

Using Two-tier Bitwise Interest Oriented QRP Data Structure For High Clustering through Interest Similarity in SWB P2P Systems

Tong-Ming, Lim, Yong Liang, Choong and Angela Lee Siew Hoong
 School of Computer Technology
 Sunway University
 5 Jalan Universiti
 Bandar Sunway,
 Selangor, Malaysia

tongmingl@sunway.edu.my, cgyl_202@yahoo.com, angelal@sunway.edu.my

Abstract - The concept of “Small-World” paradigm has been used by many peer-to-peer (P2P) systems to achieve high clustering and low number of hops to arrive to its desired target. This approach intends to improve performance of P2P systems. The Small world concept requires that the architecture of a P2P system to achieve high cluster coefficient and low average hop between any two randomly chosen peers. In this research, we propose an enhanced hierarchical overlay network by incorporating the concept of Small world into the base hierarchical architecture. This research adopts the Query Routing Protocol (QRP) data structure of a hierarchical P2P network by storing the interest information of files in the leaf peers in the network. The QRP in the leaf peers will be aggregated and propagated to the SuperPeer so that interest information could be used to form short-range, medium-range and long-range links with other SuperPeers to achieve low average hop. As peers join and leave the network, a proposed rewiring protocol is used to ensure peers are clustered by interest to form high clustering coefficient so that search activities are yielding higher relevance results in a more predictable fashion. We simulated the proposed small world P2P using test scenarios to evaluate recall rate of the small world P2P system. Our simulated results evaluate and benchmark our proposed Small-World Bee (SWB) overlay network with iCluster [3], Firework Query Model [5] and Limewire [1] to assess performance obtained and goals achieved in the research work so that future research directions could be planned.

Key Words: *QRP, Small-World paradigm, Cluster, Peer-to-Peer System*

I. INTRODUCTION

Information sharing in multi-user distributed systems is dynamic and this is a more challenging problem for peers in P2P systems. To retrieve data accurately and speedily from peers in an overlay network has always been a challenging one. P2P systems are categorized into two (2) main groups: structured and un-structured systems. Structured P2P systems or Distributed Hash Tables (DHTs) systems such as Chord

[10], CAN [9], Pastry [11], and Tapestry [15] are common solutions. They use efficient key-based data retrieval technique. Peers on structured P2P systems connect to other peers using DHT information that directs the search to specific peers that hold the desired data. However, structured P2P systems suffer from high maintenance overhead due to frequent change of membership/contents [12] and heavy peer dependency.

To overcome high peers’ dependency issue, unstructured P2P systems such as LimeWire [1], Gnutella [13], and Freenet [14] were designed to allow peers to self-organize to form overlay networks automatically to reduce maintenance overhead. However, unstructured P2P systems need to broadcast query messages to all peers to search for a desired piece of data, which ultimately cause network message flooding. The “Small-World paradigm” [3, 4, 6, 7, 8] was introduced to improve the search efficiency while maintaining highly peer clustering. In this paper, one of the two “Small-World” paradigm properties will be discussed and presented: high cluster coefficient, while the other property: low average hop between any two randomly chosen peers; will be presented in the next stage of research work. In the small word P2P system, peers in the network must maintain high cluster coefficient by similar interest [4, 14]. In a highly clustered environment, a random shortcut path is created to achieve low average hops [4, 8, and 14]. Each peer in a small world network will maintain short-range and long-range links, where short-range links can be represented as intra-cluster links that connect to peers of similar interest whereas long-range links are used as inter-cluster links as shortcut paths to other remotely related clusters.

Peers will self-organize by periodically executing **rewiring protocol** to achieve “Small-World paradigm” in unstructured P2P systems [3, 6, and 8]. This protocol will maintain intra-cluster *interest* by discarding outdated peers or dissimilar peers. The protocol will also connect to similar peers that are new to the cluster. The goal of the **rewiring protocol** is

to maintain high cluster in the P2P network so that query can route efficiently.

In this paper, we introduce algorithm that allows peers to join and leave the P2P network effectively so that the network can retain its highly clustered characteristics. By introducing “Small-world paradigm” to hierarchical P2P systems such as Limewire[1], our algorithms are able to improve the join and leave protocols for better routing path between peers. Future work will look into query search utilizing the high cluster peers in the overlay network for better search and routing performance. This will allow us to show that our search algorithms reduce the message flooding and improve the accuracy of recall rate for query search request.

This paper is organized as follows: Section 2 reviews related works, Section 3 describes an overview of our P2P systems protocol and finally, Section 4 provides a set of test scenarios. The detail of the next research directions and expected research outcomes are discussed.

II. RELATED WORKS

Several research works were reviewed on the application of Small-World paradigm for unstructured P2P networks [3, 6, 7, and 8] in this paper. Peers in small world networks are not always “neighbour” to each other. However, they are able to reach each other by using a small numbers of hops. The “Small-World paradigm” was originally introduced by Watts and Strogatz [4] in 1998. Our research work utilizes the concept of “Small-World paradigm” proposed by [4] by implementing the rewiring technique in order to achieve high cluster coefficient with low average hops. High cluster coefficient is achieved by peers through self-organization of peers by clustering using similar *interest*. This is achieved by using rewiring technique with light randomness. The peer’s routing index in our design also allows our algorithm to achieve the low average hops criteria.

Works implemented by Zhang, Goel and Govindan [7] using the Small-World paradigm to improve Freenet’s performance by distancing peers with dissimilar interest and updating new peers’ information in routing tables achieved some good performance. However, there is a small possibility of evicting a peer from the list and adding a dissimilar peer as a shortcut to other cluster. Schmitz [6] clustered peers by topic based on common ontology to achieve Small-World paradigm. In his work, each peer maintains short-range and long-range links. The short-range links are for peers of similar topic while long-range links are shortcut path to other clusters. At the same time, peer will periodically check for the similarity between neighbours. If neighbours’ similarity is less than a threshold value then the rewiring technique will be executed; they will look for new connections and discard outdated links or dissimilar peers on the routing index table.

Ng, Sia, Chan and King [5] proposed to form a cluster using the fireworks routing strategy. Peers periodically broadcast messages to retrieve other peers’ information to update its short-range links and this allows the clusters to maintain their similarity information regularly. Long-range links are selected and maintained by the user. As soon as clusters are formed, query messages can be routed through different clusters. As soon as the query message arrives at the targeted cluster, search messages will be broadcasted to its neighbours to produce the effect of fireworks exploding.

Raftopoulou and Petrakis [3, 8] used the similar rewiring technique to maintain short-range and long-range links among peers. They extended the idea of Schmitz by using document concept as the *interest* instead of the ontology concept.

However, the research works reviewed had surfaced two problems: 1) peers are always connected randomly once they join the network for the very first time. And this has made the network difficult to achieve high cluster coefficient when the peers join and leave too frequently; and 2) long-range links are randomly selected. As a result, query messages will be randomly routed to different clusters in the network. This produces unpredictable search outcomes. In our research work, we propose techniques that show how peers select cluster(s) to join the network correctly at the very first time. And we will also demonstrate how peer maintains medium-range and long-range links to overcome random long-range links which will improve the query messages to route more accurately. This lays a desirable small-world P2P architecture for the next phase of research work that focuses on high recall-rate by minimizing search messages in small-world P2P hierarchical networks.

III. AN OVERVIEW OF THE PROPOSED DESIGN

In this section, we will describe the proposed Small-World Bee (SWB) network design of our research work, aiming to improve the peers clustering of the SWB network. SWB adopts the QRP table [1] as the basic information container to capture the *interests* of peers. The Small-World clustering technique proposed by our research will periodically perform rewiring to cluster peers who have the similar *interests* [3, 4, 6, 8].

A. The SWB Architecture

In our proposed SWB design, peers in an overlay network are categorized as LeafPeers and SuperPeers to form a two-level hierarchical P2P system. The LeafPeers are located at the second level connected to few SuperPeers in an overlay network. When a LeafPeer performs a search, the SuperPeer(s) act as a *proxy* for the LeafPeers to send out query messages to others SuperPeers to perform the search task. SuperPeers are

more stable peers in terms of longer online time and better communication bandwidth. These SuperPeers must not be blocked by firewall [1]. On the other hand, LeafPeers are peers that are not-so-stable. They have short or less online time and lower communication bandwidth. To identify peers' *interests*, a peer hashed its files' name and metadata into a QRP table. Since SuperPeers are more stable peers, they have the responsibility to periodically run the peer rewiring protocol so that clusters formed are based on their likelihood to contain *similar* content. Aggregation of LeafPeers' QRP will be reconciled at the SuperPeers. The *interest* information could then be used by the SuperPeers to construct short-range, medium-range and long-range links. Short-range links are connections in the routing table that connects to other *similar interest* peers. Medium-range and long-range links are connections in a routing table that connects to *not-so-similar interest* peers. These query messages will route through different type of links based on the *interest similarity* values to look for a target cluster.

B. Basic Protocols

The main idea behind the Small-World [4] network is to let peers self-organize into clusters of *similar* contents with some light randomness. The peers of *similar* contents are clustered into group. Links are formed randomly as shortcuts to other cluster. This has enabled query request to be sent to search at the targeted clusters. In this section, the basic protocols will be explained to show how peers maintain their *interest* using QRP table. The proposed mechanism allows peers to join and leave a P2P network, to self-organize themselves into different clusters and to query data by *interest* to reduce message flooding and to improve recall rate.

1) Query Routing Protocol (QRP)



Figure 1: QRP Table [1].

Every peer in a P2P network holds a QRP table. Figure 1 shows the QRP table. A QRP table is an array of 65536 bits that consists of values 0 and 1. A QRP table will be initialized to '0', which means that the peer initially does not have any *interest*. When a query message reaches a SuperPeer, it will examine the QRP table. If the QRP tables contain '0', this indicates that files desired don't exist in that peer, and then the search will be terminated. The value '0' in a QRP table shows that the files don't exist. LeafPeers' QRP table will be aggregated to the SuperPeers' QRP table so that the SuperPeers can use the aggregated QRP to filter irrelevant queries that want to reach to their LeafPeers. At the same time, SuperPeers will periodically exchange QRP table with other SuperPeers to update their routing indexes. This will enable last hop savings when searching the desired item by visiting

all the linked SuperPeers. Last hop saving will happen when a SuperPeer receives a search message with Time-To-Live (TTL) value of one (1) before it will start checking on its neighbours' QRP. If the query message did not hit '1' in the neighbours' QRP tables then the SuperPeer that hold the query message will not send over to its neighbours. The benefit of exchanging QRP tables among SuperPeers has greatly reduced message flooding in the network.

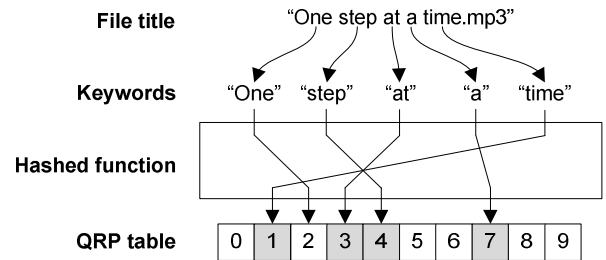


Figure 2: Peer's *interest*.

In Figure 2, it shows that the files' title and metadata are used to form the *interest* of peers in a network. For example, a file with the title of "One step at a time.mp3" will be hashed into a series of hashed QRP location. The hash function will not hash the file content. Since the query function of the search uses word(s), of the files' title, these word(s) must be separated into individual unit of word. For example, "One", "step", "at", "a" and "time".. The hash function will change all the words to lower case and the function uses locale-neutral conversion based on the UTF-16 representation [1]. The hashed values will be used to update the QRP table's entries so that the QRP table will hold the *interest* of the peer.

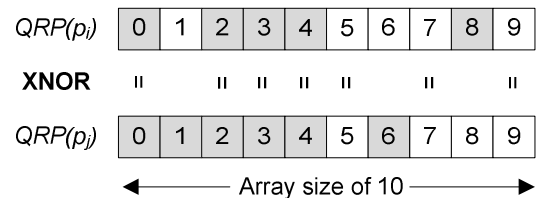


Figure 3: Two-Tier Bitwise Interest Oriented QRP.

The **Two-Tier Bitwise Interest(s) Oriented QRP** (Figure 3) is a technique where peers p_i and p_j store their *interest(s)* in their QRP tables where $QRP(p_i)$ and $QRP(p_j)$ are containers that hold their *interest(s)* values, given that $i=1..X$ and X is the total number of peers in an P2P overlay network. The **Two-Tier Bitwise Interest(s) Oriented QRP** technique then measures the *interest* value by computing the *interest similarity* value of any two peers. The proposed technique allows peer p_i and peer p_j to execute the *similarity* function, $sim(QRP(p_j), QRP(p_i))$ and the function is computed as $N/Const(Size\ of\ QRP\ table)$, where N is $XNOR$ of $QRP(p_i)$ and $QRP(p_j)$. In Figure 3, the QRP tables

for peer p_i and p_j with an array of size 10 are illustrated. In this example, the *XNOR* operation for both peers has produced 7 matches ($N = 7$). Since the *Const(Size of QRP table)* is 10, the similarity value computed is 0.7. This means that the *similarity interest* between peers p_i and p_j is 70%. When QRP table produces a hit for words of files' title then there is a high possibility that the file exists in this peer. Clustering peers by using similar interest can archive high clustering coefficient, so that peers in the network are close to each other. Therefore, network with high clustering coefficient can produce more hits for relevant queries.

2) Joining Protocol

As peer p_i joins the P2P network, the Joining protocol will manage the join operation for the joining peer. First of all, the Joining Protocol needs the peer p_i to execute the *hash(file_i)* function of all its shared files (*FILES_i*) into its QRP table as the *interests QRP(p_i)*. Then p_i will send *Bootstrap* message to UDP Host Cache (UHC) [1] to retrieve IP addresses and port numbers of the available SuperPeers (*SP*). From the list of SuperPeers, p_i will request for their QRP tables to perform *sim(QRP(p_j), QRP(p_i))* using bitwise operation. The *max(sim(QRP(p_j), QRP(p_i)))* will be chosen as the most *similar* SuperPeers for p_i . The p_i will immediately connect to the selected SuperPeers and join as its LeafPeer. Since all LeafPeers' QRP table will be aggregated to the SuperPeer's QRP table, and all the LeafPeers have very *similar interest* with their SuperPeer, a small change in their *interest* would be required when new peers join the cluster. After a p_i stays in a P2P network for a sufficiently long uptimes, and if a peer has sufficient bandwidth and no firewall blockage, then the peer p_i will be upgraded to be a SuperPeer else it will remain as a LeafPeer. To achieve efficient routing, the SuperPeer p_i will maintain a separate routing index RI_i , which contains short-range, medium-range and long-range links. The SuperPeers in the P2P network will exchange their QRP tables with each other in order to obtain the benefit of last hop savings when searching the network [1]. Each entry in the routing index RI_i is formed by $(ip(p_j), QRP(p_j))$, where $ip(p_j)$ is the IP address of p_j and $QRP(p_j)$ is the QRP table of p_j . **Algorithm 1** exhibits the Joining Protocol describes above.

Algorithm 1: Algorithm for Joining protocol

```

for all filesi ∈ FILESi do
  QRP(pi) = QRP(pi) ∪ hash(filei)
end for
forward Bootstrap message to request SP
for all QRP(pj) ∈ SP do
  compute sim(QRP(pj), QRP(pi))
end for
attempt to connect to ∃max(sim(QRP(pj), QRP(pi))) where
QRP(pj) ∈ SP

```

3) Rewiring Protocol

The Rewiring protocol will be executed periodically for each SuperPeer to maintain its short-range, medium-range and long-range links. The short-range links are intra-cluster links that connect to similar *interest* peers and long-range links are inter-cluster links that are shortcut paths to other clusters. In the Rewiring protocol, a SuperPeer p_i will periodically compute $AS_i = (1/s) \sum_{\forall p_j \in RI_i} sim(QRP(p_j), QRP(p_i))$ as the average similarity among its neighbours in its routing index (RI_i), where s is the number of short-range links. If AS_i is greater or equal to threshold θ then p_i will not continue the rewiring activity; otherwise, p_i will create a *FindPeers* message with parameters $(ip(p_i), QRP(p_i), P, t_r)$, where P is an empty list to collect all it's peers info and t_r is the time-to-live (TTL) of the message.

A peer p_j that receives the *FindPeers* message will append its IP address and QRP table into p_j by reducing t_r by one. Peer p_j then forward the message to m selected neighbours peers. Random walk [3, 4] will be applied in the rewiring protocol where messages will be randomly sent to m selected neighbours peers in Routing Index for peer j (RI_j) so that messages have the chance to explore to different peers.

When $t_r = 0$, the *FINDPEERS* message will return to the message creator p_i . Peer p_i will categorize the peer information from the list P to update its RI_i . Peer p_j with *similarity value* greater and equal to 0.7 [why 0.7?] will be used as short range-links. Outdated links or dissimilar *interest* links will be discarded. Peers with similarities within $0.5 \leq sim(p_j) < 0.7$ will be used as medium-range links and $sim(p_j) < 0.5$ will be used as long-range links. The different types of link allow shortcut paths for a peer to get connected to peers in other clusters.

The rationale for the rewiring protocol to maintain short-, medium- and long-range links is to archive the property of "six-degrees of separation" of the "Small-World" paradigm. The protocol wills periodically rewire the links in the routing indexes so that links are always up-to-date. By refining the short-range links to maintain highest similarity *interest* intra-cluster-ly will archive the high cluster coefficient requirement of the "Small-World" paradigm. The effort to maintain medium-range and long-range links is to provide several alternative paths so that future work on query search where search messages can route to the nearest cluster based on the *interest* similarity of peers to archive the low average hops between peers. **Algorithm 2** exhibits the Rewiring Protocol describes above.

Algorithm 2: Algorithm for rewiring protocol

compute $AS_i = (1/s) \sum_{\forall p_j \in RI_i} sim(QRP(p_j), QRP(p_i))$

if $AS_i < \theta$ **then**

$P = \{ \}$

create *FindPeers* message = $(ip(p_i), QRP(p_i), P, t_r)$

//forward *FindPeers* message to m selected peers

forward *FindPeers* message to p_j where $p_j \in RI_i$ and

$j = 1, \dots, m$

$P = P \cup (ip(p_j), QRP(p_j))$

end if

repeat the above procedure for p_j 's neighbours

until $t_r = 0$

IV. TESTING AND EVALUATION

The results from the simulated Small-World Bee (SWB) P2P network include tests on all our proposed protocols as discussed above. The SWB is tested base on the objectives of the research: high clustering coefficient which attempting to maintain the level of accuracy of the recall rate through high clustering coefficient. Small-World Bee or SWB P2P network will be compared with iCluster [3], Firewall Query Model [5] and Limewire [1] in the experiments carried out in our research work.

A. Experiment Set-up

PeerSim is a cycle-based simulator which is used to simulate hybrid P2P with undirected links of different network sizes and configurations. For the purpose of our experiments, the initial size of the network was designed to start with 4,000 peers and peers are assigned to two (2) SuperPeers with eight (8) LeavePeers each. For each cycle (of a specific duration), the simulated network will be reconfigured with a different set of values with a set of parameters of some specific values. In each cycle, there will be one (1) SuperPeer and two (2) LeavePeers to be removed from the network. At the same time, two (2) LeavePeers will be promoted to be SuperPeers and five (5) new peers join the network. At the end of every cycle, the network size will increase progressively to a maximum of 20000 peers. Due to the changes of the network, SuperPeers check the average similarity among its neighbors and decide whether to execute the rewiring protocol to maintain all its relevant neighbors.

SWB network was measured by using a real-world dataset from [17]. The dataset contains over 50,000 of characters that form a series of song title and simple metadata such as singer and the song type. Each peer will be randomly assigned zero (0) to four (4) song title(s) from the dataset to be stored in the Query Routing Protocol (*QRP*) tables. Each *QRP*

table will store the peer's *interest* and base on the *similarity interest*, clusters will be formed.

In this evaluation, five (5) peers will be selected to collect data in the network and get the average value. The setup is designed in such a way so that the simulation doesn't depend on one (1) peer that may give bias result. The size of the *QRP* table and similarity threshold θ among neighbor will also be experimented in our simulation.

B. Performance Measure

Local clustering coefficient has been introduced by [4] and used to measure the closeness of the p_i with its neighbors. The clustering coefficient C_i for a peer p_i is the number of links that exist between p_i 's neighbors (routing index RI_i) over the number of links to form a complete graph between p_i and its neighbors. If p_i has size of k_i with neighbors of $|RI_i|$, then $k_i(k_i - 1)/2$ will be the number of links for an undirected graph to form a complete graph. When p_i and p_j are neighbors then link of l_{ij} will exist between these two peers with the link of l_{ji} equal to l_{ij} in an undirected graph. The local clustering coefficient is defined as:

$$C_i = \frac{2|l_{jk}|}{k_i(k_i-1)} : p_j, p_k \in RI_i, p_k \in RI_j \quad (1)$$

The average local clustering coefficient is the clustering coefficient of the whole network [4]; it is the summation of the local clustering coefficient of all peers over the number n of peers in the network:

$$\bar{C} = \frac{1}{n} \sum_{i=0}^{n-1} C_i \quad (2)$$

In this research, the Local similarity was introduced to measure the similarity *interest* LS_i between p_i and its neighbors. Since short-range links are used to group similar neighbors, so short-range links S_i will take into the consideration for the measurement of local similarity and S_i is the subset or equal RI_i ($S_i \subseteq RI_i$). The Local similarity is defined as follow:

$$LS_i = \frac{\sum sim(p_i, p_j)}{k_i} : p_j \in S_i \quad (3)$$

Local clustering coefficient is used to show the closeness of the peers. And the local similarity is designed to show the peer similarity *interest* with its neighbors. Therefore, the peers in the network can be self-organized with highly cluster coefficient and surrounded by similar *interest* peers. In order to measure the average local similarity for the network, it is the summation of the local similarity of all peers over the number n peers in the network:

$$\bar{LS} = \frac{1}{n} \sum_{i=0}^{n-1} LS_i \quad (4)$$

C. Peer Organization

In order to ensure quality peer organization in a P2P network, local clustering coefficient and local similarity are tested with different size of *QRP* table and similarity threshold θ values in our experiments.

When different size of *QRP* table is used, it will affect the local clustering coefficient and the local similarity. When peer has *QRP* size of 80, the network achieves high local similarity (an average of 0.708527). Peers with small *QRP* table will increase the probability of sharing the same slots which will increase the hashed words. This will lead the network to have a lower local clustering coefficient because every peer contains a *QRP* table of general *interest*. Large *QRP* table for a peer will make the *interest* more specific, so it will be more accurate for any query activities. The reason is slots for hashed words to be shared is much low, so each of the hashed words will have a specific slot to store it.

The higher the similarity threshold θ will achieve higher local clustering coefficient and higher local similarity. If the similarity threshold is too high, it will cause the recall rate to decrease and the chances for peers to perform rewiring will increase. Query message will always get discarded when it sends to the same peers in a network that has very high local clustering coefficient; it will also lead to lower recall rate. Network with θ value of 0.6 is chosen for our setting in our experiments, because it will achieve high cluster coefficient (0.033054) and high local similarity with (0.80994). The θ value of 0.6 having higher recall rate (0.810834) compare to $\theta = 0.5$ (average recall rate = 0.785884) and $\theta = 0.7$ (average recall rate = 0.748451). SWB can archive significantly higher clustering coefficient compare to a random graph such as Limewire. The clustering coefficient for SWB is approximately 200% higher than Limewire.

D. Comparison of Different Methods

In the quest to reduce message flooding and improve recall rate, our proposed Small-World Bee (SWB), Firework Query Model [5], iCluster [3], and Limewire [1] are compared and implemented using the following configurations:

- SWB: short-range links = 9, medium-range links = 3, long-range links = 3, $\theta = 0.6$, $t_f = 4$ and $t_b = 1$.
- Firework Query Model: short-range links = 9, long-range links = 6, $\theta = 0.6$, $t_f = 4$ and $t_b = 1$.
- iCluster: short-range links = 9, long-range links = 6, $\theta = 0.6$, $t_f = 4$ and $t_b = 1$.
- Limewire: short-range links (Routing-Index) = 15 and $t_b = 3$.

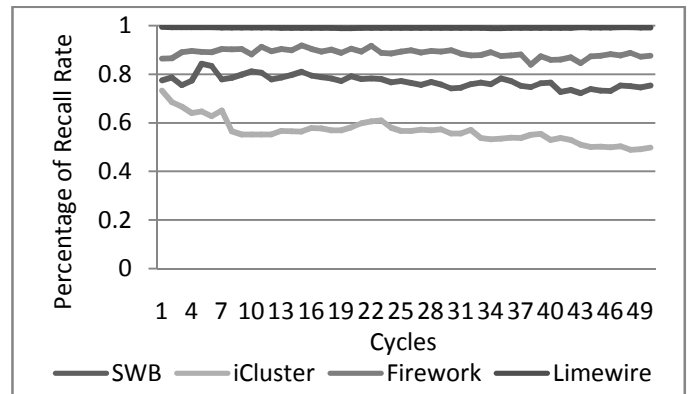


Figure 4: Percentage of recall rate for different methods.

Figure 4 showed that the percentage of recall rate for different methods in the tests. Limewire having the highest recall rate among all methods. Fireworks Query Model achieved an average of 0.886576 recall rate, where it is about the recall rate is about 11% less than Limewire. However, iCluster has low recall rate (average of 0.566097 recall-rate). Comparing to all the methods, Small-World Bee achieved the high recall-rate (average of 0.770546 recall-rate) and the recall-rate is slightly less than Limewire and Firework Query Model. However, the recall rate for Small-World Bee is approximately 13% less than Firework Query Model. Small-World Bee achieved high recall-rate. Table I summarizes test results and weaknesses and strengths of each method.

TABLE I. SUMMARY OF TEST RESULTS

	Small-World Bee (SWB)	Limewire [1]	Firework Query Model [5]	iCluster [3]
Average number of message flooding in the network	Little (442.616)	High (2339.208)	Moderate (697.328)	Little (391.436)
Average percentage of recall rate	High (0.770546)	High (0.992055)	High (0.886576)	Moderate (0.566097)

V. CONCLUSION

This paper proposes to enhance the join protocol by selecting peers of *similar interest* so that the proposed P2P system always maintains high cluster coefficient in the network. We also proposed medium-range and long-range links so that the next phase of our research work is able to improve the query route intelligently by selecting suitable path. The implemented “Small-World paradigm” in the Limewire has achieved high cluster coefficient among peers with similar interest. The experiment result shows that SWB will maintain low communication overhead and achieve high recall rate. The next phase of the research work will focus on the technique to query search with routing path planning for high recall rate and minimum query messages. Other areas of future works include

enhancing the rewiring protocol for much reduced message overhead to maintain the neighbours list.

REFERENCES

- [1] LimeWire. <http://www.limewire.com>.
- [2] K.Y.K. Hui, J.C.S. Lui, and D.K.Y. Yau. Small-world Overlay P2P Networks: Construction, Management and Handling of Dynamic Flash Crowds. *Computer Networks*, 50(15), 2006.
- [3] P. Raftopoulou and E.G. M. Petrakis. iCluster: a Self-Organising Overlay Network for P2P Information Retrieval. In *ECIR*, 2008.
- [4] D. J. Watts and S. H. Strogatz. Collective Dynamic of “Small-World” Networks. *Nature*, 393, 1998.
- [5] C. H. Ng, K. C. Sia, C. H. Chan and I. King. Peer Clustering and Firework Query Model in the Peer-to-Peer Network. In *WWW*, 2003.
- [6] C. Schmitz. Self-Organization of a Small World by Topic. In *P2PKM*, 2004.
- [7] H. Zhang, A. Goel and R. Govindan. Using the Small-World Model to Improve Freenet Performance. In *SIGCOMM*, 79, 2001.
- [8] P. Raftopoulou and E. G. M. Petrakis. A Measure for Cluster Cohesion in Semantic Overlay Networks. In *LSDS-IR*, 2008.
- [9] S. Ratnaswamy, P. Francis, M. Handley, R. Karp and S. Shenker. A scalable content-addressable network. *ACM SIGCOMM*, 2001.
- [10] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan. Chord: A peer-to-peer lookup service for internet application. *ACM SIGCOMM*, 2001.
- [11] A. Rowstron and P. Druschel. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. 18th IFIP/ACM Int. Conference on Distributed System Platforms, pg 329-350, 2001.
- [12] M. Li, W. C. Lee, A. Sivasubramaniam. Semantic Small World: An Overlay Network for Peer-to-Peer Search. In *IEEE*, 2004.
- [13] Gnutella. <http://www.gnutella.co.uk>.
- [14] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: A distributed anonymous information storage and retrieval system in designing privacy enhancing technologies. International Workshop on Design Issues in Anonymity and Unobservability, 2001.
- [15] B. Y. Zhao, J. Kubiatowicz, and A. D. Joseph. Tapestry: An infrastructure for fault-tolerant wide-area location and routing. Pages Tech. Report, UBC/VSD-01-1141, U.C. Berkeley, 2001.
- [16] D.T.Pham, A.Ghanbarzadeh, E.Koc, S.Otri, S.Rahim, M.Zaidi. The Bees Algorithm – A Novel Tool for Complex Optimization Problems, anufacturing Engineering Centre, Cardiff University, Cardiff CF24 3AA, UK, Proceedings of IPROMS 2006 conference, pp.454-461.
- [17] Magic Star Music Systems.
<http://www.magicstarmusic.com/DJsongs.htm>.