Graph-Theoretic Analysis of Kautz Topology and DHT Schemes*

Dongsheng Li, Xicheng Lu, Jinshu Su

School of Computer, National University of Defense Technology
Changsha 410073, P.R. China
leedongsh@hotmail.com

Abstract. Many proposed distributed hash table (DHT) schemes for peer-to-peer network are based on some traditional parallel interconnection topologies. In this paper, we show that the Kautz graph is a very good static topology to construct DHT schemes. We demonstrate the optimal diameter and optimal fault tolerance properties of the Kautz graph and prove that the Kautz graph is $(1+o(1))$-congestion-free when using the long path routing algorithm. Then we propose FissionE, a novel DHT scheme based on Kautz graph. FissionE is a constant degree, $O(\log N)$ diameter and $(1+o(1))$-congestion-free. FissionE shows that the DHT scheme with constant degree and constant congestion can achieve $O(\log N)$ diameter, which is better than the lower bound $\Omega(N^{1/d})$ conjectured before.

1 Introduction and Related Work

In recent years, peer-to-peer computing has attracted significant attentions from both industry and academic research. The core component of many proposed peer-to-peer systems is the distributed hash table (DHT) schemes [1] that use a hash table-like interface to publish and lookup data objects. DHT schemes for structured P2P systems have attracted much attention in academic researches for their desirable characteristics, such as scalability, robustness, self-management, and generality.

Many proposed DHT schemes are based on some traditional interconnection topology: Chord [2], Tapestry and Pastry are based on the hypercube topology; CAN [3] is based on the d-torus topology; Koorde [4] and D2B [5] are based on the de Bruijn graph; Viceroy [6], Ulysses [7] are based on the Butterfly topology. Compared with hypercube, de Bruijn or torus topology, Kautz graph has some better properties. In this paper, we demonstrate the optimal diameter and optimal fault tolerance properties of the Kautz graph and prove that the Kautz graph is $(1+o(1))$-congestion-free when using the long path routing algorithm. Then we propose FissionE, a novel DHT scheme based on Kautz graph. FissionE is a $(1+o(1))$-congestion-free DHT scheme with constant degree and $O(\log N)$ diameter.

---

*This paper is supported in part by the National Natural Science Foundation of China under the grant No. 90104001 and 69933030.

Two important measures of DHT schemes are degree, the size of routing table to be maintained on each peer, and diameter, the number of hops a query needs to travel in the worst case. In many existing DHT schemes, such as Chord, Tapestry, and Pastry, both the degree and the diameter tend to $O(\log N)$, while in CAN the degree and the diameter are $O(d)$ and $O(dN^{1/d})$ respectively. An open problem posed in [1] is whether there exists DHT scheme with $O(d)$ degree and $O(\log N)$ diameter. Recent work [4,5,6,7,8] has shown that there are DHT algorithms to achieve $O(\log N)$ diameter with $O(1)$ degree, but the algorithms cause severe congestion in P2P networks. Xu et al. [7] systematically studied the degree-diameter tradeoff of DHT schemes and defined the concept of *congestion*, and then clarified the role that *congestion-free* plays in the degree-diameter tradeoff. A conjecture posed in [7] is that "$\Omega(N^{1/d})$ is the asymptotic lower bounds for the diameter when the degree is no more than d and the network is required to be c-congestion-free for some constant c". *FissionE* is a novel constant degree and $(1+o(1))$-congestion-free DHT scheme with $O(\log N)$ diameter. FissionE can achieve better bound than the conjecture above.

FissionE is a constant-degree and $(1+o(1))$-congestion-free DHT scheme with $O(\log_2 N)$ diameter. The average degree of FissionE is 4, and the diameter of FissionE is less than $2*\log_2 N$; the average routing path length of FissionE is about $\log_2 N$. Compared with FissionE, the degree of Ulysses is $O(\log N)$ which is not constant. The expected degree of D2B is constant, but its high probability bound is $O(\log N)$, i.e., a few unlucky peers would be of degree $\Omega(\log N)$. The expected diameter of Viceroy is about $3\log_2 N$, however its $O(\log N)$ diameter is achieved not with certainty but "with high probability". Among the well-known DHT schemes, only CAN and Koorde definitely have constant degree. CAN is of $2d$ degree, but its diameter is $O(dN^{1/d})$, and so it does not scale as well as FissionE. Koorde [4] is constant degree and $O(\log N)$ diameter, but it isn't $(1+o(1))$-congestion-free and it's congestion is severer than that in FissionE.

The remainder of the paper is organized as follows. Section 2 introduces the Kautz graph and its properties. Section 3 proves the low congestion property of the Kautz graph. Section 4 describes the design of FissionE. Conclusions and future work is discussed in Section 5.

2 Static Kautz Graph

Many DHT schemes are based on the traditional interconnection network topologies. Different from dynamic P2P network, the traditional interconnection network poses some limits on the number of nodes it can support and does not support the dynamic joining or departure of nodes. To distinguish them, the traditional interconnection network is called static network in the paper. FissionE exploits Kautz graph as its static topology. This section discusses the Kautz graph and its properties.

Definition 1. The *Kautz string* ξ of length n and base d is defined as a string $a_1a_2...a_n$ where $a_j \in \{0,1,2,\ldots,d\}$ $(1 \le j \le k)$ and $a_i \ne a_{i+1}$ $(1 \le i \le k-1)$.

Definition 2. The *Kautz namespace* $KautzSpace(d,k)$ is defined as the set containing all the Kautz strings of length k and base d, i.e.,

$KautzSpace(d,k) = \{ a_1a_2...a_k \mid a_i \in \{0,1,2,\ldots,d\}\ (1 \le i \le k) \text{ and } a_i \ne a_{i+1}\ (1 \le i \le k-1)\}$.

Definition 3. The *Kautz graph* $K(d,k)$ [9] is a directed graph whose nodes are labeled with a Kautz string of length k and base d. For simplicity, we name a node with its label. Every node $U=u_1u_2...u_k$ in Kautz graph $K(d,k)$ has d outgoing edges: for each $\alpha \in \{0,1,2,\ldots,d\}$ and $\alpha \ne u_k$, node u has one outgoing edge to node $V=u_2u_3...u_k\alpha$, (denoted by $u \to v$), i.e., there is an edge from u to v iff v is a left-shifted version of u.

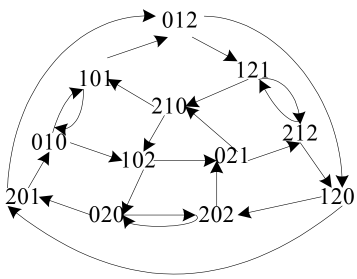

Fig. 1. Kautz graph $K(2,3)$

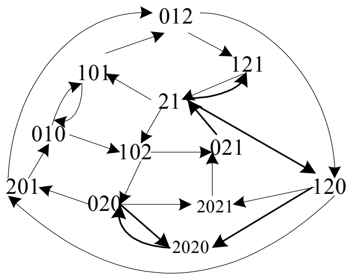

Fig. 2. Neighborhood of FissionE

Obviously there are $N=d^k+d^{k-1}$ nodes in the $K(d,k)$ graph and each node in $K(d,k)$ is of in-degree d and out-degree d. Figure 1 shows Kautz graph $K(2,3)$.

Table 1. The degree-diameter tradeoff of different topologies

Topology	Degree	Diameter	Average path length
de Bruijn	d	$\log_d N$	$\log_d N$-1/(d-1) [11]
Hypercube (Chord)	$\log_2 N$	$\log_2 N$	1/2 $\log_2 N$
d-torus (CAN)	$2d$	$1/2dN^{1/d}$	$1/4dN^{1/d}$
Butterfly	d	2 $\log_d N(1-o(1))$	about 3/2 $\log_d N$ [11]
Kautz (FissionE)	d	D=$\log_d N$- $\log_d(1+1/d)$	D-1/(d+1)

Assuming a graph of fixed degree d and diameter k, the maximum number of nodes N in the graph is the *Moore bound* [10] $1+d+d^2+\ldots+d^k$. The Moore bound is not achievable for any non-trivial graph. The number of nodes in the Kautz graph $K(d,k)$ is $d^{k-1}+d^k$, very close to the Moore bound. In fact, Kautz graph is the densest graph when the diameter is two. From the Moore bound, it is easy to see the low bound of the diameter of a graph with N nodes is $\lceil \log_d(N(d-1)+1) \rceil - 1$ and the diameter k of Kautz graph $K(d,k)$ reaches the lower bound as $\lceil \log_d(N(d-1)+1) \rceil - 1$ = $\lceil \log_d((d^k+d^{k-1})(d-1)+1) \rceil - 1$ = $\lceil \log_d(d^{k+1}-d^{k-1}+1) \rceil - 1$ = k. Thus Kautz graph $K(d,k)$ has an optimal diameter.

The Kautz graph also has optimal fault tolerance [13]. That is, Kautz graph $K(d,k)$ of degree d is d-connected (i.e., there are d node disjoint paths between any two nodes). The corresponding de Brujin graph is $(d-1)$-connected. In addition, Kautz graph $K(d,k)$ has a better load balancing feature than the de Bruijn graph as shown in [9]. Table 1 shows the degree-diameter tradeoff of different topologies.

3 Low Congestion Routing in Kautz Graph

There are many routing algorithms for Kautz graph. FissionE uses the *Long Path Routing Algorithm* in Kautz graph [9]. Long path routing in Kautz graph from node U to node V is accomplished by taking the string U and shifting in the symbols of V one at a time until the string U has been replaced by V. For instance, given two nodes $U=u_1u_2\ldots u_k$ and $V=v_1v_2\ldots v_k$, the long routing path from U to V is a path of length k shown as below:

$$U=u_1u_2\ldots u_k \rightarrow u_2u_3\ldots u_kv_1 \rightarrow u_3u_4\ldots u_kv_1v_2 \rightarrow \ldots \rightarrow u_kv_1v_2\ldots v_{k-1} \rightarrow v_1v_2\ldots v_k \quad (\text{if } u_k \neq v_1)$$

or a path of length k-1 shown as below:

$$U=u_1u_2\ldots u_k \rightarrow u_2u_3\ldots u_kv_2 \rightarrow u_3u_4\ldots u_kv_2v_3 \rightarrow \ldots \rightarrow u_kv_2\ldots v_{k-1}v_k=v_1v_2\ldots v_k \quad (\text{if } u_k=v_1)$$

For example, with the long path routing algorithm, the routing path in Kautz graph *K*(*2,3*) from node 012 to node 102 is 012→ 121→210→102, and the routing path from 012 to 202 is 012→120→202.

The long path may contain duplicate nodes and the algorithm keeps it for symmetry and simplicity. Obviously, with the long path routing algorithm, the path length between any nodes is *k* or *k*-1, and the average path length is $h=d/(d+1)*k+1/(d+1)*(k-1)=k-1/(d+1)$. Compared with the shortest path routing algorithm, the long path routing algorithm has a little longer average routing path length, while it has better load balance characteristics and the average delay is even less than the shortest routing algorithm under heavy load [9] (the severe congestion on some nodes leads to some delay). FissionE adopts the long path routing algorithm.

Now we consider the congestion characteristic of long path routing in Kautz graph. We use the concept "congestion-free" from [7].

Definition 4 [7]**.** A P2P network is *c-congestion-free* (*c* is constant and $c \geq 1$) if its static network is both *c-node-congestion-free* and *c-edge-congestion-free* under *uniform all-to-all communication* load. The *c-congestion-free* is also called *constant congestion.* A network is said to be *c-node-congestion-free* if no node is handling more than c times the average traffic per node. A network is said to be *c-edge-congestion-free* if no edge handling more than c times the average traffic per edge. The *uniform all-to-all communication* load is defined as: for each pair of nodes U,V (U≠V), there is a unit of traffic from U to V. The static P2P network is referred to the case that all nodes in the identification space exist and are alive, i.e., nodes in P2P network form the complete static topology.

Now we turn to the congestion property of the Kautz graph and some lemmas referred in the proof are shown after the Theorem 1.

Theorem 1. *When using long path routing algorithm, Kautz graph K(d,k) is* (*1*+*o*(1))*-congestion-free.*

Proof. Define $S1=\{u_1u_2...u_ku_1u_2...u_k \mid u_1u_2...u_ku_1u_2...u_k \in KautzSpace(d,2k)\}$,

$S2=\{u_1u_2...u_ku_2...u_k \mid u_1u_2...u_ku_2...u_k \in KautzSpace(d,2k-1)$ and $u_1u_2...u_k=u_ku_2...u_k\}$,

$S3=KautzSpace(d,2k)-S1$, $S4=KautzSpace(d,2k-1)-S2$, $S=S3 \cup S4$

The uniform all-to-all communication load is represented by the set M:

M={routing paths from U to V | U,V are nodes in *K*(*k*,*d*) and U≠V}

Define mapping f: $\forall \delta \in M$, assuming δ is a routing path of length *n:*

$b_1b_2...b_k \rightarrow b_2b_3...b_{k+1} \rightarrow b_3b_4...b_{k+2} \rightarrow ... \rightarrow b_nb_{n+1}...b_{n+k}$,
then f(δ)=$b_1b_2...b_k...b_{n+k}$.

From Lemma 1, f is a bijection from M to S. Thus under uniform all-to-all communication load, for any node R=$r_1r_2...r_k$, its load equals the number that the Kautz string $r_1r_2...r_k$ appears as a substring (except for the prefix) of the Kautz strings in S. From Lemma 2, the load L_n of R is:

$$L_n = \begin{cases} k * d^k + (k-1)\mathrm{d}^{k-1} - k & (r_1 \neq r_k) \\ k * d^k + (k-1)d^{k-1} - k + 1 & (r_1 = r_k) \end{cases}$$

The average path length in Kautz graph $K(d,k)$ is $h=k-1/(d+1)$, thus the average load of a node is

$$\mathrm{Aveg}(L_n)=(N-1)*h=(d^k+d^{k-1}-1)*(k-1/(d+1)) = k*d^k+(k-1)*d^{k-1}-k+1/(d+1)$$

Because $\mathrm{Max}(L_n)$- $\mathrm{Aveg}(L_n)= d/(d+1) << \mathrm{Aveg}(L_n)$, and

$$\mathrm{Max}(L_n)/\mathrm{Aveg}(L_n)<1+1/((k-1)*(d^k+d^{k-1}))=1+1/((k-1)*N)=1+O(1/N\log_d N)=1+o(1)$$

Thus the static Kautz graph is $(1+o(1))$-node-congestion-free.

In Kautz graph $K(d,k)$, the edge from $r_1r_2...r_k$ to $r_2...r_kr_{k+1}$ can be uniquely represented by the Kautz sring $r_1r_2...r_kr_{k+1}$, and each Kautz string $b_1b_2...b_kb_{k+1}$ in Kautz namespace K(d,$k+1$) can be uniquely represented by the edge from node $b_1b_2...b_k$ to node $b_2...b_kb_{k+1}$. Thus under uniform all-to-all communication load, for any edge e=$r_1r_2...r_kr_{k+1}$ in $K(d,k)$, its load equals the number that the Kautz string $r_1r_2...r_kr_{k+1}$ appears as a substring of the Kautz strings in S. From Lemma 3, the load L_n of R is:

$$L_e = \begin{cases} k * d^{k-1} + (k-1)\mathrm{d}^{k-2} - k & (r_1 = r_{k+1}) \\ k * d^{k-1} + (k-1)d^{k-2} - k + 1 & (r_1 = r_k \quad and \quad r_2 = r_{k+1}) \\ k * d^{k-1} + (k-1)d^{k-2} & (others) \end{cases}$$

In Kautz graph $K(d,k)$, the average load of edges $\mathrm{Avg}(L_e)=N*(N-1)*h/|E|$ (|E| is the number of edges in $K(d,k)$ and |E|=N*d), thus

$$\mathrm{Avg}(L_e)= N*(N-1)*h/(N*d) = (N-1)*h/d = k*d^{k-1}+(k-1)*d^{k-2}-k/d+1/(d*(d+1))$$

Because $\mathrm{Max}(L_e)$- $\mathrm{Avg}(L_e) = k/d - 1/(d*(d+1)) = h/d << \mathrm{Avg}(L_e)$, and

$$\mathrm{Max}(L_e)/\mathrm{Avg}(L_e) =1+(h/d)/((N-1)*h/d)=1+ 1/(N-1)=1+o(1)$$

Thus Kautz graph $K(d,k)$ is $(1+o(1))$-edge-congestion-free.

Therefore, Kautz graph $K(d,k)$ is $(1+o(1))$-congestion-free. □

From Theorem 1, it is easy to get that the Kautz graph is constant congestion.

Now we give the lemmas referred in the proof above.

Lemma 1 *The Mapping f is a bijection from M to S.*

Proof: Obviously, S3∩S4=∅ , S=*KautzSpace*(d,2k)∪*KautzSpace*(d,2k-1)-S1-S2.

First we prove that f is an injection.

∀δ∈M, then δ is a routing path from a certain node U=$u_1u_2...u_k$ to another node V=$v_1v_2...v_k$(U≠V):

1) if $u_k \neq v_1$, then the routing path δ would be :

$$U=u_1u_2...u_k \rightarrow u_2u_3...u_kv_1 \rightarrow u_3u_4...u_kv_1v_2 \rightarrow .... \rightarrow u_kv_1v_2...v_{k-1} \rightarrow v_1v_2...v_k=V$$

Thus f(δ)=$u_1u_2...u_kv_1v_2...v_k$, thereby f(δ) ∈ *KautzSpace*(d,2k). Since U≠V, thereafter f(δ)∉S1, thus f(δ)∈*KautzSpace*(d,2k)-S1, i.e. f(δ)∈S3.

2) if $u_k=v_1$, then the routing path δ would be :

U= $u_1u_2...u_k \rightarrow u_2u_3...u_kv_2 \rightarrow u_3u_4...u_kv_2v_3 \rightarrow ....\rightarrow u_kv_2...v_{k-1}v_k = v_1v_2...v_k$=V

Thus f(δ)=$u_1u_2...u_kv_2...v_k$, thereby f(δ) ∈ *KautzSpace*(*d*,2*k*-1).Since U≠V, i.e. $u_1u_2...u_k \neq u_kv_2...v_k$, thereafter f(δ) ∉ S2, and f(δ) ∈ *KautzSpace*(*d*,2*k*-1)-S2, i.e. f(δ) ∈ S4.

So ∀δ ∈ M, f(δ) ∈ S, that is, the range of mapping f is S, and f is a mapping from M to S. Obviously, the identical routing path can only be mapped to one Kautz string, and different routing paths will be mapped to different Kautz strings, thus f is an injection.

Then we'll prove that f is a surjection.

∀ξ ∈ S, since S3∩S4=Φ, thus we may find that ξ ∈ S3 or ξ ∈ S4.

If ξ ∈ S3, let ξ= $a_1a_2...a_{2k-1}a_{2k}$. According to the definition of S3, $a_1a_2...a_k$ and $a_{k+1}a_{k+2}...a_{2k}$ are both valid Kautz strings in *KautzSpace*(*d*,*k*) , and $a_1a_2...a_k \neq a_{k+1}a_{k+2}...a_{2k}$, $a_{k+1} \neq a_k$. Consider routes δ' in Set M with length *k* which originate from source node $a_1a_2...a_k$ to destination node: $a_1a_2...a_k \rightarrow a_2a_3...a_ka_{k+1} \rightarrow a_3a_4...a_{k+2} \rightarrow ...... \rightarrow a_k...a_{2k-1} \rightarrow a_{k+1}...a_{2k}$, we may find that ξ=f(δ'), i.e. ∃δ' ∈ M, s.t. ξ=f(δ').

If ξ ∈ S4, let ξ = $b_1b_2...b_{2k-1}$. According to the definition of S4, we can get that $b_1b_2...b_k$ and $b_kb_{k+1}...b_{2k-1}$ are all valid Kautz strings in *KautzSpace*(*d*,*k*); what's more, $b_1b_2...b_k \neq b_kb_{k+1}...b_{2k-1}$. Consider route δ' with length k-1 in set M which originates from source node $b_1b_2...b_k$ to target node $b_kb_{k+1}...b_{2k-1}$: $b_1b_2...b_k \rightarrow b_2b_3...b_kb_{k+1} \rightarrow b_3b_4...b_{k+2} \rightarrow ...... \rightarrow b_{k-1}b_k...b_{2k-2} \rightarrow b_kb_{k+1}...b_{2k-1}$. Thus we may find ξ= f(δ'), that is, ∃δ' ∈ M, s.t. ξ=f(δ').

Thus f is a bijection. □

Lemma 2 *For any Kautz string R=$r_1r_2...r_k$ in KautzSpace(d,k), the number of R appearing as the substring(except for the prefix) of Kautz strings in set S is:*

$$L_R = \begin{cases} k * d^k + (k-1)\mathrm{d}^{k-1} - k & (r_1 \neq r_k) \\ k * d^k + (k-1)d^{k-1} - k + 1 & (r_1 = r_k) \end{cases}$$

Proof. *S*= *KautzSpace*(*d*,2*k*)∪*KautzSpace*(*d*,2*k*-1)-S1-S2.

Based on theories of combinatorics, the number of times that R appears as a substring (except for the prefix) of Kautz string in *KautzSpace*(*d*,2*k*) are $k*d^k$. (R can be placed in *k* different places in a Kautz string with length 2*k,* and the other *k* places left all have *d* choices.). Similarly, the number of times that R appears as a substring (except for the prefix) of Kautz string in *KautzSpace*(*d*,2*k*-1) is $(k-1)*d^{k-1}$.

Then we calculate the number of times that R appears in S1 and S2. If R appears as a substring of the Kautz string ξ in S1={$u_1u_2...u_ku_1u_2...u_k$ | $u_1u_2...u_ku_1u_2...u_k$ ∈ *KautzSpace*(*d*,2*k*) } and R appears at No. m place of ξ, assuming ξ=$b_1b_2...b_kb_{k+1}b_{k+2}...b_{2k}=b_1b_2...b_kb_1b_2...b_k$, then U=$b_m...b_kb_1b_2...b_{m-1}$ ($b_m \neq b_{m-1}$), i.e., $r_1 \neq r_k$. Similarly, if R appears in S2, $r_1=r_k$.

Thus if $r_1 \neq r_k$, R would not appear in S2. For each *m* that satisfies 1<*m*≤*k,* we could construct a unique Kautz string ξ'=$r_{k-m+2}...r_kr_1r_2...r_kr_1r_2...r_{k-m+1}$ with length 2*k*: ξ' ∈ S1, R appears at No. *m* place of ξ' and R also appears at No. *k*+1 place of $r_1r_2...r_kr_1r_2...r_k$ that is in S1. Therefore, the number of times that R appears in S1 is *k* and the number of times that R appears in S2 is 0.

If $r_1=r_k$, for each m that satisfies $1<m\leq k$, we could construct a unique Kautz string $\xi' = r_{k-m+1}...r_k r_2...r_k r_2...r_{k-m} r_{k-m+1}$ with length $2k$-1: $\xi' \in$ S1 and R appears at No. m place of ξ'. Therefore, the number of times that R appears in S2 is k-1 and the number of times that R appears in S1 is 0.

Therefore, for any node R=$r_1r_2...r_k$ in Kautz graph $K(k,d)$,

If $r_1\neq r_k$, $L_R=k*d^k+(k-1)*d^{k-1}-k$.

If $r_1=r_k$, $L_R=k*d^k+(k-1)*d^{k-1}-(k-1)$. □

Lemma 3 *For each Kautz string e=$r_1r_2...r_kr_{k+1}$ in KautzSpace(d,k+1), the number of times that e appears as the substring of Kautz strings in set S is:*

$$L_e = \begin{cases} k*d^{k-1}+(k-1)d^{k-2}-k & (r_1=r_{k+1}) \\ k*d^{k-1}+(k-1)d^{k-2}-k+1 & (r_1=r_k, r_2=r_{k+1}) \\ k*d^{k-1}+(k-1)d^{k-2} & (else) \end{cases}$$

The proof of Lemma3 is similar to Lemma 2 and omitted here.

4 FissionE Sketch

The Kautz graph has optimal diameter and good fault tolerance characteristic. Also it is constant congestion when using long path routing algorithm. Thus the Kautz graph is a good static topology to construct DHT schemes. We propose FissionE, a novel constant degree, $O(\log N)$ diameter and $(1+o(1))$-congestion-free DHT scheme based on the Kautz graph.

FissionE adopts Kautz graph $K(2,k)$ as its static topology. Each peer in FissionE owns a zone in virtual 2-dimensional Cartesian coordinate. The identifiers of zones in FissionE are Kautz strings with base 2, and zones are organized according to their identifiers. The identifier of a peer is the identifier of the zone it owns. When peers join or leave, the "split large and merge small" policy is adopted for maintenance. Then the entire coordinate space is dynamically partitioned among the peers in the system and the identifiers of zones changes dynamically.

FissionE is somewhat similar to Fission scheme [8], and the main differences lie in the neighborhood of peers and the routing algorithm as well as the update algorithms. In FissionE, the neighborhood invariant (i.e., if zone U and V are neighbors, then $||U|-|V||\leq 1$) is kept, but there is no brother-edges. An example of FissionE neighborhood is shown in Figure 2. Routing algorithm in FissionE is much like the long path routing algorithm in the Kautz graph, while Fission adopts the short path routing algorithm. The maintenance policy is similar to that in Fission, but the procedure to find fit zone to split or merge is much more complex. Some fault-tolerant mechanisms are also proposed in FissionE. The details of FissinE and Fission are in [8, 14].

Now we show some properties of FissionE. The proof and details are in [14].

Theorem 2 (**Congestion Characteristic**) *FissionE is (1+o(1))-congestion-free.*

Theorem 3 (Performance Characteristic) *In an N-peer FissionE system,*

1. The in-degree of each peer is 2 and the out-degree is between 1 and 4. The average out-degree is 2.

*2. The diameter of FissionE systems is less than 2*log_2N.*

5 Conclusions and Future work

The Kautz graph is a good static topology to construct DHT schemes. A novel DHT scheme based on Kautz graph, FissionE, is proposed to achieve constant degree, $O(\log N)$ diameter and $(1+o(1))$-congestion-free. FissionE is a very promising DHT schemes and many topics (such as proximity, heterogeneity, etc.) on FissionE will be investigated thoroughly in our further work.

References

1. Ratnasamy, S., Shenker, S., and Stoica, I.: Routing algorithms for DHTs: some open questions. Proc. of 1st Workshop on peer-to-peer Systems (IPTPs'02), (2002)
2. Stoica, I., Morris, R., Karger. D. *et al.*: Chord: a scalable peer-to-peer lookup service for Internet applications. Proc. of ACM SIGCOMM 2001, ACM Press, New York (2001) 160-177
3. Ratnasamy, S., Francis, P., Handley, M. *et al.*: A scalable content-addressable network. Proc. of ACM SIGCOMM 2001, ACM Press, New York (2001) 149-160
4. Kaashoek, F. and Karger, D. R.. Koorde: A simple degree-optimal hash table. Proc. of 2nd Intl. Workshop on Peer-to-Peer Systems (IPTPS'2003) (2003)
5. Fraigniaud, P., Gauron, P.: The Content-Addressable Network D2B. Tech Rept. 1349, CNRS University paris-Sud, France (2003)
6. Malkhi, D., Naor, M., and Ratajczak, D.: Viceroy: a scalable and dynamic lookup network. Proc. of 21st ACM Symp. on Principles of Distributed Computing (PODC), Monterey, CA (2002)
7. Xu Jun, Kumar Abhishek, Yu Xingxing: On the fundamental tradeoffs between routing table size and network diameter in peer-to-peer networks. IEEE Journal on Selected Areas in Communications (JSAC), No.1 (2004)
8. Li Dongsheng, Fang Xinxin, WANG Yijie *et al.*: A scalable peer-to-peer network with constant degree. Proc. of APPT'2003 (5th International workshop on Advanced Parallel Processing Technologies), LNCS 2834, (2003) 414-425
9. Panchapakesan, G. and Sengupta, A.: On a lightwave network topology using Kautz digraphs. IEEE Transaction on computers, Vol. 48, No. 10, (1999) 1131-1138
10. Bridges, W.G. and Toueg, S.: On the impossibility of directed Moore graphs. Journal of Combinatorial theory, series B, 29 (1980) 330-341
11. Dmitri Loguinov, Anuj Kumar, Vivek Rai, *et al.*: Graph-Theoretic Analysis of Structured Peer-to-Peer Systems: Routing Distances and Fault Resilience. In: Proc. of ACM SIGCOMM' 2003, Karlsruhe, Germany: ACM Press, (2003) 395-406
12. Sivarajan, K. N. and Ramaswami, R.: Lightwave Networks based on de Bruijn Graphs. IEEE/ACM Trans. Networking, Vol.2 (1994) 70-79
13. Chiang Wei-kuo and Chen Rong-Jaye: Distributed Fault-Tolerant Routing in Kautz Networks. Journal of parallel and distributed computing, 20 (1994) 99-106
14. Li Dongsheng, *et al.*: FissionE: A Scalable Constant degree and constant-congestion Peer-to-Peer Network. Tech Rept. PDL-2003-14, National University of Defense technology (2003)