Pastry's Performance Evaluation Report

This report describes experimental results using the prototype implemented in JAVA. All experiments were performed on a lenovo PC with Intel(R) Core(TM) 2 Duo CPU (2.66GHz) and 2GBytes of main memory, running Windows XP Professional version 2002 (Service Pack 2). The Pastry node software was implemented in JAVA and executed using JDK 6. In all experiments reported in this paper, the Pastry nodes were configured to run in a single Java VM. The emulated network environment maintains distance information between the Pastry nodes. Each Pastry node is assigned a location in a plane; coordinates in the plane are randomly assigned in the range [0,255].

Currently two versions of Pastry protocols have been implemented: the **Original Pastry Protocol** and the **Improved Pastry Protocol**.

Two working modes are provided. One is **USER mode**, which is designed for the user to use and test Pastry. The other is **EXPERIMENT mode,** which is used for testing the performance of Pastry Protocol.

By using the **EXPERIMENT mode**, various statistics on the performance of both the **Original Pastry Protocol** and the **Improved Pastry Protocol** are collected. Using these statistics, several basic performances of Pastry routing are evaluated experimentally as follows.

1. Improved Pastry Protocol's performance

In this part, the performance of the Improved Pastry Protocol will be evaluated. (**The Improved Pastry Protocol improves the Original Pastry Protocol when the new node constructs its Routing Table, Leaf set and Neighborhood set while joining the Pastry network.** Details about the improvement can be found in **readme.doc** in the same directory.)

The first experiment shows the number of routing hops as a function of the size of the Pastry network. We vary the number of Pastry nodes from 2 to 256 in a network where b = 2, |L| = 4, |M| = 4. For a given number of nodes, 100 trials were carried out. In each trial, two Pastry nodes are selected randomly and a message is routed between the pair using Improved Pastry Protocol. Figure 1 and Figure 2 illustrate the results.

1) Averaged number of routing hops for different number of nodes:

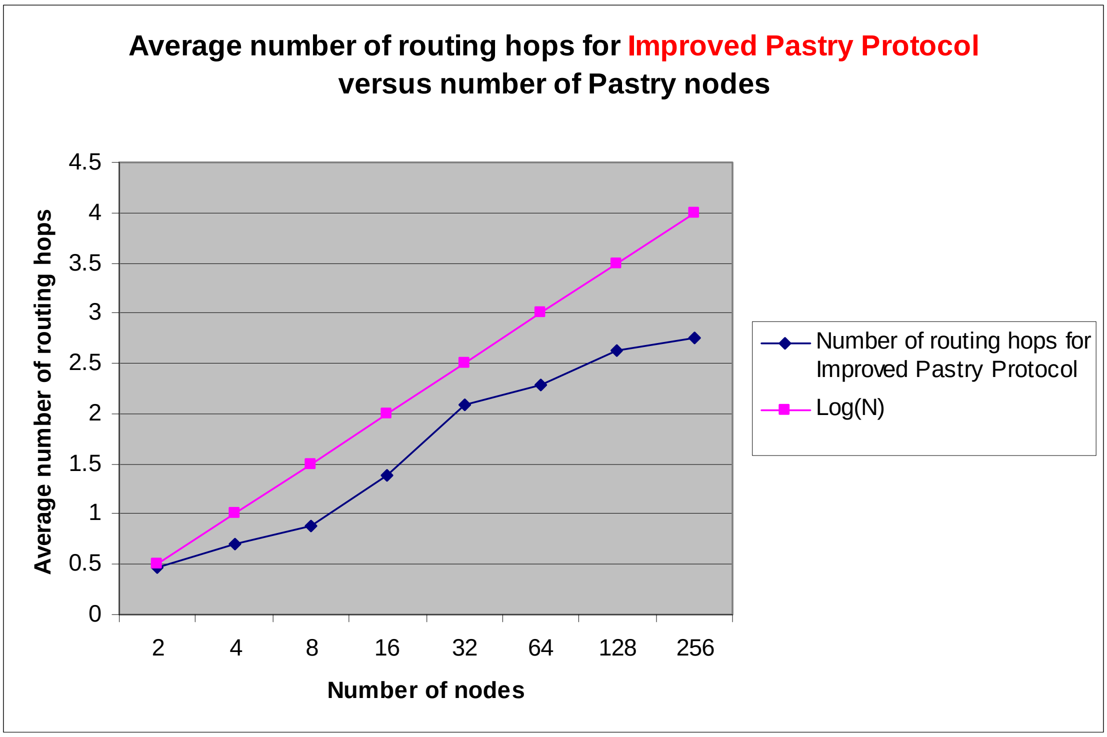

Figure 1.Average number of routing hops for different number of Pastry nodes, b = 2, |L| = 4, |M| = 4 and 100 lookups

Figure 1 shows the average number of routing hops, which is taken as a function of the network size. "LogN" shows the value $\log_{2^b} N$ and is included for comparison. ($\lceil \log_{2^b} N \rceil$ is the expected maximum number of hops required to route in a network containing N nodes). The result show that the number of routing hops scales with the size of the network as predicted.

2) Probability of different routing hops:

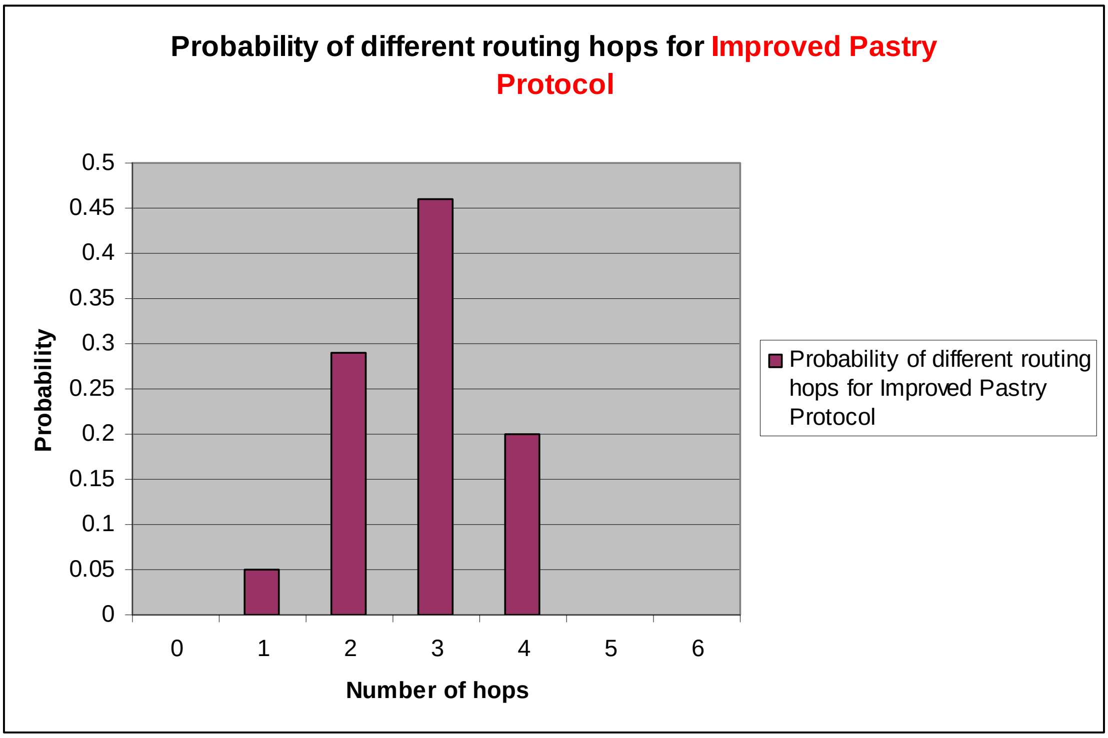

Figure 2.Probability of different number of routing hops, b = 2, |L| = 4, |M| = 4, N = 256 and 100 lookups.

Figure 2 shows the distribution of the number of routing hops taken for a network size of 256, in the same experiment. The results show that the maximum route length is $\lceil \log_2{}^b N \rceil$ ($\lceil \log_2{}^2 256 \rceil = 4$), as expected.

3) Relative distance for different number of nodes:

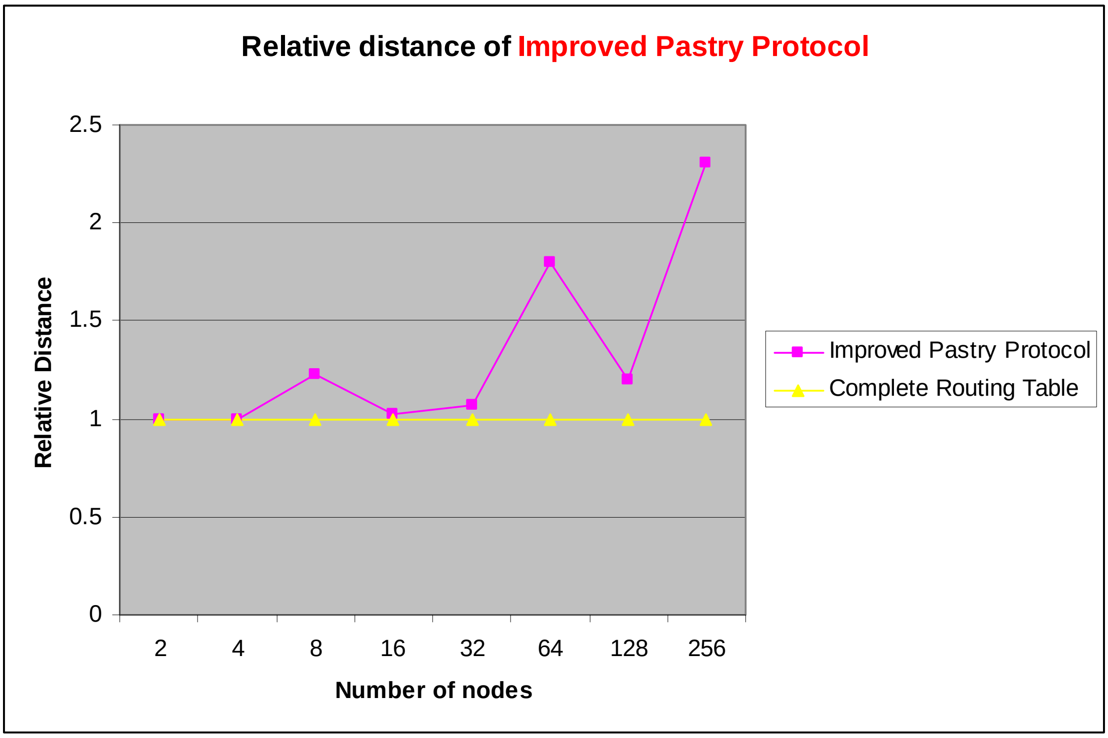

Figure 3.Route distance for different number of Pastry nodes, b = 2, |L| = 4, |M| = 4, and 100 lookups.

The second experiment evaluated the locality properties of Pastry routes. It compares the relative distance a message travels using the Pastry, according to the proximity metric, with that of a fictitious routing scheme that maintains complete routing tables. The distance traveled is the sum of the distances between consecutive nodes encountered along the route in the emulated network. For the fictitious routing scheme, the distance traveled is simply the distance between the source and the destination node. The results are normalized to the distance traveled in the fictitious routing scheme. The goal of this experiment is to quantify the cost, in terms of distance traveled in the proximity space, of maintaining only small routing tables in Pastry. Figure 3 illustrates the result.

4) Average number of hops for different number of leaf set |L|:

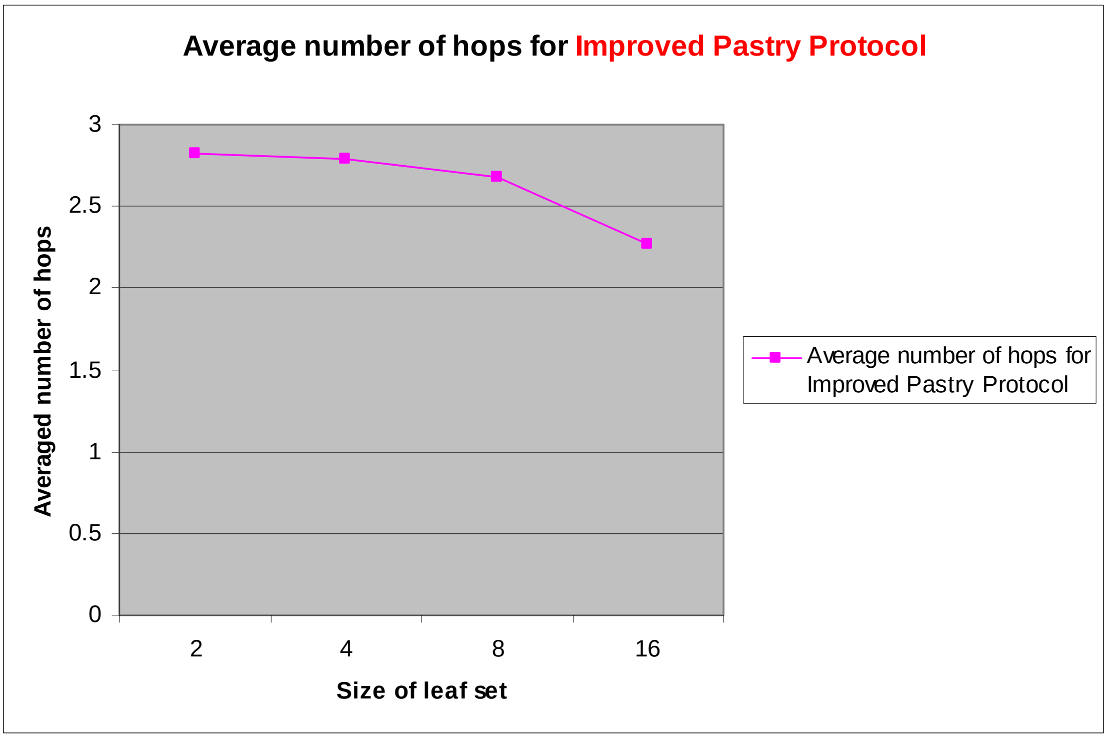

Figure 4.Average number of hops for different size of leaf set (|L|), b = 2, |M| = 4, N = 256 and 100 lookups.

The third experiment evaluated the average number of hops for different size of leaf set |L|. With b = 2, |M| = 4 and N = 256, we varied the size of leaf set from 2 to 16 and got the corresponding results as shown in Figure 4. It can be seen from the figure that the average number of routing hops drops as the size of leaf set becomes larger, as expected.

5) Relative Distance for different number of Neighborhood set |M|:

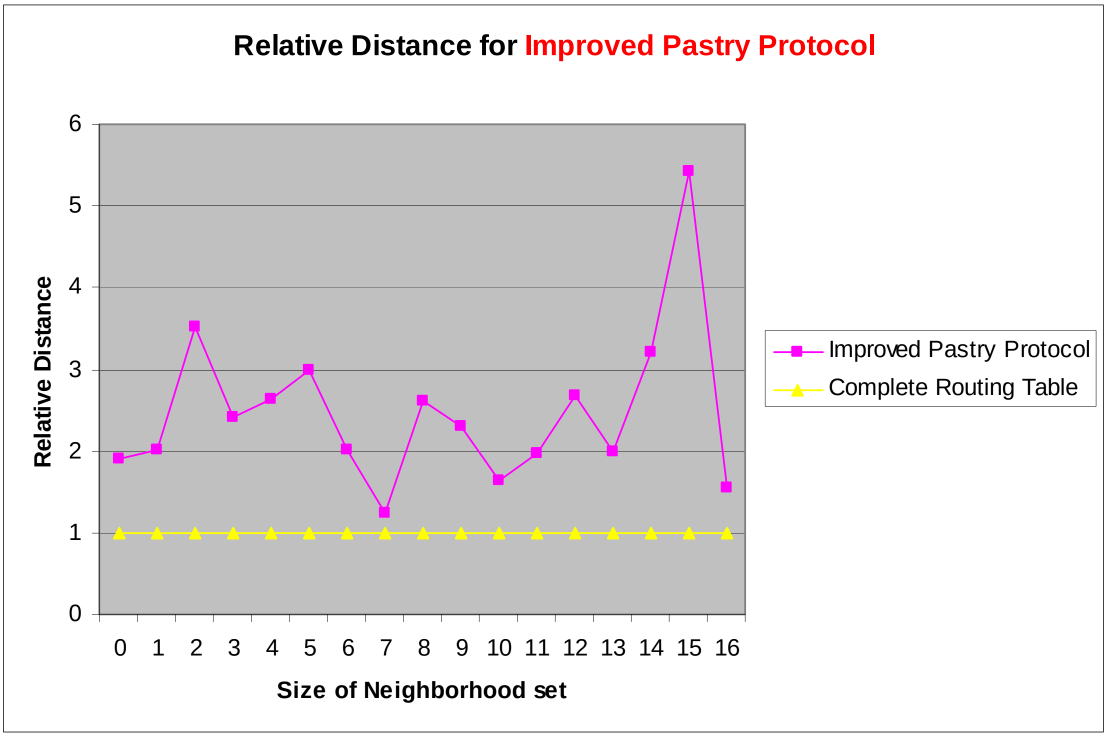

Figure 5.Route distance for different size of neighborhood set (|M|), b = 2, |L| = 4, N = 128 and 100 lookups.

The fourth experiment evaluated the route distance for different size of neighborhood set |M|. With b = 2, |L| = 4 and N = 128, we varied the size of neighborhood set from 0 to 16 and got the corresponding results shown in Figure 5.

2. Original Pastry Protocol's performance

In this part, the performance of the Original Pastry Protocol which is described in the Pastry paper will be evaluated.

The first experiment shows the number of routing hops as a function of the size of the Pastry network. We vary the number of Pastry nodes from 2 to 256 in a network where b = 2, |L| = 4, |M| = 4. For a given number of nodes, 100 trials were carried out. In each trial, two Pastry nodes are selected randomly and a message is routed between the pair using Original Pastry Protocol. Figure 6 and Figure 7 illustrate the results.

1) Averaged number of routing hops for different number of nodes:

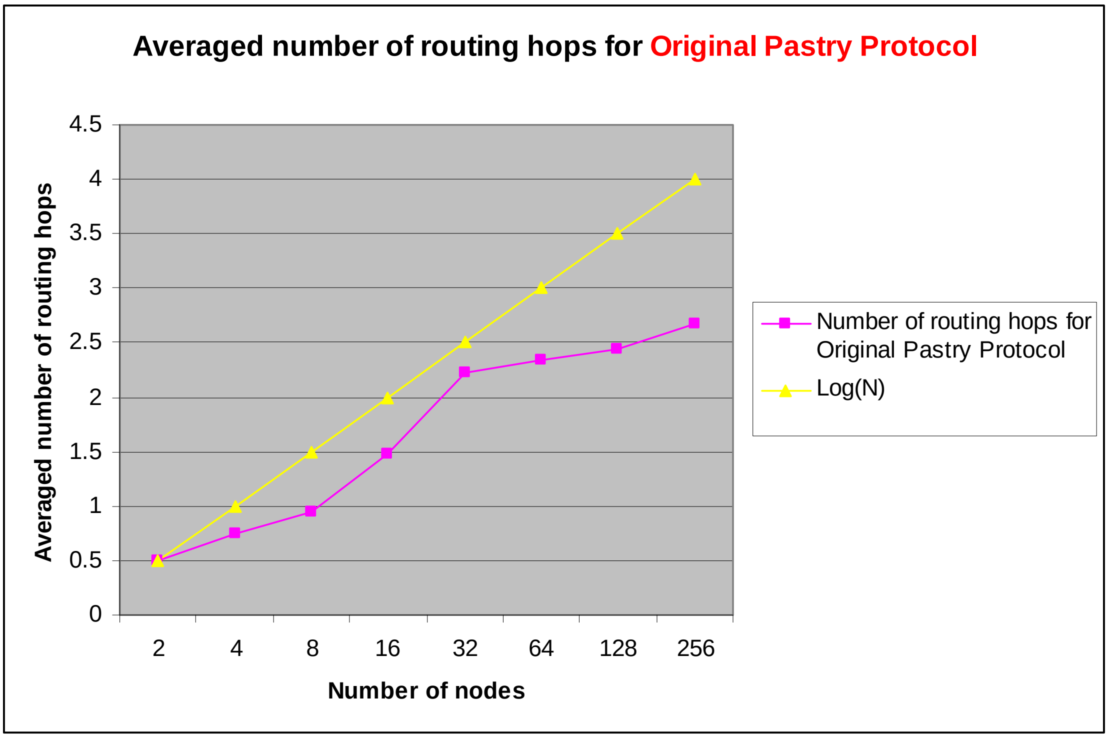

Figure 6.Average number of routing hops for different number of Pastry nodes, b = 2, |L| = 4, |M| = 4 and 100 lookups

Figure 6 shows the average number of routing hops, which is taken as a function of the network size. "LogN" shows the value $\log_2^b N$ and is included for comparison. ($\lceil \log_2^b N \rceil$ is the expected maximum number of hops required to route in a network containing N nodes). The result show that the number of routing hops scales with the size of the network as predicted.

2) Probability of different routing hops:

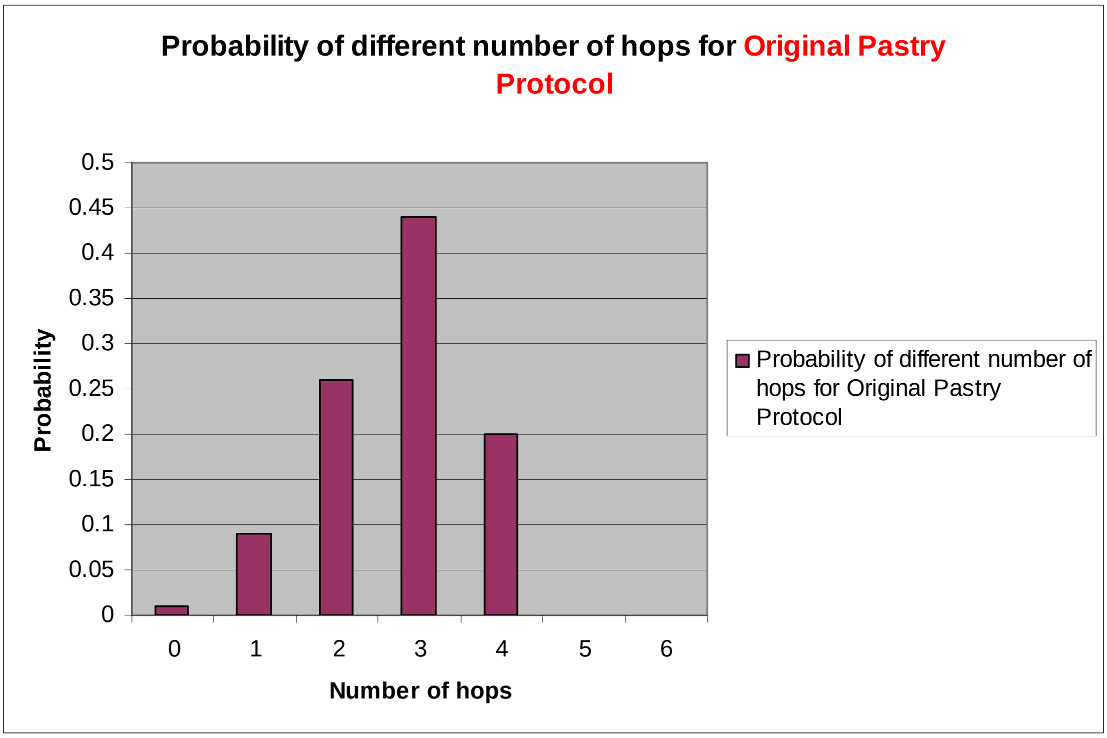

Figure 7.Probability of different number of routing hops, b = 2, |L| = 4, |M| = 4, N = 256 and 100 lookups.

Figure 7 shows the distribution of the number of routing hops taken for a network size of 256, in the same experiment. The results show that the maximum route length is $\lceil \log_{2}{}^{b}N \rceil$ ($\lceil \log_{2}{}^{2}256 \rceil = 4$), as expected.

3) Relative distance for different number of nodes:

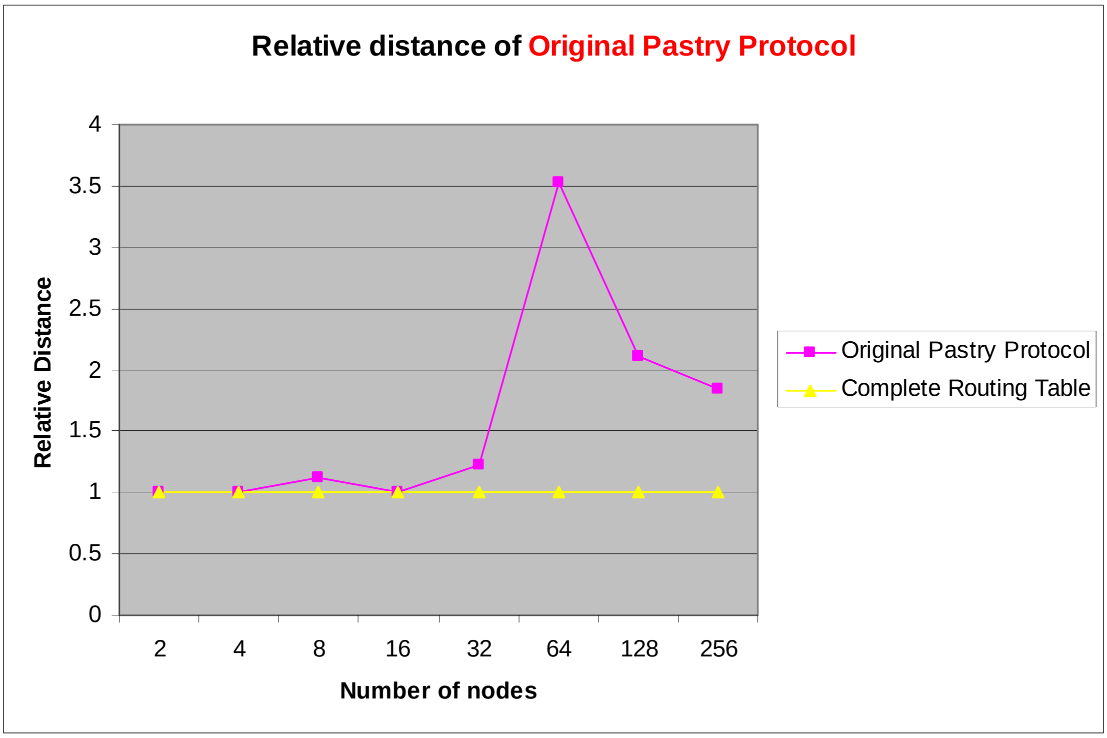

Figure 8.Route distance for different number of Pastry nodes, b = 2, |L| = 4, |M| = 4, and 100 lookups.

The second experiment evaluated the locality properties of Pastry routes. It compares the relative distance a message travels using the Pastry, according to the proximity metric, with that of a fictitious routing scheme that maintains complete routing tables. The distance traveled is the sum of the distances between consecutive nodes encountered along the route in the emulated network. For the fictitious routing scheme, the distance traveled is simply the distance between the source and the destination node. The results are normalized to the distance traveled in the fictitious routing scheme. The goal of this experiment is to quantify the cost, in terms of distance traveled in the proximity space, of maintaining only small routing tables in Pastry. Figure 8 illustrates the result.

4) Average number of hops for different number of leaf set |L|:

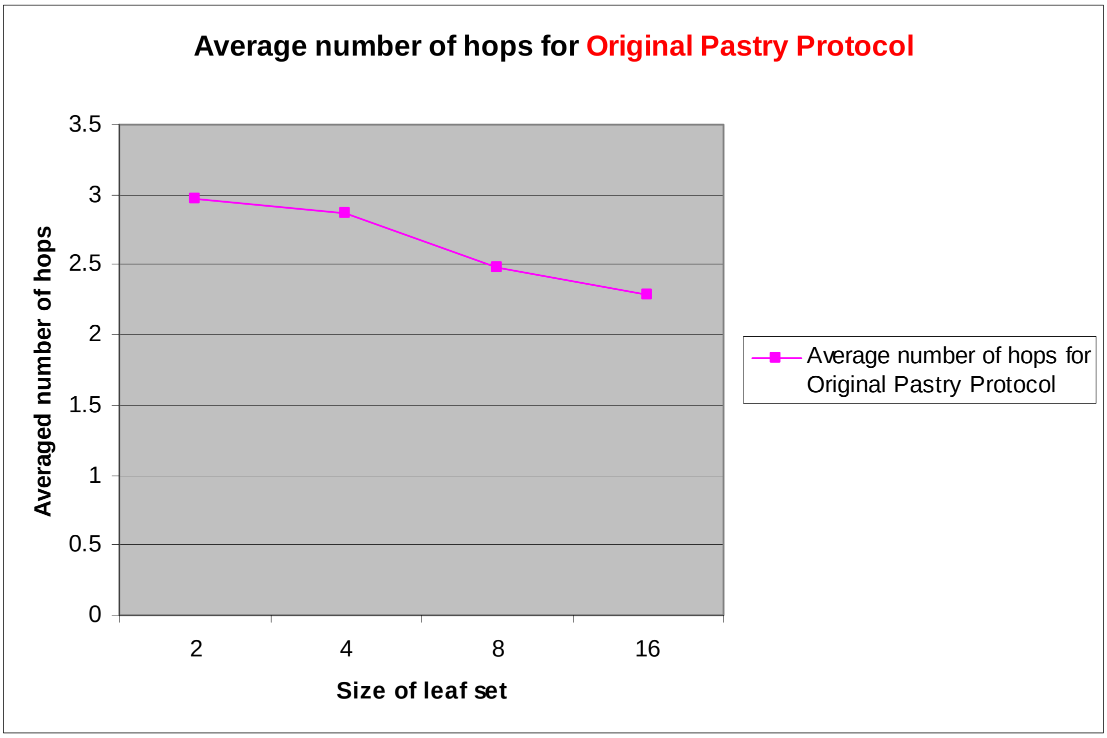

Figure 9.Average number of hops for different size of leaf set (|L|), b = 2, |M| = 4, N = 256 and 100 lookups.

The third experiment evaluated the average number of hops for different size of leaf set |L|. With b = 2, |M| = 4 and N = 256, we varied the size of leaf set from 2 to 16 and got the corresponding results as shown in Figure 9. It can be seen from the figure that the average number of routing hops drops as the size of leaf set becomes larger, as expected.

5) Relative Distance for different number of Neighborhood set |M|:

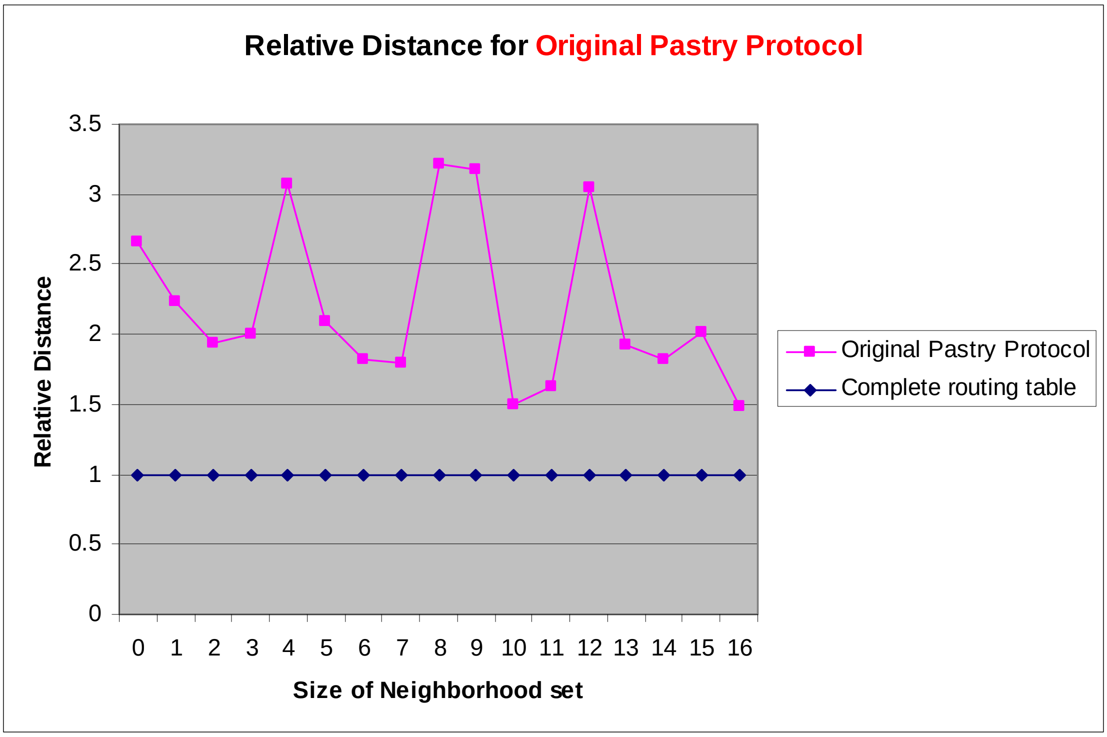

Figure 10.Route distance for different size of neighborhood set (|M|), b = 2, |L| = 4, N = 128 and 100 lookups.

The fourth experiment evaluated the route distance for different size of neighborhood set |M|. With b = 2, |L| = 4 and N = 128, we varied the size of neighborhood set from 0 to 16 and got the corresponding results as shown in Figure 10.

3. Performance Comparison between Original Pastry Protocol and Improved Pastry Protocol

In this part, performance comparison between the Original Pastry Protocol and Improved Pastry Protocol is carried out with respect to the average routing hops and the locality properties.

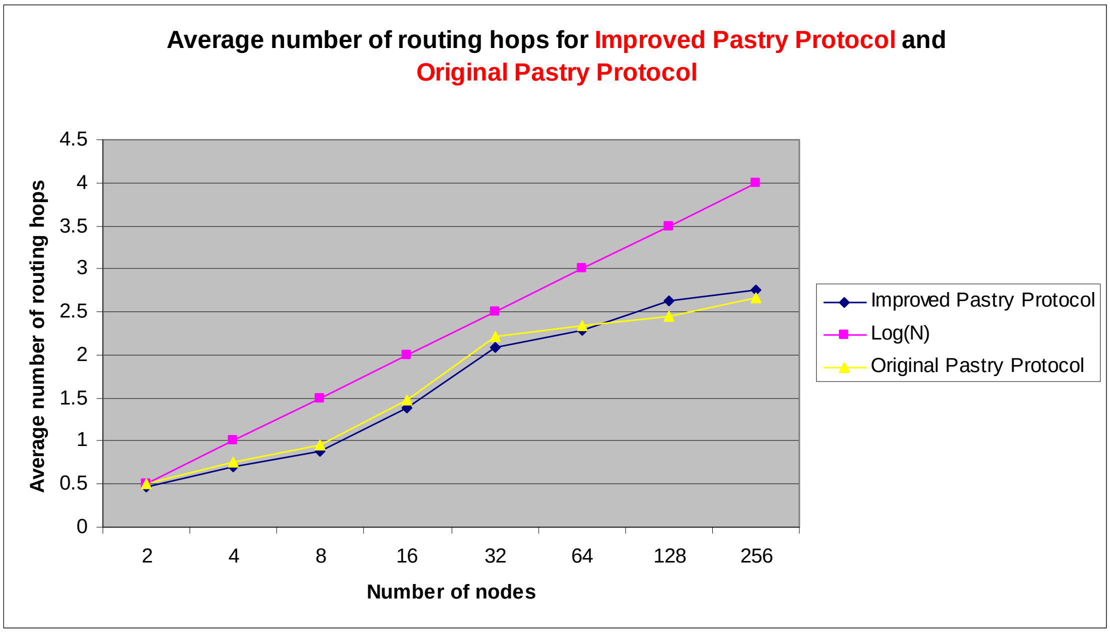

Figure 11.Average number of routing hops for different number of Pastry nodes, b = 2, |L| = 4, |M| = 4 and 100 lookups.

Figure 11 shows the average number of routing hops, which is taken as a function of the network size. "LogN" shows the value $\log_2{}^b N$ and is included for comparison. ($\lceil \log_2{}^b N \rceil$ is the expected maximum number of hops required to route in a network containing N nodes). The result show that the number of routing hops scales with the size of the network in both the Original Pastry Protocol and the Improved Pastry Protocol. At the same time, the figure shows that Improved Pastry Protocol has smaller routing hops than the Original Pastry Protocol most of the time.

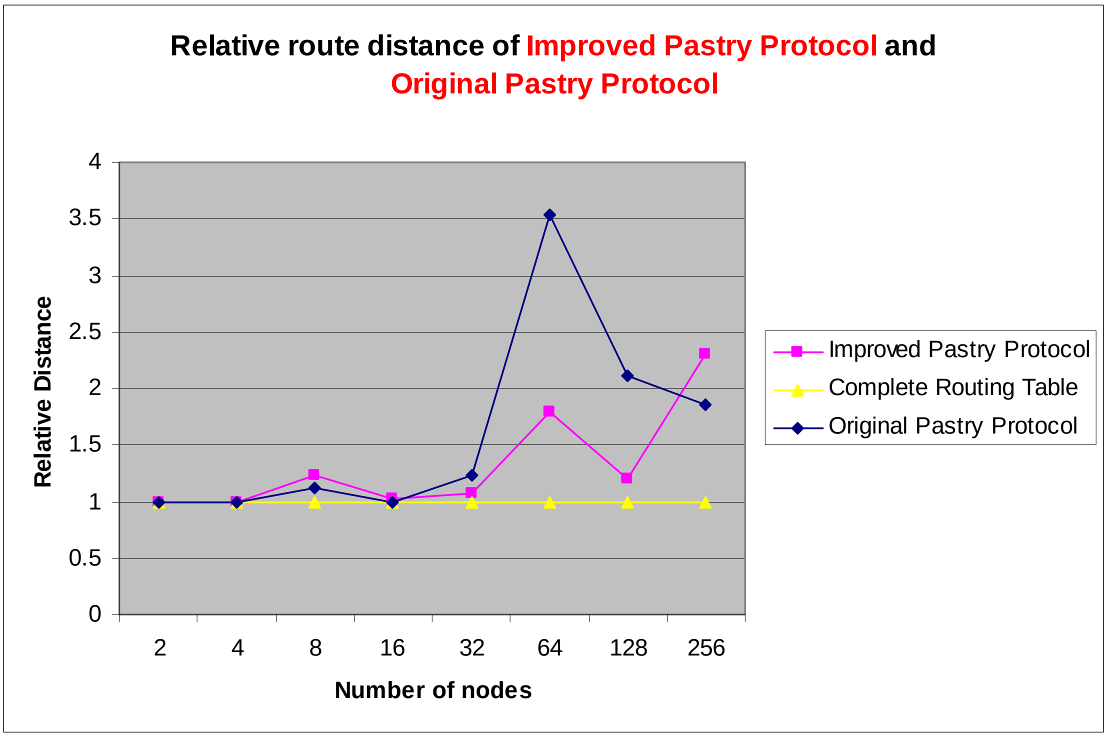

Figure 12.Route distance for different number of Pastry nodes, b = 2, |L| = 4, |M| = 4, and 100 lookups.

We also evaluated the route distance for different size of neighborhood set |M| for both the Original Pastry Protocol and the Improved Pastry Protocol. With b = 2, |L| = 4 and N = 128 in both cases, we varied the size of neighborhood set from 0 to 16 and got the corresponding results shown in Figure 12.

Reference:

[1] Rowstron, Antony and Druschel, Peter (2001) *Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems.* Lecture Notes in Computer Science.