IEICE TRANS. INF. & SYST., VOL.E95–D, NO.12 DECEMBER 2012

PAPER Special Section on Parallel and Distributed Computing and Networking

Analytical Modeling of Network Throughput Prediction on the Internet

Chunghan LEE[†a)], **Hirotake ABE**[††b)], **Toshio HIROTSU**[†††c)], ***Nonmembers, and*** **Kyoji UMEMURA**[††††d)], ***Member***

SUMMARY Predicting network throughput is important for network-aware applications. Network throughput depends on a number of factors, and many throughput prediction methods have been proposed. However, many of these methods are suffering from the fact that a distribution of traffic fluctuation is unclear and the scale and the bandwidth of networks are rapidly increasing. Furthermore, virtual machines are used as platforms in many network research and services fields, and they can affect network measurement. A prediction method that uses pairs of differently sized connections has been proposed. This method, which we call *connection pair*, features a small probe transfer using the TCP that can be used to predict the throughput of a large data transfer. We focus on measurements, analyses, and modeling for precise prediction results. We first clarified that the actual throughput for the connection pair is non-linearly and monotonically changed with noise. Second, we built a previously proposed predictor using the same training data sets as for our proposed method, and it was unsuitable for considering the above characteristics. We propose a throughput prediction method based on the connection pair that uses ν-support vector regression and the polynomial kernel to deal with prediction models represented as a non-linear and continuous monotonic function. The prediction results of our method compared to those of the previous predictor are more accurate. Moreover, under an unstable network state, the drop in accuracy is also smaller than that of the previous predictor.

key words: *network measurement, virtualization, PlanetLab, Support Vector Regression (SVR)*

1. Introduction

Network throughput prediction is a challenging issue in the network research field. A predicted network throughput can be used to enhance grid task scheduling, path selection on multiple paths, and the efficiency of data transfer. Various throughput prediction methods have been proposed, but many of them are suffering from the fact that the scale and bandwidth of networks are rapidly increasing. Additionally, the current traffic consists of mice and elephants [1], [2]. A spike that corresponds to large and abrupt throughput is occasionally caused by elephants. Such spikes are obstacles in determining a model of probability distribution for the prediction. A prediction method that uses pairs of different-sized connections was previously proposed by Wolski et al. [3]. This method, which we call *connection pair*, features a small probe transfer using the Transmission Control Protocol (TCP) that can be used to predict the throughput of a large data transfer.

Virtualization technology has been widely applied in many network research fields, and virtual machines have recently been used as platforms for grid computing [4], network testbeds [5], [6], and cloud services [7]. We should particularly consider the impact of virtualization because it can affect network measurement. PlanetLab [6] is a virtualized network testbed, and Linux-Vserver is used to virtualize resources on a node. In the testbed, a platform called a *sliver* is provided as a virtualized environment to users, and multiple slivers can be run simultaneously at each node. A set of these slivers participating in the same activity at different nodes is called a *slice*. Thus, PlanetLab consists of virtualized nodes on the Internet.

The aims of our research are to predict network throughput and to improve the predicted throughput results in comparison with those of an existing prediction method. The prediction results can be improved by considering historical measurements, and such prediction can be formulated as a regression problem. We first selected an appropriate probe size (256 KB) [8] through Spearman's rank correlation coefficient (ρ) before the prediction. Second, we built a previously proposed predictor [9] for comparison, and developed a throughput prediction method that uses ν-support vector regression (SVR) and the polynomial kernel to deal with prediction models represented as a non-linear and continuous monotonic function. Next, we compared the prediction results of our proposed method with those of the previously proposed predictor for the same data sets. Finally, we composed additional input data sets below the mean throughput of the probe transfer to evaluate the prediction results under an unstable network state.

In this paper, we present our throughput prediction method and show its improved prediction results. The contributions of this work are as follows.

- We find that the previous prediction method is unsuitable when actual throughput is non-linearly and monotonically changed with noise.
- We propose a network throughput prediction method with improved accuracy. The prediction results of the

Manuscript received January 6, 2012.
Manuscript revised April 5, 2012.

[†]The author is with the Dept. of Electronic and Information Eng., Toyohashi University of Tech., Toyohashi-shi, 441–8530 Japan.
[††]The author is with Cybermedia Center, Osaka University, Ibaraki-shi, 567–0047 Japan.
[†††]The author is with the Faculty of Computer and Information Sciences, Hosei University, Koganei-shi 184–0002 Japan.
[††††]The author is with the Dept. of Computer Science and Eng., Toyohashi University of Tech., Toyohashi-shi, 441–8530 Japan.
a) E-mail: lch@ss.cs.tut.ac.jp
b) E-mail: habe@cmc.osaka-u.ac.jp
c) E-mail: hirotsu@hosei.ac.jp
d) E-mail: umemura@tut.jp
DOI: 10.1587/transinf.E95.D.2870

proposed method are more accurate than the previous one for the same data sets.
- The proposed method is shown to be more robust than the previous one. In an evaluation under an unstable network state, the range of any drop in accuracy is smaller than that of the previous method.

The rest of this paper is organized as follows. First, we describe related work in Sect. 2 and explain our measurement methodology in Sect. 3. Second, we present the previously proposed method and our prediction method using the same data sets in Sects. 4 and 5. Third, we discuss the prediction results of these two methods in Sect. 6. Finally, we conclude by summarizing the main points in Sect. 7.

2. Related Work

Many throughput prediction methods have been based on historical data. He et al. [10] proposed a history-based (HB) prediction. It is based on the moving average and Holt-Winter models. Wolski et al. [3] empirically established the basic probe size as 64 KB for the Network Weather Service (NWS). They used the connection pair to predict the throughput of data transfer on NWS and focused on only the connection pair where the probe size was 64 KB and data size was 16 MB. However, they selected the size of the connection pair empirically and generated connection pairs in limited networks, so the probe size might be unsuitable for other networks. Moreover, Yousaf et al. [11] reported the requirement of a large-sized probe, but they did not select an appropriate probe size for the Internet. Vazhkudai et al. [12] proposed a linear regression model using a combination of the 64-KB probe and past measurements; their model uses the least squares method. However, there were less data transfers than probe transfers, meaning the data size was not determined precisely. Swany et al. [9] proposed a prediction method using the cumulative distribution function (CDF) of network throughput for probe and data transfers. The throughput for data transfer was predicted by using the CDF of a probe transfer. In particular, we build in this work the CDF predictor, and compare the prediction results of our method with those of the CDF predictor for the same data sets.

Support vector machines (SVMs) and support vector regression (SVR) have been used in various network research areas. Bermolen et al. [13] used an SVR for link load prediction. Beverly et al. [14] considered an SVM for predicting round-trip latency. Moreover, Feng et al. [15] proposed WLAN traffic prediction using an SVM. Mirza et al. [16] have proposed a throughput prediction method using an SVR that combines prior data transfers and measurements of network metrics, such as packet loss, queuing delay, and available bandwidth. However, our method depends only on measurements of the connection pair. Thus, our method uses bivariate data while their method uses multivariate data. A radial basis function (RBF) for the kernel trick is used to consider non-linear and multivariate regression. They used their laboratory testbed [16] for passive and active measurements of these network metrics. In evaluations, the prediction results with the passive measurements were more accurate than those with the active measurements because of the accurate network metrics for the passive measurements. However, the network metrics are normally undisclosed to users, and it is hard to estimate them on end nodes precisely. Next, they used the Resilient Overlay Networks (RON) testbed [17] to evaluate their method with active measurements on the Internet, but nodes that should have little or no other CPU or network load were restricted and the operating system was also limited to FreeBSD 4.7 for the active measurements of the network metrics. Thus, these limitations may be unsuitable for evaluations on the Internet.

3. Measurement Methodology

We present here an overview of our prediction method. We first selected an appropriate probe size through Spearman's rank correlation coefficient (ρ). However, an incorrect probe size was selected due to the impact of virtualization. After filtering the negative effects, a 256-KB probe was selected for the prediction model. We then developed a throughput prediction method that uses ν-SVR and the polynomial kernel. In this section, we describe how to select the appropriate probe size to gather training data sets, and we show our throughput measurement method. Finally, we present the characteristics of the training sets and input sets for the prediction methods.

3.1 Selection of Probe Size

To find the appropriate sizes for better prediction performance, we evaluated various combinations of probe and data sizes of connection pairs. The test transfers of connection pairs are generated every 5 minutes on PlanetLab. The details of the measurements and results are described in another paper [8]. For convenience, we summarize the main points of that paper below. We gathered a data set of approximately 15,000 connection pairs over 2 weeks, and selected the appropriate probe size through Spearman's rank correlation coefficient (ρ) before the prediction. Thus, we assumed only monotonicity between probe and data throughputs. For example, if probe throughput is decreased, data throughput will be decreased monotonically. A high ρ value implies that the probe will have a high predictability and that it can therefore be regarded as appropriate for prediction. The 32-KB probe had the highest ρ value (0.69) in our evaluation.

A packet spacing is an idle period between the reception of a packet and the sending of the next packet. It is approximately 0.000010 [s] over a non-virtualized environment. However, we found oversized packet spacings, which can result from CPU scheduling latency, even when no significant changes occur in well-known network metrics. These packet spacings are a major cause of throughput fluctuations in the best condition (Fig. 1), and they are un-

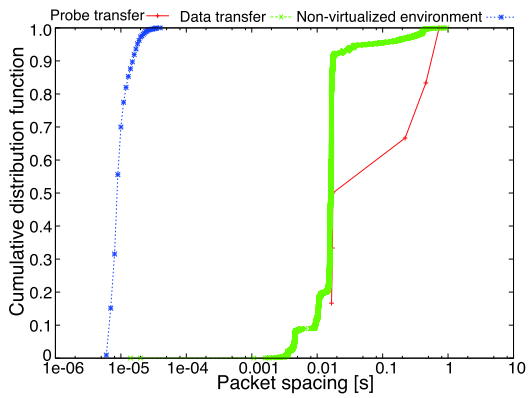


Fig. 1 CDF of packet spacing at anomalous case.

Table 1 Geographic location and mean RTT at node pairs.

Node name	Geographic location	Node pair (arrow is transfer direction)	Mean RTT [s]
α, β	Europe	$\alpha \leftarrow \beta$	0.0283
γ, δ	Europe	$\gamma \leftarrow \delta$	0.0477
ϵ, ζ	Europe	$\epsilon \leftarrow \zeta$	0.0511
η, θ	North America	$\eta \leftarrow \theta$	0.0370
κ, λ	North America	$\kappa \leftarrow \lambda$	0.0598
μ, ν	North America	$\mu \leftarrow \nu$	0.0392

usual anomalies in a virtualized network environment [18]. An anomalous case is that throughput instability occurs despite a stable network state, which can be observed through round-trip time (RTT), packet loss rate, and so on. Such anomalous cases are unnecessary for a precise prediction model, and we should review the actual throughput carefully. After filtering the anomalous cases, we selected a 256-KB probe (ρ = 0.67) [8], instead of the 32-KB probe (ρ = 0.31), as the appropriate probe size.

3.2 Training Data Sets

For training data sets, we empirically selected six pairs of nodes from PlanetLab nodes located in both North America and Europe, which we refer to as nodes (α, β), (γ, δ), (ϵ, ζ), (η, θ), (κ, λ), and (μ, ν). The geographic location and mean RTT using ping for all the pairs are shown in Table 1. We simultaneously generated connection pairs at the sender every 5 minutes. Each of that consists of two simultaneous TCP/IP streams in different sizes. Smaller one is called 'probe' and larger one is called 'data'. This time we used the 256-KB probe and the 16-MB data. If the measured size is smaller than expected or if the transfer time is more than 5 minutes, we judge at the receiver that the measurement has failed. Thus, network throughput was measured using the connection pair. The measurement methodology is shown

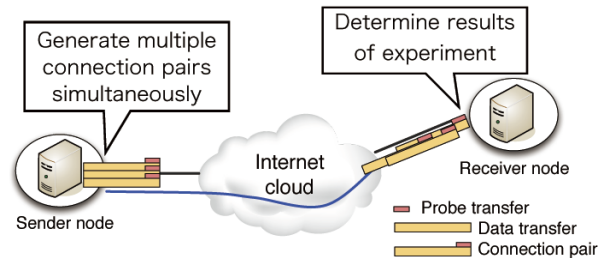


Fig. 2 Measurement methodology.

Table 2 Statistics of training data sets. (NT is network throughput).

Node pair	Min NT [KBps]	Mean NT [KBps]	Max NT [KBps]	Total counts
α, β	138.1	2008.0	2107.8	4705
γ, δ	323.7	1280.5	1290.7	3690
ϵ, ζ	0.2	1140.9	1168.0	4535
η, θ	304.3	1467.6	1616.3	4941
κ, λ	56.3	975.0	1032.1	4936
μ, ν	845.9	1335.4	1512.3	5340

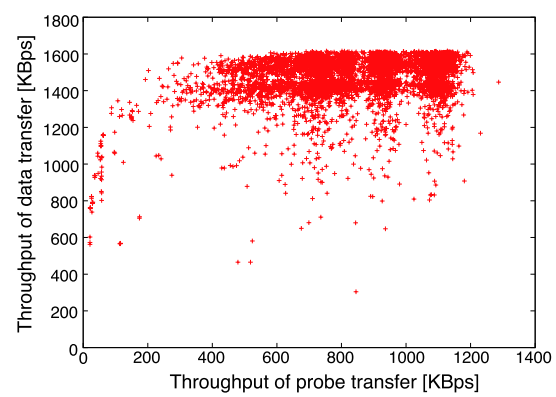


Fig. 3 Actual throughput for connection pair at node pair (η, θ).

in Fig. 2. We gathered training data sets for all the node pairs over seven days. There were no anomalous cases in the training sets. The statistics for the training sets are shown in Table 2. The actual throughput for the connection pair at (η, θ) is described in Fig. 3. It is shown to be non-linear with noise. Thus, probe and data throughputs are monotonically changed. Finally, we should consider noise and non-linear characteristics for a precise prediction model.

3.3 Input Data Sets

To evaluate prediction methods, the input data sets per node pair were collected over 36 hours. The actual throughput of the input data set at node pair (η, θ) is shown in Fig. 4. The actual throughput widely fluctuated with noise. Prediction results under an unstable network state are more important than those under a stable one. If the network state is stable or stationary, we do not have to predict network throughput. However, the scale and the bandwidth of networks are rapidly increasing and the network state is dynamically changing. To evaluate the prediction results under an unsta-

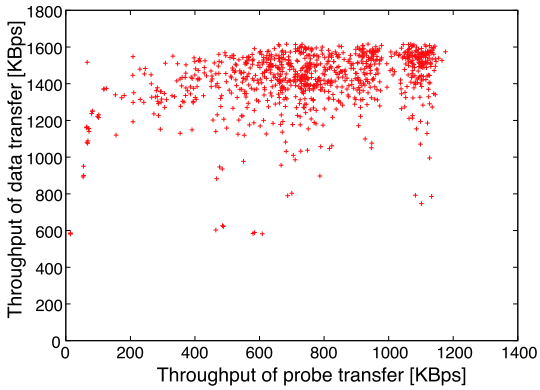


Fig. 4 Actual throughput of input data set at node pair (η, θ).

Table 3 Composition of input data sets. (NT is network throughput).

Node pair	Mean NT [KBps]	Total counts	Below mean
α, β	1116.0	1134	328
γ, δ	699.2	779	146
ϵ, ζ	586.0	1130	325
η, θ	760.5	929	378
κ, λ	364.9	654	259
μ, ν	656.5	992	377

ble network state, we determined the mean throughput of a probe transfer as a threshold value, and additional input data sets consisted of the actual throughput below the threshold value. The composition of the data sets is described in Table 3. For example, at node pair (α, β), the number of connection pairs at the input set is 1134 and that at the additional set is 328.

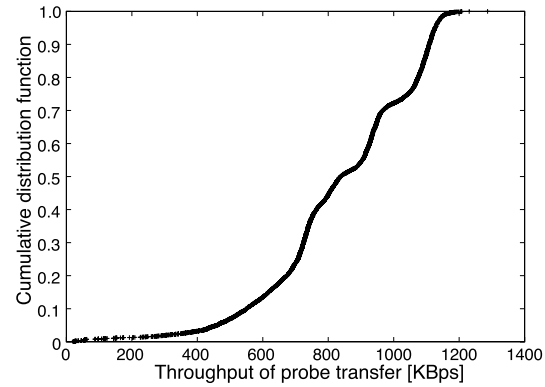
4. Previous Prediction Method (CDF Predictor)

4.1 Building CDF Predictor

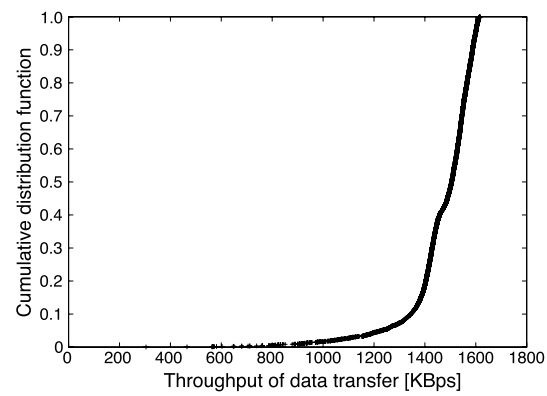
While other predictors require network metrics, such as RTT, packet loss, and so on, it is possible to build the CDF predictor [9] using only the connection pair. Thus, it is appropriate for the comparison of prediction results under the same condition. We first built the CDF predictor with the training sets to evaluate whether this predictor produces precise prediction results. It computed the CDF of throughput for probe and data transfers. The throughput for data transfer was predicted by using the CDF of a probe transfer. If there is no noise, throughput can be predicted precisely. The CDF of the connection pair at node pair (η, θ) is shown in Fig. 5. In the previous work [9], the curve shape of the CDF of the connection pair was similar, but that of the CDF at all the node pairs was different. The cause of the different shape is noise.

4.2 Prediction Results

Prediction results of the CDF predictor at node pair (κ, λ) are shown in Fig. 6. Because the CDF predictor deals with all data, the results are far from those for the input data set.



(a) CDF of probe transfer



(b) CDF of data transfer

Fig. 5 CDF of connection pair at node pair (η, θ).

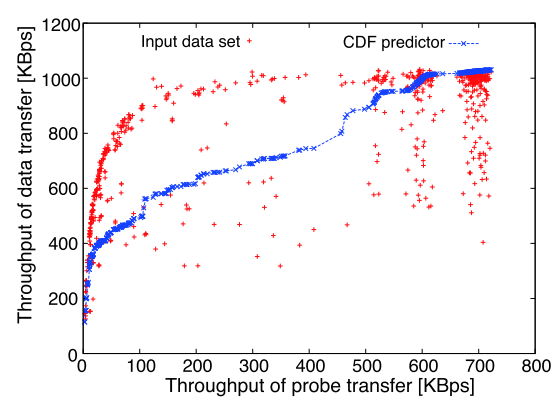


Fig. 6 Prediction results of CDF predictor at node pair (κ, λ).

Thus, a major cause of the difference is noise. Moreover, the difference between the actual throughput and the predicted throughput becomes large when the probe throughput is below the mean probe throughput (364.9 KBps). Therefore, the prediction results would be inaccurate under an unstable network state. The other results are similar to the above results. We should consider the noise and monotonicity between probe and data throughputs for a precise prediction model.

5. Proposed Prediction Method (SVR Predictor)

5.1 SVR Overview

Support vector regression (SVR) is a version of a support vector machine (SVM) [19] for regression. The concept of SVR is to maximize margins. Assume we have a training data set $\{(x_1, y_1), ..., (x_i, y_i), ..., (x_l, y_l)\} \in R^n \times R$, where R^n is the space of the input features x_i, and y_i is a symbol value. Here, we give an overview of two types of SVR: ϵ-SVR [19] and ν-SVR [20]. ϵ-SVR finds a function $f(x)$ that approximates future values accurately. The function is defined as

$$f(x) = w\phi(x) + b \tag{1}$$

where $w \in R^n, b \in R$, and ϕ is a non-linear transformation from R^n to high-dimensional spaces. An ϵ-insensitive loss function is used to measure an empirical error and is defined as

$$L_\epsilon(f(x_i), y_i) = \begin{cases} 0, & \text{if} |f(x_i) - y_i| \leq \epsilon \\ |f(x_i) - y_i| - \epsilon, & otherwise \end{cases} \tag{2}$$

ϵ-SVR can be written as

$$min \frac{1}{2}||w||^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \tag{3}$$

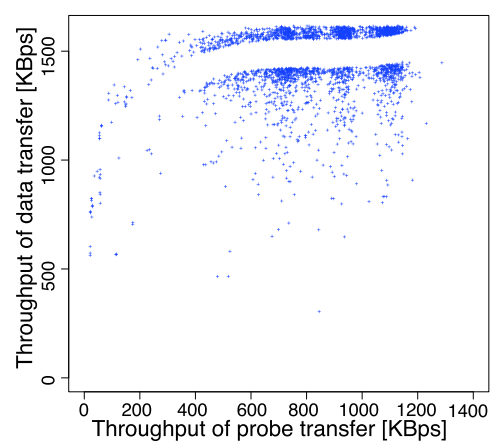
where C is a weight parameter. The constant $C > 0$ is used to determine the trade-off between training error and model flatness. Slack variables ξ_i and ξ_i^* are allowed to lie outside of an ϵ-insensitive tube. Thus, ϵ-SVR calculates the distance of data points on the tube to determine the shape of the tube. However, the shape of the tube would be changed inappropriately if there was an outlier, such as noise. ν-SVR is a modified version of ϵ-SVR. It has the advantage that a parameter ν, which replaces C, can be interpreted as both an upper bound on the fraction of margin errors and a lower bound on the fraction of support vectors. Thus, there is no calculation of distance on the outside tube in ν-SVR. Even if there is an outlier, we can determine the appropriate shape of the tube. We thus selected ν-SVR for our prediction method. We can apply the kernel trick [21] in SVR without ever having to compute the mapping explicitly. The value of the kernel is equal to the inner product of two vectors x_i and x_j in the feature space $\phi(x_i)$ and $\phi(x_j)$. Commonly used kernel functions are linear, polynomial, and radial basis. In our prediction method, we use the polynomial function to consider the non-linear characteristics of traffic. It is given by

$$k(x_i, x_j) = (< x_i \cdot x_j > +1)^d \tag{4}$$

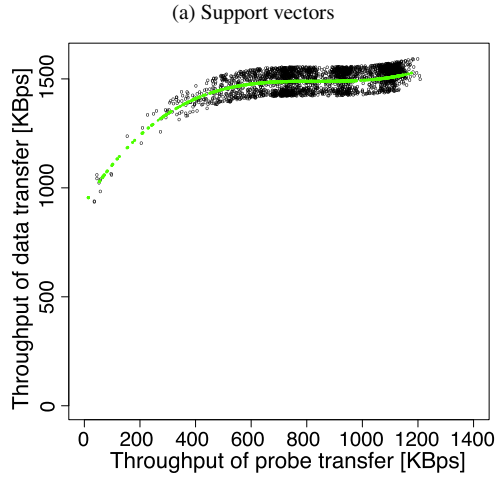
where d is degrees.

5.2 Building SVR Predictor

We introduce here our prediction method using ν-SVR and the polynomial kernel. A linear regression curve for throughput prediction would be inappropriate. Various types of traffic, such as mice and elephants [1], [2], co-exist in current networks, and spikes that correspond to large and abrupt throughput are occasionally caused by the elephants. The distribution of traffic fluctuation is close to long-tail by the above characteristics. In particular, the marginal distribution of the traffic is not Gaussian [1]. Again, the actual throughput was changed monotonically and there was noise. Then, ϵ-SVR would be inappropriate because it calculates the distance of data points for the shape of the tube. The proposed method uses ν-SVR to deal with noise. The radial basis function (RBF) has been introduced for the purpose of function interpolation, and it can also be used for non-linear characteristics. In comparison with the polynomial kernel, the RBF would be undesirable when there is noise or a paucity of data. Because of this, we apply the polynomial kernel of degree 3 into the proposed method to consider a non-linear and continuous monotonic function. The SVR predictor for node pair (η, θ) is shown in Fig. 7. The number of support vectors is 2472, and the other data points are used for the tube of the regression curve. Although there is noise, the SVR predictor reflects the characteristics of a non-linear and continuous monotonic function. The other node pairs are similar to the above case. The e1071 package [22] in



(a) Support vectors



(b) Tube of regression curve

Fig. 7 SVR predictor at node pair (η, θ).

R [23] is used for the predictor. It offers an interface to the libsvm library [24], which is a popular SVM tool. The other parameters in the package are set to the default values.

6. Comparison of Prediction Results

To evaluate the accuracy of an individual throughput prediction result at the predictors, we used the relative prediction error (RPE) [10], which is defined as

$$RPE = \frac{\hat{R} - R}{min(\hat{R}, R)} \tag{5}$$

where $\hat{R}$ is the predicted throughput and R is the actual throughput. We show the fraction of RPE within 10% or less, which is written as

$$Fraction\ of\ RPE = \frac{\sharp\{\boldsymbol{r} | -0.1 < RPE(\boldsymbol{r}) < 0.1\}}{\sharp\{\boldsymbol{r}\}} \tag{6}$$

where $\boldsymbol{r}$ is the input set. To evaluate the entire input data sets and the entire additional input data sets, we used the root-mean-square error (RMSE). It provides an error for the entire input sets. The minimum error value would be one key criterion in selecting a precise predictor.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{R} - R)^2} \tag{7}$$

where n is the number of connection pairs at the input set, and $\hat{R}$ and R are the same as for RPE. To summarize, we compare the individual prediction result through RPE and the entire prediction results through RMSE. The fraction of RPE within 10% or less for the input data sets is shown in Fig. 8. At node pair (μ, ν), 49.8% of the CDF predictor has an RPE of 10% or less while 89.3% of the SVR predictor has an RPE of 10% or less. In the other input data sets, the fraction of RPE with SVR was higher than that with CDF. Moreover, the RMSE with SVR (Table 4) was also smaller than that with CDF. From these results, the SVR predictor is more precise than the CDF predictor. Next, prediction results at node pair (η, θ) are shown in Fig. 9. The regression curve of the SVR predictor is more accurate than that of the CDF predictor. Thus, the noise had little effect on the SVR predictor. The fraction of RPE within 10% or less and the RMSE for the additional sets are shown in Fig. 10 and Table 5 respectively. For the additional sets, the fraction of RPE with SVR was higher than that with CDF. Although the fraction of RPE with the SVR predictor decreased, the range of the drop was small in comparison with that of the CDF predictor. In the fraction of RPE with CDF, the range of the drop at node pair (κ, λ) was 33.9%, the largest value. Furthermore, the RMSE value with SVR was also smaller than that with CDF. These results are sufficient to show that

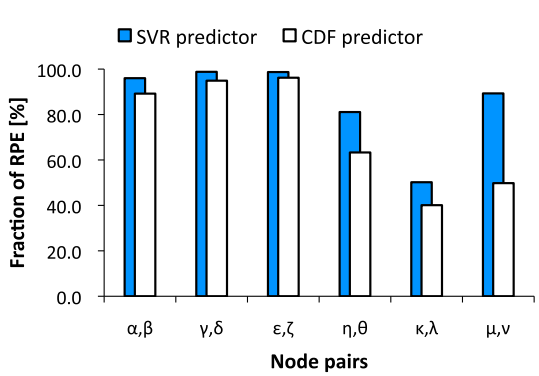


Fig. 8 Fraction of RPE within 10% or less for input data sets.

Table 4 Root-Mean-Square error of input data sets.

Node pair	SVR Predictor	CDF Predictor
α, β	94.6	113.3
γ, δ	27.0	51.4
ϵ, ζ	40.9	79.7
η, θ	155.7	187.5
κ, λ	168.1	203.0
μ, ν	102.7	157.2

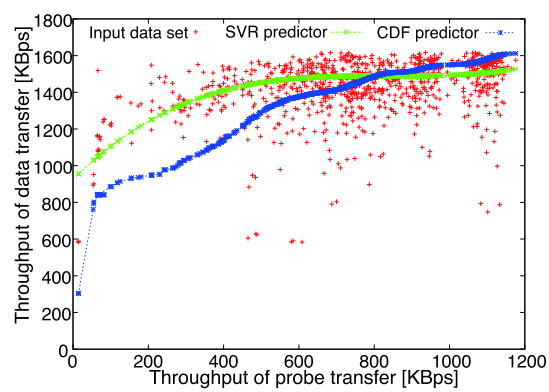


Fig. 9 Prediction results at node pair (η, θ).

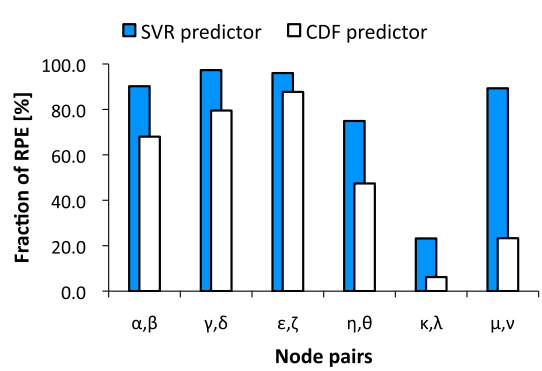


Fig. 10 Fraction of RPE within 10% or less for additional input data sets.

Table 5 Root-Mean-Square error of additional input data sets.

Node pair	SVR Predictor	CDF Predictor
α, β	137.7	180.3
γ, δ	35.0	98.2
ϵ, ζ	56.1	137.7
η, θ	183.0	230.0
κ, λ	182.6	251.9
μ, ν	100.8	192.5

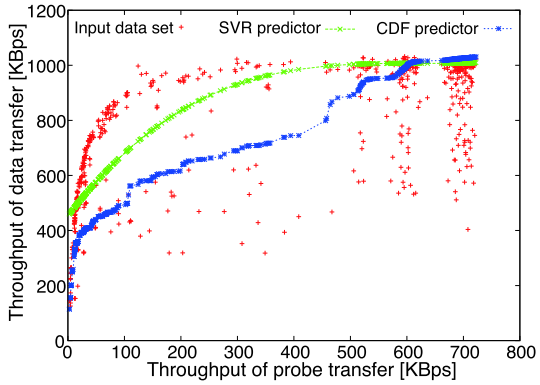


Fig. 11 Prediction results at node pair (κ, λ).

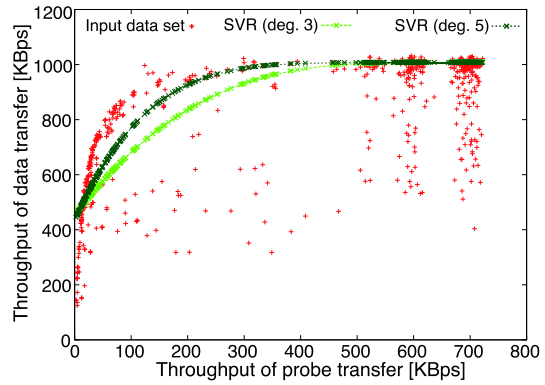


Fig. 12 Prediction results with degree 3 and 5 at node pair (κ, λ).

Table 6 Fraction of RPE of degrees for input data sets. (Deg. is degree).

Node pair	Fraction of RPE [%]			
	Deg. 2	Deg. 3	Deg. 4	Deg. 5
α, β	96.0	96.0	96.0	96.0
γ, δ	98.8	98.8	98.8	98.5
ϵ, ζ	98.7	98.7	98.3	98.3
η, θ	81.5	81.1	80.2	79.9
κ, λ	48.9	50.2	51.8	53.8
μ, ν	89.0	89.3	89.6	89.6

our SVR predictor is precise, robust, and better performing than the CDF predictor.

While the CDF predictor deals with all data that include noise, our SVR predictor uses only characteristic features in the training set. This explains its better prediction results. The regression curve of the SVR predictor at node pair (κ, λ) (Fig. 11) was also closer to the input data set in comparison with that of the CDF predictor. Because computational resources and the I/O device of the node on the virtualized testbed are shared by many slices, the sharing can affect the prediction results. Moreover, a congested network state for the node pair can also affect the prediction results. We should clarify what effects lead to the changes in the prediction results, and investigation of this is one of our future works.

To investigate the adequacy of degree 3, we performed the same experiments by varying the degree from 2 to 5. We omitted the polynomial kernels of degree 6 and above because the regression curve could be fitted to a complicated curve, consequently resulting in overfitting. Moreover, it is time-consuming to determine the regression curve with kernels of high degree. The evaluation results are summarized in Table 6. These results show that there are no significant differences in the RPE, except in the case at node pair (κ, λ). Thus, in this study, we concluded that the 3-degree polynomial kernel, which is the default value, is a reasonable choice for our prediction method. Next, from the results obtained with degree 5 at node pair (κ, λ), we found that the 5-degree kernel could achieve better RPE results than the 3-degree kernel. Figure 12 shows the regression curves of both 3- and 5-degree kernels. From this figure, we observe that although both the curves do not fit ideally, the discrepancy is small when we use the 5-degree kernel. In future, we intend to investigate the reason why kernels of higher degree can outperform in such cases.

To summarize, the actual throughput for the connection pair had noise and non-linear characteristics, and the CDF predictor was unsuitable for considering the above characteristics. In our prediction method, ν-SVR and the polynomial kernel are used to deal with a non-linear and continuous monotonic function. These lead to the improved prediction results. In the evaluation, we showed that our proposed method is better performing than the CDF predictor through RPE and RMSE.

7. Conclusion

In this work, we focused on measurements, analyses, and modeling for precise prediction results. The appropriate probe for the connection pair was selected through the rank correlation coefficient. Due to the impact of virtualization, an incorrect probe size was first selected. After filtering the negative effects, the 256-KB probe was selected as the appropriate probe size. The actual throughput with the 256-KB probe was non-linearly and monotonically changed with noise. We first built a predictor based on an existing prediction method [9] for the same data sets as for our method to evaluate whether it produces precise prediction results. We found that the existing prediction method was unsuitable when actual throughput was non-linearly and monotonically changed with noise. We thus proposed a throughput prediction method for precise prediction results. The proposed method uses ν-SVR and the polynomial kernel to deal with prediction models represented as a non-linear and continuous monotonic function. The prediction results of our proposed method are more accurate than those of the existing one. Furthermore, it is more robust than the existing one under an unstable network state. To summarize, 256-KB probes are appropriate for the current networks, and our SVR predictor is accurate, robust, and suitable for its purpose.

In future work, we intend to investigate what effects lead to the change in the prediction results and find appropriate parameters for SVR using grid search for the more

precise results. Next, we will gather data sets from multiple sites on the virtualized network testbed and non-virtualized environments to improve our prediction method, and compare the prediction results with those of other prediction methods. We will also design and implement a network throughput prediction system for network-aware applications.

Acknowledgments

We would like to express our gratitude to those who reviewed our paper and provided many comments that helped to improve our paper. This work was supported in part by JSPS KAKENHI (22500072 and 22700029) and Global COE Program "Frontiers of Intelligent Sensing" from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

[1] T. Mori, R. Kawahara, S. Naito, and S. Goto, "On the characteristics of Internet traffic variability: Spikes and elephants," IEICE Trans. Inf. & Syst., vol.E87-D, no.12, pp.2644–2653, Dec. 2004.
[2] Y. Zhang, L. Breslau, V. Paxson, and S. Shenker, "On the characteristics and origins of internet flow rates," Proc. 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '02, pp.309–322, New York, NY, USA, ACM, 2002.
[3] R. Wolski, N. Spring, and J. Hayes, "The network weather service: A distributed resource performance forecasting service for metacomputing," J. Future Generation Computing Systems, vol.15, pp.757–768, 1999.
[4] O. Khalid, R.J. Anthony, P. Nilsson, K. Keahey, M. Schulz, K. Parrot, and M. Petridis, "Enabling and optimizing pilot jobs using xen based virtual machines for the hpc grid applications," Proc. 3rd International Workshop on Virtualization Technologies in Distributed Computing, VTDC '09, pp.1–8, New York, NY, USA, ACM, 2009.
[5] StarBED, http://www.starbed.org/
[6] PlanetLab, https://www.planet-lab.org/
[7] Amazon EC2, http://aws.amazon.com/ec2/
[8] C. Lee, H. Abe, T. Hirotsu, and K. Umemura, "A statistical approach for selecting throughput prediction parameters on the Internet," Ubiquitous Information Technologies and Applications (CUTE), 2011 Proc. 6th International Conference, Korea Information Processing Society (KIPS), pp.37–40, Dec. 2011.
[9] M. Swany and R. Wolski, "Multivariate resource performance forecasting in the network weather service," Supercomputing '02: Proc. 2002 ACM/IEEE Conference on Supercomputing, pp.1–10, Los Alamitos, CA, USA, IEEE Computer Society Press, 2002.
[10] Q. He, C. Dovrolis, and M. Ammar, "On the predictability of large transfer tcp throughput," SIGCOMM '05: Proc. 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, pp.145–156, New York, NY, USA, ACM, 2005.
[11] M.M. Yousaf, M. Welzl, and M.M. Junaid, "Fog in the network weather service: A case for novel approaches," Proc. First International Conference on Networks for Grid Applications, GridNets '07, ICST, pp.23:1–23:6, Brussels, Belgium, Belgium, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007.
[12] S. Vazhkudai and J.M. Schopf, "Predicting sporadic grid data transfers," HPDC '02: Proc. 11th IEEE International Symposium on High Performance Distributed Computing, p.188, Washington, DC, USA, IEEE Computer Society, 2002.
[13] P. Bermolen and D. Rossi, "Support vector regression for link load prediction," Comput. Netw., vol.53, pp.191–201, Feb. 2009.
[14] R. Beverly, K. Sollins, and A. Berger, "Svm learning of ip address structure for latency prediction," Proc. 2006 SIGCOMM workshop on Mining network data, MineNet '06, pp.299–304, New York, NY, USA, ACM, 2006.
[15] H. Feng, Y. Shu, and M. Ma, "Wlan traffic prediction using support vector machine," IEICE Trans. Commun., vol.E92-B, no.9, pp.2915–2921, Sept. 2009.
[16] M. Mirza, J. Sommers, P. Barford, and X. Zhu, "A machine learning approach to tcp throughput prediction," IEEE/ACM Trans. Netw., vol.18, pp.1026–1039, Aug. 2010.
[17] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient overlay networks," SIGOPS Oper. Syst. Rev., vol.35, pp.131–145, Oct. 2001.
[18] C. Lee, H. Abe, T. Hirotsu, and K. Umemura, "Traffic anomaly analysis and characteristics on a virtualized network testbed," IEICE Trans. Inf. & Syst., vol.E94-D, no.12, pp.2353–2361, Dec. 2011.
[19] V.N. Vapnik, The nature of statistical learning theory, Springer-Verlag New York, New York, NY, USA, 1995.
[20] B. Schölkopf, A.J. Smola, R.C. Williamson, and P.L. Bartlett, "New support vector algorithms," Neural Comput., vol.12, pp.1207–1245, May 2000.
[21] A. Aizerman, E.M. Braverman, and L.I. Rozoner, "Theoretical foundations of the potential function method in pattern recognition learning," Automation and Remote Control, vol.25, pp.821–837, 1964.
[22] e1071:Misc Functions of the Department of Statistics, http://cran.r-project.org/web/packages/e1071/
[23] The R Project for Statistical Computing, http://www.r-project.org/
[24] C.C. Chang and C.J. Lin, "LIBSVM: A library for support vector machines," ACM Trans. Intelligent Systems and Technology, vol.2, pp.27:1–27:27, 2011. Software available at url http://www.csie.ntu.edu.tw/~cjlin/libsvm

Chunghan Lee received the B.Eng. from Korea University of Technology and Education, Korea in 2007 and the M.Eng. from Toyohashi University of Technology, Japan in 2010. He is a doctoral course student in the Dept. of Electronic and Information Engineering at Toyohashi University of Technology. His current research interests include network measurement and system software for network virtualization areas.

Hirotake Abe received the B.Eng. degree in 1999, the M.Eng. degree in 2001, and the Ph.D. degree in 2004, all from University of Tsukuba, Japan. From 2004 to 2007, he was a research staff of Japan Science and Technology Agency. From 2007 to 2010, he was an Assistant Professor in Toyohashi University of Technology, Japan. He is currently an Assistant Professor in Cybermedia Center, Osaka University, Japan. His research interests include system software, distributed systems and computer security. He received the distinguished paper award from IPSJ in 2005.

Toshio Hirotsu received the Ph.D. degree in computer science from Keio University in 1995. From 1995 to 2004, he worked in NTT Laboratories, Japan. He was in the Dept. of Information and Computer Science at Toyohashi University of Technology as an Associate Professor from 2004 to 2009. He is currently a Professor of the faculty of the Computer and Information Science at Hosei University. His research interests include system software for Internet and the ubiquitous environment.

Kyoji Umemura received the B.Eng., the M.Eng. and the Ph.D. degrees from the University of Tokyo, Japan in 1981, 1983 and 1991 respectively. Currently he is a Professor in the Dept. of Computer Science and Engineering at Toyohashi University of Technology. His research interests include Information Retrieval, Lisp and Symbolic Computation, Compiler, Operating System and Natural Language Processing.