ComNET: A P2P Community Network

Zhentao Sun and Wenjun Xiao

Department of Computer Science, South China University of Technology, Guangzhou 510641, P.R. China
robinvane@163.com, wjxiao@scut.edu.cn

Abstract. In addition to searching, browsing is yet another requirement of P2P file sharing systems. Nevertheless, none of the recent P2P DHTs can closely connect peers with the same interests together so that it is not practical to provide browsing service in such systems. In this paper, we define a new cayley graph to support logical grouping, and based on this cayley graph, a set of P2P DHT protocols which is suitable for providing file browsing service is also designed. Performance evaluation indicates that the new protocols can reach the theoretical lower bound of routing table size and query path length. Furthermore, the robustness of ComNET is also better than most of the P2P DHTs recently proposed.

Keywords: P2P, ComNET, Grouping, Browsing service, Cayley graph, Small-world.

1 Introduction

Almost all recent researches of P2P DHT [1] [2] [3] concentrate on how to lower the length of query path and reduce the size of routing table. Therefore the users download behavior are not taken into consideration when they design the systems. In addition to searching, browsing is one of other important requirements when people use P2P file-sharing systems, but as we know, none of current P2P DHT can support efficient file browsing service.

To remedy the disadvantages of the recent structured P2P system, we introduce small-world phenomena [4] into the overlay network. The phenomena of small-world lead to the phenomena of community, which means that people with the same interests know each other with high probability (i.e. highly clustered). In order to introduce small-world features into P2P DHT, we define a new cayley graph Γ, and then based on Γ, a new P2P DHT called ComNET is also designed. In addition to efficient resource searching mechanism, ComNET supports explicit peer grouping, and thus supports effective resource browsing service. Both theoretical analysis and experimental evaluation show that ComNET can reach the lower bound[3] of routing table size and query path length at the same time. Moreover, its robustness is also better as compared to Ulysses[1].

M. Xu et al. (Eds.): APPT 2007, LNCS 4847, pp. 271–281, 2007.

2 Related Research and Design Objectives

2.1 Related Research

The motivation for our research stems from the following fields:

1. Ulysses[1]. Ulysses is a P2P DHT based on the famous butterfly graph and can reach a diameter of $\mathcal{O}(\log n/\log\log n)$. Nevertheless, in order to cluster similar peers, Ulysses has to introduce another overlay which will cause performance issues.
2. Content-based shortcut Gnutella[6]. Built on Gnutella, this P2P network adds links between similar peers and removes rarely used links. However, this system is still unstructured which means that the overloading problem can not be solved and explicit peer grouping can not be provided in this system.
3. Cayley graph as models of small-world networks. W. Xiao and B. Parhami propose a model of deterministic small-world graph in [7]. They use a algebraic method to construct cayley graphs which display small-world features.

2.2 Design Objectives

We are primarily interested in the following features of P2P DHT network.

1. *Short query path*: The average length of query path would not increase significantly as the number of peers in the overlay network become large.
2. *Reasonable size of routing table*: The minimal routing table size is beneficial to ensure fault-tolerance while the maximal one is relevant for ensuring bounded maintenance cost[5].
3. *Reasonable cluster coefficient*(CC1)[4]: Non-zero cluster coefficient leads to the phenomena of clustering and community. Our resulting P2P system should have a reasonable CC1.
4. *Peer Grouping*: Performance of file sharing system can be remarkably improved by structured grouping, and with which, users can easily browse the files they are more interested in.
5. *Self-configuration*: It is not possible for a large scale P2P system to employ centralized server to provide joining, departing and searching services. For the sake of scalability, distributed network services are preferred.
6. *Robustness*: P2P systems should have the ability to deal with high dynamic environment so that its performance will not drop dramatically when there are faulty peers in the system.

3 The Definition of Static ComNET

Every cayley graph is vertex transitive, and thus using cayley graph as the static graph of P2P DHT has the benefit of distributing loading to all peers evenly[8]. The static graph of ComNET Γ is defined in this section and then some properties of this graph which are essential to P2P systems are explored.

3.1 Terminology and Notation

In this section, x and y are assumed to be strings composed of digital or asterisk "*". We define the following operations and predicates on them:

1. $|x|$: the length of string x
2. $x[i]$: the i^{th}(left to right, counting from index 0) character of x
3. $lock(x, y, i)$:$|x| \leq i \vee |y| \leq i \vee x[i] =$ "*" $\vee y[i] =$ "*" $\vee x[i] = y[i]$. For example, if $x =$ "210", $y =$ "2*11", then $lock(x, y, 0)$, $lock(x, y, 1)$, $lock(x, y, 3)$, but $\neg lock(x, y, 2)$
4. $lockall(x, y)$:$\forall j \in \mathbb{N}, lock(x, y, j)$
5. $AP(x, i, r)$[1], $i \in \mathbb{Z}_r$ and x is a binary string: A sub-string of x which is composed of $x[i], x[i+r], x[i+2r], \cdots$. For example $AP($"010*2"$, 1, 2) =$"1*".
6. $APlock(x, y, i, r)$: defined as $lockall(AP(x, i, r), AP(y, i, r))$. For example, if $x=$"010*1"$, y=$"110111"$, i=1, r = 2$, then $AP(x, i, r) =$ "1*"$, AP(y, i, r)=$ "111", and thus $APlock(x, y, i, r)$. In this paper, $APlock(x, y, i, r)$ is referred to as *"x locks on y along dimension i"*.
7. $APlockset(x, y, r)$: is the maximal subset of $\mathbb{Z}_r$ which satisfies that $\forall l \in APlockset(x, y, r), APlock(x, y, l, r)$. This set is called *the locking set of x on y* or in brief, *the locking set of x* if the context is clear.
8. $APlockbut(x, y, i, r)$:$\forall j \in \mathbb{Z}_r \wedge j \neq i, APlock(x, y, j, r)$
9. $APlockall(x, y, r)$:$\forall j \in \mathbb{Z}_r, APlock(x, y, j, r)$
10. $m(\alpha, \beta, \gamma)$: $m(\alpha, \beta, \gamma)$ is obtained by replacing the γ^{th} character of α with the γ^{th} character of β.

3.2 The Definition of Cayley Graph Γ

Definition 1. *Let $H = (G, \bullet)$, where $G = (\mathbb{Z}^k_{r_c}, \mathbb{Z}^k_{r_p}, \mathbb{Z}_k)$. The operation $\bullet$ on G is defined as $\forall(\mathbf{c_1}, \mathbf{p_1}, r_1), (\mathbf{c_2}, \mathbf{p_2}, r_2) \in G$,*

$$(\mathbf{c_1}, \mathbf{p_1}, r_1) \bullet (\mathbf{c_2}, \mathbf{p_2}, r_2) = (\mathbf{c}_1 \oplus \sigma^{r_1}(\mathbf{c}_2), \mathbf{p}_1 \oplus \sigma^{r_1}(\mathbf{p}_2), r_1 + r_2)$$

σ is cyclic right shift operation. $\oplus$ is component-wise addition mod r_c and r_p. Unless noted otherwise, $+$ is modulo-k addition throughout this paper.

Corollary 1. *$(G, \bullet)$ is a group.*

For any $(\mathbf{c}, \mathbf{p}, r) \in G$, $\mathbf{c}$ is referred to as *group identifier*, $\mathbf{p}$ as *intra-group identifier*, r as *region identifier*, and $(\mathbf{c}, \mathbf{p}, r)$ as *vertex identifier*.

Group identifier make it possible for P2P DHT to provide peer grouping mechanism. Moreover, if the peers who share similar interests are clustered with the same group identifier, it would be more easier for them to reach each other, and therefore improve the efficiency of browsing operation.

Definition 2. *Let $S = S_p \cup S_c \cup S_r$, where $S_p = \{(\mathbf{0}, p0^{k-1}, 0) | p \in \mathbb{Z}_{r_p} \backslash 0\}$ $S_c = \{(c0^{k-1}, \mathbf{0}, 0) | c \in \mathbb{Z}_{r_c} \backslash 0\}$ $S_r = \{(\mathbf{0}, \mathbf{0}, r) | r \in \mathbb{Z}_k \backslash 0\}$, then $\Gamma = Cay(G, S)$ is a cayley graph.*

Links in Γ can be categorized to 3 types: $Link_p$, links between two vertices in the same group and region; $Link_c$, links between two vertices in different groups but in the same region; and $Link_r$, links between two vertices in different regions.

[1] From [1].

3.3 Some Properties of *Γ*

The degree of a cayley equals the cardinality of S, that is $|S| = |S_p| + |S_c| + |S_r| = r_p - 1 + r_c - 1 + k - 1 = r_p + r_c + k - 3$.

Proposition 1. *Γ is a $(r_p + r_c + k - 3)$-regular graph.*

The routing from $(\mathbf{c}, \mathbf{p}, r)$ to $(\mathbf{c_3}, \mathbf{p_3}, r_3)$ proceeds in two phases. In the first phase $\mathbf{c}$ and $\mathbf{p}$ successively change to $\mathbf{c}_3$ and $\mathbf{p}_3$. In the second phase, one step is required to correct the region identifier to r_3. The pseudo-code for forwarding in a vertex is shown in algorithm 1.

```
Input : Current vertex (c_1, p_1, r_1) and destination vertex (c_3, p_3, r_3)
Output: Identifier of next-hop vertex (c_2, p_2, r_2)
if (c_1 = c_3 ∧ p_1 = p_3 ∧ r_1 = r_3) then
    the destination has been reached
else
    if (c_1 = c_3 ∧ p_1 = p_3) then
        (c_2, p_2, r_2) := (c_3, p_3, r_3)
    else
        if (lock(c_1, c_3, r_1) ∧ lock(p_1, p_3, r_1)) then
            r_2 is an integer that does not satisfy lock(c_1, c_3, r_2) or
            lock(p_1, p_3, r_2). Non-r_3 integers are preferred;
            c_2 := c_1, p_2 := p_1
        else
            if (¬(lock(p_1, p_3, r_1)) then
                p_2 := m(p_1, p_3, r_1), c_2 := c_1, r_2 := r_1
            else
                c_2 := m(c_1, c_3, r_1), p_2 := p_1, r_2 := r_1
            end
        end
    end
end
```

Algorithm 1. Routing algorithm at a vertex in *Γ*

We can obtain from the routing algorithm that,

Proposition 2. *The diameter of Γ is $2k + k = 3k$.*

Note that the number of vertices n of Γ is $k(r_c r_p)^k$. If we let $k = \log n / \log\log n$, then $r_c r_p = (n/\frac{\log n}{\log\log n})^{1/\frac{\log n}{\log n \log n}} < \log n$. According to proposition 1, the degree of a vertex is $r_p + r_c + k - 3 < \log n / r_c + r_c + \log n / \log\log n < \log n + 1 + \log n / \log\log n$, thus

Proposition 3. *The degree and diameter of Γ can reach a complexity of $\mathcal{O}(\log n)$ and $\mathcal{O}(\log n / \log\log n)$ respectively.*

Proposition 3 shows that Γ reaches the theoretical lower bounds proposed by paper [3].

Proposition 4. *CC1 of Γ equals* $(C^2_{r_c-1} + C^2_{r_p-1} + C^2_{k-1})/C^2_{r_c+r_p+k-3}$

We can see that Γ is highly clustered with CC1 $\gg CC1_{random\ graph}$, which shows the small-world phenomenon.

4 ComNET Protocols

Our P2P DHT uses Γ as its static graph. In this section, algorithms on how to embed peers into Γ to construct a P2P DHT overlay called ComNET are presented.

4.1 ComNET Basics

1. *The Identifier Space*: Every peer in ComNET is identified by a unique 3-tuple $(\mathbf{c}, \mathbf{p}, r)$:

$$\mathbf{c} \in \{(c_0 c_1 \cdots c_s) | -1 \le s < l_c, c_i \in \mathbb{Z}_2 \cup \{*\}\}$$
$$\mathbf{p} \in \{(p_0 p_1 \cdots p_t) | -1 \le t < \infty, p_i \in \mathbb{Z}_2\}$$
$$r \in \mathbb{Z}_k$$

   l_c and k are two integral parameters of ComNET.
2. *The Topology of ComNET*: The topology of ComNET captures the link structure of Γ. Geometrically, $(\mathbf{c_1}, \mathbf{p_1}, r_1)$ is adjacent to $(\mathbf{c_2}, \mathbf{p_2}, r_2)$ *if and only if*:
   (a) $r_1 = r_2 \wedge APlockall(\mathbf{c}_1, \mathbf{c}_2, k) \wedge APlockbut(\mathbf{p}_1, \mathbf{p}_2, r_1, k)$, link between them corresponds to $Link_p$ in Γ
   (b) $r_1 = r_2 \wedge APlockall(\mathbf{p}_1, \mathbf{p}_2, k) \wedge APlockbut(\mathbf{c}_1, \mathbf{c}_2, r_1, k)$, link between them corresponds to $Link_c$ in Γ
   (c) $APlockall(\mathbf{c}_1, \mathbf{c}_2, k) \wedge APlockall(\mathbf{p}_1, \mathbf{p}_2, k)$, link between them corresponds to $Link_r$ in Γ

   Figure 1 shows a ComNET composed of 10 peers
3. *Distribute the Hash Table*: File names in ComNET are hashed to 3-tuples (α, β, γ), where $\gamma \in \mathbb{Z}_k$, α and β are fixed strings which satisfy $|\alpha| = l_c$ and $|\beta| \gg |\mathbf{p}|$. For any key (α, β, γ) and peer identifier $(\mathbf{c}, \mathbf{p}, r)$, if $lockall(\mathbf{c}, \alpha) \wedge lockall(\mathbf{p}, \beta) \wedge r = k$, we say that $(\mathbf{c}, \mathbf{p}, r)$ is responsible for the key (α, β, γ), and thus hash items with this key is stored at this peer.

4.2 Routing in ComNET

The routing problem is to find a path to a peer with specified identifer or a peer that is responsible for a given hash key of a file. Similar to routing in Γ, routing in ComNET also proceed in 2 phases. Nevertheless, due to high dynamic of P2P network, fault tolerance should be introduced. Forwarding operations at a peer in ComNET are shown in algorithm 2 - 5.

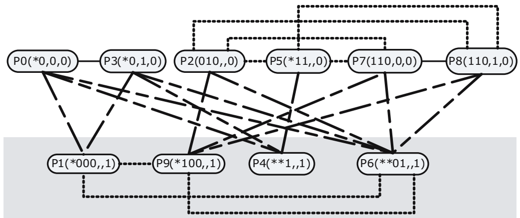

Fig. 1. Example of ComNET topology, where $k = 2, l_c = 4$

1. *Basic Routing Algorithm*: The routing algorithm corrects one bit of differences between the identifier of the source and the destination per step. The routing from P0 to P8 in figure 1 can be visualized as P0 $\rightarrow$ P3 $\rightarrow$ P1 $\rightarrow$ P9 $\rightarrow$ P8. Note that the resulting query path might not be the shortest one. But simple optimization can be obtained based on our routing algorithms. Since there is always more than one neighbor of the current peer whose identifier satisfies these algorithms. In this case, if we choose the neighbor with maximal *locking set* as next-hop, the length of query path can be reduced.

Input : The current peer $(\mathbf{c_1}, \mathbf{p_1}, r_1)$ and destination identifier (α, β, γ)
Output: The identifier $(\mathbf{c_2}, \mathbf{p_2}, r_2)$ of the next-hop peer along $Link_r$
r_2 is an integer that does not satisfies $\texttt{APlock}(\mathbf{c}_1, \alpha, r_2, k)$ or $\texttt{APlock}(\mathbf{p}_1, \beta, r_2, k)$. Non-$\gamma$ integers are preferred;
$\mathbf{c}_2$ satisfies $\texttt{APlockset}(\mathbf{c}_2, \alpha, k) \supseteq \texttt{APlockset}(\mathbf{c}_1, \alpha, k)$;
$\mathbf{p}_2$ satisfies $\texttt{APlockset}(\mathbf{p}_2, \beta, k) \supseteq \texttt{APlockset}(\mathbf{p}_1, \beta, k)$;

Algorithm 2. Finding a next-hop along $Link_r$ in ComNET: LinkRNextHop

Input : The current peer $(\mathbf{c_1}, \mathbf{p_1}, r_1)$ and destination identifier (α, β, γ)
Output: The identifier $(\mathbf{c_2}, \mathbf{p_2}, r_2)$ of the next-hop peer along $Link_p$
$r_2 = r_1$;
$\mathbf{c}_2$ satisfies $\texttt{APlockset}(\mathbf{c}_2, \alpha, k) \supseteq \texttt{APlockset}(\mathbf{c}_1, \alpha, k)$;
$\mathbf{p}_2$ satisfies $\texttt{APlockset}(\mathbf{p}_2, \beta, k) \supseteq \texttt{APlockset}(\mathbf{p}_1, \beta, k) \cup \{r_1\}$

Algorithm 3. Finding a next-hop along $Link_p$ in ComNET: LinkPNextHop

Input : The current peer $(\mathbf{c_1}, \mathbf{p_1}, r_1)$ and destination identifier (α, β, γ)
Output: The identifier $(\mathbf{c_2}, \mathbf{p_2}, r_2)$ of the next-hop peer along $Link_c$
$r_2 = r_1$;
$\mathbf{c}_2$ satisfies $\texttt{APlockset}(\mathbf{c}_2, \alpha, k) \supseteq \texttt{APlockset}(\mathbf{c}_1, \alpha, k) \cup \{r_1\}$;
$\mathbf{p}_2$ satisfies $\texttt{APlockset}(\mathbf{p}_2, \beta, k) \supseteq \texttt{APlockset}(\mathbf{p}_1, \beta, k)$

Algorithm 4. Finding a next-hop along $Link_c$ in ComNET: LinkCNextHop

```
   Input  : The current peer's identifier (c1, p1, r1) and destination identifier
            (α, β, γ)
   Output: The next-hop's identifier (c2, p2, r2)
1  if (lockall(c1, α) ∧ lockall(p1, β) ∧ r1 = γ) then
2      The destination has been reached
3  else
4      if (lockall(c1, α) ∧ lockall(p1, β)) then
5          c2 satisfies APlockset(c2, α, k) ⊇ APlockset(c1, α, k) ;
6          p2 satisfies APlockset(p2, β, k) ⊇ APlockset(p1, β, k);
7          r2 := γ;
8          if ((c2, p2, r2) is faulty) then
9              The destination can not be reached temporarily
10         end
11     else
12         if (APlock(c1, α, r1, k) ∧ APlock(p1, β, r1, k)) then
13             (c2, p2, r2) := LinkRNextHop();
14             if ((c2, p2, r2) is faulty) then
15                 Find a non-faulty peer along Link_r that satisfies
                   |APlockset(c2, α, k)| + |APlockset(p2, β, k)| ≥
                   |APlockset(c1, α, k)| + |APlockset(p1, β, k)| −
                   RetreatThreadhold
16             end
17         else
18             if (¬APlock(p1, β, r1, k)) then
19                 (c2, p2, r2) := LinkPNextHop();
20                 if ((c2, p2, r2) is faulty ∧ ¬APlock(c1, α, r1, k)) then
21                     (c2, p2, r2) := LinkCNextHop();
22                     if ((c2, p2, r2) is faulty ∧ |APlockset(c1, α, k)| +
                       |APlockset(p1, β, k)| < (2k − 2)) then
23                         (c2, p2, r2) := LinkRNextHop()
24                     end
25                 end
26             else
27                 (c2, p2, r2) := LinkCNextHop();
28                 if ((c2, p2, r2) is faulty ∧ |APlockset(c1, α, k)| +
                   |APlockset(p1, β, k)| < (2k − 2)) then
29                     (c2, p2, r2) := LinkRNextHop()
30                 end
31             end
32         end
33     end
34 end
```

Algorithm 5. Routing in ComNET routeDHT

2. *Robustness*: In order to improve the robustness of our system, line 8-10, 14-16, 20-25 and 28-30 in algorithm 5 are added. The robustness-ensuring mechanism is triggered when the next-hop peer identified by the basic routing algorithm

is not working. The basic idea of our robustness-ensuring algorithm is to find a non-faulty next-hop peer with non-descending locking set, that is to say the size of the locking sets of the next-hop peer's group and intra-group identifiers should be greater or at least equal to those of the current peer respectively.

5 Performance Evaluation

In this section, some important performance metrics are measured by system simulation. All evaluation is performed within a single process with no network communication actually exists. System parameters $k = 3$ and $l_c = 8$ are used for all network size because they are fixed at programming time, and thus could not be adjusted according to the network size at runtime.

5.1 Query Path Length

Figure 2 plots the average and maximum query path length of ComNET as a function of the number of peers. There are two types of simulation: one is to find the number of hops required for routing between two randomly selected peers; the other is to find the length of routing path between two random selected peers with the same group identifier. We can see from figure 2 that when the number of peers reaches around 4k, the maximum length of routing path and intra-group routing path are fixed at 9 and 6 respectively. It can be explained as follows. According to proposition 2, the diameter of ComNET is related to k through the equation $Diameter = 3k = 9$. Also it is not difficult to verify that the maximum length of intra-group routing is 6. Note that, the parameter k is fixed at programming time, which means that the upper bound of routing path length in ComNET is a constant $\mathcal{O}(3k)$. Figure 3 plots the distribution of routing path length for a network size of 2^{22}. It can be seen from this figure that the length of 89.39% random routing varies from 6-9, and the length of 94.88%

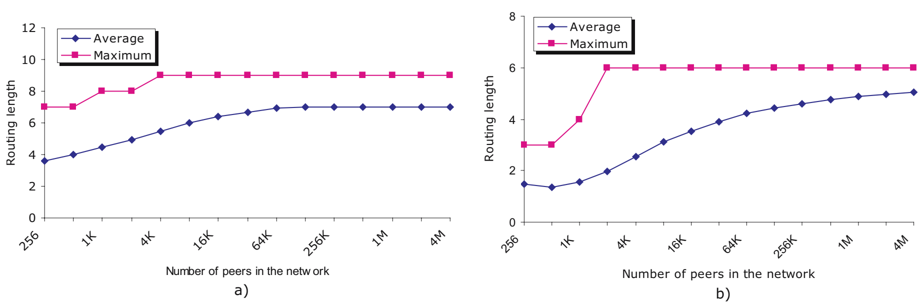

Fig. 2. Routing path length with different network size. a) plots the length of routing path between randomly selected peers and b) plots the length of routing path between randomly selected peers with the same group identifiers.

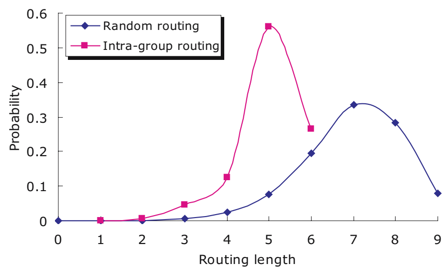

Fig. 3. The distribution of query length with 2^{22} peers

intra-group routing varies from 4-6. Low variance of routing path length may indicate low network jitter which is essential to real time P2P applications.

5.2 Size of Routing Table

Figure 4 plots the average size of routing table with different number of peers. According to proposition 1, the size of the routing table of a peer, is related to r_c and r_p. Thus the size of routing table would increase as the network size increase. When 2^{20} peers exist in ComNET, the average size of routing table is 21.3.

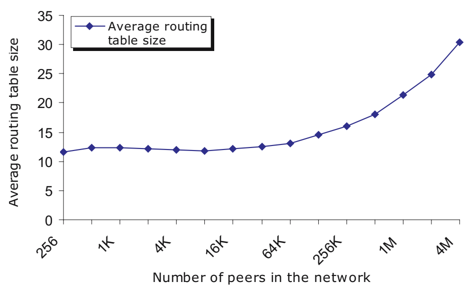

Fig. 4. The size of routing table

5.3 Robustness

In this section, we compute the probability that a query ends in failure and the average length of the successful routing in ComNET. All these tests are run in a network with 2^{22} peers. Figure 5a) plots the probability that routing ends in failure as a function of the percentage of peer failures. For the same failure rate, the probability that the ComNET routing algorithm exits in failure is lower than in Ulysses and Chord. Figure 5a) also plots the percentage of intra-group routing failure. As compared to random routing, the failure rate of intra-group routing is higher (57.2% when the peer failure rate is 20%), but is still low as compared

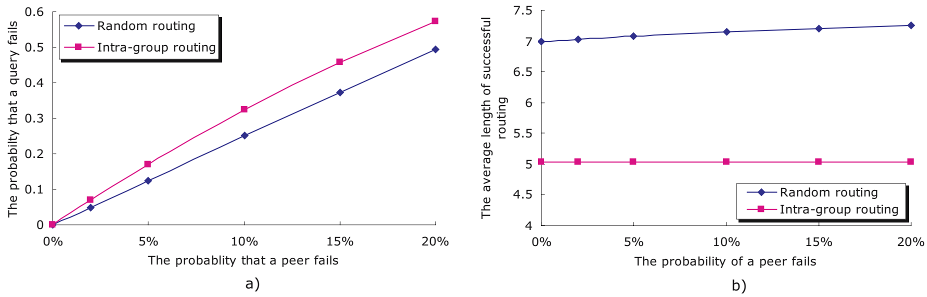

Fig. 5. a) the probability that routing ends in failure. b) the average length of successful routing. The network size is 2^{22}.

to that in Ulysses and in Chord. Figure 5b) plots the average hops required for the successful routing. The even curve in this figure indicates that the routing length is rarely influenced by faulty peers. As compared to ComNET, the routing length increases very remarkably in Ulysses (from 6.8 to 8.8, growing by 30%).

6 Conclusion and Future Work

In this paper, we define a new cayley graph Γ with small-world features. The essential properties including degree, diameter and CC1 show that Γ is a small-world graph and a suitable static model for P2P DHT network. In the latter sections, the protocols of P2P DHT network ComNET are proposed to capture the static structure in high dynamic environment. In ComNET, excellent routing performance is obtained while keeping the routing table small. Furthermore, the robustness of ComNET is also better than that of other protocols like Ulysses and Chord. And what is more, explicit peer grouping mechanism in ComNET enables us to implement effective resource browsing service in P2P DHT network.

Our further work will focus on how to adjust the number of regions (that is k) according to the network size, balancing zone splitting, finding a way to cluster a peer to more than one group and implementation of ComNET.

References

1. Kumar, A., Merugu, S., Xu, J., Yu, X.: Ulysses: A robust, low-diameter, low-latency peer-to-peer network. European transaction on telecommunications 15(6), 571–587 (2004)
2. Malkhi, D., Naor, M., Ratajczak, D.: Viceroy: A scalable and dynamic emulation of the butterfly. In: Proc. of ACM PODC 2002, ACM Press, New York (2002)
3. Xu, J.: On the fundamental tradeoffs between routing table size and network diameter in peer-to-peer networks. In: Proc. of IEEE Infocom 2003, vol. 1-3, pp. 2177–2187 (2003)
4. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature (1998)

5. Aberer, K., Alima, L.O., Ghodsi, A., Girdzijauskas, S., Haridi, S., Hauswirh, M.: The essence of p2p: a reference architecture for overlay networks. In: Fifth IEEE International Conference on Peer-to-Peer Computing (2005)
6. Sripanidkulchai, K., Maggs, B., Zhang, H.: Effcient content location using interest-based locality in peer-to-peer systems. In: Proc. of IEEE Infocom 2003: The conference on computer communications, vol. 1-3, pp. 2166–2176 (2003)
7. Xiao, W., Parhami, B.: Cayley graph as models of deterministic small-world networks. Information Processing Letters 97(3), 115–117 (2006)
8. Qu, C., Nejdl, W., Kriesell, M.: Cayley dhts - a group-theoretic framework for analyzing dhts based on cayley graphs. In: Cao, J., Yang, L.T., Guo, M., Lau, F. (eds.) ISPA 2004. LNCS, vol. 3358, Springer, Heidelberg (2004)