Using the Small-World Model to Improve Freenet Performance

IEEE INFOCOM 2002

by wusf

- Intorduction
- Freenet and Small-World Model
- Simulatiing Freenet Performance Under Heavy Load
- Enhanced-Clustering Cache Placement Scheme
- Analysis
- Conclusions

Introduction

In a peer-to-peer networked system:

- usually employ a naming scheme that allows them to address a node without knowing its exact whereabouts

  < key, data > tuples →facilitates scalable access to these tuples using the key

- possess a routing mechanism that allows

  each node to meaningfully communicate with the rest of the system

Introduction

- Hit ratio in Freenet is dependent on the local policy used to manage the cache of data and the routing table
- A standard LRU like cache replacement policy can result in significantly low hit ratio under high load
- Freenetlike P2P systems will always have limits on cache sizes. → examining the performance degradation of such systems under high load is important

Introduction

- Freenet routing algorithm relies on a high degree of clustering of the routing table entries
- solution to this performance :

1. the routing tables can be shaped by changing the cache replacement policy at each node

2. use intuition from the small world model →the routing distance in a graph is small if each node has pointers to its geographical neighbors as well as some randomly chosen far away nodes.

FREENET AND THE SMALL-WORLD MODEL

- In Freenet, each node maintains a routing table which is a set of <*key* , *pointer* > pairs, where *pointer* points to a node that has a copy of the file associated with *key*.

- A steepest-ascent hillclimbing search with backtracking is used to locate a document

An Example

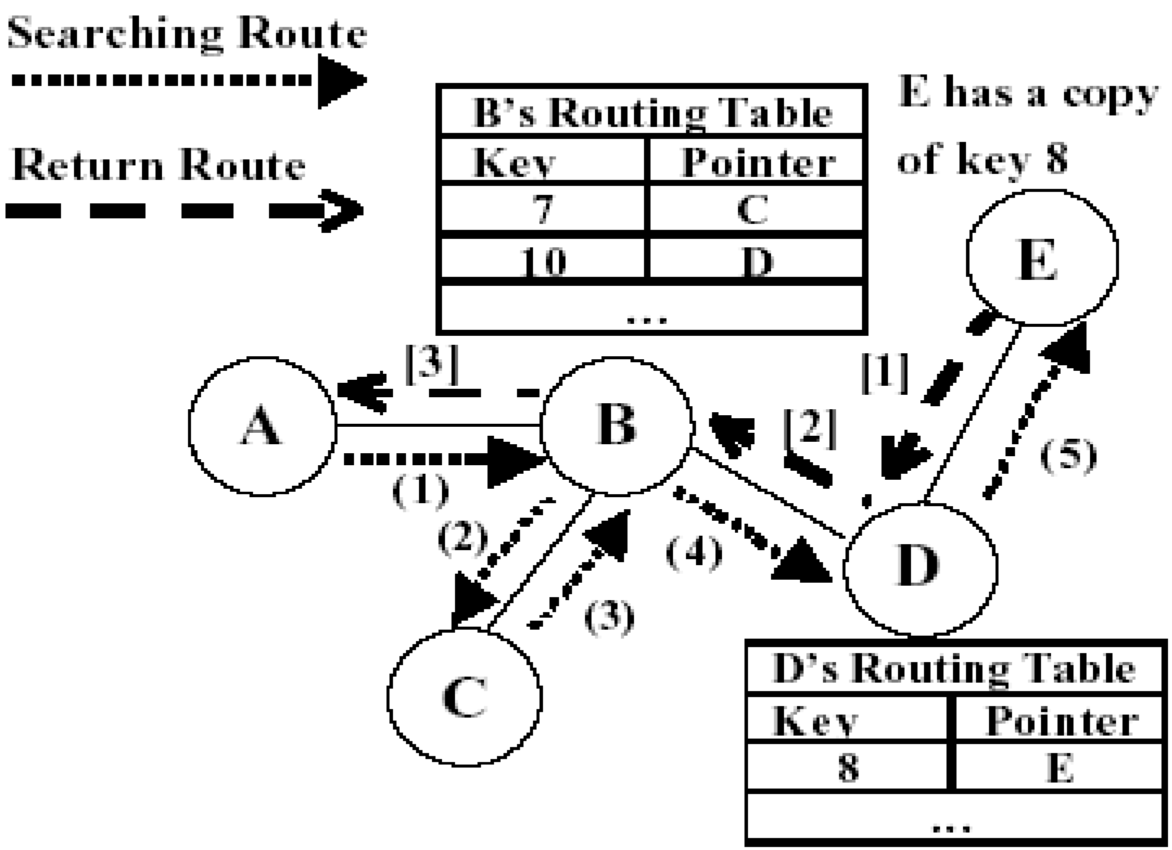

Fig.1 An example of a search in a Freenet network. Node A searches for Key 8 and finally finds it in E.

Small-World Model

- Small-world behavior : each node in the network knows its physical neighbors as well as a small number of randomly chosen distant nodes.

- Shortcut:

leading to a small routing distance between any two individuals the probability of a random shortcut being a distance x away from the source is proportional to 1/x

SIMULATING FREENET PERFORMANCE UNDER HEAVY LOAD

- the performance of the system is represented by two metrics:

1. the request hit ratio
2. the average hops per request.

Better performance :

Higher hit ratio

Lower average hops per request

- Hit Ratio:

  the ratio of the number of successful requests to the total number of requests made

- Average hops per request:

  the ratio of the total number of hops incurred across all requests to the number of all requests.

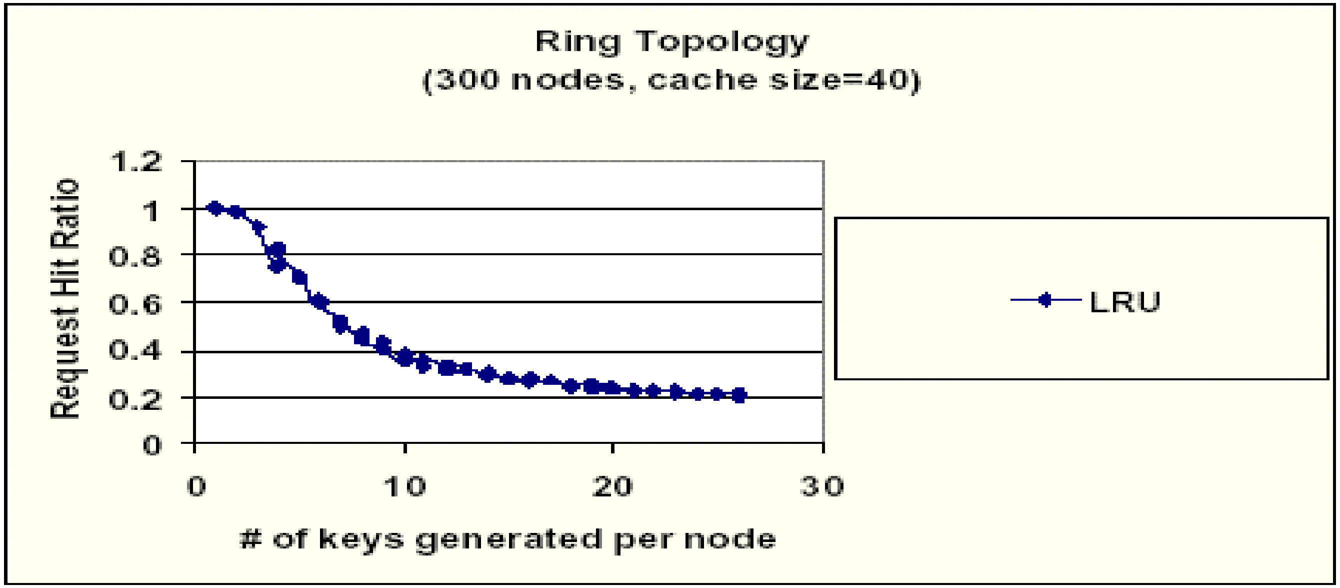

Fig. 2. The curve of hit ratio Vs. Load for Freenet with LRU scheme

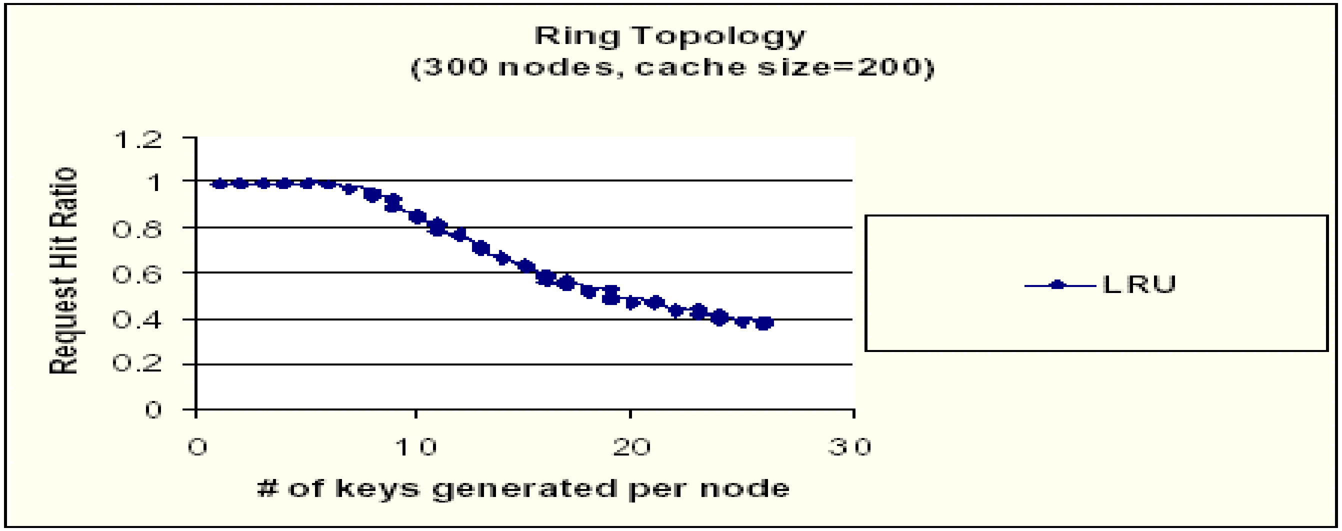

Fig.3. Simulation with a larger datastore and routing table. We see the hit ratio still decreased rapidly with the increasing of the load.

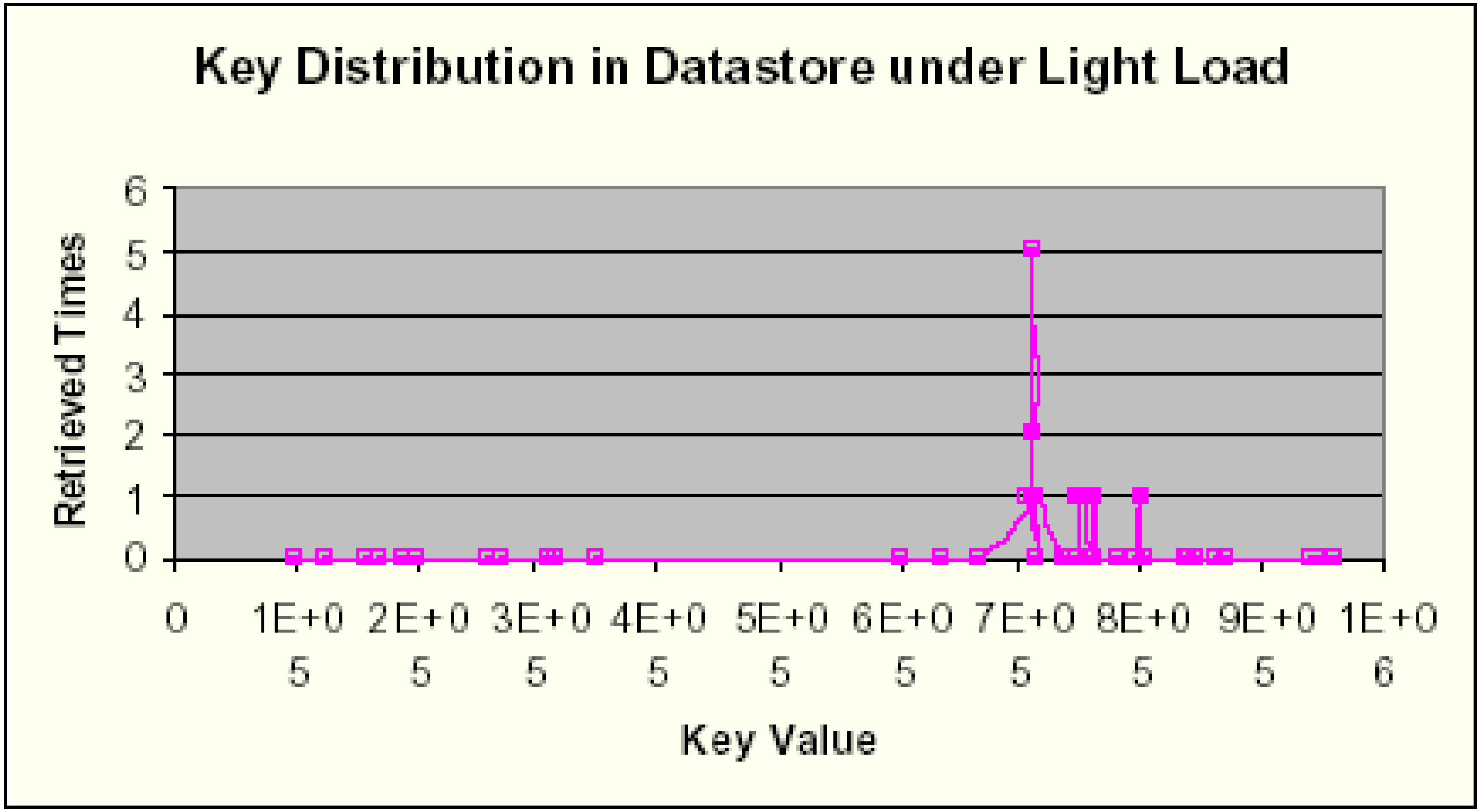

Fig.4. The files stored in the datastore cluster around the two keys generated by the node itself. Local clustering is obvious under light load (in this case average number of keys generated per node =2)

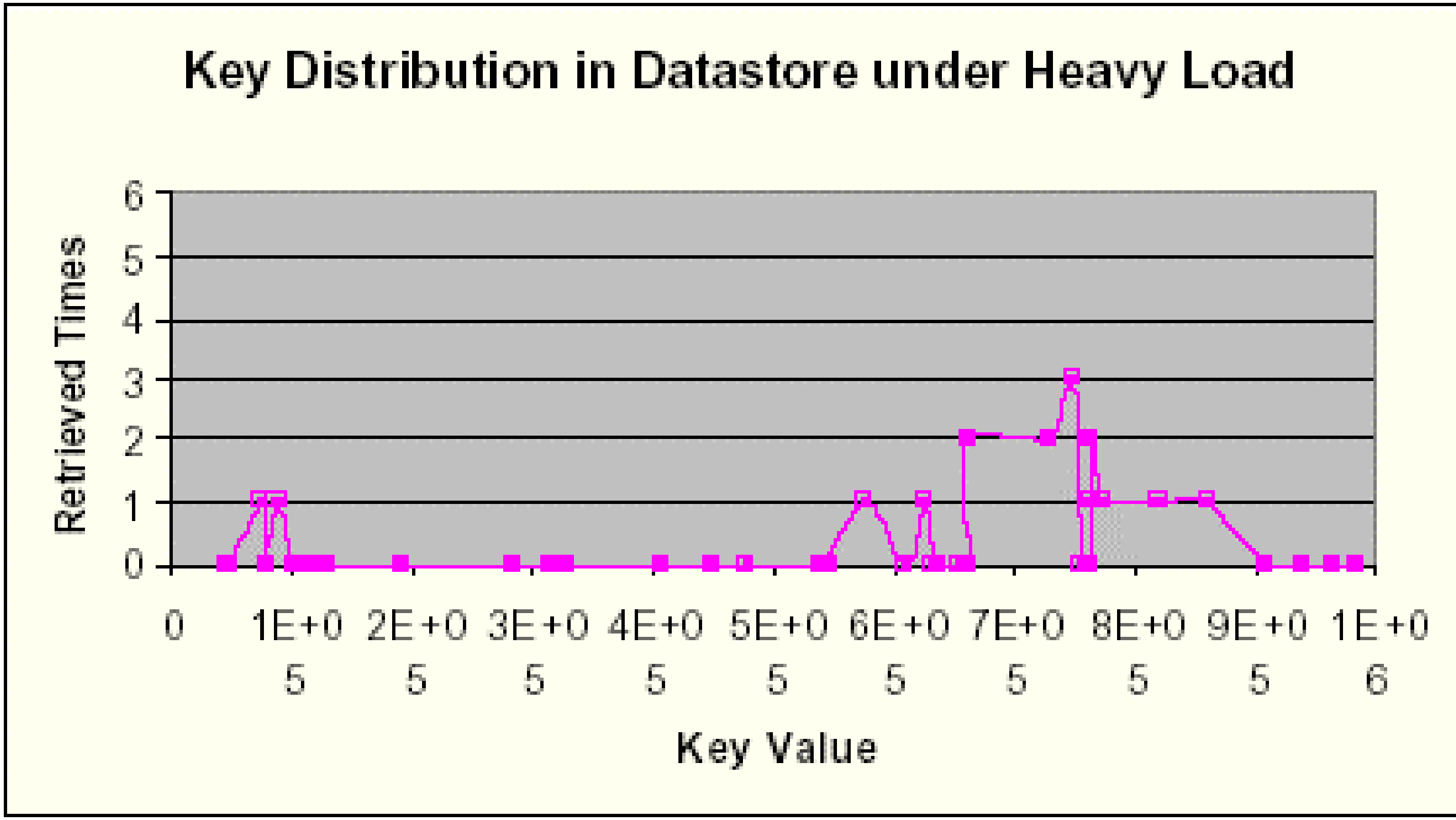

Fig.5. Due to the large number of keys generated by the node itself, the local clustering phenomenon becomes weak under heavy load (in this case average number of keys generated per node =20)

ENHANCED-CLUSTERING CACHE REPLACEMENT SCHEME

- Each node x chooses a seed s(x) randomly from the key space *S* when it joins the system.

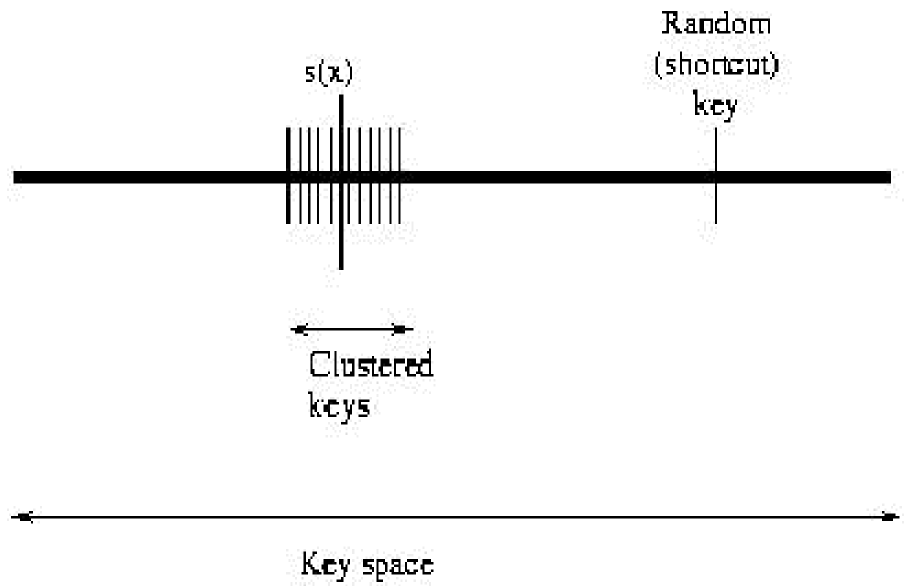

Fig.6. For the routing table at node x to conform to the small-world model, we need a set of key entries clustered around some key s(x) and one or more randomly chosen shortcut keys

- When the datastore at a node is full and a new file with key u arrives

Distance (v, seed) = *Datastore w Max* $\in$ Distance (w, s(x))

Then compare the value of

Distance(u, seed) and Distance (v, seed)

Enforced-clustering scheme

- If Distance(u, seed) $\leq$ Distance (v, seed), cache u and evict v. Create an entry for u in the routing table.

→This has the effect of clustering the keys in the routing table around the seed of the node.

Enforced-clustering with random shortcuts scheme

- If Distance(u, seed) > Distance (v, seed), cache u, evict v and create an entry for u in the routing table with a probability p (randomness).

→Creating a few random shortcuts.

Comparison of Three Cache Replacement Schemes

- LRU always throws out the least recently used file from the datastore
- Enforced-clustering implements the scheme outlined above with $p=0$
- Enforced-clustering with random shortcuts implements the scheme outlined above with $p = 0.03$ and still keeps small number of random shortcuts in the routing table

The adv. of Enforced-clustering with random shortcuts

- make Freenet look more like a small-world network

- The probability p=0.03 was chosen in order to achieve the highest hit ratio and the fewest Average hops per request

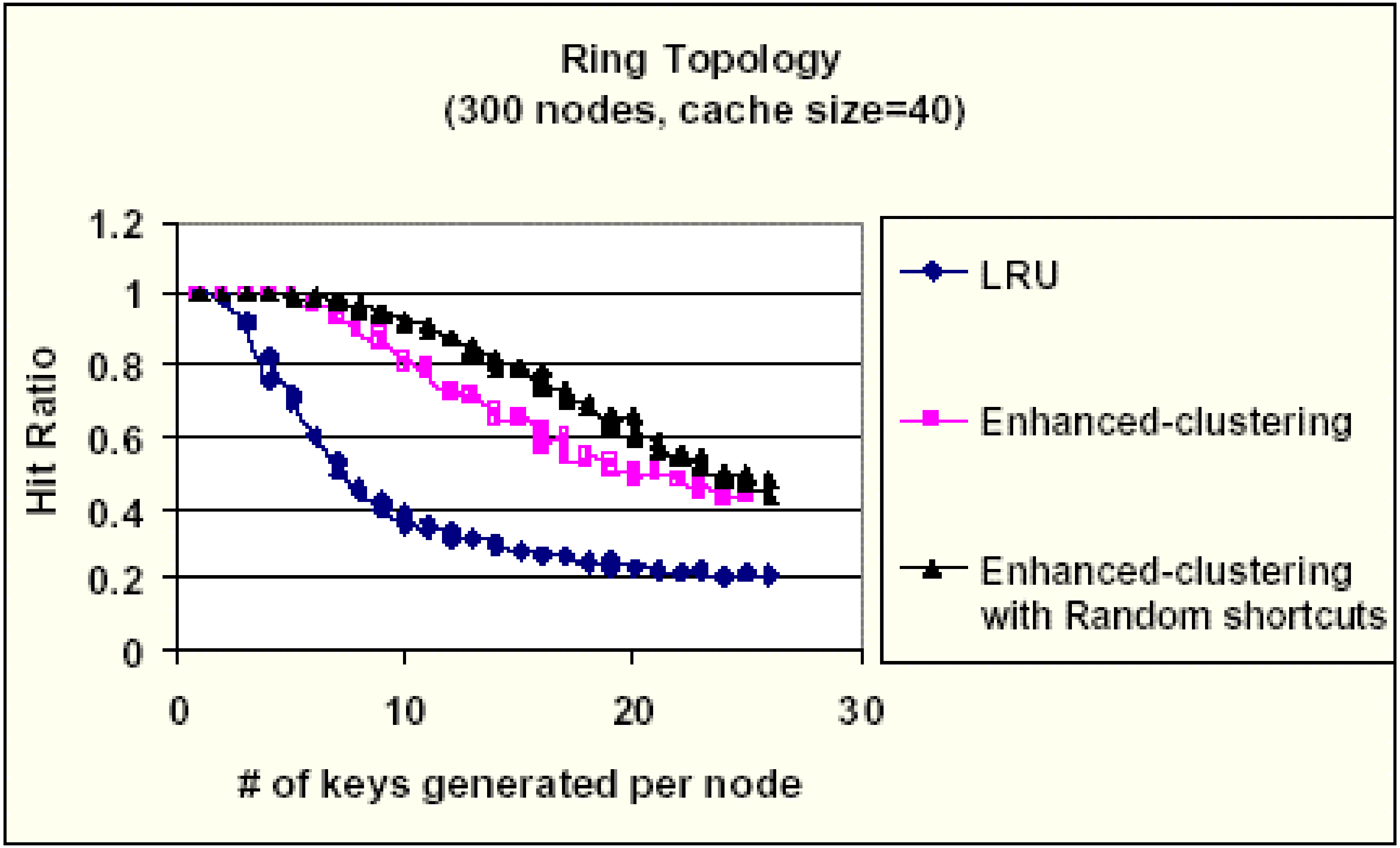

Fig. 7. Availability of Freenet with LRU, enforced-clustering, and enforced-clustering with random shortcuts

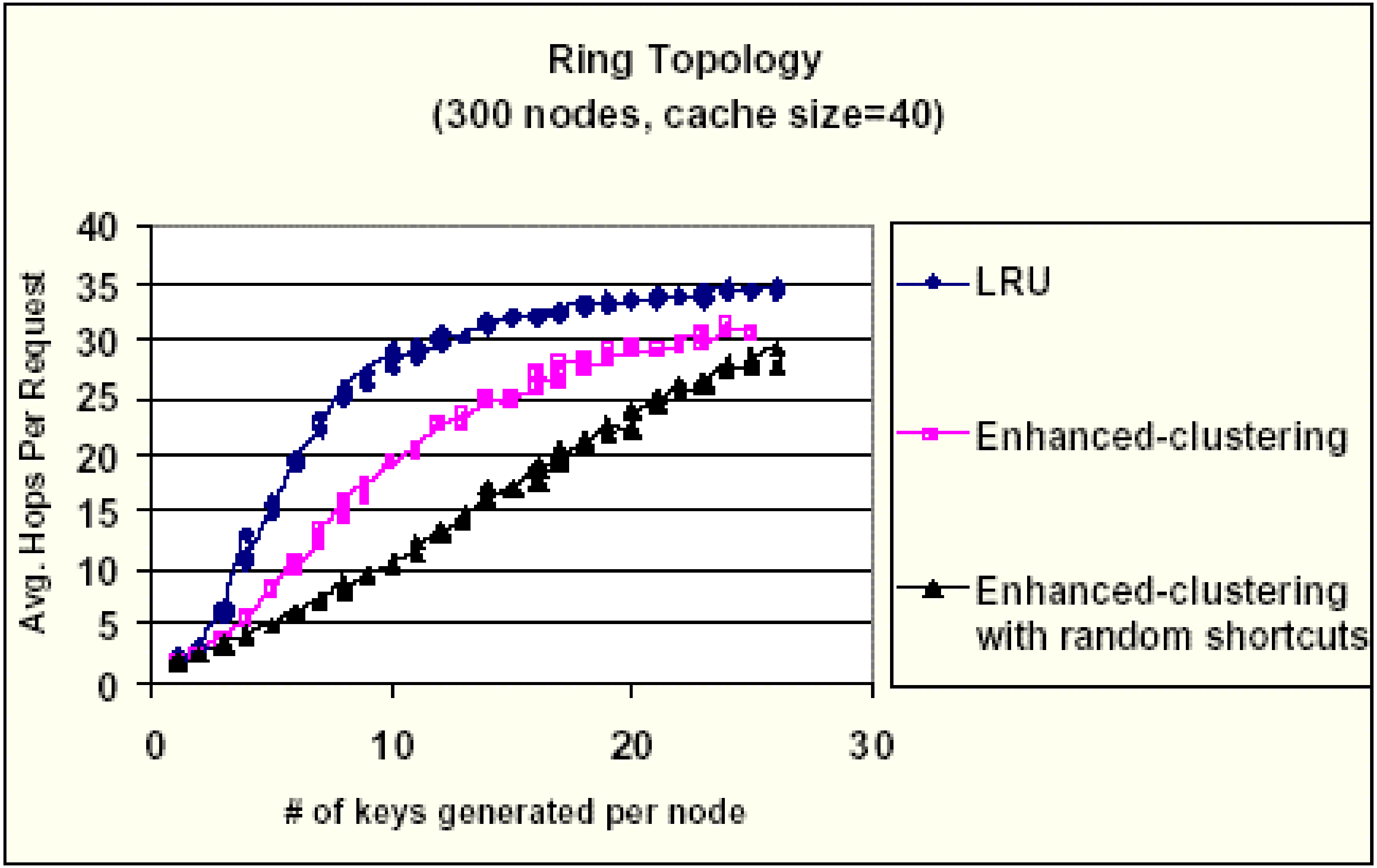

Fig. 8. Average hops per request for Freenet with LRU, enforced-clustering, and enforced-clustering with random shortcuts

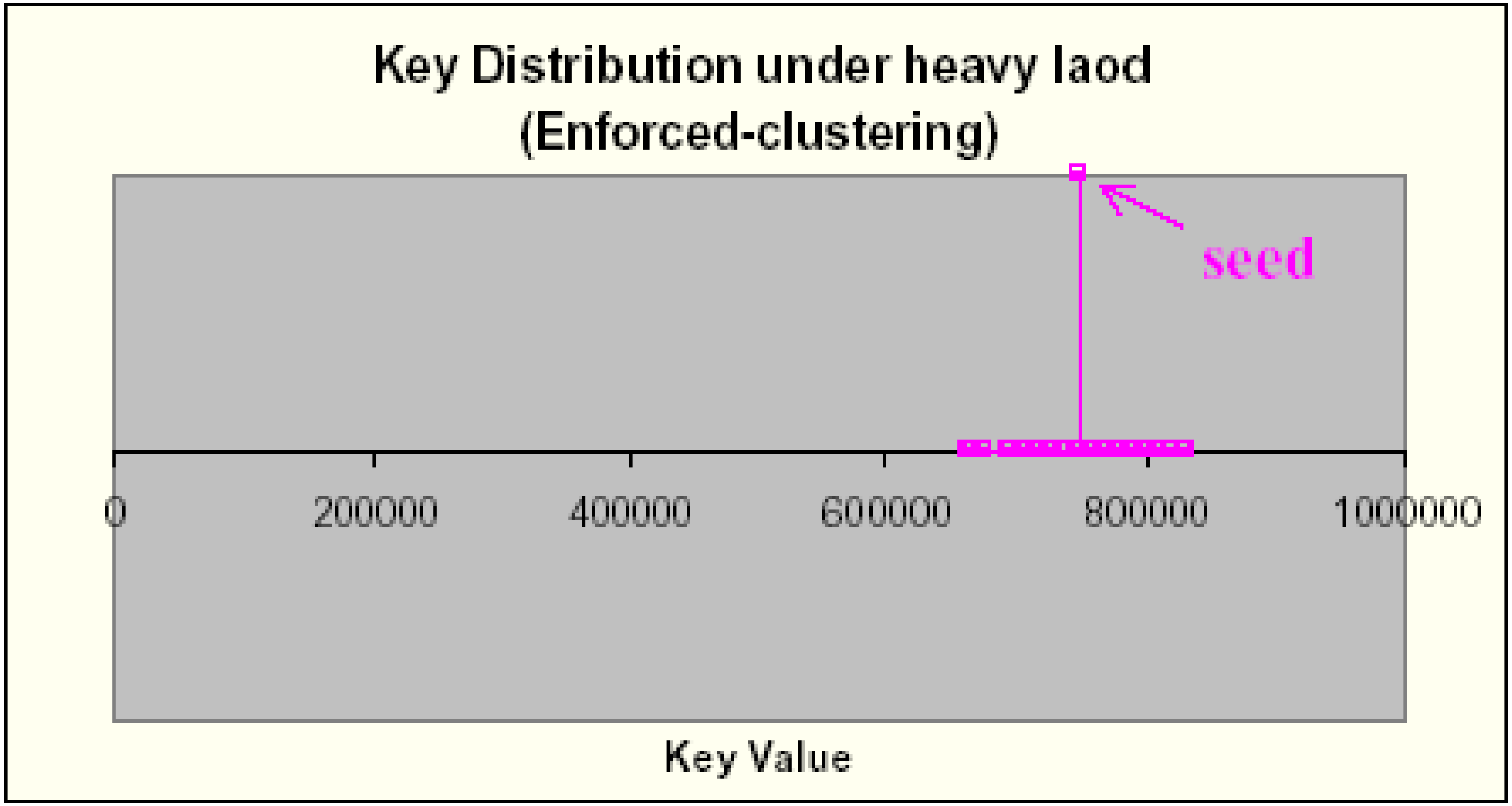

Fig.12. The keys in the routing table cluster around the seed of this node. All keys generated by the node itself are removed in this graph. (in this case average number of keys generated per node =20)

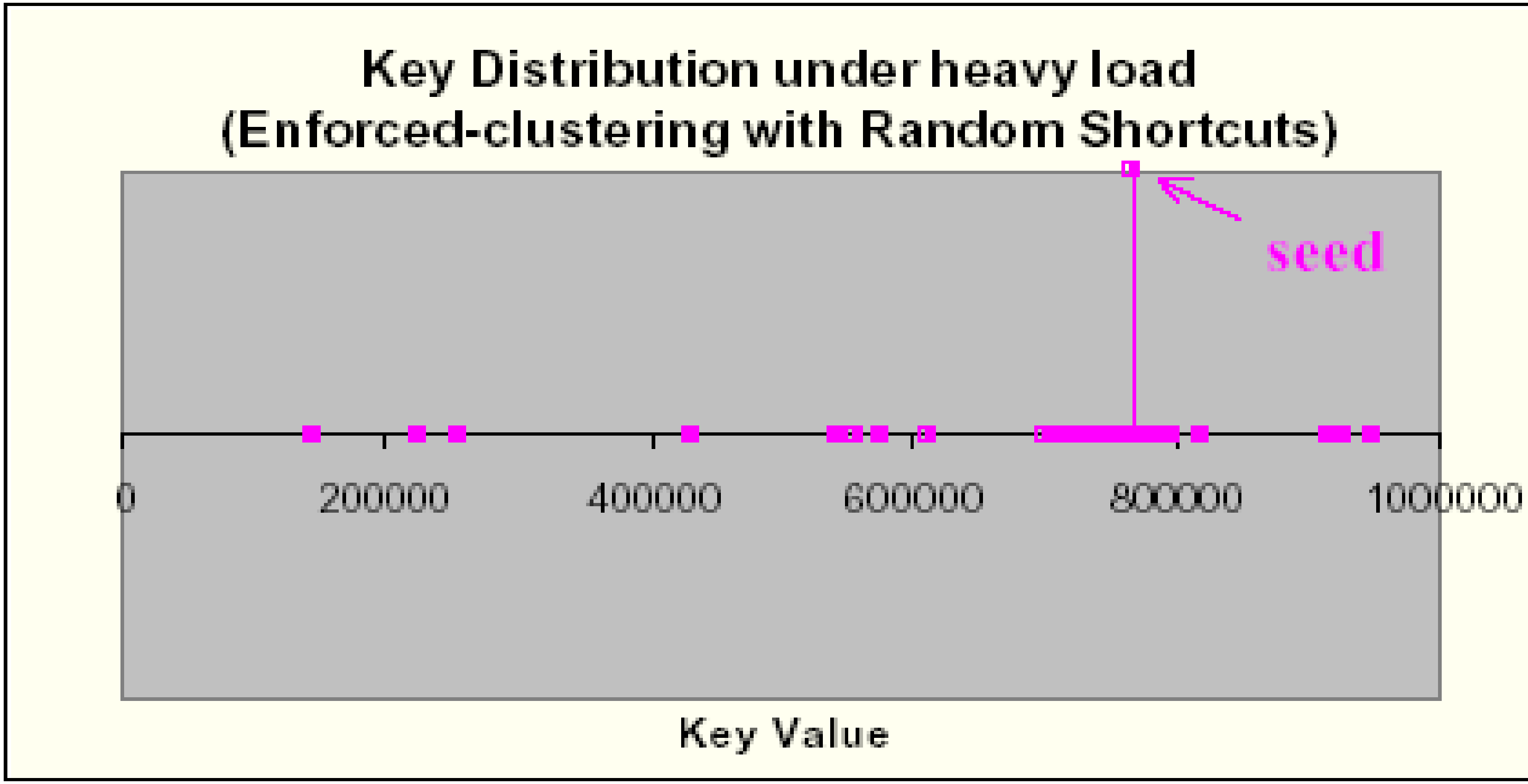

Fig.13. The keys in the routing table cluster around the seed of this node and some random keys distribute in the key space . All keys generated by the node itself are removed in this graph. (in this case average number of keys generated per node =20)

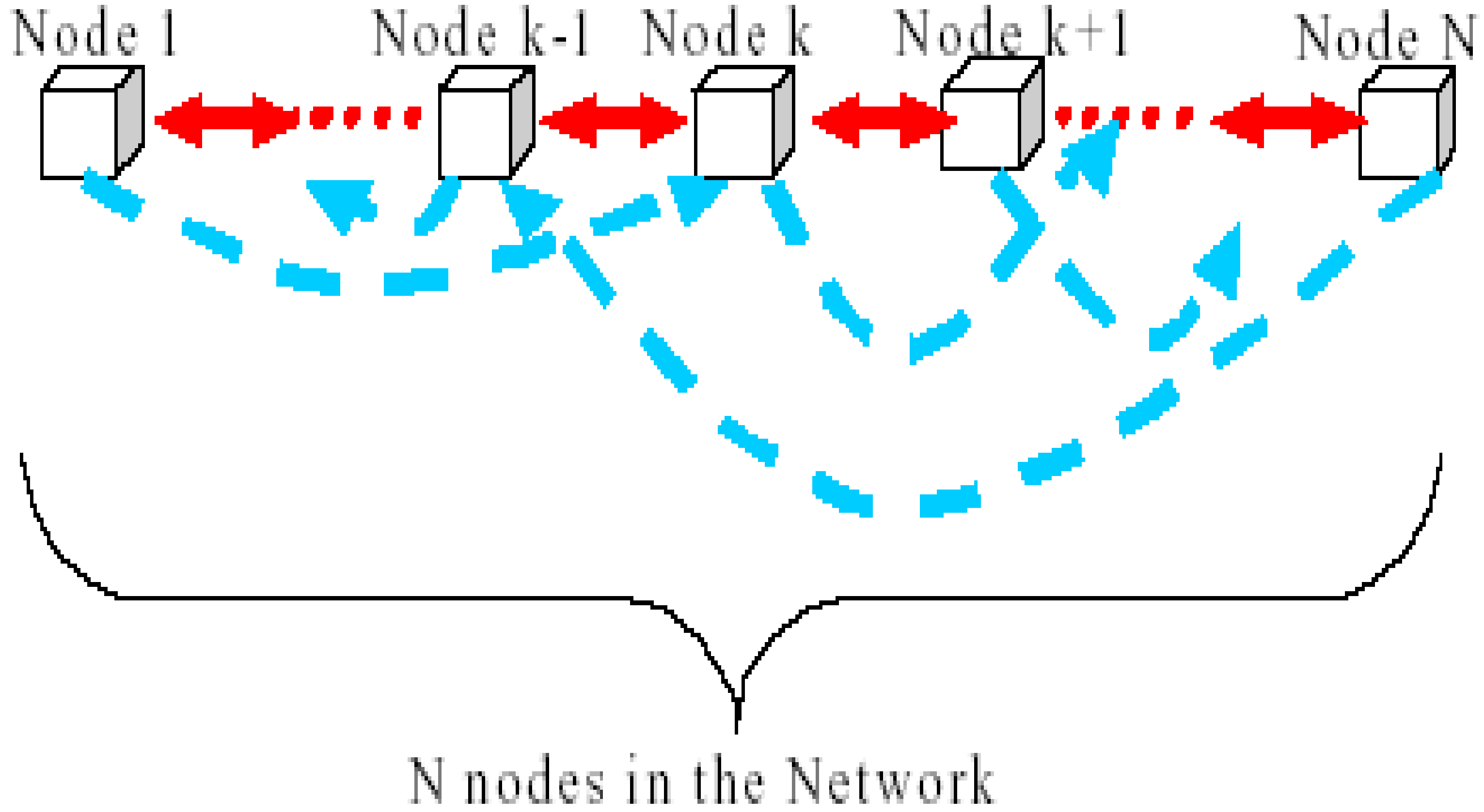

Fig. 15. An idealized model of Freenet from the node level. The long edges correspond to shortcut connections.

Review

- The freenet and small-world model
- To evaluate the performance:
  the request hit ratio
  the average hops per request
- Three Cache Replacement Schemes
  1.LRU
  2.Enforced-clustering scheme
  3.Enforced-clustering with random shortcuts scheme

Conclusion

- We improving the performance of unstructured systems such as Freenet by using intuition from the small.world model

- The simulations show a significant better performance when we used the enhanced.clustering caching with random shortcuts

- It is still a problem to find the best value of p