社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

トラヒックマトリクス推定との協調による段階的な VNT 再構成

大下　裕一[†]　宮村　崇[††]　荒川　伸一[†††]　阿多　信吾[††††]　大木　英司[††]
塩本　公平[††]　村田　正幸[†††]

† 大阪大学 大学院経済学研究科
††† 大阪大学 大学院情報科学研究科
†††† 大阪市立大学 大学院工学研究科
†† 日本電信電話株式会社 ネットワークサービスシステム研究所
E-mail: †y-ohsita@econ.osaka-u.ac.jp, ††{miyamura.takashi,oki.eiji,kohei.shiomoto}@lab.ntt.co.jp,
†††{arakawa,murata}@ist.osaka-u.ac.jp, ††††ata@info.eng.osaka-cu.ac.jp

あらまし　トラヒックを効率的に収容する方法として、IP/光ネットワーク上で、光パスによって構築される論理トポロジ (VNT) を動的に再構成する手法の研究が進められている。しかしながら、VNT を適切に再構成するためには、対地間のトラヒック量を把握することが必須であるが、ネットワークの規模が大きくなるとともに、すべての対地間トラヒック量を測定することは困難となる。そのため、リンク負荷などの一部の測定情報から対地間トラヒック量を推定するトラヒックマトリクス推定手法の適用が望まれるが、トラヒックマトリクス推定を考慮に入れていない従来の VNT 再構成手法では、推定誤差の影響を大きく受けてしまう。そこで、本稿では、トラヒックマトリクス推定を考慮に入れた新しい VNT 再構成の手法を提案する。提案手法では、VNT 再構成を複数ステージに分け、前のステージでの測定情報を推定に反映させることにより、推定誤差を削減しつつ VNT 再構成を行う。また、提案手法では、各ステージで追加・削除される光パスの本数に制約をもうけることにより、推定誤差の影響を受ける範囲を制限した VNT 再構成を行う。本稿では、シミュレーションを用い、提案手法が誤差を削減し、トラヒックエンジニアリングへの誤差の影響を緩和できることを示す。
キーワード　トラヒックエンジニアリング、GMPLS、トラヒックマトリクス推定

Gradually Reconfiguring Virtual Network Topologies based on Estimated Traffic Matrices

Yuichi OHSITA[†], Takashi MIYAMURA[††], Shin'ichi ARAKAWA[†††], Shingo ATA[††††], Eiji OKI[††], Kohei SHIOMOTO[††], and Masayuki MURATA[†††]

† Graduate School of Economics, Osaka University
††† Graduate School of Information Science and Technology, Osaka University
†††† Graduate School of Engeneering, Osaka City University
†† NTT Network Service Systems Laboratories
E-mail: †y-ohsita@econ.osaka-u.ac.jp, ††{miyamura.takashi,oki.eiji,kohei.shiomoto}@lab.ntt.co.jp,
†††{arakawa,murata}@ist.osaka-u.ac.jp, ††††ata@info.eng.osaka-cu.ac.jp

Abstract　In this paper, we present a practical VNT (virtual network topology) reconfiguration method for large-scale IP and Optical networks with traffic matrix estimation considerations. We newly introduce a partial VNT reconfiguration algorithm with multiple transition stages. By dividing the whole VNT transition sequence into multiple transitions, estimation errors are calibrated at each stage by using network state information of prior stages. Because estimation errors are mainly due to less-constraint conditions in the estimated traffic matrix calculation, our approach tries to increase the constraint conditions for traffic matrix estimation by introducing partial reconfiguration, and to relax the impact of estimation errors by limiting the number of optical-paths reconfigured at each stage. We also investigate the effectiveness of our proposal through extensive simulations and clarify the robustness against estimation errors by using partial reconfiguration.
Key words　Traffic engineering, GMPLS, Traffic matrix estimation

1. Introduction

In a backbone network, optical layer traffic engineering (TE) is one efficient way for accommodating traffic that fluctuates unpredictably, which optimizes resource usage by dynamically reconfiguring the VNT (virtual network topology). Here, a VNT is defined as a set of optical paths that carry packet traffic between edge routers. Packet traffic between two IP routers is routed over the VNT. VNT reconfiguration approaches are to configure the optimal VNT for a given traffic demand, which improves the efficiency of network resource utilization [1–8]. This approach dynamically reconfigures the optical-layer topology, so it can accommodate traffic that fluctuates widely.

The VNT reconfiguration process consists of 1) the VNT computation phase and 2) the VNT transition phase, and for each phase, several algorithms can be considered. First, VNT computation algorithms can be categorized into a) full and b) partial reconfiguration algorithms. The former computes the new optimal VNT configuration without considering any operational limitations regarding from the current VNT to the new one, while the latter computes with consideration of a limit of the number of controlled optical paths. Second, VNT transition algorithms compute the control sequence of added/deleted optical paths in order to move from the current state to the new state. In designing a VNT reconfiguration method, we need to consider adequate VNT computation and transition algorithms in order to suit the underlying network conditions and operational policies.

The VNT reconfiguration requires traffic matrix information in computing the optimal VNT, but it is infeasible to collect traffic matrix information of a large-scale network truly due to the N-square problem. To avoid this issue, we introduce a traffic matrix estimation method, which presume the whole traffic matrix information by using the limited (i.e., not a whole) information collected in the network. Several papers have investigated the performance of existing TE methods in conjunction with estimated traffic matrices, and revealed that existing TE methods are less robust and do not work properly due to estimation errors [9, 10].

We thus develop a practical VNT reconfiguration method for optical networks with taking traffic matrix estimation into consideration. We newly introduce a partial VNT reconfiguration algorithm with multiple transition stages. By dividing the whole VNT transition sequence into multiple transitions, estimation errors are calibrated at each stage by using network state information of prior stages. Because estimation errors are mainly due to less-constraint conditions in estimated traffic matrix calculation, our approach tries to increase the constraint conditions for traffic matrix estimation by introducing partial reconfiguration, and to relax the impact of estimation errors by limiting the number of optical-paths reconfigured at each stage.

In this paper we propose two estimation methods: First, we utilize the variation of traffic between before and after the reconfiguration. It increases the number of equations in calculating estimated traffic. However, it may decrease estimation accuracy due to traffic variations between the previous stage and the current stage. To cope with this issue, we second extend the first one to be more robust against the time-dependent variation of the traffic. Furthermore, we investigate the effectiveness of our proposal through extensive simulations and demonstrate the robustness of the proposed control method.

The rest of this paper is organized as follows; Section 2 proposes a VNT control algorithm with traffic matrix estimation. Next, we introduce simulations that demonstrate the limitations of the conventional method and evaluate the performance of our algorithm in Section 3. Finally, a brief conclusion is provided in Section 4.

2. Partial reconfiguration method with traffic matrix estimation

2.1 Design Principle and Method Overview

Our goal is to provide robust and scalable VNT reconfiguration methods based on traffic matrix estimation; performance degradation due to estimation errors is minimized and an adequately level of resource usage is realized. The objective function of our VNT reconfiguration algorithm is to minimize resource consumption for a given traffic matrix while avoiding excessive network congestion. Here, network congestion is measured by the maximum link utilization in the network, thus maximum link utilization should always be held below a given upper bound (e.g., 80% of link bandwidth).

The main challenges are as follows; i) if we deploy the full reconfiguration approach, errors in a few elements can severely degrade the performance of the whole network according to [10]; and ii) it is extremely hard to eliminate estimation errors, therefore we need to estimate traffic matrices as accurate as possible to perform sophisticated VNT reconfiguration.

To solve these issues, we deploy the partial reconfiguration approach with multiple transition stages. This tactic provides robust and safe network control by reducing the negative impact of estimation errors. By introducing partial reconfiguration, network congestion caused by estimation errors will be limited to the small portion of the network. In addition, existing traffic matrix estimation methods use only instantaneous traffic information, which inevitably contains estimation errors because of the fewer constraint conditions imposed on the estimation. To avoid this issue, the proposed reconfiguration approach divides the whole VNT transition sequence into multiple transition stages and estimation errors are calibrated at each stage by using network state information of prior stages.

In this paper, we assume two models. In the *basic model*, the traffic demand of each source-destination pair was assumed to be constant within the reconfiguration period (Fig. 1). If the reconfiguration period is short enough, changes in traffic demand can be negligible. Under the assumption that the traffic demands do not vary during the reconfiguration period, we improve the estimation accuracy by using the whole information from the beginning of the reconfiguration period. We also consider time-varying traffic within the same reconfiguration period as shown in Fig. 2 as the general form of the basic model. We refer to this as the *extended model* in this paper. Unlike the basic model, the extended model performs both traffic matrix estimation and VNT reconfiguration periodically to follow changes in traffic demands. For traffic matrix estimation, we use the traffic measurements and routing information from the last m stages to improve the estimation accuracy.

2.2 A Heuristic Algorithm for Partial Reconfiguration

We propose a heuristic algorithm for calculating the VNT, which is simple enough to adopt the large-scale ISP backbones. The heuristic algorithm consists of two phases: i) addition phase and ii) deletion phase. It adds new optical paths to mitigate congestion and deletes existing underutilized optical paths if possible for reclamation. Note that the algorithm below is applied to both of basic and extended models.

In adding / deleting optical paths, we consider minimizing the total amount of IP router ports used for accommodating the given traffic. IP router is much more expensive than OXC systems. Multiple new optical paths may contend for the same resource of wavelength links or router ports before some path candidates are deleted. To avoid this situation, the heuristic algorithm adds new optical path candidates first without relying on resources freed up by deleting optical paths, and then removes existing underutilized optical paths.

The heuristic algorithm uses three parameters to define the congested or underutilized state of each optical path; T_H and T_L denote thresholds for path congestion and underutilization, respectively,

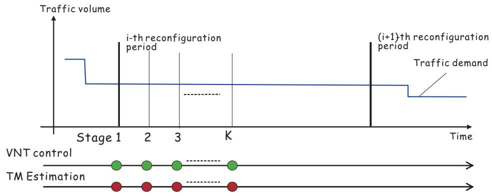


Fig. 1 Basic Model: We assume that the traffic demand of each source-destination pair is constant within the reconfiguration period.

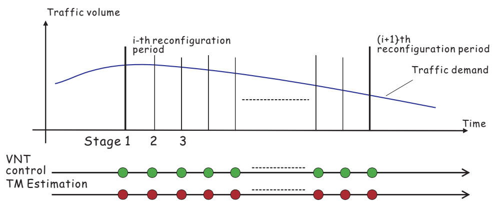


Fig. 2 Extended Model: We assume that the traffic demand of each source-destination pair shows time-variation regardless of the reconfiguration period.

and T_M denotes a threshold for moderately loaded optical paths. We consider that the network is moderately used (i.e., there is no path to be deleted) if the utilization of at least one optical path exceeds T_M. The general sequence of the heuristic algorithm is as follows:

Step 1 First we check the utilization of all optical paths. If at least one congested optical paths is found, go to the optical path addition phase (Step 2). Else if the utilization of an optical path is less than T_L and maximum load of optical paths is less than T_M, then go to *Step 3*.

Step 2 Perform the optical path addition phase of the algorithm, which is described below, then go to *Step 4*.

Step 3 Perform the optical path deletion phase of the algorithm, which is described below, then go to *Step 4*.

Step 4 The number of controlled paths is updated, $Nr = Nr - 1$. If $Nr > 1$, go to *Step 1*. Otherwise, go to *End*.

End

The detail of the heuristic algorithm is described as follows.

(1) *Optical path addition phase:*
If the utilization rate of an optical path exceeds T_H, a new optical path is set up to reroute traffic away from the congested optical path. The ingress and egress nodes of the newly added optical path are selected from those of a packet path accommodated by the congested optical path, and then the packet path is rerouted via the newly added optical path. If there are more than one packet path in the congested path, the highest-rate packet path with multiple hops is selected. In this way, we can efficiently avoid network congestion by adding optical paths.

(2) *Optical path deletion phase:*
If the maximum load is less than T_M ($\leqq T_H$), network resources are used to a moderate level. However, if the utilization of an optical path is less than T_L and the maximum load of each optical path is less than T_M, it is torn down for resource reclamation if possible in oder to improve resource efficiency. To ensure that congestion does not occur after the deletion of the optical path, we add the condition regarding T_M. If there is more than one candidate of deleted optical paths, each candidate optical path is tested in decreasing order of the utilization rate, so the most under-utilized optical path is selected among the candidates. In this way, we can adequately select removal candidate and delete optical paths without causing new congestion.

2.3 Traffic Models and TM Estimation Method

In this section, we propose a new estimation method which collaborates closely with partial VNT reconfiguration. As described in Subsection 2.1, the objective of our TM estimation method is to increase the accuracy of estimation by effectively utilizing the feedback information from the results of the previous stage in partial VNT reconfiguration.

Suppose that the estimated traffic matrix at the current stage has a number of estimation errors. As the result of VNT reconfiguration, some optical path additions/deletions are performed. Under the assumption that the traffic demand of each source-destination (i.e., true traffic matrix) is constant over the period from before to after the reconfiguration, we focus on two observations in order to improve the accuracy of the estimation.

(1) *Differences of link utilization rates between before and after VNT reconfiguration:* Given a fixed traffic matrix, the addition/deletion of an optical path directly impacts the utilization rate of links that carry the optical path. That is, if the measured utilization rate of an optical link changes with VNT reconfiguration, the difference is caused by optical path addition/deletion. This information can yield additional equations for solving the traffic matrix calculation.

(2) *Difference in link utilization rates between estimated and measured rates:* Since VNT reconfiguration is performed by using traffic matrices that essentially have estimation errors, the estimated utilization of optical links may be altered by the errors. After reconfiguration, there should be some difference between the estimated and measured link utilization rates. This information is used as feedback to subsequent VNT reconfiguration.

In the proposed method, we consider both pieces of information above in order to improve estimation accuracy. As described before, we consider two (i.e., basic and extended) models. The following subsections introduce our proposed TM estimation methods for basic and extended models.

2.3.1 Basic model

We denote the actual traffic matrix, which is considered to be constant during the reconfiguration period, by T. At Stage i, we denote the routing matrix and the matrix of link utilization rate by A_i and X_i, respectively.

Because link utilization is the sum of the traffic demands for the source-destination pairs using the links, we have

$$X_i = A_i T. \tag{1}$$

In the basic model, we assume that the traffic demand is constant throughout the reconfiguration period. At Stage n, therefore, by combining all relations from X_0 to X_n, we also have

$$\begin{bmatrix} X_0 \\ \vdots \\ X_i \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} A_0 \\ \vdots \\ A_i \\ \vdots \\ A_n \end{bmatrix} T. \tag{2}$$

The calculation of T from Eq. (2) requires huge computational overhead. However, partial reconfiguration changes only a small number of routes at each stage. That is, most entries of $\Delta A_i = A_i - A_{i-1}$ are to be expected to be 0. By applying this phenomenon, we introduce following procedure to reduce the size of the matrix. First, we transform X_i and A_i into the matrices,

$$\bar{X}_n = \begin{bmatrix} X_1 - X_0 \\ \vdots \\ X_i - X_{i-1} \\ \vdots \\ X_n - X_{n-1} \\ X_n \end{bmatrix} \tag{3}$$

and

$$\Delta\bar{A}_n = \begin{bmatrix} \Delta A_1 \\ \vdots \\ \Delta A_i \\ \vdots \\ \Delta A_n \\ A_n \end{bmatrix}. \tag{4}$$

Second, we remove rows ΔA_i in $\Delta\bar{A}_n$ if $\Delta A_i = O$. We also remove the same position of the row $X_i - X_{i-1}$ in $\bar{X}_n$. We denote $\bar{X}_n$ and $\Delta\bar{A}_n$ after row removal by $\bar{X}'_n$ and $\Delta\bar{A}'_n$, respectively. Finally, we use

$$\bar{X}'_n = \Delta\bar{A}'_n T, \tag{5}$$

instead of Eq. (2).

To estimate the traffic matrix $\hat{T}$ from Eq. (5), we apply the pseudoinverse calculation method described in [11]. The traffic matrix $\hat{T}$ is obtained from

$$\hat{T} = \mathrm{pinv}(\Delta\bar{A}'_n)\bar{X}'_n, \tag{6}$$

where $\mathrm{pinv}(\Delta\bar{A}'_n)$ is the pseudoinverse of matrix $\Delta\bar{A}'_n$. If we simply apply [11] to calculate the pseudoinverse of $\Delta\bar{A}'_n$, some entries in $\hat{T}$ may have negative values, which are nonexistent as regards the traffic matrix. The following iteration eliminates such negative values. We define the estimated traffic matrix for the i-th iteration as $\hat{T}^{(i)}$.

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

（2） Calculate $\hat{T}^{(i)}$ from $\hat{T}^{(i-1)}$ by using

$$\hat{T}^{(i)} = \hat{T}^{(i-1)} + \mathrm{pinv}(\Delta\bar{A}'_n)(\bar{X}'_n - \Delta\bar{A}'_n\hat{T}'^{(i-1)}), \tag{7}$$

where $\hat{T}'^{(i)}$ is a matrix in which we replace all negative values of $\hat{T}^{(i)}$ with zero.

（3） If all entries in $\hat{T}^{(i)}$ are non-negative values, go to Step 4, else back to Step 2.

（4） Let $\hat{T}^{(i)}$ be the final result of traffic matrix $\hat{T}$

We discuss here why the above method offers better accuracy. As discussed before, we use two types of information to improve the estimation accuracy. In the basic model, both are included in Eq. (2). More specifically, we use the constant value of T for all stages to increase the number of relations in Eq. (2). Also, Eq. (7) offers an adjustment by keeping Eqs. (1) and (5). Though the errors are included in estimated traffic matrix $\hat{T}_i$, they do not appear if the routing is not changed, because the estimated traffic matrix and link utilization rates even satisfy Eqs. (1) and (5). However, because routing changes are created by reconfiguration, the impact of estimation error on the new paths appears in Eq. (5).

2.3.2 Extended model

In the extended model, we consider the time-dependent variation of traffic demands to estimate the traffic matrix more accurately. Moreover, we can reconfigure the VNT adaptively to meet the current traffic demand at each stage.

We denote the actual traffic matrix at Stage i as T_i and the difference of the traffic matrices between Stages $i-1$ and i as $\Delta T_i = T_i - T_{i-1}$. Unlike the basic model, both VNT reconfiguration and traffic matrix estimation are performed consistently. Therefore, in extended model, we use the link utilization rates monitored at last m stages.

The actual traffic matrix also changes at each stage, so we use

$$X_i = A_i T_i \tag{8}$$

instead of Eq. (1). By using $\Delta T_i = T_i - T_{i-1}$, we can derive the relation between T_i and T_n as

$$T_i = T_n - \sum_{k=i+1}^{n} \Delta T_k. \tag{9}$$

From Eqs. (8) and (9), the relation between X_i and T_n is given by

$$X_i = A_i T_n - A_i \sum_{k=i+1}^{n} \Delta T_k. \tag{10}$$

By using Eq. (10), we can derive the following equation in the same way as Eq. (5).

$$\bar{X}_n = \Delta\bar{A}_n T_n - W_n \tag{11}$$

where

$$\bar{X}_n = \begin{bmatrix} X_{n-m+2} - X_{n-m+1} \\ \vdots \\ X_j - X_{j-1} \\ \vdots \\ X_n - X_{n-1} \\ X_n \end{bmatrix}, \tag{12}$$

$$\Delta\bar{A}_n = \begin{bmatrix} \Delta A_{n-m+2} \\ \vdots \\ \Delta A_j \\ \vdots \\ \Delta A_n \\ A_n \end{bmatrix} \tag{13}$$

and

$$W_n = \begin{bmatrix} \Delta A_{n-m+2} \sum_{k=n-m+3}^{n} \Delta T_k - A_{n-m+2}\Delta T_{n-m+2} \\ \vdots \\ \Delta A_j \sum_{k=j+1}^{n} \Delta T_k - A_j \Delta T_j \\ \vdots \\ \Delta A_n \Delta T_n - A_n \Delta T_n \\ 0 \end{bmatrix}. \tag{14}$$

In Eq. (14), W_n cannot be measured directly. Our solution is to apply Gauss-Markov estimation, which can solve simultaneous equations including unobservable noise by weighted fitting. We use Gauss-Markov estimation to determine the traffic matrix from

$$\hat{T}_n = (\bar{A}_n^T B_n^{-1} \bar{A}_n)^{-1} \bar{A}_n^T B_n^{-1} \bar{X}_n \tag{15}$$

where B_n is the covariance matrix of W_n, i.e.,

$$B_n = E[W_n W_n^T]. \tag{16}$$

B_n can be calculate from the covariance matrix of $\Delta\bar{T}_n$. According to [12], the variance of traffic demands can be modeled by

$$\mathrm{Var}(t) = \phi \mathrm{Avg}(t)^c \tag{17}$$

where ϕ and c are constant parameters. Therefore, we set value $\mathrm{Var}(\Delta T)$ from the estimated traffic matrix at the previous stage by using Eq. (17). Note that the value of parameter ϕ doesn 't affect the estimation results because Gauss-Markov estimation use not the values of the variances themselves, only relative values of the variances. Therefore, we set ϕ to 1 to simplify the calculation. We set parameter c to 1.5 following the fitting results described in [12].

Eq. (11) includes only the differences in link utilization rates due to VNT reconfiguration. As described before, we can also use the differences between estimated and measured link utilization rates. In what follows, we describe how we consider this difference. Unlike the basic model, the influence of this difference is not included

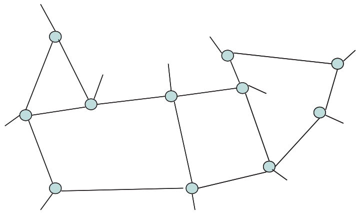


Fig. 3 Abilene topology

in Eq. (11) because the equations used in estimating the traffic matrix at the previous stage are not reflected in Eq. (11). Hence we need a method that can adjust the difference independently from above equations. The difference between the true traffic at Stage i and the estimated traffic at Stage $i-1$ can be considered as the summation of estimation errors at Stage $i-1$ and traffic variation from Stage $i-1$, i.e.,

$$T_i - \hat{T}_{i-1} = E_{i-1} + \Delta T_i. \tag{18}$$

where $\hat{T}_{i-1}$ is the estimated traffic matrix at stage $i-1$ and E_{i-1} is the estimation errors included in $\hat{T}_{i-1}$.

From Eqs. (18) and (8), the differences between the estimated and measured link utilization rates at Stage i are given by

$$X_i - A_i\hat{T}_{i-1} = A(E_{i-1} + \Delta T_i). \tag{19}$$

From Eqs. (18) and (19), we can estimate the traffic matrix at Stage i in the following steps.

(1) Estimate $\hat{D}_i = E_{i-1} + \Delta T_i$ satisfying Eq. (19) using

$$\hat{D}_i = \mathrm{pinv}(A_i)(X_i - A_i\hat{T}_{i-1}). \tag{20}$$

(2) By using $\hat{D}_i$, we can estimate the traffic matrix $\hat{T}_i$ that satisfies Eq. (18) as

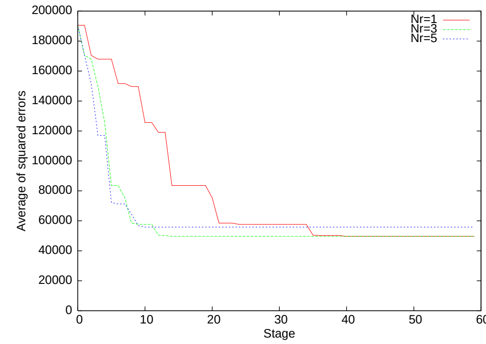
$$\hat{T}_i = \hat{T}_{i-1} + \hat{D}_i. \tag{21}$$

3. Performance Evaluation

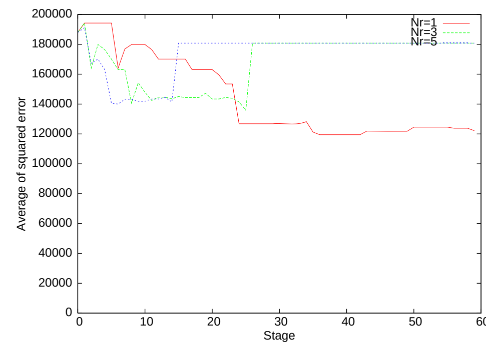
To demonstrate the effectiveness of the partial reconfiguration method with traffic matrix estimation under actual traffic matrices, we perform extensive simulations. In this simulation, we use two types of traffic models; the basic model and extended model as described in Section 2.3.

Other simulation conditions used in our performance evaluation is as follows; we use the Abilene backnone as the network topology as shown in Fig. 3. The number of wavelengthes for each optical link is set to be eight. For true traffic, we adopt real data collected from the Abilene backbone network [13]. However, the actual traffic demand does not make a significant impact to VNT reconfiguration due to low variations of the actual traffic. In order to verify the effect of VNT reconfiguration, we scale the monitored traffic enough to make most of links to be congested. Furthermore, we add 1 Gbps of traffic to four randomly picked up source-destination pairs to cause significant route changes after VNT reconfiguration. On the simulation, we first create the initial VNT topology by using our VNT reconfiguration algorithm for a given traffic demand, we then change the traffic to above-mentioned one to evaluate how our proposed method works.

For performance measures, we evaluate followings: i) average of squared estimation error, and ii) maximum number of output ports additionally required for the VNT reconfiguration. The former is the metric which directly indicates the accuracy of the traffic matrix estimation method. The latter is a useful to consider the network cost, i.e., if the required number of ports is larger, we need more cost to construct the network.



(a) Our method



(b) Tomogravity

Fig. 4 Average squared errors of traffic matrix estimation

3.1 Basic model

We first compare the average squared errors of traffic matrix estimation between tomogravity [14] and our method in Fig. 4, where we set the threshold Nr to be 1, 3 and 5. From this figure, we can observe that our method can decrease the estimation error dramatically compared to the tomogravity. This is caused by the difference of number of equations used in the traffic matrix calculation. In the tomogravity model, the accuracy is improved only because the part of actual traffic matrix can be directly measured by setting up additional source-destination paths, while our method can improve the accuracy not only by the direct measurements but also by the increase of the number of equations. However, in some stages, even our method could not decrease the estimation error (e.g., Stages 14–18 for $Nr = 1$). The reason is because our method can improve the accuracy only if at least one route is changed from the previous stage. In cases such as when we set an additional path for the already connected source-destination pair, or when we delete one of multiple paths for the same source-destination pair.

Fig. 4 also shows the effect of the parameter Nr for the traffic matrix estimation. We found from this figure that there is a tradeoff between the convergense time and the accuracy of estimation. That is, when $Nr = 5$ the VNT becomes stable faster (i.e., smaller number of stages) than the case when $Nr = 1$, while the accuracy becomes worse when $Nr = 5$. To improve the accuracy, the smaller Nr is better, however, small value of Nr requires a lot of stages to be converged, which lead a heavy measurement overhead or long time to make the VNT stable. One possible solution is to set Nr as small as possible for given acceptable maximum number of stages for partial reconfiguration which is determined by the traffic monitoring period and preferable duration for the reconfiguration.

We next compare the total number of output ports additionally required for the VNT reconfiguration. Fig 5 shows the variation of difference of the number of paths from the beginning of VNT

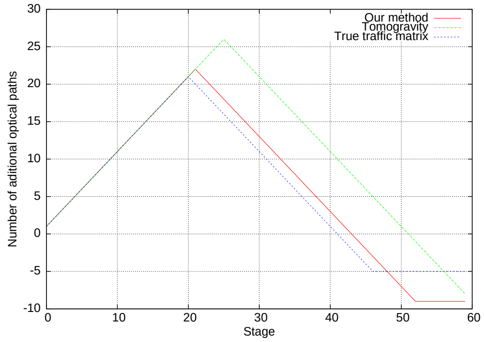


Fig. 5 Variation of difference of the number of paths from the beginning of VNT reconfiguration

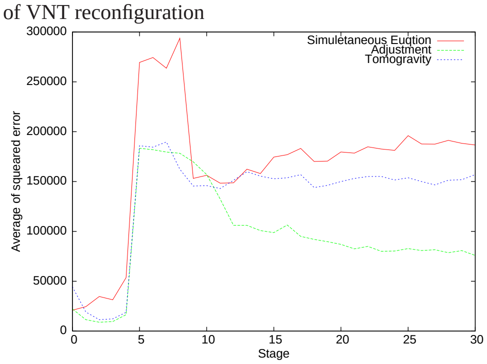


Fig. 6 Time-dependent average of squared estimation errors among tomogravity and our methods

reconfiguration. The number of paths is increasing linearly by passing stages. When the VNT satisfies the condition that the maximum link utilization is less than T_h, the reconfiguration algorithm tries to delete as many unused paths as possible. The number of paths is therefore decreasing linearly. From this figure, more output ports are required when we use the tomogravity due to larger estimation errors. Also, tomogravity requires more stages to reach the stable VNT. It is because the VNT reconfiguration algorithm is required to delete more paths which are unnecessarily added caused by estimation errors.

3.2 Extended model

To evaluate the extended model, we simulate our methods with time-dependent changes in traffic demands. We use the traffic data recorded by NetFlow with a 5 minutes interval on the Abilene backbone network for the time-dependent change in traffic. However, this traffic data includes only gradual changes though the VNT reconfiguration is needed especially when the traffic changes significantly. Therefore we evaluate our method by adding the significant change to the traffic data.

Fig. 6 compares the time-dependent average of squared estimation errors among tomogravity and our methods. In this figure, "Simultaneous Equation" uses the traffic matrix estimated by Eq. (11), and "Adjustment" uses Eqs. (20) and (21) for the estimation. In this simulation, we add the significant change into traffic demands at 5-th stage. Moments after the significant change of traffic, estimation errors of all method become large. However, unlike tomogravity, both "Simultaneous Equation" and "Adjustment" methods can decrease the estimation error a few stages after the change. We can also see that our method of extended model especially "Adjustment" can decrease the estimation errors significantly as the stage goes on while the estimation errors of tomogravity method remain high. As a result, a few stages after the significant change in traffic, the estimation errors of our method become smaller than those of tomogravity method. This is because our method can use more informations monitored at the stages after the significant change in traffic as the stage goes on.

4. Concluding Remarks

In this paper, we proposed a practical VNT reconfiguration method with traffic matrix estimation considerations. We presented the heuristic algorithms for partial reconfiguration. The key technology lies in a partial VNT reconfiguration algorithm with multiple transition stages. By introducing the partial reconfiguration, network congestion caused by estimation errors can be limited to small portion of the network. We also investigated the effectiveness of our proposal through extensive simulations and clarified the robustness of the proposed control method under estimated traffic.

Acknowledgement

We are highly indebted to all the people that allowed the Abilene Netflow data to be available on-line, and especially Mark Fullmer for his help.

References

[1] B. Mukherjee, D. Banerjee, S. Ramamurthy, and A. Mukherjee, "Some principles for designing a wide-area WDM optical network," *IEEE/ACM Transactions on Networking*, vol. 4, pp. 684–696, Oct. 1996.

[2] D. Banerjee and B. Mukherjee, "Wavelength-routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study," *IEEE/ACM Transactions on Networking*, vol. 8, pp. 598–607, Oct. 2000.

[3] J. Wei, C.-D. Liu, S.-Y. Park, K. Liu, R. Ramamurthy, H. Kim, and M. Maeda, "Network control and management for the next generation Internet," *IEICE Transactions on Communications*, vol. 83-B, pp. 2191–2209, Oct. 2000.

[4] L. Zhang, K. Lee, and C.-H. Youn, "Adaptive virtual topology reconfiguration policy employing multi-stage traffic prediction in optical Internet," in *Proceedings of Workshop on High Performance Switching and Routing*, pp. 26–29, May 2002.

[5] 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.

[6] 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, Oct. 2003.

[7] S. Gieselman, N. Singhal, and B. Mukherjee, "Minimum-cost virtual-topology adaptation for optical WDM mesh networks," in *Proceedings of IEEE ICC*, vol. 3, pp. 1787–1791, June 2005.

[8] R. Hayashi, T. Miyamura, M. Aoki, and S. Urushidani, "Simulation of a dynamic multi-layer optimization algorithm with SRLG consideration," in *Proceedings of OECC/COIN 2004*, July 2004.

[9] M. Roughan, M. Thorup, and Y. Zhang, "Traffic engineering with estimated traffic matrices," in *Proceedings of ACM SIGCOMM Internet Measurement Conference 2003*, pp. 248–258, Oct. 2003.

[10] Y. Ohsita, T. Miyamura, S. Arakawa, S. Ata, E. Oki, K. Shiomoto, S. Urushidani, and M. Murata, "Traffic engineering strategies in IP and optical multi-layer networks," *Submitted for publication*, 2006.

[11] J. C. Nash, *Compact Numerical Methods for Computers: Linear Algebra and Function Minimisation*. Adam Hilger, 1990.

[12] 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 2004*, pp. 149–160, Oct. 2004.

[13] "Abilene network status." available at `http://abilene.internet2.edu/observatory/`.

[14] 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.