedsada Phengsuwan and Naiyana Sahavechaphan
Large Scale Simulation Research Laboratory
National Electronics and Computer Technology Center (NECTEC)
112 Thailand Science Park, Pahon Yothin Rd.
Klong 1, Klong Luang, Pathumthani 12120 Thailand
Email: jedsada.phengsuwan|naiyana.sahavechaphan@nectec.or.th

Abstract

Rich information from various data sources has been recognized for enabling people to perform an efficient decision-making and problem-solving. According to thousands of data sources and numerous information types available at the edge of the Internet, the discovery of all *potential data sources are needed to facilitate the procurement of* rich *information. However, current P2P approaches are limited to the discovery of potential data sources in which the number of data sources does not cause tremendous consequences for decision-making and problem-solving. In this paper, we thus propose* WindChimer*, a partially centralized and controlled P2P system that hybridly combines network topology of structured and unstructured P2P systems. Our experiments showed that the partially centralized and controlled topology enables an efficient discovery of* all *potential peers as per a given information type. This was accomplished via (i)* acceptable *number of messages sent throughout the network; (ii)* high *messaging accuracy for propagating the message to potential peers; and (iii)* effective *scalability when adding more peers. However, it is currently applicable for* low *degree of transient peer population, and* sustainable *data sources of organizations only.*

Keywords: Peer-to-Peer, P2P, information-sharing, information discovery

1 Introduction

Information from various data sources has been recognized for enabling people to perform an efficient decision-making and problem-solving. For example, NRCT, NSTDA and HSRI are organizations that fund research projects proposed by researchers in a country. Based on a constraint that no similar projects would get funded, each organization should thus need project information from the rest organizations to facilitate its decision whether to fund a given project. For simplicity, we term information collected (integrated) from different data sources as *rich* information.

Currently, there are thousands of data sources and hence numerous information types[1] as well as their corresponding information content (or information, for short) available at the edge of the Internet. Each data source can independently provide information as per any information types regarding to its business. For each information type, on the other hand, its corresponding information can be uniquely or similarly published by one or more data source(s). According to this, we believe that the discovery for *all* possibly potential data sources as per a given information type would be the *first step* towards the realization of its *rich* information. In this work, we thus assume that each data source provides *information types* while their corresponding information would be focused in the future work.

Today, peer-to-peer (P2P) has rapidly grown to become one of the most interesting technology that promotes the *discovery* and *sharing* of information available at the edge of the Internet [12]. Various P2P approaches ranging from *unstructured* to *structured* networks have thus been proposed in the literature. Specifically, *unstructured* P2P [1, 2, 3, 4, 13] is much appropriate for accommodating highly-transient peer populations wherein peers can provide information as per any information types. However, its search mechanisms [11, 6] still contact a large number of peers. To address this, *structured* P2P [7, 8] was introduced. Conversely to unstructured P2P, the potential peers can be discovered in O(ln n), while the structure is hard to be maintained in the face of a very transient peer populations [10]. In addition, structured P2P is solely applica-

[1] Examples of information types are researcher, project, traffic, crime and media information types. Information of a project information type would be any instances in the form of <id, name, description, status>.

ble for peers containing unique keys of information types (see Figure 1(a)), while peers containing duplicated keys are prohibited (see Figure 1(b)).

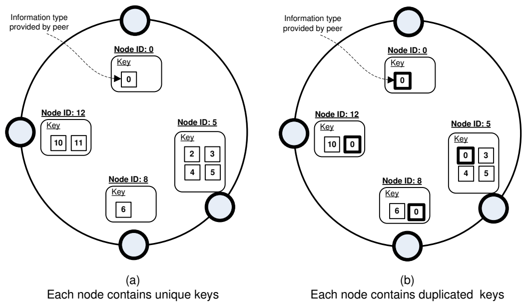

Figure 1. Structured P2P for Different Case Studies.

While these P2P approaches take steps into the right direction, they are limited to the discovery of potential data sources in which the number of data sources does not cause tremendous consequences for decision-making and problem-solving. To achieve an efficient discovery for *all* possibly potential peers that provide any information, we believe that the network topology of data sources is a significant aspect and hence must be well-designed. In this work, we thus propose a novel partially centralized and controlled P2P system, namely *WindChimer*[2]. Particularly, *WindChimer* hybridly combines network topology of structured and unstructured P2P systems – "*Some peers are connected in a controlled manner and act as gateways for the rest peers. The rest peers, on the other hand, are connected to their corresponding gateways in an uncontrolled manner*". Our experiments showed that the partially centralized and controlled topology enables an efficient discovery of *all* potential peers as per a given information type. This was accomplished via (i) *acceptable* number of messages sent throughout the network; (ii) *high* messaging accuracy for propagating the message to potential peers; and (iii) *effective* scalability when adding more peers.

Roadmap: The rest of this paper is organized as follows: Section 2 defines our novel P2P system along with the search algorithm. Experimental evaluation is given in Section 3. We conclude in Section 4.

2 WindChimer

In this section, we detail topology of *WindChimer* together with the techniques to efficiently discover all potential data sources having the desired information type.

[2] It is named for our novel P2P system as the figure of its network topology looks like *Windchime*.

2.1 Network Topology and Architecture

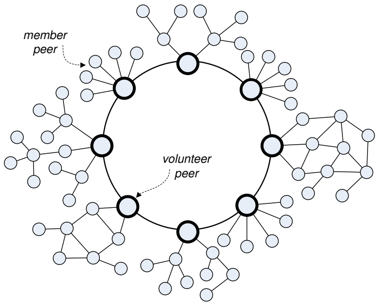

Figure 2. The Conceptual *WindChimer* Topology.

Figure 2 shows the conceptual network topology of *WindChimer* that hybridly combines network topology of structured and unstructured P2P systems – "*Some peers are connected in a controlled manner and act as gateways for the rest peers. The rest peers, on the other hand, are connected to their corresponding gateways in an uncontrolled manner*". Specifically, there are two significant roles each peer can play as per each individual information type i contained in itself: *volunteer* or *member* role. For simplicity, we term a physical peer (or peer) p acting the volunteer or member role for an individual information type i as a *logical volunteer peer* (or volunteer peer) p_v or a *logical member peer* (or member peer) p_m, respectively. The overall number of volunteer and member peers of a peer corresponds to the number of information types contained in it.

Volunteer Peer. A *volunteer peer* p_v is a logical peer that be a representative of an information type i contained in a peer p. Essentially, a peer p is selected as a volunteer peer p_v for an information type i based on the conditions that: (i) there must be an information type i contained in a peer p; (ii) a volunteer peer representing an information type i has not yet appeared in the network; and (iii) the performance of a peer p (i.e. bandwidth and reliability) is above a specified threshold – the more performance, the more chance it to be a volunteer peer.

Technically, a volunteer peer p_v is responsible for efficiently discovering the potential volunteer peer that represents the desired information type i by sending queried messages to some volunteer peers. To achieve this, a volunteer peer p_v should connect to a set of certain volunteer peers based on a specified rule. For example, like Chord [8], volunteer peers are connected in term of ring topology. In addition, a volunteer peer p_v for an information type i should contain (i) the *finger table* that provides information used for peer looking up mechanism based on DHT; and (ii) a *neighbor table* that provides a collection of its neighbor peers publishing such information type i.

Recall that a volunteer peer p_v represents a unique information type i. To facilitate an efficient information discovery, it is therefore necessary that the number of volunteer peers in the network should be equivalent to the number of information types published throughout the network.

Member Peers. A *member peer* p_m is a logical peer that be a representative of an information type i contained in a peer p. Essentially, a peer p is selected as a member peer p_m for an information type i based on the conditions that (i) there must be an information type i contained in a peer p; and (ii) a volunteer peer representing an information type i has already appeared in the network.

Technically, a member peer p_m is responsible for navigating the queried message to its volunteer peer p_v as well as propagating the message to potential neighbor peers in an efficient manner. To achieve this, a member peer p_m should connect to its corresponding volunteer peer p_v and other member peers that represent an information type i, similarly to what it represents. In particular, a member peer p_m can either *directly* connect to its volunteer peer p_v, or *indirectly* connect to its volunteer peer p_v via one or more member peer(s) that represent(s) similar information type i in the way that the member peer p_m itself should be traced back to its volunteer peer p_v via the shortest path. In addition, a member peer p_m for an information type i should contain a *neighbor table* that provides a collection of its neighbor peers publishing such information type i.

2.2 Network Reconfiguration

The *WindChimer* network needs to be reconfigured, to reflect its architecture as described in Section 2.1, when a peer (i) joins the network; (ii) removes the network; and (iii) updates its configuration. Due to the space constraint, in this section we detail the addition of peers while others can be seen in [5].

Based on the $joinTopology$ algorithm (see details at [5]), Figure 3 illustrates the network reconfiguration when a new peer p_4 that contains a set of information types $\{i_4, i_8\}$ wants to join the network. Assuming that an existing topology is shown in the Figure 3(a). Here, there are five peers $\{p_0, p_1, p_2, p_8, p_{10}\}$ and three information types $\{i_0, i_8, i_{10}\}$. The peers p_0, p_8 and p_{10} are volunteer peers representing information types i_0, i_8 and i_{10}, respectively. Each volunteer peer has a *Chord* based *finger table* which is used for DHT mechanism. Peers p_1 and p_2 are member peers that contain information i_8. Their neighbor tables provide a list of neighbors connecting to them. Figure 3(b) illustrates the new network configuration when a peer p_4 joins the network. For an information type i_4, a volunteer peer representing an information type i_4 is not available. The peer p_4 will then be added to the topology as a volunteer peer that represents an information type i_4. Its *finger table* for an information type i_4 is thus created and assigned to a peer p_4. In addition, based on *Chord* algorithm, only peers p_0 and p_{10} will be notified for updating their *finger tables*. Lastly, for an information type i_8, there exists a volunteer peer p_8 representing information for i_8. A peer p_8 then selects for an appropriate neighbor peer p_1 using random walking technique through its neighbor list that contains an information type i_8. Once obtaining a peer p_1 as an appropriate neighbor peer, peers p_4 and p_1 will add the information of each other to their *neighbor tables*.

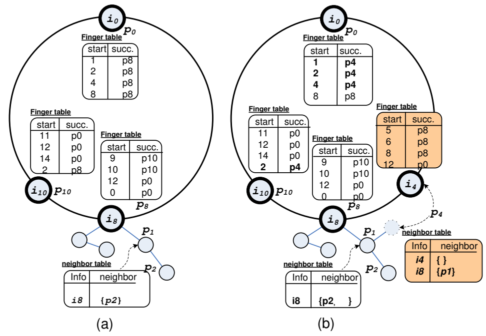

Figure 3. The Network Reconfiguration.

2.3 The Discovery of Potential Peers

The goal of peer discovery is to propagate the queried message to *all* potential peers that provide the desired information type in an efficient manner. Based on the network topology of *WindChimer* as defined in section 2.1, the discovery mechanism of potential peers must thus combine (i) *DHT algorithm* for seeking a potential volunteer peer representing the desired information type; and (ii) *flooding algorithm* for propagating a queried message to all potential member peers of such volunteer peer.

Based on the above *discoverPeers* algorithm (see details at [5]), Figure 4 illustrates the path for discovering potential peers having the desired information type i_d. Here, the queried message m_q is initially sent to the member peer p_3 that does not contain the desired information type i_d. The peer p_3 then discovers for its volunteer peer using the shortest path technique. The volunteer peer p_1 is selected as the path length from p_3 to p_1 is shorter than it from p_3 to another volunteer peer p_4. Once obtaining the volunteer peer p_1, the peer p_1 utilizes its finger table to find the volunteer peer, p_7, representing the desired information i_d. Next, the volunteer peer p_7 propagates the message m_q to its neighbors p_8 and p_{10}. The peers p_8 and p_{10} further distributes the message m_q to their neighbors in a similar manner. Particularly, the peer p_8 sends the message m_q to its neighbor peer

p_9, while the peer p_{10} sends it to peers p_9 and p_{11}.

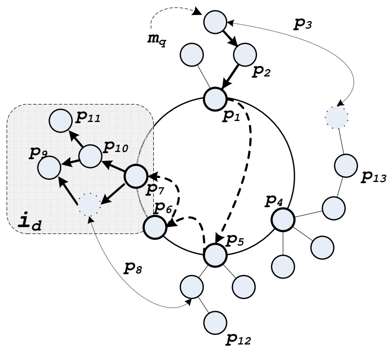

Figure 4. The Discovery of Potential Peers.

3 Preliminary Experimental Evaluation

The goal of the *WindChimer* is to provide an efficient discovery of *all* potential peers that provide the desired information type. We conducted several experiments to evaluate the potential benefit of the *WindChimer* over existing topologies with respect to unstructured P2P systems alone[3]. In this section, we describe our experimental setup and methodology together with our experimental results.

3.1 Experimental Setup and Methodology

To evaluate our approach, we developed the $p2pEval$ system that simulated 3 essential topologies: (i) purely decentralized P2P (i.e. Gnutella) – in that all peers in the network perform exactly the same [10]; (ii) partially centralized P2P (i.e. Kazaa) – in that some peers act as local central indexes for file shared by local peers [10]; and (iii) partially centralized and controlled P2P, *WindChimer* – in that some peers act as gateways and are connected in a controlled manner while the rest peers are connected to their corresponding gateways in an uncontrolled manner. The $p2pEval$ system together with their underlying search algorithms is implemented by Java (JDK 1.5) and deployed on a standalone PC Pentium IV 3.2 GHz with 1 GB RAM running Microsoft Windows XP. Specifically, the three topologies are configured based on the same set of peers and information types. In this work, we use MDL instances[4] [9] to represent information types. Based on each topology, the $p2pEval$ system receives as input (i) the queried message that specifies the desired information type in term of a MDL instance; and (ii) the starting peer that receives the queried message, and returns a set of peers having such information based on the desired MDL instance as output. These potential peers are discovered using existing search algorithms such as flooding, modified-BFS [11], random walk [6] and DHT-flooding, our proposed search algorithm defined in Section 2.3. The first three algorithms are applicable for purely decentralized and partially centralized P2P while the last one is for *WindChimer*.

Network Topology Configuration. In this evaluation, we simulated the three topologies based on the same set of $10,000$ - $50,000$ peers and 100 MDL instances (information types). Here, each peer was assigned to publish information with respect to a random MDL instance, with the exception that a certain set of 100 peers providing information based on the desired MDL instance mdl_d. Additionally, in

- purely decentralized P2P, each peer was connected to a random number, ranging from 1 to 3, neighbor peers[5].
- partially centralized P2P, a set of 100 superpeers were randomly chosen. Each superpeer was connected to 2 random neighbor superpeers[6]. In addition, each leaf peer was connected to a random superpeer in the *star*-like fashion.
- partially centralized and controlled P2P (*WindChimer*), a set of 100 volunteer peers[7] providing the distinct 100 MDL instances were randomly selected. The rest peers were connected to the corresponding volunteer peers in 3 patterns: *star*, *tree* and *graph*.

Measure Of Efficiency. To evaluate the benefit of *WindChimer* over existing topologies, we measured (i) the number of messages sent throughout the network. In particular, three different message types are evaluated: *request*, *success* and *failure* messages [10]; and (ii) the accuracy for propagating the queried message to potential peers having the desired MDL instance. Here, we compared the number of relevant peers R, that had the desired information and received the messages, with the overall number of peers N that the messages were sent to. Based on the cardinalities of these sets, the messaging accuracy $\frac{R}{N}$ was computed.

[3]Structured P2P is not applicable if peers can provide any information types

[4]A MDL instance provides the standard information schema as per a certain information type. It has been used in our Information Grid project to facilitate an integration of any information.

[5]The maximum of 3 neighbor peers were selected as it performed the best (see [5]).

[6]A set of 100 superpeers were needed to make it corresponded to the number of volunteer peers in *WindChimer*, and the maximum of 2 neighbor superpeers were selected as it performed the best (see [5]).

[7]A set of 100 volunteer peers were needed as per the specification of *WindChimer* that the number of volunteer peers should correspond to the number of MDL instances.

3.2 Experimental Result

A series of experiments were conducted to evaluate the potential benefit of *WindChimer*.

3.2.1 The Topology Efficiency

The first set of experiments measured the effectiveness of different topologies – purely decentralized PD, partially centralized PC, and partially centralized and controlled PCC (*WindChimer*) – in order to discover all peers having an information with respect to the desired MDL instance. The effectiveness was defined as the ability of topologies to propagate less number of messages – the less number of messages, the less network bandwidth utilization. The experiments were conducted 10 times for each $10,000$-peer topology alone wherein each iteration was initiated with the random starting peer and the queried message with the desired MDL instance mdl_d. Additionally, the partially centralized and controlled topology was constructed with the member peer connection in the star-like fashion.

Figure 5 shows the average number of messages – request, success and failure messages – sent throughout three topologies with their best search algorithms to discover potential peers having the desired MDL instance at the high hit rate (see details in [5]). In particular, flooding algorithm was selected for PD and PC as 3 and 2 neighbors were applied respectively, while the PCC was performed on the combination of DHT and flooding techniques as defined in the *discoverPeers* algorithm (see Section 2.3). The x-axis has the topologies and the y-axis is the number of messages. Here, PD performed the worst with the total of $32,786$ messages. The PC required $1,140$ messages, performing 28 times better than PD. This was due to the fact that the messages were not propagated to most non-potential peers filtered out by every single superpeer. The PCC performed the best with 401 messages, offering approximately 3 times better than PC. This was mainly because the messages were intentionally sent to all potential peers via visiting at most $ln(n)$ volunteer peers wherein n is the number of volunteer peers.

3.2.2 The Topology Accuracy

The second set of experiments measured the accuracy of three different topologies – PD, PC, and PCC (*WindChimer*) – in order to discover all peers having the desired MDL instance. The accuracy was defined as the ability of topologies to distribute the messages toward the relevant peers having the desired MDL instance. It is thus evaluated based on the messaging accuracy as previously defined. These experiments were conducted in a similar manner to the first set of experiments.

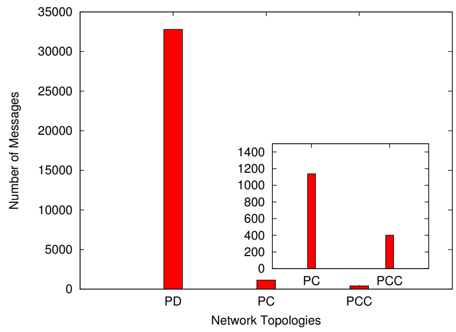

Figure 5. The Topology Efficiency.

Figure 6 shows the messaging accuracy of three topologies with their best search algorithms to discover all potential peers. The x-axis has the topologies and the y-axis is the messaging accuracy. Here, PD performed the worst with nearly 0% accuracy. PC offered 50% accuracy. PCC performed the best, providing 98% accuracy. This was mainly because messages were potentially sent to relevant peers via firstly propagating messages to irrelevant peers at the minimum.

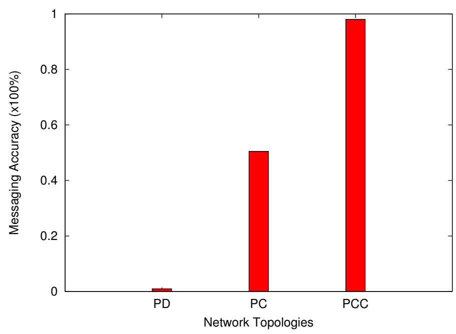

Figure 6. The Topology Accuracy.

3.2.3 The Topology Scalability

The large set of experiments measured the scalability of three different topologies – PD, PC, and PCC (*WindChimer*) – in order to discover all peers having the desired MDL instance. The scalability was defined as the ability of topologies to be practical for large number of peers in the network. This was indicated by the growing number of messages – the less growing number of messages with respect to the increasing number of peers, the more network scalability. The experiments were conducted using various set of the number of peers – $10,000$, $20,000$, $30,000$, $40,000$ and $50,000$ while other set up remained the same with 100 potential peers.

Figure 7 shows the average number of messages – request, success and failure messages – sent throughout three topologies with their best search algorithms to discover potential peers having the desired MDL instance at the high hit rate. The x-axis has the number of peers and the y-axis is the number of messages. Here, PD performed the worst with the number of messages growing in an exponential manner. The PC and PCC were similarly performed, with almost the constant number of messages. However, PCC required less number of messages than PC.

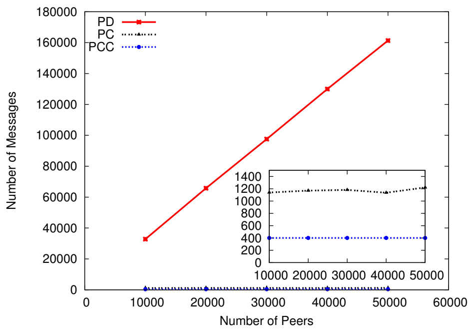

Figure 7. The Topology Scalability when Adding Non-Potential Peers.

Additionally, we measured the topology scalability with the addition of 100-400 more potential peers in $10,000$-peer PC and PCC. Figure 8 shows the average number of messages – request, success and failure messages – sent throughout two topologies – PC and PCC with their best search algorithms to discover potential peers having the desired MDL instance at the high hit rate. The x-axis has the number of potential peers and the y-axis is the number of messages. Here, PC performed the worst with the number of messages increased at the rate of 8.5. PCC performed the best, with an approximate increasing rate of 4. This was mainly because $ln(100)$ instead of 100 volunteer peers were involved in the message propagation, resulting in shorter path from the requested peer to each potential peer – the shorter path, the smaller number of request messages and hence the smaller number of success messages.

4 Conclusion

In this paper, we proposed a novel P2P topology, namely the $WindChimer$, to facilitate an efficient discovery of *all* potential data sources as per an information type. Such discovery is recognized as the *first step* towards the procurement of rich information to support an efficient decision-making and problem-solving. Particularly, $WindChimer$ is a partially centralized and controlled P2P system that

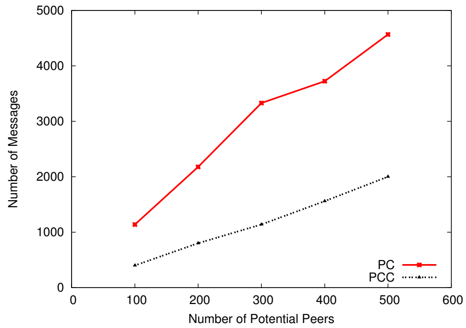

Figure 8. The Topology Scalability when Adding Potential Peers.

hybridly combines network topology of structured and unstructured P2P systems. To obtain the rich information from *all* potential data sources (peers), our experiments showed that $WindChimer$ provides benefits than purely decentralized and partialy centralized topology, in term of (i) number of messages sent throughout the network; (ii) the messaging accuracy for propagating the message to potential peers; and (iii) the scalability when adding more peers. However, it is applicable for *low* degree of transient peer population, and *sustainable* data sources of organizations only.

Future work focuses on (i) the volunteer peer selection based on not only the peer performance but also the network configuration; (ii) the self-organization of network when an existing volunteer (or member) peer joins, downs or removes; and (iii) an efficient peer discovery based on not only information type but also information content.

References

[1] Edutella. http://www.edutella.org.
[2] Gnutella. http://www.gnutella.com.
[3] Kazaa. http://www.kazaa.com.
[4] Napster. http://www.napster.com.
[5] WindChimer. http://www.informationgrid.org.
[6] C. L. et al. Search and Replication in Unstructured Peer-to-Peer Networks, 2001.
[7] I. C. et al. Freenet: A Distributed Anonymous Information Storage and Retrieval System. *Lecture Notes in Computer Science*, 2009:46+, 2001.
[8] I. S. et al. Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications. In *Proceedings of the 2001 ACM SIGCOMM Conference*, pages 149–160, 2001.
[9] J. P. et al. MDL: Marker Description Language. In *Proceedings of the 4th International Conference on Computer Science and Software Engineering (JCSSE)*, 2007.
[10] S. A.-T. et al. A Survey of Peer-to-Peer Content Distribution Technologies. *ACM Computing Surveys*, pages 335–371, 2004.

[11] V. K. et al. A Local Search Mechanism for Peer-to-Peer Networks. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 300–307. ACM Press, 2002.
[12] C. Shirky. What is P2P and What isn't, 2000.
[13] S. Waterhouse. JXTA Search: Distributed Search for Distributed Networks. http://search.jxta.org/, 2001.