A Method for Overlay Network Latency Estimation from Previous Observation

Weihua Sun
Nara Institute of
Science and Technology
Ikoma, Japan 630–0192
sunweihua@is.naist.jp

Naoki Shibata
Nara Institute of
Science and Technology
Ikoma, Japan 630–0192
n-sibata@is.naist.jp

Keiichi Yasumoto
Nara Institute of
Science and Technology
Ikoma, Japan 630–0192
yasumoto@is.naist.jp

Masaaki Mori
Shiga University
1-1-1 Banba, Hikone,
Shiga, Japan 522–8522
mori@biwako.shiga-u.ac.jp

Abstract*—Estimation of the qualities of overlay links is useful for optimizing overlay networks on the Internet. Existing estimation methods requires sending large quantities of probe packets between two nodes, and the software for measurements have to be executed at both of the end nodes. Accurate measurements require many probe packets to be sent, and other communication can be disrupted by significantly increased network traffic. In this paper, we propose a link quality estimation method based on supervised learning from the previous observation of other similar links. Our method does not need to exchange probe packets, estimation can be quickly made to know qualities of many overlay links without wasting bandwidth and processing time on many nodes. We conducted evaluation of our method on PlanetLab, and our method showed better performance on path latency estimation than estimating results from geographical distance between the two end nodes. *(Abstract)

Keywords-link quality; PlanetLab; Estimation; Learning Algorithm.

I. INTRODUCTION

In order to construct an efficient Peer-to-peer (P2P) overlay network, we need to know the link quality of overlay links, and several methods for estimating link qualities such as available bandwidth, packet-loss rate and latency between peers on the Internet have been proposed. This kind of estimation methods are also useful in client-server applications.

Existing estimation methods requires sending large quantities of probe packets between two nodes. Pathload[1] assumes that a periodic packet stream shows an increasing trend when the stream's transmission rate is higher than the available bandwidth, and it measures the available bandwidth between two nodes. Abing [2] estimates the capacity of a path (bottleneck bandwidth) based on the observed the dispersion experienced by two back-to-back packets. These methods require measurement software to be executed at both of the end nodes. Since accurate measurements require many probe packets to be sent, other communication can be disrupted by significantly increased network traffic. Moreover, in order to make more accurate measurement of link qualities, more probe packets need to be sent into the network. If we could estimate link qualities between each pair of nodes on the Internet. Estimation of link qualities is useful for optimizing overlay networks on the Internet. However, the number of overlay links is the square of the number of peers, it is difficult to estimate all the link qualities using the tools discussed above, since the packets for estimation between a pair of nodes can disrupt measurements between other nodes. In general, the network delay is considered to increase as the geographical distance or the number of routers in the route increases. However, due to the disproportionate data flow, large delay occurs at some specified routers. Also, there are detour of physical communication links by geographical or political reasons. Because of these reasons, link qualities are considered to be attributed to the geographical positions of the two end nodes, rather than just the geographical distance between the nodes. We also need to consider the varying conditions of congestion, and that the situation can suddenly change. However our observation tells that most of the links usually have relatively stable available bandwidths and delays. Since most people use the Internet in the daytime, there should be constant periodical changes of link qualities. Thus, we assume that we can estimate the degree of congestion of an overlay link from periodical observation of the link qualities in the past.

We first discuss these assumptions by conducting experiments on observing link qualities of PlanetLab nodes, and show that the assumptions stated above are probable. Then, we explain our proposed method based on supervised learning for estimation of overlay link qualities from qualities observed in the past. Our method takes account of the geographical locations of end nodes to estimate the link qualities. Our experiments on PlanetLab showed that our method has good performance on path latency estimation. The estimation based on just geographical distance showed large error, especially when the distance is shorter than 2000 km. The proposed method achieved high estimation accuracy in that range. We have shown a part of our results in a work-in-progress paper [3], and we show detailed experimental results and discussion in this paper.

In Section II, we provide an overview of related works, while in Section. III we present a preliminary discussion on how accurately we are able to estimate link qualities from those previously observed. We propose a method for estimating link qualities based on a supervised learning algorithm in Section IV and present the results of experiments on PlanetLab to demonstrate the accuracy of the proposed method in Section

V. Finally, our conclusions are given in Section VI.

II. Related Research and Contribution

Previously, in the field of wide area networks, many approaches were proposed to measure and estimate the delay and bandwidth between end nodes. Accurate estimation of the available bandwidth is important for throughput optimization between end nodes, overlay network routing, peer-to-peer file distribution, traffic engineering, and capacity planning. In this section, we discuss the measurement and prediction methods with respect to the available path bandwidth between end nodes.

A. Bandwidth Measurement Method

There are three different metrics for path bandwidth between end nodes: (1) capacity (maximum bandwidth), (2) available bandwidth (maximum unused bandwidth), (3) TCP throughput/bulk transfer capacity (maximum achievable bandwidth). The existing four measurement methodologies are:

- **VPS (Variable Packet Size probing)** is a method to estimate link capacity by measuring the round-trip time; that is, calculating the serialization delay of various sized packets sent from a sender node to a receiver node.
- **PPTD (Packet Pair/Train Dispersion)** is a method for measuring the capacity of the path between end nodes. Letting a sender node continuously send uniform sized packet pairs or trains to a receiver node, this approach calculates the maximum link serialization delay in the path to estimate the minimum link capacity (bottleneck) by measuring the dispersion of the received packet times.
- **SLoPS (Self-Loading Periodic Streams)** is a method for measuring available bandwidth. While a sender node continuously sends uniform sized packets to a receiver node with transmission rate R, SLoPS observes the variation in delay for each packet at the destination node, and measures whether R is greater than A. By adjusting the transmission rate R, SLoPS estimates the available bandwidth A.
- **TOPP (Trains of Packet Pairs)** measures capacity and available bandwidth by transmitting data at a particular transmission rate for a specified number of packet pairs. Unlike SLoPS, TOPP estimates the available bandwidth by increasing the transmission rate linearly and observing the arrival delay.

Other tools that have been proposed and implemented are **Pathchar, Clink**, and **Pchar** for measuring link capacity, **Brpobe, Nettimer, Pathrate** and **Sprobe** for measuring path capacity, **Cprobe, Pathload, IGI**, and **pathChirp** for measuring available bandwidth, and **Treno, Cap, TTCP, NetPerf, Iperf** for measuring TCP throughput. As reported in [4], **Pathload** and **pathChirp** showed better performance than **Abing, Spruce**, and **Iperf** on a high-speed network testbed.

Most of the above tools focus on measuring the average available bandwidth, but do not consider bandwidth variation. Therefore, the authors in [6] proposed a method to measure bandwidth variation. Moreover, with the goal of estimating the bandwidth without causing excessive traffic, a method was proposed in [5] to estimate capacity and available bandwidth without congesting the minimum capacity link in the path.

Most of the existing bandwidth measurement methods and tools work by exchanging probe packets between sender and receiver nodes. Although these methods are useful for accurate bandwidth measurement, they generate traffic while measuring bandwidth. SLoPS and TOPP, in particular, cause temporary congestion of the minimum capacity link. Consequently, in a large scale P2P network with millions of nodes, these methods may cause serious deterioration in the network performance.

B. Bandwidth/Latency Prediction Method

Various network traffic prediction models have been proposed. In networks, similar traffic patterns with long time intervals are said to be self-similar, while those patterns with short time intervals are called multi-fractal. In [8], a method was proposed to predict network traffic at several time steps in advance, based on past measured traffic information. Moreover, the authors in [7] improved the method in [8], by proposing a new ARIMA/GARCH model that predicts network traffic with higher accuracy. In this model, self-similarity and multi-fractals can be predicted by utilizing short-range and long-range dependencies. Through comparison experiments with real network traffic, the authors showed that network traffic can be predicted with reasonable accuracy.

These models aim to predict future traffic from previous detailed measurements. Moreover, the models can be used to predict the available bandwidth and latency by separately measuring the capacity of the path between the end nodes. Similar to the above methods, the method in [9] accurately estimates the latency of the path between end nodes based on traffic measurement. However, because detailed measurements are needed in advance, the models are not suitable for estimating bandwidth/latency at low cost owing to the additional traffic generated.

C. Contribution

The traffic prediction model makes use of the self-similarity and multi-fractal properties of traffic. By applying these characteristics to the different nature of similar paths, link qualities (including end to end delay, available bandwidth, and so on) can be predicted using fewer a priori measurement results.

In this paper, by considering the similarity of paths, we propose an overlay link quality prediction method, which assumes that similar paths have similar characteristics. To the best of the authors' knowledge, there is no other prediction method that, like ours, does not require much bandwidth. Moreover, we have implemented the proposed method in PlanetLab and evaluated the performance thereof.

III. Preliminary Experiments And Observation

In this section, we first describe the results of two preliminary experiments. In the first experiment, we observed the fluctuations in link quality over time, while in the second, we investigated the relation between route (overlay link) similarity

 ISBN: 978-1-61208-245-5

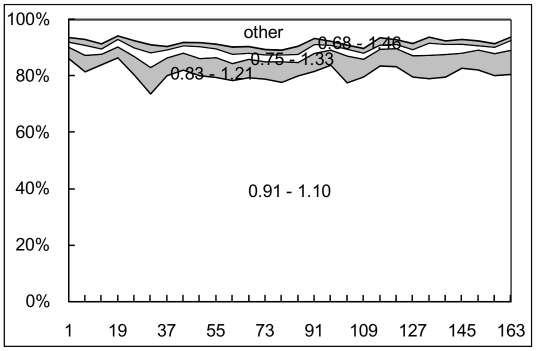


Fig. 1. Observed fluctuation of latency (X axis = time)

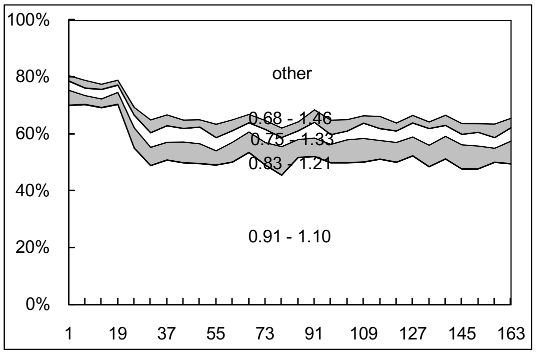


Fig. 2. Observed fluctuation of available BW (X axis = time)

and the difference in link qualities. The amount of traffic on the Internet changes continuously, influenced both by the day of the week and the season. We observed the actual fluctuations in link quality on PlanetLab. In the subsequent subsections, we describe the configuration of the experiments, the definition of route similarity, and the results of these experiments.

Observation of PlanetLab: We observed the fluctuations in available bandwidth and latency between nodes in PlanetLab over 7 days starting on 20th January 2011. We created 500 random pairs among the nodes in PlanetLab and measured the available bandwidth and latency using Pathload and ping every hour. About 63000 valid data records were obtained.

Fig. 1 shows a stacked bar graph of the observed latency at each time divided by the latency observed at the beginning. The bottom series indicates the ratio of routes where the observed latency divided by the latency observed at the beginning was between 0.91 and 1.1. The second series indicates the ratio of overlay links with latencies between 0.83 and 1.21 times. Fig. 2 shows the results for bandwidth. From Fig. 1 it is clear that for 80% of the routes, the fluctuation in latency was less than 10%, and this ratio did not change for the whole week. Fig. 2 shows that for 70% of the routes, the fluctuation in bandwidth was less than 10% for 20 hours from the beginning of the experiment. It also shows that for half the routes, bandwidth fluctuation was less than 10% for the week. We did not observe daily periodic fluctuations in bandwidth or latency.

A. Relation between Route Similarity and the Difference in Link Quality

It would be convenient if we could estimate the link quality of an unknown overlay link on which no link quality observations have been made. To realize such a method, we first define similarity between two overlay links based on geographical distance. There are free databases from which we can find the geographical location of nodes from their IP addresses, and thus it is easy to locate the geographical position of nodes on the Internet. We also show the measurement results for link similarity and the difference in link quality.

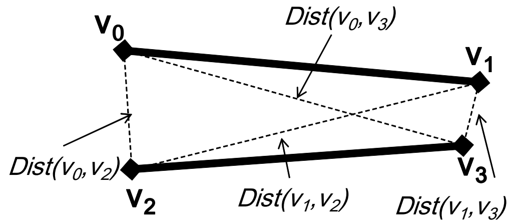
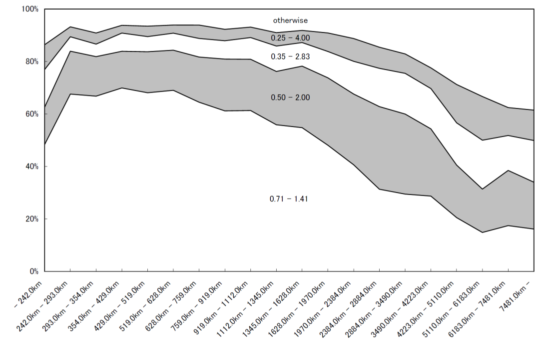
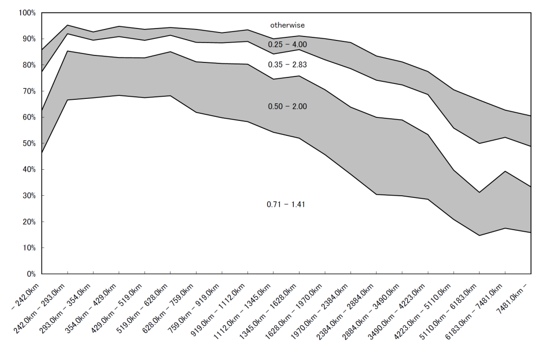


Fig. 3. Similarity Definition



(a) After 1 hour



(b) After 6 days

Fig. 4. Estimated latency by Proposed method, X axis = distance(Km)

Route Similarity: As shown in Fig. 3, the route similarity $geo(v_0, v_1, v_2, v_3)$ between two routes is defined as the minimum value between $dist(v_0, v_2) + dist(v_1, v_3)$ and $dist(v_0, v_3) + dist(v_1, v_2)$, where $dist(v_0, v_1)$ denotes the geographical distance between v_0 and v_1.

Measurements on PlanetLab: We created 500 random pairs of nodes on PlanetLab, and investigated the relation between similarity as defined above and the observed latency. Fig. 4(a) shows the relation between link similarity and latency fluctuation one hour after the first measurement was made, while Fig. 4(b) shows the results obtained six days after the first measurement. We can see that these two graphs are almost identical, and that there is almost no change in the fluctuation over time. We can also see that the amount of fluctuation decreases with more similar routes. With the sum of the distance less than 600 km, the fluctuation is within 50% to 200% for 80% of the routes.

We also performed similar experiments on bandwidth, but did not observe any relation between route similarity and fluctuation. This seems to be due to the fact that the available bandwidth is usually limited by the bandwidth for the last hop rather than that for the entire backbone. However, we are still investigating finding an appropriate similarity definition for estimating the correct bandwidth.

IV. Overlay Link Quality Estimation Method

In this section, we propose an overlay link quality estimation method based on the results of the preliminary experiments in Section III. In the proposed method, (1) a centralized server periodically collects, from various peers in the P2P network, quality information of overlay links they have observed, and (2) the quality of a given overlay link is estimated from the information of previously observed overlay links based on the weighted k-nearest neighbor ($WKNN$) algorithm, which is one of the supervised learning techniques.

A. Preliminaries

1) Weighted k-nearest neighbor algorithm: The WKNN method uses training samples expressed as pairs of an object and a real number, and learns a function that maps an arbitrary object to a real number. In our proposed method, the object and real number correspond to an overlay link and latency, respectively.

To use the WKNN algorithm, the following two functions must be given: (1) a function to calculate the distance between two objects; and (2) a function that assigns a weight to each object.

In the WKNN algorithm, learning is carried out using all training samples (the training set) stored in memory. When estimating a real number for an input object, WKNN selects the k samples in the training set geographically closest to the input object, and estimates a real number by calculating the weighted average of the k samples with their weights.

2) Assumptions, estimation target, and algorithm outline: We aim to apply the proposed method to estimate overlay link quality in a P2P application such as video streaming. We assume that the P2P application consists of a central server and many peers (users). In the application, each peer observes the quality of the overlay links directly connected to other peers and periodically sends the observed information to the server. In this study, we have designed the learning algorithm as a centralized one, but it could easily be implemented as a distributed algorithm using, e.g., a distributed hash table.

The proposed algorithm is executed on the server and estimates the quality of a given overlay link by applying the WKNN method to the previously observed quality information collected by the server. As described in Section III-A, we could not find any correlation between link similarity and the observed available bandwidth. Thus, we focus mainly on overlay link latency as the quality estimation target in this study.

Each peer sends the server a query to estimate the quality of the specified overlay links. When the server receives a query, it estimates the quality of the given links based on the proposed algorithm and sends the estimated result back to the peer.

The server carries out learning and estimation. In the WKNN algorithm, the server performs learning using all training samples stored in its memory. As time progresses, the number of training samples increases and more memory space is required. To limit the required memory size, when the number of training samples exceeds a predefined threshold, the oldest samples are deleted from memory.

The size of a message that a peer exchanges with the server (to upload the observed link quality, send a query for link quality estimation, or receive the estimation result) is at most 200 bytes since it contains only an overlay link together with the associated quality.

The server has a table that maps IP addresses to geographic coordinates as explained in Section III-A.

B. Learning algorithm

The proposed algorithm consists of two phases: (i) a learning phase, and (ii) an estimation phase. We describe these phases in detail below.

1) Learning phase: We assume that each peer participating in a target application communicates frequently with other peers participating in the same application, e.g., to realize video P2P streaming.

In the proposed algorithm, each peer performs the following steps:

- When the peer communicates with other peers, it measures the quality of the overlay links to those peers.
- The peer periodically sends the quality of overlay links observed during the current period to the server. The message contains the IP addresses of both ends of each overlay link and the measured latency.

When the server receives the observed quality of an overlay link from a peer, it stores the data –that is, the IP addresses of the end nodes of the overlay link and the latency, in its memory.

When the amount of data exceeds a predefined threshold, the server removes the oldest data from its memory.

2) Estimation phase: When a peer wishes to know the quality of an overlay link, it sends the server a query specifying the IP addresses of the end points of the link. When the server receives the query, it estimates the quality of the specified link as follows:

- The server selects the k closest training samples from the training set.
- It calculates the weight of each selected sample as explained in Section IV-B4.
- It calculates the weighted average of the latency of the selected k samples.
- It sends the calculated result to the peer that originally sent the query.

 ISBN: 978-1-61208-245-5

3) Estimation example: Let us suppose that peer n_0 has sent the server a query regarding the overlay link between itself and peer n_1. When the server receives the query, it selects the k training samples closest to the overlay link between n_0 and n_1 based on the distance function defined in Section III-A. Let us suppose that $k = 2$ and overlay links r_1 and r_2 have been selected. Then, the server calculates the weights of r_1 and r_2 according to the method in Section IV-B4. Let the weights for r_1 and r_2 be 1 and 2, respectively. Let us also suppose that the previously observed latencies of r_1 and r_2 are 3 and 4, respectively. Finally, the server obtains the value $(1 \times 3 + 2 \times 4)/(3 + 4) = 1.57$ as the weighted average and sends this value as a reply to peer n_0.

4) Weight function: In Section III-A, we defined the similarity between two overlay links observed at the same time. In general, this similarity should be defined between two links observed at different times. However, as explained in Section III-A, the variation in latency with time is rather small. Thus, we use the similarity function defined for two links observed at the same time in the proposed algorithm.

According to the measurement results presented in Section III, more than 80% of overlay links experience a latency variation between 0.71 and 1.41 times the initial measured latency. Thus, we define the weight function as follows:

$$Weight(u_s, u_d, v_s, v_d)0.7 - \frac{0.3}{5000} \cdot geo(u_s, u_d, v_s, v_d) \quad (1)$$

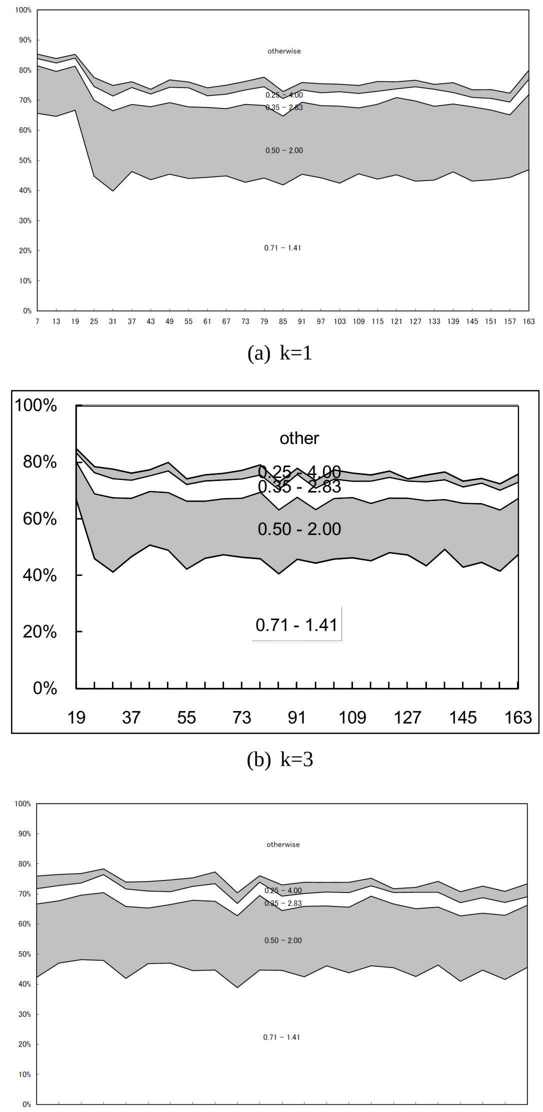
where (u_s, u_d) and (v_s, v_d) are the geographic coordinates of the target overlay link and the training sample, respectively.

V. EVALUATION

In this section, we evaluate the estimation accuracy of the proposed method. According to the underlying principle of the proposed method, the estimation accuracy depends on the distance and time from the measured path. The greater the difference in time or distance is, the worse is the estimation accuracy. With respect to available bandwidth, we investigated the relationship over time and estimation accuracy. With respect to latency, we investigated the relationship over the distance between paths and estimation accuracy.

A. Evaluation of Available Bandwidth

As described above, despite the paths being similar, no correlation with available bandwidth was observed. In this experiment, using the k measured results of both cases of one measurement per day and one measurement per hour on a certain path, we investigated the estimation accuracy when varying k and the elapsed time from the last measurement. The results are shown in Figs. 5(a)-5(c). According to these figures, the observed estimation accuracy corresponds to the results of the preliminary experiments. However, the estimation accuracy did not improve even when increasing k.



(a) k=1

(b) k=3

(c) k=5

Fig. 5. Estimated bandwidth, X axis = time

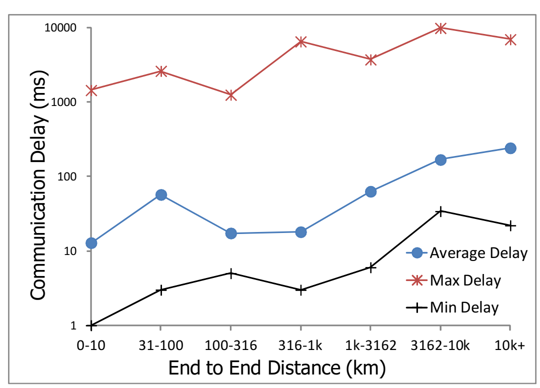


Fig. 6. The average, maximum and minimum path delay

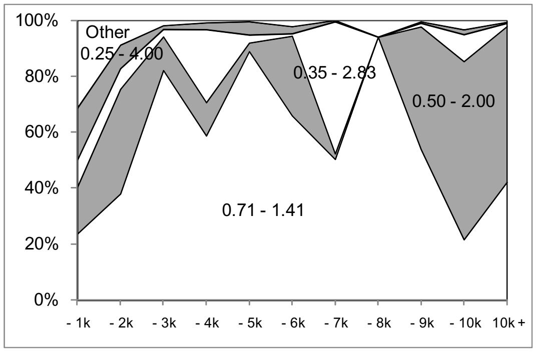


Fig. 7. Accuracy of Estimated delay based on path length (X axis = distance (KM))

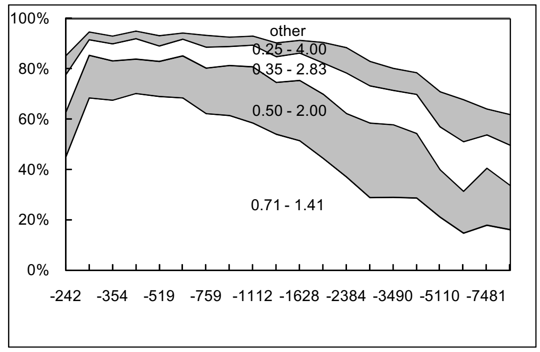
1) Estimation based on link distance: For comparison with the proposed method, we used a delay estimation method based on link distance. Fig. 6 shows the relationship between link length and delay. According to this result, the average delay of a path increases roughly in proportion to the distance. However, the maximum and minimum delays do not follow this trend. The delay calculated from the average delay is 0.019 ms/km. The results of applying this value to the delay estimation method are shown in Fig. 7. Obviously, the estimation accuracy is low when the link distance is less than 2000 km.

2) The proposed method: In this experiment, we investigated the accuracy of measuring path latency based on the measured latency results of k different paths six days previously. We investigated the estimation accuracy for a number of distance functions by varying k. The results are shown in Figs. 8(a) – 8(c).

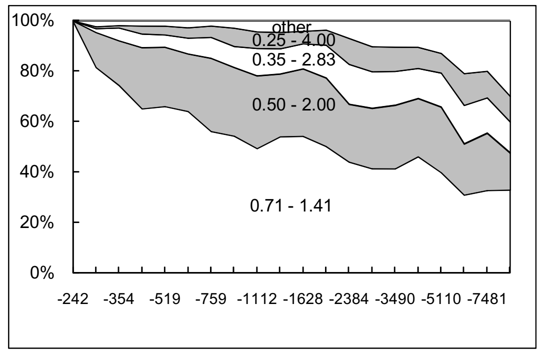
According to these figures, accurate estimation was observed. The estimation accuracy improved as k increased. In particular, we confirmed that the estimation accuracy (0.71–1.41 and 0.5–2.0) is very high for medium and short distances, respectively.

VI. Conclusion

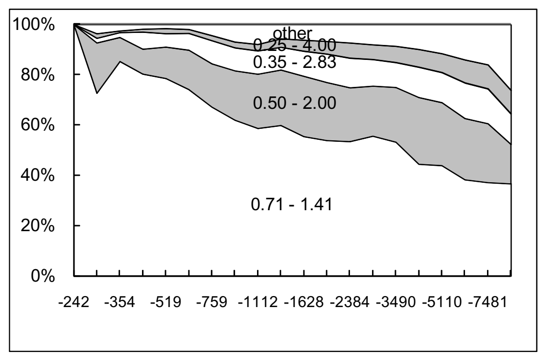
We proposed a learning-based overlay link quality estimation method that uses the quality observed for other links in the past. With respect to latency, by defining geographical similarity between overlay links, the proposed method achieves good estimation accuracy. With respect to bandwidth, we found that there is no correlation between overlay links with close geographical similarity. In the future, we intend devising a new similarity metric to accurately estimate overlay link bandwidth taking into account domain type, connecting ISP, and so on.



(a) k=1



(b) k=2



(c) k=3

Fig. 8. Estimated latency by Proposed method, X axis = distance(Km)

References

[1] M. Jain and C. Dovrolis, "End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput," in IEEE/ACM Transactions on Networking, vol. 11, Issue 4, pp. 537–549, 2003.

[2] J. Navratil and R. L. Cottrell, "ABwE:A practical approach to available bandwidth estimation," in Proc. of Passive and Active Measurement Workshop (PAM'03), in-cdrom, 2003.

[3] W. Sun, N. Shibata, K. Yasumoto, and M. Mori, "Estimation of overlay link quality from previously observed link qualities," in Proc. of The 10th Annual IEEE Consumer Communications Networking Conference (CCNC'13), pp.582-585, Jan. 2013.

[4] A. Shriram, M. Murray, Y. Hyun, N. Brownlee, A. Broido, M. Fomenkov, and K. Claffy, "Comparison of public end-to-end bandwidth estimation tools on high-speed links," in Proc. of of Passive and Active Measurement Workshop (PAM'05), pp.306–320, 2005.

[5] S. Kang, X. Liu, M. Dai, D. L, and D. Loguinov, "Packet-pair bandwidth estimation: stochastic analysis of a single congested node," in Proc. of 20th IEEE International Conference on Network Protocols (ICNP'04), pp.316–325, 2004.

[6] M. Jain and C. Dovrolis, "End-to-end estimation of the available bandwidth variation range," in Proc. of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems, pp.265-276, 2005.

[7] B. Zhou, D. He, Z. Sun, and W. H. Ng, "Network traffic modeling and prediction with ARIMA/GARCH," In Proc. of 3rd international working conference: performance modelling and evaluation of heterogeneous networks (HET-NETs'05), in-cdrom, 2005.

[8] A. Sang and S. Li, "A predictability analysis of network traffic," in Proc. of IEEE INFOCOM 2000, pp.342–351, 2000.

[9] N. Hariri, B. Hariri, and S. Shirmohammadi, "A distributed measurement scheme for Internet latency estimation," in Proc. of IEEE Trans. on Instrumentation and Measurement, 60 (5): pp.1594–1603, 2011

 ISBN: 978-1-61208-245-5