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.

大規模ネットワークのための階層型トラヒックエンジニアリング

大下　裕一† 　宮村　崇†† 　荒川　伸一††† 　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
†† 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,kohei.shiomoto}@lab.ntt.co.jp,
†††{arakawa,murata}@ist.osaka-u.ac.jp

あらまし　時間変動が大きなトラヒックを効率的に収容するには、各時刻のトラヒックに合わせて経路を動的に変更するトラヒックエンジニアリング (TE) が有効である。各時刻のトラヒックに合わせてネットワーク全体の経路を変更するためには、経路計算を行う PCE と呼ばれるサーバで、ネットワーク全体のトラヒック情報を収集する必要がある。しかしながら、大規模ネットワークでは、ネットワーク全体のトラヒック情報を頻繁に収集することは困難であり、トラヒック変動によって輻輳等の問題が生じてから経路変更を行うまで時間がかかる。そこで、本稿では、ネットワークを地理的に分割、階層化を行い、頻繁に行うことが可能な局所的な制御と、長い周期で行う広い範囲の制御を組み合わせることにより、トラヒック変動発生後、すばやく、適切な経路に移行する手法を提案する。本稿では、シミュレーションにより、提案手法の有効性を確認する。
キーワード　トラヒックエンジニアリング、階層化、トラヒックマトリクス、Oblivious Routing

Hierarchical dynamic traffic engineering for a large-scale network

Yuichi OHSITA†, Takashi MIYAMURA††, Shin'ichi ARAKAWA†††, Kohei SHIOMOTO††, and
Masayuki MURATA†††

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

Abstract　Traffic engineering (TE) is one efficient way of accommodating traffic that changes unpredictably. The traffic information of the whole network is essential to reconfigure the routes of the whole network. However, it is difficult to collect the traffic information of the whole network in a short period of time. Thus, the reconfiguration of the routes of the whole network cannot be performed in a short period of time. In this paper, we develop a method that can handle the traffic changes in a short time in a large-scale network. In our method, we hierarchically divide the network into several ranges. Our method reconfigures the routes within small ranges in a short period of time to handle the traffic changes that occur in a short period of time. In addition, we also reconfigure the routes of the whole network to handle the significant traffic change that cannot be handled by the reconfiguration within small ranges. In this paper we evaluate our method by simulation and clarify that our method achieves the similar maximum link utilization to the method using traffic information of the whole network.
Key words　Traffic engineering, Hierarchization, Traffic matrix, Oblivious Routing

1. はじめに

近年、Peer-to-Peer、Video-on-Demand、SaaS や PaaS などのネットワークを介した様々なアプリケーションが普及するにつれ、ネットワークを流れるトラヒックの時間変動が大きくなっている。ネットワークの管理者は、大きなトラヒック変動が生じた場合でも、輻輳を生じることなく、全トラヒックを収容する必要がある。経路変更を行うことなく、起こりうるすべての

トラヒック変動に対応するような経路を設計することもできる [1] ものの、各時刻のトラヒックに合わせて経路設計をした場合の 2 倍以上の帯域が必要となる。

トラヒック変動に効率的に対応する方法としては、ネットワーク内の経路を、各時刻のトラヒックに合わせて動的に変更することが有効である。ネットワーク内の経路を動的に変更する手法は、トラヒックエンジニアリング (TE) と呼ばれ、様々な手法が提案されている [2–5]。これらの手法では、ネットワーク内の経路を管理する Path Computation Element (PCE) と呼ばれるサーバを配置する。PCE は、ネットワーク内の各ノードからトラヒック情報を収集、収集した各地点間を流れるトラヒック量を輻輳なく収容できるように、ネットワーク内の経路を再設計する。しかしながら、大規模ネットワークでは、PCE が収集をしなければならない情報量、PCE が問い合わせを行う必要がある機器数が著しく大きいため、ネットワーク内の全地点間を流れるトラヒック量に関する情報を頻繁に収集することは困難である。そのため、分単位で発生するようなトラヒック変動 [6] に合わせた経路変更を行うことができず、トラヒック変動による輻輳発生後、輻輳の解消まで時間がかかってしまう。

そこで、本稿では、大規模ネットワークにおいて、分単位といった短いタイムスケールで発生するトラヒック変動にも対応し、輻輳を回避することができる階層型 TE 手法を提案する。提案手法では、最下位層では狭い範囲、上位層は下位層の複数の範囲を束ねたより広い範囲となるように、制御対象のネットワークを階層的に分割する。各範囲には、対応する PCE を配置する。各 PCE は、上位層の PCE・下位層の PCE と集約した情報を交換することにより、自身が制御する対象の範囲内のトラヒック状況を把握する。そして、把握したトラヒック状況に応じて、制御対象の範囲を経由するトラヒックの経路を変更することにより、制御対象の範囲内で発生した問題を解消する。提案する階層型 TE 手法では、局所的な制御であり、制御の影響を受ける範囲が小さい下位層の制御は、頻繁に行うことが可能である。そのため、トラヒック変動等により輻輳等の問題が発生した際は、すばやく問題を解消することができる。また、下位層の局所的な制御のみでは十分に問題を解消することができない場合であっても、上位層の制御を行うことにより、問題を解消することができる。

以降、2. では、提案する階層型 TE の概要、3. で階層型 TE で用いる集約情報の作成方法、集約情報を用いた経路設計方法を述べる。そして、4. で提案手法を評価し、5. でまとめを述べる。

2. 階層型トラヒックエンジニアリングの概要

提案手法では、ネットワークを地理的に分割した範囲を階層的に構築する。提案手法における階層化の概要を図 1 に示す。図 1 に示すように、最下位層では、ノードを境界とし、全リンクがいずれかの範囲に含まれるように、ネットワークを地理的に複数の範囲に分割する。範囲の境界に位置するノードを以降、境界ノードと呼ぶ。最下位層の範囲に対して、対応する PCE を設置する。最下位層の各 PCE は、定期的に自身の制御対象の範囲内に存在するノードに問い合わせを行い、制御対象の範囲内の全リンクのトラヒック状況を把握する。

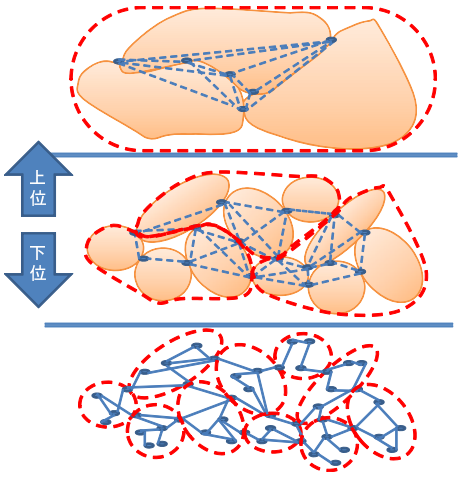

図 1　大規模ネットワークの階層化の概要

上位層の範囲は、下位層の範囲を複数を束ねて一つの範囲とすることで構成され、最下位層と同様、各範囲に対応する PCE を配置する。上位層の PCE では、下位層では境界ノードとなっていたノードのみからなるトポロジを扱い、そのノード間の輻輳状態に関する集約情報を下位層の PCE から受け取ることにより、境界ノード間の輻輳状態を把握する。

各 PCE は、直接観測した情報や下位層から取得した集約された情報をもとに、自身の制御対象の範囲内で、輻輳等の問題の有無を検知する。そして、自身の制御対象の範囲内を経由しているトラヒックの制御対象の範囲内の経路や、制御対象の範囲内が始点または終点となるトラヒックの出入口となる境界ノードを変更することにより、検出された問題の解消を図る。

経路の移設を行う際には、各 PCE は、自身の制御対象の範囲内のみではなく、自身の制御範囲内の境界ノードと、他の範囲の境界ノード間の輻輳状態に関する情報を上位層の PCE から取得する。他の範囲の境界ノードと自身の境界ノード間の輻輳状態を把握することにより、他の範囲で新たな輻輳を発生させることを回避させつつ、制御対象範囲内が始点または終点となるトラヒックの出入口となる境界ノードの変更を行うことができる。

提案手法では、各 PCE が収集・交換するトラヒック情報は、局所的なもの、あるいは、集約されたもののみである。そのため、収集・交換が必要な情報量が少なく、頻繁な情報交換が可能であり、各 PCE は、短いタイムスケールのトラヒック変動にも追随して、トラヒック状況を把握することができる。また、下位層の PCE での経路変更は局所的であり、経路変更の影響を受ける範囲が小さいため、頻繁な経路変更が可能である。そのため、提案手法では、下位層の制御周期を短くすることにより、輻輳等の問題に素早く対応することができる。

具体的なトラヒック情報の集約方法と、集約情報を用いた経路変更方法は、3. で述べる。

3. トラヒック情報集約と集約した情報を用いたトラヒックエンジニアリング手法

本節では、リンク使用率が閾値を超える状態を輻輳とし、輻

表 1　リンク l に関する集約情報

記号	説明
b_l	帯域
x_l^{all}	リンク上のトラヒック量
$x_l^{\max}$	リンク上のトラヒック量のうち上位層で制御可能なトラヒック量の上限
$x_l^{\min}$	リンク上のトラヒック量のうち上位層で制御可能なトラヒック量の下限
P_l	リンク l を経由する境界ノードペアの集合
$f_{p,l}^{\mathrm{lower}}$	境界ノードペア間のトラヒック $p \in P_l$ のうち、リンク l を経由する割合

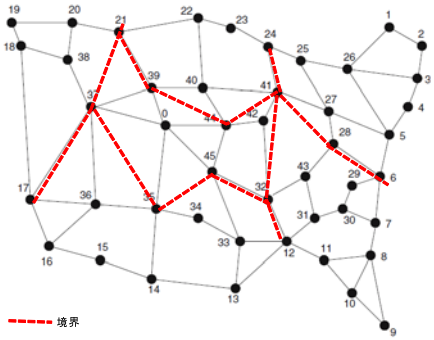

図 2　US topology (2 layers)

輻輳を回避することをトラヒックエンジニアリングの目的とする。以降、この目的を達成するための、階層型トラヒックエンジニアリングにおける、トラヒック情報の集約手法、集約したトラヒック情報を用いて経路変更を行う手法について述べる。

3.1　トラヒック情報の集約手法

3.1.1　上位層への集約情報

提案手法では、各境界ノード間のトラヒックが経由するリンクのうち、最もリンク使用率の高いリンクに関する情報を集約情報として、上位層の PCE に渡す。これにより、上位層の PCE は、集約情報のみから、各境界ノード間の輻輳の有無を把握することができる。

また、集約情報として含める各リンク l について、表 1 の情報を集約情報として上位層の PCE に渡す。表 1 の情報のうち、$x_l^{\max}$、$x_l^{\min}$ は、付録の式 (A·1) の最適化問題を解くことにより得ることができる。これらの情報を用いることにより、上位層では、リンク l を経由するトラヒックの特定、経路変更後のリンク l の負荷の最大値を求めることができる。

3.1.2　下位層への集約情報

上位層から下位層には、情報送信先の PCE が制御を行う範囲内の各境界ノードと別の境界ノード間の輻輳状態に関する集約情報を、3.1.1 と同様の手順で生成して渡す。この情報を用いることにより、下位層の制御で、範囲内からのトラヒックの出入口となる境界ノードを変更した際にも、他の範囲で新たな輻輳を防ぐことができる。

3.2　集約されたトラヒック情報を用いた経路設計手法

提案手法では、輻輳発生を検出した後、輻輳箇所を経由している各トラヒックを移設することにより、輻輳の解消を試みる。各トラヒックの移設先の経路は、PCE が把握しているトポロジ G とトラヒックの集約情報をもとに、以下の手順で確定する。

手順 1　トポロジ G 上で、最短ホップ経路で経路を定める。

手順 2　手順 1 で計算された経路上にある各リンクについて、リンク使用率の取りうる上限を集約情報から計算する。

手順 3　リンク使用率の上限が閾値を超えるリンクが存在した場合、閾値を超えるリンクをトポロジ G から削除した上で、手順 1 へ戻る。リンク使用率の上限が閾値を超えるリンクが存在しない場合は、経路を確定する。

上記の手順 2 の、リンク使用率の取りうる上限は、付録の式 (A·2) の最適化問題を解くことにより得ることができる。

上記の手順での経路計算は、制御対象の範囲内の経路計算にも、制御範囲内が始点・終点となるトラヒックの制御対象範囲からの出入口を計算する際にも適用することができる。提案手法では、まず、上記の手順で制御対象範囲内のトラヒックの経路変更を試み、制御対象範囲内の経路変更のみでは輻輳を十分に回避できない場合に、制御対象範囲からの出入口となる境界ノードの変更を行う。

4.　評　　価

本節では、提案手法の有効性について評価する。本評価では図 2 に示されるように US topology (46 ノード、70 リンク）を 6 つの範囲に分け、提案手法により、各範囲内の制御と、ネットワーク全体の制御の 2 階層の制御を行った。経路制御前の経路は、[7] の結果に基づき対数正規分布に従って生成した対地間トラヒックが流れる環境において、リンク使用率を最小化するように設定した。その後、対地間トラヒック量を新たに対数正規分布に従う乱数として生成し、著しいトラヒック変動が発生した環境を生成した。この環境において、下位層の経路変更を 1 分間に 1 回、上位層の経路変更を 13 分に一回動作させた場合について、評価を行った。

4.1　達成可能なリンク使用率

まず、輻輳とみなすリンク使用率を変化させながら、30 分以内の制御で達成可能なリンク使用率を評価した。本評価では、70 種類のトラヒックを生成した。図 3 に結果を示す。図中には、2 階層の制御を行った場合、下位層のみの制御を行った場合、ネットワーク全体の情報を用いて制御を行った場合に達成可能なリンク使用率と、経路変更前の最大リンク使用率の分布を示している。

図 3 より、多くの場合は、下位層のみの制御であっても、経路変更前と比べ、最大リンク使用率を著しく削減することができることがわかる。つまり、多くの場合は局所的な経路変更のみで、輻輳を解消することができる。また、図 3 より、下位層の TE のみでは、最大リンク使用率を十分に削減できない場合であっても、上位層の制御を加えることにより、ネットワーク全体の情報を用いた制御と同程度までリンク使用率を削減することができることがわかる。

4.2　リンク使用率を閾値以下に削減するまでにかかる時間

次に、輻輳とみなすリンク使用率の閾値を 0.4 とし、経路変更前のリンク使用率が 0.4 を超えた場合について、リンク使用率を 0.4 以下まで削減するまでにかかる時間を調べた。図 4 に、

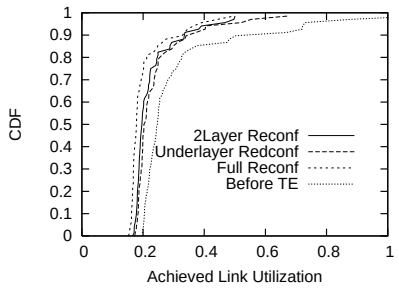

図 3　達成可能なリンク使用率の分布

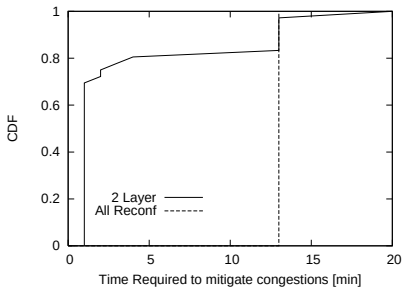

図 4　リンク使用率を 0.4 以下にするまでにかかる時間

リンク使用率を 0.4 以下にするまでにかかる時間の分布を示す。また、図中では、上位層と同じく 13 分に 1 回の頻度で、ネットワーク全体の情報を用いた TE を行った場合に、リンク使用率を 0.4 以下にするまでにかかる時間の分布も示す。

図 4 より、8 割以上の箇所で、ネットワーク全体の情報を用いた TE よりも早くリンク使用率を 0.4 以下まで削減することができることがわかる。これは、図 3 にも示されているように、多くの場合は、下位層の経路変更のみでも、十分にリンク使用率を削減することができるためである。つまり、階層型 TE を行うことにより、トラヒック変動が発生後、短い時間で輻輳を回避させることができる。

5. まとめと今後の課題

本稿では、ネットワークを地理的に分割、階層化した動的な経路制御を行うことにより、素早く、輻輳を解消することができる手法を提案した。また、シミュレーションにより提案手法の有効性を確認した。今後の課題としては、階層数を増やした場合の提案手法の評価があげられる。

謝辞　本研究の一部は、文部科学省科学研究費補助金基盤研究 (B) 22300023 および、若手研究 (B)21700074 によっている。ここに記して謝意を表す。

付　　録

以下に提案手法内で用いる最適化問題について記載する。以下の最適化問題を PCE で解くに当たり、PCE が把握している情報を表 A·1 に示す。また、以下の最適化問題おける変数を表 A·2 に示す。

PCE では、以下の最適化問題を解くことにより、集約情報に含めるリンク l を経由するトラヒックのうち、上位層で制御可能なトラヒックの上限 $x_l^{\max}$ を求めることができる。また、目的関数を最小化とすることにより、下限 $x_l^{\min}$ を求めることができる。

表 A·1　最適化問題の入力

記号	説明
P^{current}	制御対象の範囲内のノードペアの集合
P^{upper}	上位でトラヒックを制御可能なノードペアの集合
L	PCE が集約情報によりトラヒック量を把握しているリンクの集合
$f_{p,l}$	ノードペア $p \in P^{\text{current}}$ 間のトラヒックがリンク $l \in L$ 間を経由する割合
b_l、x_l^{all}、$x_l^{\max}$、$x_l^{\min}$、P_l、$f_{s,d,l}^{\text{lower}}$	集合 L に含まれるリンクのトラヒックに関する集約情報

表 A·2　最適化問題の変数

変数名	説明
t_p	ノードペア p 間のトラヒック量。非負の変数。
t_l^{loewer}	リンク l 上の現在の階層では制御できないトラヒック量。非負の変数。

できる。

$$\text{maximize} \quad \sum_{p \in P^{\text{upper}}} f_{p,l} t_p \tag{A·1}$$
$$s.t \quad \forall l \in L\text{: } x_l^{\min} \leqq \sum_{p \in P^{\text{current}}} f_{p,l} t_p \leqq x_l^{\max}$$

経路制御を行う際には、経路変更後にノードペア p のトラヒックがリンク l を経由する割合 $f_{p,l}^{\text{new}}$ を入力として与え、以下の最適化問題を解くことにより、リンク l のリンク使用率の取りうる最大値を求める。

$$\text{maximize} \quad \frac{1}{b_l}\left(\sum_{p \in P^{\text{current}}} f_{p,l}^{\text{new}} t_p + t_l^{\text{loewer}}\right) \tag{A·2}$$
$$s.t \quad \forall l \in L\text{: } x_l^{\min} \leqq \sum_{p \in P^{\text{current}}} f_{p,l} t_p \leqq x_l^{\max}$$
$$\forall l \in L\text{: } \sum_{p \in P^{\text{current}}} f_{p,l} t_p + t_l^{\text{loewer}} = x_l^{\text{all}}$$

文　　献

[1] D. Applegate and E. Cohen, “Making routing robust to changing traffic demands: algorithms and evaluation,” *IEEE/ACM Transactions on Networking*, vol. 14, pp. 1193–1206, Dec. 2006.

[2] B. Fortz and M. Thorup, “Internet traffic engineering by optimizing OSPF weights,” in *Proceedings of IEEE INFOCOM*, vol. 2, pp. 519–528, Mar. 2000.

[3] 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,” *IEEE/ACM Transactions on Networking*, vol. 18, pp. 177–189, Feb. 2010.

[4] I. Juva, “Robust load balancing,” in *Proceedings of GLOBECOM*, pp. 2708–2713, Nov. 2007.

[5] G. Retvari and G. Nemeth, “Demand-oblivious routing: distributed vs. centralized approaches,” in *Proceedings of IEEE INFOCOM*, pp. 1217–1225, Mar. 2010.

[6] R. Teixeira, N. Duffield, J. Rexford, and M. Roughan, “Traffic matrix reloaded: Impact of routing changes,” in *Proceedings of Passive and Active Network Measurement*, pp. 251–264, Mar. 2005.

[7] A. Nucci, A. Sridharan, and N. Taft, “The problem of synthetically generating IP traffic matrices: Initial recommendations,” *ACM SIGCOMM Computer Communication Review*, vol. 35, pp. 19–32, July 2005.