A Novel Content Distribution Architecture Utilizing Network Distance Prediction

Naoto SHIMIZU, Wataru KAMEYAMA, Hiroshi WATANABE

Waseda University
Graduate School of Global Information and Telecommunication Studies
1-3-10 Nishiwaseda Shinjuku-ku, Tokyo, Japan

ABSTRACT

Contents Delivery/Distribution Network (CDN) and Peer-to-Peer (P2P) network have been proposed for a large-scale contents distribution. We have been investigating what mechanisms are required for CDN, P2P, and both of them. It turns out that 1) optimal server selection and 2) content discovery are key role functions. In this paper, we propose a method for optimal server selection. The optimal server is defined to minimize end-to-end delay between a requesting host and the server. Here, end-to-end delay is regarded as network distance. Round trip time (RTT) can be used as a metric to predict end-to-end delay. However, it takes much cost to send a probe packet to a whole set of hosts in order to measure all RTT. Our proposed method send a probing packet to a set of servers called Landmarks, and predict which server is optimal based on these RTT. Simulation result shows the validity of the proposed method. We confirm that this approach can adapt to scalable architecture.

Keywords: Contents distribution, Multimedia system, CDN, P2P

1. INTRODUCTION

A mechanism to provide efficient contents distribution has been a major research topic in recent years. It can be classified into two categories: 1) Network-based, which requires network layer functions,
2) Overlay-based, which requires application layer functions. IP multicast is a typical network layer function and almost all network-based contents distribution utilizes it. It is possible for IP multicast to diffuse contents to large population efficiently. However, group management, distributed multicast address allocation, security, and support for network management, have not been provided for large scale deployment of IP multicast [1]. Network-based contents distribution without IP multicast will require initial large cost. It makes a service model complicated. Therefore, it seems to be difficult to implement network-based contents distribution at the present time.

Overlay-based contents distribution does not require network layer functions, because its service is provided on the Internet. The performance of overlay-based contents distribution, such as client-perceived latency, may be lower than the one of network-based contents distribution. However, it can achieve cost saving compared to network-based contents distribution. Therefore, overlay-based contents distribution is considered to be practical. In this paper, we propose a method to provide efficient overlay-based contents distribution.

This paper is organized as follows. In Section 2, we review some overlay-based contents distribution architectures and point out that optimal server selection and content discovery are key role functions for them. In Section 3, we discuss optimal server selection and present our proposed contents distribution architecture. Content discovery remains as a future work. In Section 4, we present numerical results by network simulator. In Section 5, we present our conclusions.

2. OVERLAY-BASED CONTENTS DISTRIBUTION ARCHITECTURES

Overlay-approach Contents Delivery/Distribution Network (CDN) [2] and Peer-to-Peer (P2P) network are representative overlay-based contents distribution architectures. CDN employs a dedicated set of machines, which are called edge servers or surrogate servers, to store and distribute contents. The contents are deployed among edge servers and each of clients can get contents from appropriate edge servers. Therefore, low client-perceived latency, scalability, fault tolerance, and load balancing are achieved.

P2P relies on clients to hold and distribute contents to other clients. No central server that hold contents is necessary. Therefore, P2P provides better-cost performance and robust properties compared to CDN. Napster and Guntella are examples of P2P.

However, each architecture has its limitations. CDN costs a great deal to deploy and maintain surrogate servers, although its cost may be less than network-based contents distribution. Current solutions to cut cost usually compromise the service quality provided. In P2P, each peer offers very low out-bound bandwidth compared to edge servers in CDN. It will take a long time to distribute contents globally. Therefore, P2P needs a sufficient number of supplying peers. Hybrid architecture that integrates CDN and P2P will be a prospective solution for above-mentioned problems, because it compensate the defect of them with each others [3].

We have been investigating what mechanisms are required for CDN, P2P, and hybrid architecture. On CDN, the following two mechanisms are most important. They are 1) request routing: redirecting client requests to appropriate edge servers, and 2) content replication: deciding a strategy of distributing contents in order to deal with user's request efficiently. For request routing, 1) a method specifying which surrogate servers contain the requested contents and 2) a method specifying which server is most suitable in these servers must be considered. The former is called content discovery and the latter is called optimal server selection. For the efficient content replication, content discovery and optimal server selection are also required for P2P. In this paper, we focus on a method for optimal server selection.

3. OPTIMAL SERVER SELECTION

3.1. Metrics

For optimal server section, there are some metrics to be used. They are fallen into 1) network metrics and 2) host metrics. Network metrics are as follows.

1. IP path length: the number of routers through packet traversal
2. AS path length: the number of Autonomous System through packet traversal
3. location information: geographic metric, such as geographic distance between hosts
4. Round trip time (RTT)

RTT can be the most useful metric for optimal server selection [4]. Therefore, we adopt RTT as network metrics. The host's load factors, such as CPU usage and HDD throughput, are examples of host metrics. In this paper, we focus on network metrics, namely RTT only for simplicity. The combination of RTT and other network metrics, network metrics and host metrics is our future work.

3.2. RTT prediction method

It takes much cost to measure all RTT, since it needs to send probing packets to a whole set of hosts. Therefore, we do not measure all RTT. Instead, we use RTT prediction method for optimal server selection. This approach is to predict RTT using available prior information by some scalable ways without sending many probing packets from each host.

As a example of RTT prediction method, coordinate-based approach called "Global Network Positioning (GNP)" has been proposed [5]. This approach models the Internet as a geometric space and computes geometric coordinates to characterize the position of hosts on the Internet. The procedure of GNP is comprised of 1) Landmark procedure and 2) Host procedure. Landmark procedure is completed before Host procedure is started. Landmark procedure is as follows:

1. Some hosts are prepared to be used as the cardinal points for mapping the Internet to a geometric space. These hosts are called Landmarks.
2. Each RTT among Landmarks is measured.
3. Coordinates of each Landmark on geometric space S and a distance function $f_s(x, y)$, where x and y are coordinates on S, is determined by the measured RTTs. The dinstance function provides a distance between two coordinate points on S.

4. An identifier of S and $f_s(x, y)$ are distributed to all hosts participating in the session.

Host procedure is as follows:

1. Each of ordinary hosts sends a packet to all Landmarks, and a set of RTT is obtained.

2. Coordinates of each ordinary host on S is determined based on each RTT between itself and Landmark.

RTT between ordinary hosts is predicted by using coordinates of each host and $f_s(x, y)$. Some simulation results showed that GNP could predict RTT accurately [5]. This approach can be used as an optimal server selection method. However, it has some limitations on computational cost and robustness. The reasons are as follows:

1. It requires high computational cost to determine the coordinates.

2. The service will be stopped when Landmark encounters system down.

3.3. Proposed method

To avoid high computational cost, we focus on the fact that determination of the most optimal server in terms of RTT should be based on their relation rather than accuracy. Our proposal is based on this idea.

The procedure of proposed method is as follows:

1. Some Landmarks are prepared.

2. Each of hosts sends a packet to all Landmarks and a set of RTT is measured. This set of RTT is called location vector (Fig.1).

3. Each of hosts attaches the location vector to the information, which describes holding contents.

4. When a requesting host find that multiple hosts are holding desired contents, it compares own location vector with theirs. This comparison is done by Eq.(1), where A and B represent location vector of hosts to be compared.

$$\alpha = \arccos\left(\frac{A \cdot B}{|A||B|}\right) \quad (1)$$

The host having the smallest α is selected as a optimal target.

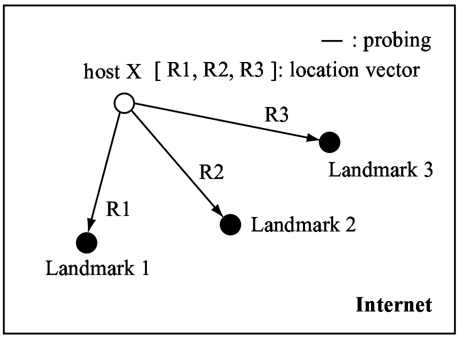

Figure 1: Definition of location vector

This is because Eq.(1) represents correlation coefficient between A and B. Consequently, smaller value of Eq.(1) indicates the similarity of location vector. This implys that a client locates close to the optimal host.

A service provided by GNP will be stopped when Landmark encounters system down. In our proposed method, the location vector comparision can be completed by eliminating failed Landmarks. For example, we assume that location vector A is [Ra1, Ra2, Ra3, Ra4, Ra5, Ra6] and location vector B is [Rb1, Rb2, Rb3, Rb4, Rb5, Rb6]. If the element of location vector B, Rb4, is not determined because of a breakdown of Landmark 4, the comparison is done with eliminating Ra4 and Rb4. This makes the accuracy of optimal server selection lower. However, the service can be provided in succession.

4. SIMULATION

4.1. Condition

Simulation is carried out to confirm validity of the proposed method using the Network Simulator - ns-2 [6]. Fig.2 shows the simulation model. Many hosts constructs overlay-network for contents distribution. Landmarks are picked up randomly from all hosts in a simulation topology. Each host holds some contents. We assume that each host knows the deployment of all contents.

Evaluation items are as follows.

Download time: Time which it takes to download all contents by a host

Traffic: The amount of traffic on the network

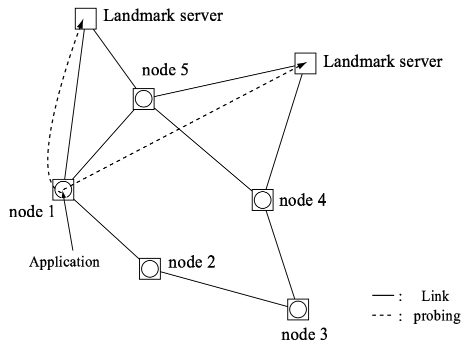

Figure 2: Simulation model

"Download time" represents an evaluation of users' side and "Traffic" represents an evaluation of network provider's side.

These evaluations are provided in case that 1) an optimal host is selected randomly, 2) an optimal host is selected by probing all target hosts, and 3) an optimal host is selected by our proposed method.

The parameters used in the simulation are depicted in Table 1. Network topology is generated by BRITE [8].

We simulate 15 configurations by changing the position of Landmarks and the deployment of all contents.

Table 1: Simulation parameters

Parameter	Value
The number of hosts	1000
The number of Landmark	10
The number of contents	100
The size of each content	10Mbyte
The number of hosts holding each content	5
Bandwidth between each of hosts	1-11Mbps uniform
Propagation delay between each of hosts	1ms
Network topology	waxman [7]

We set propagation delay between each of hosts shown in Table 1. In ns-2, bandwidth and delay must be set on each link. The total delay is expressed the following Eq.(2).

$$\begin{aligned} total\ delay &= propagation\ delay \\ &\quad + \frac{packet\ size}{bandwidth} \\ &\quad + queue\ waiting\ time \end{aligned} \tag{2}$$

The Eq.(2) has two variables, propagation delay and bandwidth. If propagation delay is much larger than (packet size/bandwidth), each host will determine the optimal host based on propagation delay only. However, the optimal host must be superior in respect to bandwidth. RTT is most useful metric for optimal server selection [4]. This means that the path between two hosts which has lower RTT might have higher bandwidth. Therefore, we set propagation delay between each of hosts constant. However, this assumption may not be correct in some cases. It is our future work to set propagation delay and bandwidth properly. This is also related to the problem about how to design the real Internet [9].

4.2. Numerical results

Fig.3 and Fig.4 shows the result of "Download time" and "Traffic" respectively.

Table 2 and Table 3 shows the average value of "Download time" and "Traffic" respectively.

Table 2: The average value of "Download time"

Random	Best	Proposed
3396	2533	3196

[sec]

Table 3: The average value of "Traffic"

Random	Best	Proposed
4386	3259	4077

[Mbyte]

From Table 2 and 3, compared with the random selection, our proposed method gives faster "Download time" and fewer "Traffic". These results shows validity of our proposed method. However, our proposed method is sometimes worse than the random

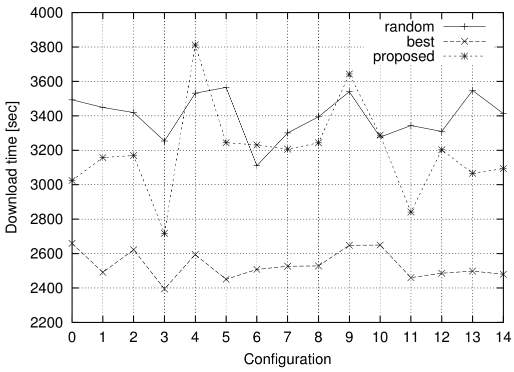

Figure 3: The result of “Download time”

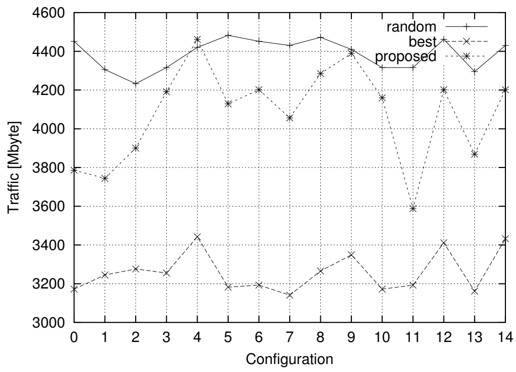

Figure 4: The result of “Traffic”

selection from Fig.3. We consider that such result was caused by a simulation network topology. In flat topology, such as waxman, all hosts may have similar location vector. Therefore, the possibility to select a wrong host may be high.

On the other hand, Transit-Stub model [10] has a hierarchical topology. In a hierarchical topology, many hosts tend to have different location vector compared with waxman. The Internet is considered to have a hierarchical stucture, and it is more similar to Transit-Stub model than waxman. Therefore, we may achieve better result by using Transit-Stub model for the Internet.

5. CONCLUSION

In this paper, we proposed the optimal server selection method based on RTT prediction. It can be performed without probing all hosts. This can avoid high network cost. Though a RTT prediction method is proposed in the previous work, our proposed method improves it pertaining to high computational cost and robustness. We confirmed the validity of the proposed method by network simulation. The contents distribution architecture using this proposed method adapts to scalable architecture.

6. REFERENCES

[1] C. Diot, B. N. Levine, B. Lyles, H. Kassem, and D. Balensiefen, “Deployment issues for the IP Multicast Service and Architecture,” IEEE Network magazine, Jan. 2000.

[2] I.Lazar and W. Terrill, “Exploring Content Delivery Networking,” IT Professional vol.3 No.4, pp.47-49, 2002.

[3] D. Xu, H. Chai, C. Rosenberg, and S. Kulkarni, “Analysis of a Hybrid Architecture for Cost-Effective Streaming Media Distribution,” in Proc. of SPIE/ACM Conference on Multimedia Computing and Networking (MMCN 2003), Jan. 2003.

[4] B. Huffaker, M. Fomenkov, D. J. Plummer, D. Moore and k claffy, “Distance Metrics in the Internet,” IEEE International Telecommunication Symposium (ITS2002), Sep. 2002.

[5] T. S. E. Ng and H. Zhang, “Predicting Internet Network Distance with Coordinates-Based Approaches,” Proceedings of IEEE INFOCOM’02, Jun. 2002.

[6] “The Network Simulator - ns-2,” http://www.isi.edu/nsnam/ns/

[7] B. Waxman, “Routing of Multipoint Connections,” IEEE J. Select. Areas Commun., Dec. 1988.

[8] “BRITE,” http://www.cs.bu.edu/brite/

[9] H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, “Network Topology Generators: Degree-Based vs. Structural,” ACM SIGCOMM 2002, Aug. 2002.

[10] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, “How to Model an Internetwork,”, IEEE INFOCOM ’96, 1996.