

# サーバー負荷情報に基づく OpenFlow を用いた経路分散方式の提案

## Proposal of Route Distribution System Based Server Load Information using OpenFlow

吉村 悠<sup>†</sup>      木村 真乃介<sup>‡</sup>      佐藤 健哉<sup>†</sup>  
Haruka Yoshimura   Shinnosuke Kimura   Kenya Sato

### 1 はじめに

近年, Youtube などに代表される動画共有サイトの普及により, 動画トラフィックが増大しインターネット通信網の多くを占有している. さらに, これらの動画共有サイトは動画の高品質化を推進しており, 今後ますます動画トラフィックが増大しネットワークの輻輳が起りやすくなる事が予想される. 対応策として, トラフィックの伝送経路を分散する事が考えられている. 現在の OSPF 等のルーティングプロトコルを用いる一般的な IP 通信網では, 伝送されるトラフィックが特定のリンクに集中する傾向があり輻輳が起きやすくなる. 経路分散を行う事でネットワークの使用効率を向上できる.

本研究では, サーバからの動画トラフィック伝送において, 動的なネットワークの伝送制御を可能にする OpenFlow を用いた経路分散によりネットワークの輻輳を抑制するシステムを提案する.

### 2 関連研究

#### 2.1 経路分散に必要な機能

複数経路による伝送を実現するために必要な機能として以下の 3 つが挙げられる [1].

##### 複数の経路を発見する機能

経路を分散するためには同一の宛先ノードに対して複数の経路を用意しなければならない.

##### 経路の空き帯域を周期的に収集する機能

伝送する経路を選択するためには各経路の負荷情報を収集し, 各経路の空いている帯域幅を把握しなければならない.

##### 負荷情報から経路を選択する機能

各経路の負荷情報と伝送に必要とする帯域から経路を動的に選択し, その経路にそってトラフィックを伝送しなければならない.

これらの機能を従来の, 経路途中の各々のノードが経路を判断する IP 通信網で実現するのは難しく, ノードに対して多くの機能を付加した複雑な実装となる上, 経路を絞る, 経路を固定するなどの制限が必要になる.

#### 2.2 パケット順序逆転問題

従来の複数経路通信では, パケット単位で経路を分散するため, 同一の TCP フローのパケットも異なる経路で伝送されパケットの到達順序の逆転が頻発してしまう. TCP 通信においてこのパケット順序の逆転が頻発すると, TCP の再送処理などが発生しかえってスルー

プットの減少してしまう. 動画共有サイトでは基本的に TCP 通信が使われるので, このパケット順序の逆転を防ぐ必要がある.

本稿では, ここまでに挙げた経路分散の課題をネットワークの集中管理を可能とする OpenFlow を用いて解決し TCP 通信の経路分散の実現を目指す. その上で経路分散に必要な機能が OpenFlow によってどう実現されるか示す, またパケット順序逆転問題の解決法を示す.

### 3 提案システム

#### 3.1 概要

図 1 に提案システムの概要を示す. OpenFlow を利用し, ネットワークの各経路の帯域情報を把握し, 適切な経路分散を行い, ネットワークの輻輳を抑制する. 次に OpenFlow を用いた, 経路分散に必要な機能の実現方法とフロー単位の経路分散について述べる.

#### 3.2 複数の経路を用意する機能

OpenFlow ではコントローラは各スイッチから経路情報を収集し, ネットワーク全体の構造を把握する事ができる. そこから宛先に対しての複数経路を効率的に見つけることができる.

#### 3.3 負荷情報を周期的に把握する機能

周期的に各スイッチから各リンクに流れているトラフィック量を収集する事で各リンクの空き帯域幅を把握できる.

#### 3.4 負荷情報から経路を選択する機能

OpenFlow は各スイッチにあるパケットをどのリンクに伝送するか動的に設定できるので, 空き帯域幅から経路を決定し, その経路に沿ってパケットを伝送できる.

#### 3.5 TCP フロー単位の経路分散

OpenFlow はコントローラとスイッチからなりコントローラがスイッチに対しフローエントリの追加, 削除, 更新を行う. フローエントリは, 送信元 IP アドレスなどパケットの条件を表すマッチング部, 何番ポートから出力する, 宛先 IP アドレスを書き換えるといったパケットに対する動作を表すアクション部, フローエントリの使用回数などの統計情報を表すカウンタで構成され, スイッチはフローエントリの集合をフローテーブルとして保持する. スイッチがパケットを伝送する時, パケットとマッチするフローエントリのアクションを実行し, カウンタを更新する. 無い場合はコントローラにこのパケット用のフローエントリを要求する. この OpenFlow の伝送制御により同一の TCP フロー毎に経路を設定する TCP フロー単位の経路分散ができる. TCP フロー単位の経路分散では同一の TCP フローのパケットは同一経路で伝送されるので, パケット順序の逆転を抑制する事が可能である. その場合, 同一の TCP フロー内のパ

<sup>†</sup> 同志社大学 理工学部 情報システムデザイン学科

<sup>‡</sup> 同志社大学大学院 理工学研究科 情報工学専攻

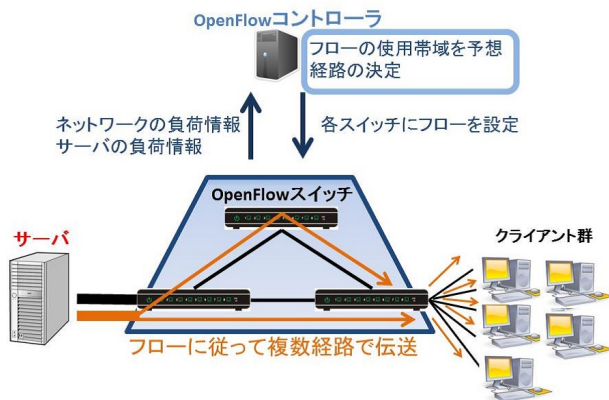


図1 提案システム概要

ケットの経路分散ができなくなるが、TCP 通信では使用する帯域を動的に増減させる仕組みがあり [2]、経路分散を行うとこの仕組みが正常に機能しなくなるので、同一の TCP フロー内のパケットの経路分散は考慮しない。また TCP コネクションが終了したら該当するフローエントリを削除する必要があるが、TCP コネクションの終了をパケットから明示的に判定するのは難しいので一定時間そのコネクションで通信が行われないと、TCP コネクションが終了したとみなしスイッチからフローエントリを削除する。ここでは Windows OS における TCP/IP 通信のデフォルトのタイムアウト時間である 21 秒を規定時間とする。

### 3.6 フローが使用する帯域の把握

データ伝送を始める為、新たにフローエントリを設定する際にその TCP フローが必要とする帯域を把握する必要があるが、必要とする帯域はそのフローがどんなサービスを提供するために使われるかによるので、OpenFlow 側で判断するのは難しい。提案システムではあるサーバから提供されるサービスは同様のものとして、サーバの負荷情報からフローが必要とする帯域を割出す。サーバにつながるスイッチの統計情報からサーバから伝送されているトラフィックの帯域とその伝送のフローの数を調べる事ができるので、ここからフローが使用している帯域の平均値を割出し、新たなフローが必要とする帯域とする。

### 3.7 システム構成と条件

システムの構成機器と条件を以下に示す。

#### OpenFlow コントローラ

スイッチにネットワークの帯域情報に応じてフローエントリを設定し伝送制御を行う。

#### OpenFlow スイッチ

フローテーブルに従いパケットを伝送する。またネットワークの帯域情報を収集する。

#### サーバ

TCP 通信によるデータ伝送を行いクライアントにサービスを提供する。宛先の端末には複数の経路が設置されているものとする。

### 3.8 動作手順

コントローラは周期的にスイッチからネットワークの帯域情報とサーバの負荷情報を収集し、この情報から伝送経路を決定、伝送を行う。提案システムの動作手順を

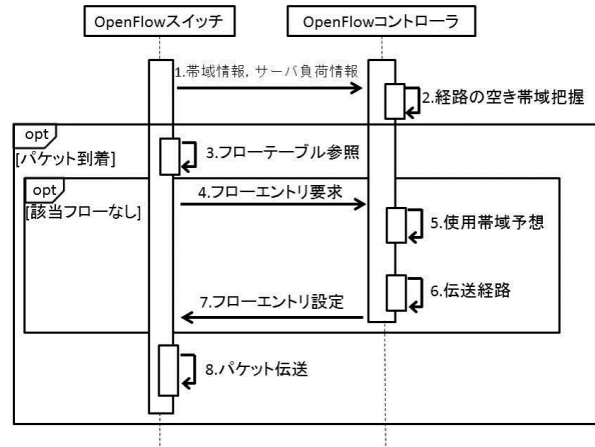


図2 動作手順

図2に示す。

1. スイッチからネットワークの帯域情報とサーバの負荷情報を収集する。
2. ネットワークの帯域情報から各経路の空き帯域を把握しておく。
3. パケットがスイッチに届くと、スイッチはフローテーブルを参照する。
4. 該当するフローエントリが存在しない場合、コントローラにフローエントリを要求する。
5. コントローラは、サーバの負荷情報から、設定するフローエントリが使用する帯域幅を予想する。
6. 予想した使用する帯域幅と経路の帯域情報から、伝送経路を決定し、フローエントリを作成する。
7. 各スイッチにフローエントリを設定する。
8. フローエントリに従ってパケットを伝送する。

## 4 考察

### 4.1 フロー単位での経路分散

フロー単位で経路分散を行う事で、同一の TCP フローのパケットの順序逆転を防ぎつつ、ネットワークの利用効率を上昇させ輻輳の抑制ができる。

### 4.2 サーバ負荷情報の利用

サーバの負荷情報により、新たなフローが使用する帯域幅を予想する事により、より適切に経路設定が行える。

## 5 まとめ

本研究では、OpenFlow を用いた経路分散について検討した。フロー単位で複数の経路にトラフィックを分散し、ネットワークの輻輳を抑制した。サーバの負荷情報からフローが使用する帯域を予想し、フローの経路設定の指標として利用した。

## 参考文献

- [1] 宮本正和, 家永憲人: 動的負荷分散型 IP 網の構成法と経路制御アルゴリズムの性能に関する研究, 電子情報通信学会技術研究報告, CS, 通信方式, 2002.
- [2] 宮川湧太郎: TCP 各バージョンの輻輳制御の比較 <http://www.net.c.dendai.ac.jp/yutaro/> (2015/6/26 アクセス).