Estimating current traffic matrices accurately by using long-term variations information

Yuichi Ohsita[†], Takashi Miyamura[§], Shin'ichi Arakawa[‡], Eiji Oki[§],
Kohei Shiomoto[§], Masayuki Murata[‡]

[†] Graduate School of Economics, Osaka University
[‡] Graduate School of Information Science and Technology, Osaka University
[§] NTT Network Service Systems Laboratories, NTT Corporation

***Abstract*— Obtaining current traffic matrices is essential to traffic engineering (TE) methods. Because it is difficult to monitor traffic matrices, several methods for estimating them from link loads have been proposed. The models used in these methods, however, are incorrect for some real networks. Thus, methods improving the accuracy of estimation by changing routes also have been proposed. However, existing methods for estimating the traffic matrix by changing routes, however, can only capture long-term variations and cannot obtain current traffic matrices accurately. In this paper, we propose a method for estimating current traffic matrices by using route changes introduced by a TE method. In this method, we first estimate the long-term variations of traffic by using the link loads monitored the last M times. Then, we adjust the estimated long-term variations so as to fit the current link loads. In addition, when the traffic variation trends change and the estimated long-term variations cannot match the current traffic, our method detects mismatches. Then, so as to capture the current traffic variations, the method re-estimates the long-term variations after removing information about the end-to-end traffic causing the mismatches. For this paper, we evaluated our method through simulation. The results show that our method can estimate current traffic matrices accurately even when some end-to-end traffic changes suddenly.**

I. Introduction

Obtaining current traffic matrices accurately is essential to traffic engineering (TE) methods [1–4]. By using the current traffic matrices, TE methods configure routes on a network so as to fit the current traffic. As a result, even when traffic changes unpredictably, by reconfiguring routes, the network can efficiently accommodate all traffic without congestion.

One approach for obtaining traffic matrices is to construct fully meshed label-switched paths using Multiprotocol Label Switching (MPLS) and directly measure the traffic amounts over each path. This approach, however, does not scale because it requires N-squared label-switched paths. Another approach is to tally the numbers of packets of each end-to-end traffic flow at all edge nodes. This, however, is also difficult to apply in large-scale networks, because tallying these numbers requires a non-negligible amount of CPU resources at the edge nodes, and gathering the tallied data for all end-to-end traffic also consumes a non-negligible amount of network resources such as bandwidth.

Therefore, several methods for estimating traffic matrices from limited information have been proposed [5–14]. In such methods, an entire traffic matrix is estimated using link loads that can be collected much more easily than by directly monitoring end-to-end traffic. Because the link load is the sum of the traffic using a link, we have

$$X(n) = A(n)T(n), \tag{1}$$

where $X(n)$ is a matrix indicating the amount of traffic on each link at time n, $T(n)$ is the traffic matrix at time n, and $A(n)$ is the routing matrix (i.e., a matrix in which an element corresponding to an instance of end-to-end traffic and a link is 1 if the end-to-end traffic passes the link or 0 if it does not). However, because the number of links is much smaller than the number of elements of the traffic matrix, Eq. (1) has multiple solutions in which true traffic matrix is included.

Therefore, several methods use traffic matrix models to estimate the traffic matrix. For example, the tomogravity method [5] uses a model called the gravity model, in which the amount of traffic from a source node to a destination node is proportional to the total incoming or outgoing traffic for each edge node. According to [15], however, the gravity model does not fit the actual traffic in some real networks. Traffic matrices estimated by this method include estimation errors, such as underestimates of end-to-end traffic whose amounts are actually large. As a result, when a TE method uses traffic matrices estimated by this method, these underestimates can cause high link utilizations.

Recently, several methods increasing the accuracy of estimation by using additional measurements have been proposed [12–14]. These methods obtain the additional information by changing the routing matrices and observing the differences between the link loads before and after the route changes. For example, Ref. [12] obtains additional measurements by using the route changes performed by a TE method. By performing TE a sufficient number of times, this approach obtains a sufficient number of measurements and then estimates the traffic matrix by assuming that the true traffic matrix does not change throughout the TE method execution. It takes a long time to change routes sufficient times, however, so the current traffic can differ from the initial traffic monitored before the first route change. Therefore, we need a traffic matrix estimation method that considers the time variations of traffic matrices. Ref. [14] proposes a method for modeling traffic variations by using periodic functions and estimates these functions' parameters. However, when traffic changes unpredictably, the traffic matrices estimated by this approach cannot fit the current traffic matrices since it can only estimate the average variations of traffic for a period of a day by monitoring link loads for several days. As a result, a TE method cannot configure routes suitable for the current traffic.

Therefore, in this paper, we propose a new estimation method, with which we can accurately estimate current traffic matrices by using the route changes introduced via a TE method. Unlike in Ref. [14], the purpose of our method is to estimate not the long-term variations of traffic but the current traffic matrix, which consists of both long-term variations and short-term variations. By using the accurate traffic matrix, a TE method can properly work to configure routes suitable for the current traffic.

In our method, we first estimate the long-term variations of traffic by using the link loads monitored the last M times. Then, we adjust the estimated long-term variations so as to fit the current link loads. In addition, when the traffic variation trends change and the estimated long-term variations cannot match the current traffic, our method detects mismatches between the estimated long-term variations and the current traffic. Then, our method re-estimates the long-term variations after removing information about the end-to-end traffic causing the mismatches, so as to capture the current traffic variations.

The rest of this paper is organized as follows. Section II describes the proposed method for estimating current traffic matrices by using route changes. Then, in Section III, we give the results of evaluating our method through simulation. Finally, Section IV provides a conclusion.

II. Method for Estimating Current Traffic Matrix by Using Changes in Routes

A. *Overview of estimation method*

Obtaining current traffic matrices is essential to TE methods. The existing estimation methods, however, cannot estimate current traffic matrices accurately. As a result, if a TE method uses traffic matrices estimated by the existing estimation methods, the TE method cannot reconfigure routes suitable to the current traffic.

Therefore, in this paper, we propose a new method for estimating current traffic matrices accurately. We assume that a TE method sometimes changes routes in the network. Under this condition, we can obtain additional information, which can be used in estimating the traffic matrices, by monitoring link loads while some routes are changed.

Because it takes a long time to change routes enough times to obtain a sufficient amount of additional information, however, the current traffic might be very different from the initially monitored link loads. Therefore, we need to consider long-term variations. By using the link loads monitored the last M times, our method estimates the long-term variations of traffic instead of estimating the current traffic matrices directly. Then, we obtain the current traffic matrices by adjusting the estimated long-term variations so as to fit the current link loads.

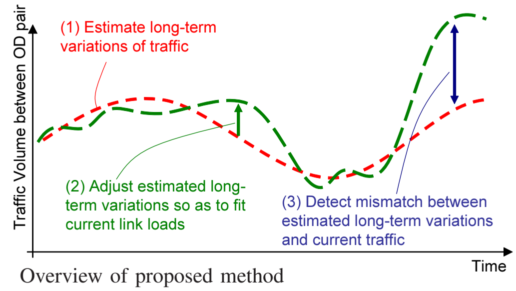


Fig. 1. Overview of proposed method

In addition, when the traffic variation trends change, the changes may cause significant estimation errors if we also use information obtained before the changes, since this information can be very different from the current traffic. Therefore, in our method, we check whether the estimated long-term variations match the current link loads. Then, if we detect a mismatch between the estimated long-term variations and the current link loads, we re-estimate the long-term variations after removing the traffic information causing the mismatch, so as to follow the current variations of traffic.

Fig. 1 shows an overview of the proposed estimation method. Our method estimates the traffic matrix through the following steps.

- Step 1 Estimate the long-term variations of the traffic matrices by using the link loads monitored the last M times.
- Step 2 Obtain estimation results of the current traffic matrix by adjusting the estimated long-term variations so as to fit the current link loads.
- Step 3 Check whether the estimated long-term variations fit the current link loads. If they do not match the current link loads, return to Step 1 after removing the previous information about the end-to-end traffic causing the mismatch. Otherwise, proceed to Step 4.
- Step 4 Designate the estimation results from Step 2 as the final estimation results.

In the above steps, estimating the long-term variations requires the longest computation time because this step uses the link loads monitored the last M times while other steps use only the currently monitored link loads. However, we do not need to estimate the long-term variations every time we estimate the current traffic matrices because we can estimate the current traffic matrices accurately if only the estimated long-term variations match the current traffic. Setting the interval of estimating the long-term variations independently of the interval of estimating the current traffic matrices is one of our future works.

In subsection II-B, we describe the method for estimating the long-term traffic variations. Subsection II-C explains how to adjust the estimated long-term variations so as to fit the current link loads. Subsection II-D describes how to the detect mismatches between the estimated long-term variations and the current traffic, and how to re-estimate the long-term variations and the current traffic matrix after mismatch detection.

B. Estimating long-term traffic variations

1) Traffic variation model: According to [14], the amount of traffic between each node pair varies periodically with a certain cycle, such as one day or one week. Therefore, in this paper, we model the traffic amount between nodes i and j as

$$t_{i,j}(n) = f_{i,j}(n) + \delta_{i,j}(n), \quad (2)$$

where $t_{i,j}(n)$ is the traffic volume between nodes i and j at time n, $f_{i,j}(n)$ is a function modeling the periodic variation, and $\delta_{i,j}(n)$ is the variation not included in $f_{i,j}(n)$. In our method, we model the long-term variations by $f_{i,j}(n)$ and estimate them by estimating the parameters of $f_{i,j}(n)$.

We model $f_{i,j}(n)$ by applying the model used in [14]. This approach models the periodic traffic variation by using sin and cos functions. With this model, the periodic variation is represented as

$$\begin{aligned} f_{i,j}(n) &= \sum_{h=0}^{N_f} \alpha_{h,i,j} cos\left(\frac{2\pi nh}{N^{\text{cycle}}}\right) \\ &+ \sum_{h=0}^{N_f} \alpha_{h+N_f,i,j} sin\left(\frac{2\pi nh}{N^{\text{cycle}}}\right). \end{aligned} \quad (3)$$

where N^{cycle} is the number of times monitoring link loads in each cycle, N_f is a parameter determining the number of terms in Eq. (3), and the $\alpha_{h,i,j}$ are the variables to be estimated by our estimation method. With N_f set to a large value, the traffic variation modeled by Eq. (3) captures more of the short-term variation, but the number of variables to be estimated also increases. In our method, we only have to roughly model the traffic variations, because we can estimate the current traffic matrix by adjusting the roughly estimated long-term variations. That is, in our method, a small N_f is sufficient.

2) Method for estimating long-term variations: In the model described by Eq. (3), the variables $\alpha_{h,i,j}$ determine the long-term variations. Therefore, our method estimates the long-term variations by estimating the $\alpha_{h,i,j}$. We estimate the $\alpha_{h,i,j}$ by using the link loads monitored the last M times. At any time n, the link loads and the traffic matrix have a relation described by Eq. (1). Therefore, we estimate all variables so as to satisfy Eq. (1) in any time. In this paper, we use a least square algorithm to estimate the variables. That is, when the number of nodes is N, the variables are basically estimated as

$$minimize \sum_{k=n-M+1}^{n} |X(k) - A(k)\hat{T}^{\text{est}}(k)|^2 \quad (4)$$

where

$$\hat{T}^{\text{est}}(k) = \begin{bmatrix} f_{0,0}(k) \\ \vdots \\ f_{i,j}(k) \\ \vdots \\ f_{N,N}(k) \end{bmatrix}. \quad (5)$$

By using Eq. (4), when some routes are changed, we can use additional equations for estimating the variables.

With Eq. (4), however, we may not be able to estimate the long-term variations accurately because of the effects of traffic variations that cannot be modeled by Eq. (3). Because the actual traffic variations do include variations that cannot be modeled by Eq. (3) (i.e., $\delta_{i,j}(n)$ in Eq. (2)), long-term variations modeled by Eq. (3) cannot completely fit all the monitored link loads. With Eq. (4), however, we estimate the long-term variations so as to completely fit all the monitored link loads. As a result, estimation results from Eq. (4) can be affected by traffic variations that cannot be modeled by Eq. (3), making the results very different from the actual traffic.

To mitigate the impact of $\delta_{i,j}$ on the estimated long-term variations, in our method, by placing constraints on the variables themselves, we avoid estimating the long-term variations so as to completely fit all the monitored link loads. We thus use the following equation instead of Eq. (4):

$$\begin{aligned} minimize \quad & \sum_{k=n-M+1}^{n} |X(k) - A(k)\hat{T}^{\text{est}}(k)|^2 \\ & + \Phi \sum_{i,j} \left(m_{i,j} \sum_{h=0}^{2N_f} (\alpha_{h,i,j} - \alpha'_{h,i,j})^2 \right), \end{aligned} \quad (6)$$

where the $\alpha'_{h,i,j}$ are the variables estimated the previous time, $m_{i,j}$ is the amount of information monitored before, and Φ denotes a parameter by which we can set the weight to the constraints on the variables themselves. Using this equation, we estimate all the $\alpha_{h,i,j}(0 \leq h \leq 2N_f)$ of $f_{i,j}(n)$ so as to fit all the monitored link loads while keeping the values close to the values estimated the previous time.

When we estimate the long-term variations the first time, however, we have not obtained the $\alpha'_{h,i,j}$. Thus, in such cases, we set the $\alpha'_{0,i,j}$ to the elements of traffic matrices estimated by other methods [5–11], and we set the $\alpha'_{h,i,j}(1 \leq h \leq 2N_f)$ to 0. By using this approach, we can avoid estimating traffic variations as having significantly larger values than the actual variations.

In addition, even if the initial $\alpha'_{h,i,j}$ are not accurate, we can estimate the long-term variations more accurately by using link loads monitored at multiple times as additional information. Then, when we estimate the long-term variations the next time, we can use more accurate $\alpha'_{h,i,j}$. That is, as we estimate the long-term variations more times, the accuracies of these estimations increase.

C. Adjustment of estimated long-term variations

In described in subsection II-B, we estimate the long-term variations. Because these estimates do not include the $\delta_{i,j}(n)$ in Eq. (2), however, they do not fit the current link loads. Therefore, we adjust the long-term variations estimated as given in subsection II-B so as to fit the current link loads.

The adjustment is performed through the following steps. First, by assigning n to the functions corresponding to the estimated long-term variations, we obtain a roughly estimated traffic matrix $\hat{T}^{\text{est}}(n)$. Then, we obtain a traffic matrix $\hat{T}(n)$ that is close to $\hat{T}^{\text{est}}(n)$ and fits the link loads monitored at time n. That is, we obtain the estimation results by applying a least square algorithm so as to satisfy the following conditions:

$$minimize |\hat{T}(n) - \hat{T}^{\text{est}}(n)|^2 \quad (7)$$

where

$$A(n)\hat{T}(n) = X(n). \tag{8}$$

A traffic matrix estimated by a least square algorithm, however, can include negative values, which are meaningless in the context of a traffic matrix. Therefore, we eliminate negative values through the following steps. We denote the estimated traffic matrix for the i-th iteration as $\hat{T}^{(i)}(n)$.

Step 1 Let $\hat{T}^{(0)}(n) \leftarrow \hat{T}(n)$.

Step 2 Obtain the matrix $\hat{T}'^{(i)}(n)$, in which we replace all the negative values of $\hat{T}^{(i)}(n)$ with zero.

Step 3 Obtain $D^{(i)}(n)$ satisfying the following condition:

$$\mathit{minimize}|D^{(i)}(n)|^2 \tag{9}$$

where

$$A(n)\left(\hat{T}'^{(i)}(n) + D^{(i)}(n)\right) = X(n). \tag{10}$$

Step 4 Let $\hat{T}^{(i+1)}(n) \leftarrow \hat{T}'^{(i)}(n) + D^{(i)}(n)$.

Step 5 If all elements of $\hat{T}^{(i+1)}(n)$ are non-negative, proceed to Step 6. Otherwise, return to Step 1.

Step 6 Let $\hat{T}^{(i+1)}(n)$ be the final result for the traffic matrix $\hat{T}(n)$.

D. Re-estimation of traffic matrix after mismatch of estimated long-term variations

When traffic variation trends change, long-term variations estimated by using all the link loads monitored the last M times can exhibit mismatches with the current traffic. This is because the long-term variations are estimated so as to fit the link loads before the change, which can be very different from the current traffic variations. In such cases of mismatch, we cannot estimate the current traffic matrices accurately even after adjustment, because the adjustment uses only the current link loads, which are insufficient for estimating the traffic matrices accurately.

Therefore, in our method, when the estimated long-term variations exhibit mismatches with the current traffic, we detect the mismatches and re-estimate the long-term variations without using link loads that do not match the current traffic. In this subsection, we describe how to detect mismatches and identify the end-to-end traffic causing the mismatches, as well as how to re-estimate the long-term variations after mismatch detection.

1) Detecting mismatches and identifying end-to-end traffic causing mismatches: When the estimated long-term variations are very different from the current traffic, the differences between the current link loads and the link loads calculated using the estimated long-term variations are large. In this case, because the results of adjusting $\hat{T}(n)$ must satisfy Eq. (8), while $A(n)\hat{T}^{\mathrm{est}}(n)$ is very different from the current link loads $X(n)$, the elements of $\hat{T}^{\mathrm{est}}(n) - \hat{T}(n)$, corresponding to the traffic causing the mismatches, become large. Therefore, we detect mismatches and identify the end-to-end traffic causing the mismatches by evaluating $\hat{T}^{\mathrm{est}}(n) - \hat{T}(n)$.

Because the size of traffic variation that cannot be included in Eq. (3) depends on the end-to-end traffic [14], if we set a single threshold for the elements of $\hat{T}^{\mathrm{est}}(n) - \hat{T}(n)$, traffic with large variations that cannot be modeled by Eq. (3) will be erroneously detected as traffic causing mismatches.

Therefore, we detect mismatches and identify their sources by comparing $\hat{T}^{\mathrm{est}}(n) - \hat{T}(n)$ with its previous values. Our method performs the comparison by using the Smirnov-Grubbs method [16], which can easily detect outliers in sampled data.

Here, we define the elements of $\hat{T}^{\mathrm{est}}(n)$ and $\hat{T}(n)$ corresponding to the traffic between nodes i and j as $\hat{t}^{\mathrm{est}}_{i,j}(n)$ and $\hat{t}_{i,j}(n)$ respectively. In the Smirnov-Grubbs method, we detect whether $|\hat{t}^{\mathrm{est}}_{i,j}(n) - \hat{t}_{i,j}(n)|$ is an outlier by calculating

$$d_{i,j} = \frac{|\hat{t}^{\mathrm{est}}_{i,j}(n) - \hat{t}_{i,j}(n)| - \mu_{i,j}}{\sigma_{i,j}}, \tag{11}$$

where $\mu_{i,j}$ and $\sigma_{i,j}$ are the average and standard deviation of $|\hat{t}^{\mathrm{est}}_{i,j}(k) - \hat{t}_{i,j}(k)| (n - M + 1 \le k \le n)$, respectively. Then, $|\hat{t}^{\mathrm{est}}_{i,j}(n) - \hat{t}_{i,j}(n)|$ is detected as an outlier if $d_{i,j}$ is larger than the theshold

$$\tau = (M-1)\sqrt{\frac{\tau^2_{\theta,M+2}}{M(M-2) + M\tau_{\theta,M+2}}} \tag{12}$$

where M is the number of samples, θ is a parameter specifying the detection sensitivity, and $\tau_{\theta,M}$ is a value corresponding to the top $\theta/M\%$ points of the T distribution with $M-2$ degrees of freedom.

Too small $\sigma_{i,j}$ causes detection of points where $|\hat{t}^{\mathrm{est}}_{i,j}(n) - \hat{t}_{i,j}(n)|$ is small. We do not, however, need to detect such points, because the estimated long-term variations there fit the current traffic, since $|\hat{t}^{\mathrm{est}}_{i,j}(n) - \hat{t}_{i,j}(n)|$ is small. Therefore, to avoid detecting such points, we introduce a parameter s and set $\sigma_{i,j}$ to s if $\sigma_{i,j}$ is smaller than s.

2) Re-estimation of long-term variations after detection: When mismatches between the estimated long-term variations and the current traffic are detected, we need to re-estimate the long-term variations so as to fit the current traffic. Because such mismatches occur when we estimate the long-term variations by using previously monitored link loads that are very different from the current traffic variations, we re-estimate the long-term variations by using link loads and routing matrices in which information about the end-to-end traffic causing the mismatches has been removed.

Our method removes previous information corresponding to the end-to-end traffics causing mismatches at time n through the following steps. We first remove such information from the routing matrices $A(i)(n - M + 1 \le i < n)$ by setting elements corresponding to the identified end-to-end traffic to 0. We denote the routing matrix after such replacement as $A'(i)$.

Then, we create a link load matrix $X'(i)(n - M + 1 \le i < n)$ from which information about the identified end-to-end traffic has been removed. The sum of the elements of traffic matrix T corresponding to the identified end-to-end traffic traversing each link at time i is calculated as $(A(i) - A'(i))\,T$. Therefore, X'_i is given by

$$X'(i) = X(i) - (A(i) - A'(i))\,\hat{T}'^{\mathrm{est}}(i). \tag{13}$$

where $\hat{T}'^{\mathrm{est}}(i)$ is the traffic matrix at time i calculated using the estimated long-term variations. In calculating $\hat{T}'^{\mathrm{est}}(i)$, we use the long-term variations estimated at time $n-1$, since the long-term variations estimated at time n can be affected by changing trends.

Next, our method re-estimates the long-term variations by using Eq. (14), which is refined from Eq. (6) to use $X'(k)$ and $A'(k)$:

$$\begin{aligned} \textit{minimize} \quad & \sum_{k=n-M+1}^{n-1} |X'(k) - A'(k)\hat{T}^{\text{est}}(k)|^2 \\ & + |X(n) - A(n)\hat{T}^{\text{est}}(n)|^2 \\ & + \Phi \sum_{i,j} \left(m_{i,j} \sum_{h=0}^{2N_f} (\alpha_{h,i,j} - \alpha'_{h,i,j})^2 \right). \end{aligned} \tag{14}$$

We only have to re-estimate the long-term variations so as to fit the current traffic, because the purpose of our method is to estimate the current traffic matrix. Moreover, in estimating the traffic amounts of the identified end-to-end traffic by using Eq. (14), we do not need to consider the related traffic variations, because the traffic amounts corresponding to the identified traffic are included only in $X(n)$.

Therefore, in re-estimating the long-term variations, we model the amounts of the identified end-to-end traffic by

$$f_{i,j}(n) = \alpha_{0,i,j}, \tag{15}$$

instead of using Eq. (3). By using Eq. (15), we can minimize the number of variables to be estimated.

3) Re-estimation of traffic matrix after re-estimation of long-term variations: After re-estimating the long-term variations, we re-estimate the current traffic matrix through the same steps described in subsection II-C.

III. Evaluation

A. Metrics

In this section, we describe an evaluation of our method by simulation. In the simulation, we evaluated our method by two general metrics: (1) the accuracy of estimation, and (2) the performance of a TE method using the estimated traffic matrices.

To evaluate the accuracy, we used two specific metrics – the root mean squared error (RMSE), and the root mean squared relative error (RMSRE) – as defined below:

$$\text{RMSE} = \sqrt{\frac{1}{N^2} \sum_{1 \le i,j \le N} (\hat{t}_{i,j}(n) - t_{i,j}(n))^2} \tag{16}$$

$$\text{RMSRE} = \sqrt{\frac{1}{N_{\tilde{t}}^2} \sum_{1 \le i,j \le N, t_{i,j} > \tilde{t}} \left(\frac{\hat{t}_{i,j}(n) - t_{i,j}(n)}{t_{i,j}(n)} \right)^2} \tag{17}$$

The RMSE gives an overall measure for the errors in estimation, while the RMSRE gives a relative measure. For small matrix elements, however, the relative errors are not really important. Thus, in computing the RMSRE, we consider only matrix elements greater than a threshold $\tilde{t}$. $N_{\tilde{t}}$ is the number of elements greater than $\tilde{t}$ in a traffic matrix. In the following simulation, $\tilde{t}$ was set so that the sum of the end-to-end traffic whose actual rate was greater than $\tilde{t}$ composed 75 % of the total traffic.

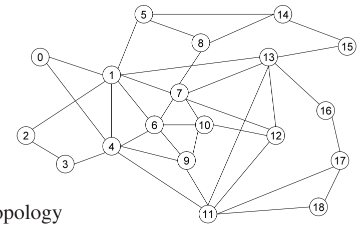


Fig. 2. EON topology

To evaluate the performance of a TE method using the estimated traffic matrices, we investigated whether the purpose of the TE method was achieved. The next subsection describes the purpose of the TE method used in our simulation.

B. Environment used in evaluation

In our method, we assume that a TE method changes routes sometimes. In this evaluation, we used the optical layer TE as an example of a TE method. The optical layer TE establishes optical layer paths between two IP routers over a physical network consisting of IP routers and optical cross-connects (OXCs). A set of optical layer paths forms a virtual network topology (VNT). Traffic between two routers is carried over the VNT by using IP layer routing. Under these conditions, the optical layer TE accommodates traffic that fluctuates widely by dynamically reconfiguring the VNT.

In our simulation, we used the European Optical Network (EON) (19 nodes, 37 links) shown in Fig. 2 as the physical topology and executed the optical layer TE method proposed in [12] once an hour. The purpose of this method is to keep the maximum link utilization under the threshold T_H. In this method, optical layer paths are added or deleted with a limitation on the number of optical layer paths reconfigured at one time. Optical layer paths are added if at least one path whose utilization exceeds the threshold T_H exists. Otherwise, if there is an optical layer path whose utilization is less than a threshold T_L, the path is deleted. In this simulation, we set the maximum number of optical layer paths reconfigured at one time to 30, T_H to 0.7 and T_L to 0.4.

In the simulation, we investigate the accuracy of the estimation when the some traffic change suddenly. Therefore, we generate end-to-end traffic by adding sudden changes to the traffic generated by adding variations to sin functions whose amplitudes and phases were randomly generated. We added sudden changes to the traffic from nodes 2 to 4, 9 to 1, and 0 to 12 at times 70, 110, and 140, respectively. The rates of the sudden traffic changes from nodes 2 to 4, 9 to 1, and 0 to 12 were, respectively, 120 % , 150 %, and 160 % of the maximum rate of traffic before the addition.

In our estimation method, we use parameters M, N_f, Φ, θ and s. In this simulation, we set M to 160, N_f to 2, Φ to 0.01, θ to 0.01, and s to 1.

C. Accuracy of the estimation

In our method, we obtain estimation results by adjusting the estimated long-term variations so as to fit the current link loads. In addition, when the trends of traffic variations change and the estimated long-term variations do not match the current traffic, our method detects mismatches and identifies the end-to-end traffic causing them, after which it re-estimates the long-term variations.

Therefore, we investigated the effectiveness of adjusting the estimated long-term variations and effectiveness of re-estimation, by comparing the accuracy of our estimation method with the accuracies of the following methods:

- A method using only the current link loads. By comparison with this method, we investigated the effectiveness of using the link loads monitored at previous times. For this method, we used the tomogravity method with the simple gravity model [5]. Although the simple gravity model does not fit the traffic matrices used in our simulation, because we use randomly generated traffic matrices, this model also is not incorrect in some real networks [15]. The focus of this comparison is the effectiveness of using link loads monitored at previous times when the simple gravity model is not correct.
- A method using the link loads monitored at previous times but not considering the time variations of traffic. By comparison with this method, we investigated the effectiveness of modeling long-term traffic variations. For this method, we used the additional equation method proposed in [12].
- Our method without re-estimation. For our method without re-estimation, we estimated the long-term variations and adjusted them but did not re-estimate them even when the variation trends changed. By comparing with this method, we investigate the effectiveness of the re-estimation after detection of the mismatches between the estimated long-term variations and the current link loads.

Figures 3 and 4 show the RMSE and RMSRE when we added these sudden traffic changes. The results show that the errors for the tomogravity method are the largest. This is because the tomogravity method uses only the current link loads, which is an insufficient amount of information.

The errors for the additional equation method are also large. This is because that method does not consider traffic variations but assumes instead that the true traffic matrix does not change during TE execution. Therefore, this method cannot estimate traffic matrices accurately when traffic varies, even while monitoring the link loads a sufficient number of times.

On the other hand, the errors for our methods are relatively small. That is, by including the link loads monitored at previous times in considering the time variations of traffic, we can estimate traffic matrices accurately. However, the RMSE for our method without re-estimation increases after time 70, whereas the RMSE for our method with re-estimation remains small after time 70. This difference is caused by the sudden changes, whose impact we discuss in detail later. The RMSRE for our method without re-estimation does not increase as much as the RMSE. This is because the end-to-end traffic whose estimation errors increase due to the sudden change have large amounts of traffic in this case. So, the increases of the relative errors are small.

The results shown in Figs. 3 and 4 also verify that the impact of false positives (i.e., the cases of mistakenly detecting end-to-end traffic with no changes) is small. In this case, some end-to-end traffic without changes in the traffic variation trends

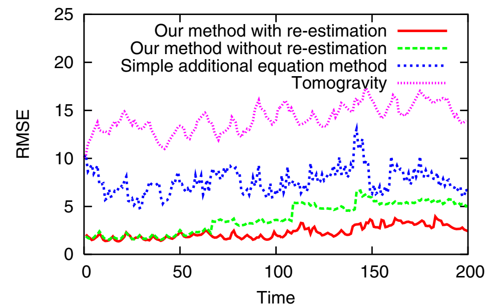


Fig. 3. Time variation of RMSE (when some traffic variations change)

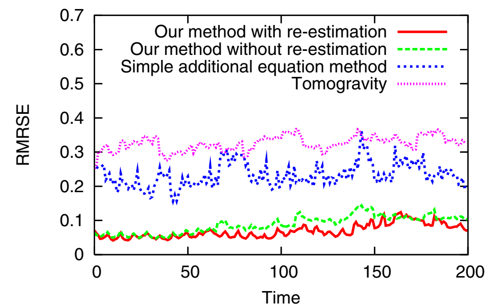


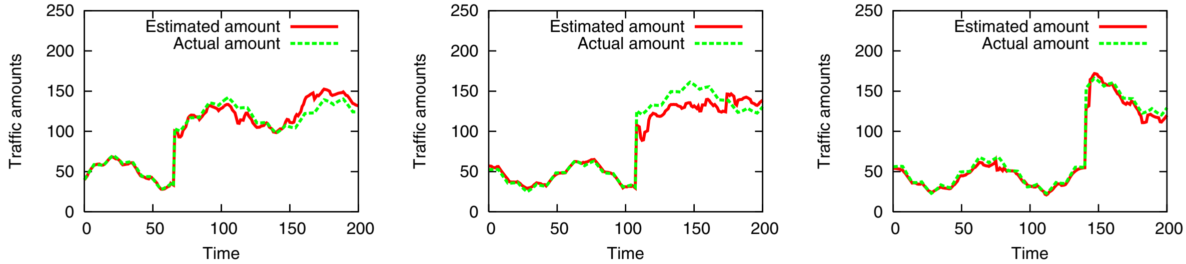
Fig. 4. Time variation of RMSRE (when some traffic variations change)

will be mistakenly identified as causing mismatches between the estimated long-term variations and the current traffic. For example, at time 76 in Fig. 3, the end-to-end traffic between nodes 14 and 0 is mistakenly identified as causing a mismatch. From the figure, however, we can see that the RMSE and RMSRE for our method do not become significant even when such false positives occur; they always remain the smallest among the four methods. This is because we have sufficient information to estimate the long-term variations and traffic matrices accurately even when some false positives occur and information about the mistakenly identified end-to-end traffic is removed.

To investigate the impact of sudden changes in detail, we compared the estimation results obtained for traffic with sudden changes added. Figures 5 and 6 show the estimation results for our method with and without re-estimation, respectively.

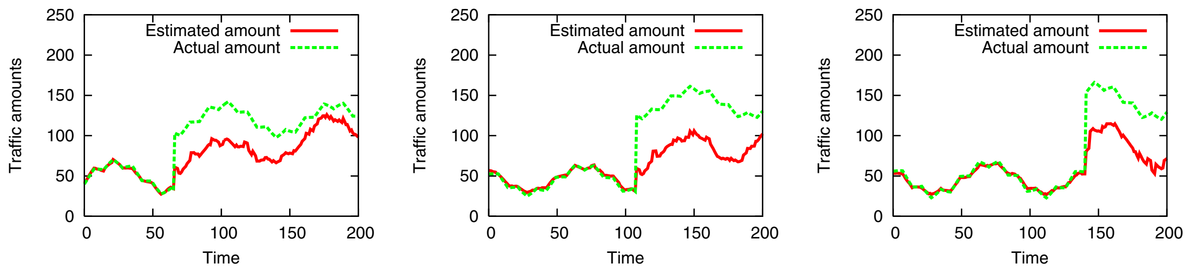
These figures show that both methods can accurately estimate all the traffic amounts before adding the sudden changes. After adding the changes, however, the traffic rate estimated by our method without re-estimation cannot capture the changes. This is because that method also uses the link loads monitored before adding the sudden changes, which are very different from the current traffic variations. Therefore, because of this information that does not fit the current traffic variations, the long-term variations cannot be estimated accurately. Even though we adjust the estimated long-term variations so as to fit the current link loads, the adjusted results still do not capture the sudden changes, because the adjustment process can use only the current link loads, which is insufficient information for estimating the traffic matrices accurately.

On the other hand, our method with re-estimation can estimate all the traffic amounts accurately even after adding the sudden changes. This is because by re-estimating the long-term variations after removing information about the end-to-



(a) Traffic between nodes 2 and 4 (b) Traffic between nodes 9 and 1 (c) Traffic between nodes 0 and 12

Fig. 5. Estimation results for our method with re-estimation



(a) Traffic between nodes 2 and 4 (b) Traffic between nodes 9 and 1 (c) Traffic between nodes 0 and 12

Fig. 6. Estimation results for our method without re-estimation

end traffic causing the mismatches between the estimated long-term variations and the current traffic, we avoid the impact of information that is very different from the current traffic variations.

D. Impact on performance of TE methods

Finally, we evaluate the performance of TE methods using traffic matrices estimated by our method. The TE method used in our simulations configured the VNT and routes over the VNT so as to keep the maximum link utilization under the threshold T_H. When we use traffic matrices including estimation errors, however, these errors can cause the maximum link utilization to be above T_H. Therefore, in this evaluation, we investigated the maximum link utilization after TE was performed. For this simulation, we used the same traffic described in the previous subsection.

Figure 7 shows the results of this simulation. The figure shows that when using the tomogravity method or the additional equation method, the maximum link utilization becomes significantly larger than the threshold T_H. This is because the estimation errors of these methods are large, as described above. When the estimation errors are large, the link utilizations after executing the TE method, as calculated using the estimated traffic matrix, can be very different from the actual link utilizations. As a result, the link utilizations after TE are mistakenly regarded as being lower than T_H, even though the actual link utilizations are still high and the necessary optical layer paths have not been added.

This figure also shows that the maximum link utilizations in the case of using our method without re-estimation sometimes become significantly larger than the threshold, as well. This is caused by significant underestimation of the traffic including the sudden changes. As shown in Fig. 6, our method without re-estimation cannot capture the added sudden changes and significantly underestimates their amounts. Because of these underestimates, when the TE method changes the routes of the underestimated traffic, it does not reserve enough bandwidth. As a result, since the actual traffic rates are much higher than expected, the link utilizations become high.

On the other hand, in the case of using our method with re-estimation, we can reduce the maximum link utilization to around T_H at all times. This is because, with re-estimation, our method can estimate traffic matrices accurately even when the traffic changes suddenly.

Although the maximum link utilization is reduced to around T_H with traffic matrices estimated by our method with re-estimation, however, it is not always smaller than T_H. This is because estimation errors can still be included in the results of our method with re-estimation, even though this method is the most accurate of the four methods considered here.

Especially when multiple instances of end-to-end traffic are identified as causing mismatches between the estimated long-term variations and the current traffic, the estimation errors increase, because removing the previous information about these multiple instances decreases the amount of information used for estimation. In the case of Fig. 7, two instances of end-to-end traffic are erroneously identified as causing the mismatch at time 183. These false positives cause a slight increase in

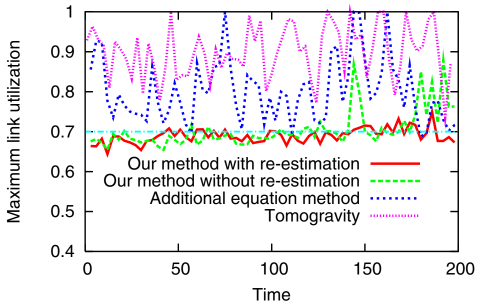


Fig. 7. Variation in maximum link utilization after TE execution

the estimation error, leading to a link utilization higher than T_H. Thus, to estimate traffic matrices more accurately, we need to minimize the number of false positives by setting parameters optimally or using a more sophisticated detection method. These considerations remain for our future work.

Minimizing the number of false positives is insufficient, however, because it is possible for multiple instances of end-to-end traffic to actually change suddenly, causing mismatches between the estimated long-term variations and the current traffic. When this happens, increases in estimation errors are difficult to avoid, because we cannot obtain sufficient information about such traffic changing suddenly. Therefore, to avoid the impact of such errors on methods using estimated traffic matrices, TE methods also need to consider estimation errors. This is another topic for our future work.

IV. Concluding remarks

In this paper, we have proposed a method for estimating current traffic matrices by using the changes in routing matrices introduced via a TE method. In this method, we first estimate the long-term variations of traffic matrices by using the link loads monitored the last M times. Then, we obtain the current traffic matrix by adjusting the estimated long-term variations of traffic so as to fit the current link loads. In addition, when the traffic variation trends change and the estimated long-term variations cannot fit the current variations, our method detects mismatches and identifies the end-to-end traffic causing them. Then, our method re-estimates the long-term traffic variations after removing information about the end-to-end traffic causing the mismatches.

For this paper, we evaluated our method through simulation. According to the results, our method can obtain accurate traffic matrices by adjusting the estimated long-term variations. In addition, when some end-to-end traffic changes suddenly and the estimated long-term variations do not match the current traffic, our method can detect mismatches accurately. Then, by re-estimating the long-term variations after removing information about the end-to-end traffic causing the mismatches, the method can estimate current traffic matrices accurately even when some end-to-end traffic changes suddenly.

In addition to evaluating the proposed method, we evaluated a TE method using traffic matrices estimated by our method. According to these results, by using the traffic matrices estimated by our method, a TE method can reduce the maximum link utilization to around its target value, whereas the maximum link utilization becomes high with other methods considered here.

Our future work will include optimally setting parameters such as M, Φ and N_f. In particular, M needs to be set to small value so as not to require too many resources (e.g., CPU and memories) especially in a large network, though setting M to smaller value may cause larger estimation errors. Thus, we need to set M, taking such trade-off into account. Additionally, although we used fixed values of N_f in this work, it might be possible to estimate traffic matrices more accurately by setting N_f dynamically according to the current measurement. Another future work is evaluation of our method by using the actual traffic traces.

Acknowledgement

This work was partially supported by Japan Society for the Promotion of Science (JSPS), Grant-in-Aid for Young Scientists (Start-up) 19800023.

References

[1] B. G. Jozsa, D. Orincsay, and L. Tamasi, "Multi-hour design of dynamically reconfigurable MPLS networks," in *Proceedings of Networking 2004*, pp. 502–503, May 2004.

[2] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights," in *Proceedings of IEEE INFOCOM 2000*, pp. 519–527, Mar. 2000.

[3] K. Shiomoto, E. Oki, W. Imajuku, S. Okamoto, and N. Yamanaka, "Distributed virtual network topology control mechanism in GMPLS-based multiregion networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, pp. 1254–1262, Oct. 2003.

[4] A. Gencata and B. Mukherjee, "Virtual-topology adaptation for WDM mesh networks under dynamic traffic," *IEEE/ACM Transactions on Networking*, vol. 11, pp. 236–247, Apr. 2003.

[5] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," in *Proceedings of ACM SIGMETRICS 2003*, pp. 206–217, June 2003.

[6] A. Medina, K. Salamatian, N. Taft, I. Matta, and C. Diot, "A two-step statistical approach for inferring network." Technical Report BUCS-2004-11, Computer Science Department, Boston University, Mar. 2004.

[7] Y. Zhang, M. Roughan, C. Lund, and D. Donoho, "An information-theoretic approach to traffic matrix estimation," in *Prodceedigs of ACM SIGCOMM*, pp. 301–312, Aug. 2003.

[8] C. Tebaldi and M. West, "Bayesian inference of network traffic using link count data," *Journal of the American Statistical Association*, vol. 93, pp. 557–576, June 1998.

[9] J. Cao, D. Davis, S. V. Wiel, and B. Yu, "Time-varying network tomography," *Journal of the American Statistical Association*, vol. 95, pp. 1063–1075, Feb. 2000.

[10] Y. Vardi, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of the Americal Statistical Association,*, pp. 365–377, Mar. 1996.

[11] I. Juva, S. Vaton, and J. Virtamo, "Quick traffic matrix estimation based on link count covariances," in *Proceedings of IEEE ICC 2006*, vol. 2, pp. 603–608, June 2006.

[12] Y. Ohsita, T. Miyamura, S. Arakawa, S. Ata, E. Oki, K. Shiomoto, and M. Murata, "Gradually reconfiguring virtual network topologies based on estimated traffic matrices," in *Proceedings of INFOCOM 2007*, pp. 2511–2515, May 2007.

[13] A. Nucci, R. Cruz, N. Taft, and C. Diot, "Design of IGP link weight changes for estimation of traffic matrices," in *Proceedings of INFOCOM 2004*, vol. 4, pp. 2341–2351, Mar. 2004.

[14] A. Soule, A. Nucci, R. Cruz, E. Leonardi, and N. Taft, "Estimating dynamic traffic matrices by using viable routing changes," *IEEE/ACM Transactions on Networking*, vol. 13, pp. 485–498, June 2007.

[15] A. Gunnar, M. Johansson, and T. Telkamp, "Traffic matrix estimation on a large IP backbone –a comparison on real data," in *Proceedings of Internet Measurement Conference*, pp. 149–160, Oct. 2004.

[16] S. Burke, "Missing values, outliers, robust statistics & non-parametric methods," *LC-GC Europe Online Supplement, Statistics & Data Analysis*, vol. 2, pp. 19–24, Jan. 2001.