

3U-02

大規模災害時におけるトラフィック制御システムの実アプリケーションによる評価

平久 紬[†] 中尾 彰宏[‡] 山本 周[‡] 山口 実靖^{††} 小口 正人[†]
[†]お茶の水女子大学 [‡]東京大学 ^{††}工学院大学

1. はじめに

地震などの大規模災害時にネットワーク輻輳が発生すると、現在のネットワークでは全てのアプリケーションのトラフィックが同一に扱われているために人々が本当に必要とする情報を取得するのが困難になるという問題が発生する。そこで本研究では大規模災害時に通信障害が発生した場合、ソーシャル情報に基づき通信障害を検知すると共にトラフィックの種別を判定し優先度を高くすべきアプリケーションを優先できるネットワーク制御システムを提案する。自動制御を実現するプラットフォームとして FLARE[1] を使用し、実アプリケーションを用いたアプリケーション毎の SDN 制御実験を行うことにより提案システムの有効性を示す。

2. アプリケーション毎の SDN 制御システム

大規模災害時における通信障害発生時のソーシャル情報に基づいたアプリケーション毎のネットワーク自動制御システムの概要は以下の通りである (図 1)。 (1)-(3) の流れで、自動でネットワーク制御を行う。

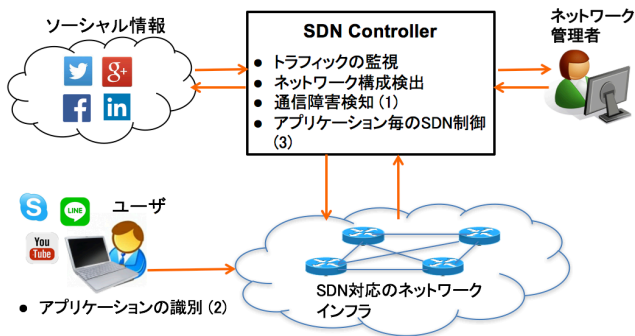


図 1: アプリケーション毎の SDN 制御システム

(1) 通信障害検知

ソーシャルデータとして Twitter を用いる。リアルタイムにツイートを検査し、通信障害が発生している地域に関する情報を含むツイートを取得することにより、どこで障害が起こっているのかを検知する。

(2) アプリケーションの識別

アプリケーション識別のためのソフトウェアをインストールしたユーザ端末を用いる。プロセステーブルと宛先ポート番号からアプリケーションを識別する。ア

プリ識別子としてパケットの後ろにトレーラバイトを付加する。本システムではアプリケーション名とアプリケーション名の長さを付加している。

(3) アプリケーション毎の SDN 制御

SDN 対応のネットワークインフラを制御するコントローラ上で、クライアント PC 上で付加したトレーラバイトを確認しアプリケーションを識別する。トレーラバイトの代わりに Table id を付加することにより、OpenFlow の REST-API を用いコントローラからスイッチに帯域制限や経路切替の命令を行う。本システムではアプリケーション毎に制御可能だが、本研究では災害時に重要度が高いと考えられるリアルタイムに情報交換を行うアプリを優先するように制御する。

3. 実験

本章では、アプリケーション毎の制御を行うため FLARE のスライスを用いて、スライス毎にネットワーク制御を行う。

3.1 実験概要

FLARE のスライスを複数枚使用し、アプリケーション毎にスライス切替を行う。各スイッチ上にスライスを 2 枚用意し、各スライスに OpenFlow 機能を実装する。実験ネットワークを図 2 に示す。クライアント PC とプロキシサーバ間で iperf を用い 2 種類のトラフィックを発生させ、各トラフィックを各スライスに流す。トラフィック 1 は災害時に重要度が高いアプリケーション、トラフィック 2 は重要度が低いアプリケーションとして考える。スイッチ間で通信障害を検知した後、システムはトラフィック 1 を優先して流すためにトラフィック 2 のみ 300Kbps に帯域制限を行う。

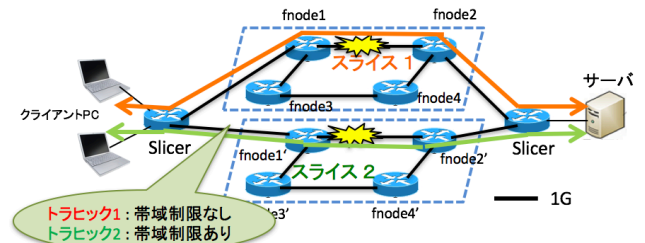


図 2: アプリケーション毎のスライス切り替え

3.2 性能評価

帯域制御後のスライスを 1 枚用いた場合とスライスを 2 枚用いた場合の各トラフィックのスループット値を図 3 に示す。スライス 1 枚を用いた場合、トラフィック 2 のみ帯域制限が行われており提案システムがアプリケーション毎にネットワーク制御できていることがわかる。しかし、トラ

Evaluation of Traffic Control System in Large-Scale Disasters using Real applications

[†] Tsumugi Tairaku, Masato Oguchi

[‡] Akihiro Nakao, Shu Yamamoto

^{††} Saneyasu Yamaguchi

Ochanomizu University ([†])

University of Tokyo([‡])

Kogakuin University (^{††})

ヒック 1 はトラヒック 2 を帯域制限することによりリンク帯域に余裕ができたのにも関わらず 480Mbps のスループット値しか出ていない。スライス 2 枚を用いた場合、トラヒック 2 のスループット値が 610Mbps にまで上がることが確認できた。一スライス内の帯域制御ではフロー数で割った値以上にならないが、スライスを複数枚使用することによりリンク帯域を効率よく使用しつつアプリ毎に帯域制御を行うことが可能となる。

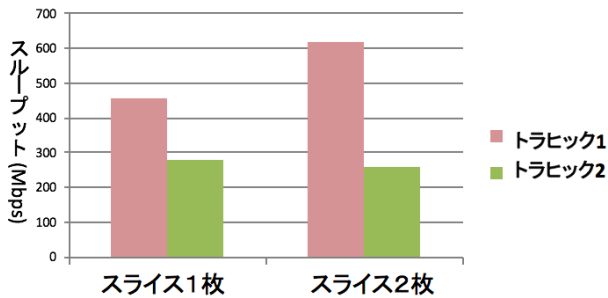


図 3: 帯域制御後のスライス 1 枚と 2 枚用いた時の各トラヒックのスループット値

4. 実アプリケーションによる実験

本章では、実インターネットアプリケーションを用いた実験を行うことにより、提案システムの有効性を示す。

4.1 実験概要

実アプリケーションとして Skype と YouTube を用いて提案システムの有効性を示す実験を行う。Skype は情報交換をリアルタイムに行うことが可能なため災害時に人々にとって重要度が高いアプリケーション、YouTube は帯域を多く使用してしまい娯楽目的であるため重要度が低いアプリケーションとして考えられる。実験ネットワークを図 4 に示す。各スイッチが東京、宮城、新潟にあると想定し本システムの実験を行う。スイッチ間のリンク帯域に余裕があるため、輻輳状態を作るためにリンク帯域を FLARE の Click を用いて絞る。クライアント PC 上で Skype のテレビ電話を行うと同時に YouTube を流す。ソーシャルデータとして東日本大震災時の実際のツイートに基づき本システムを動作させたところ、システムは東京と宮城の間で通信障害を検知し YouTube のみ利用可能帯域が 100Kbps になるように帯域の制限を自動的に行なった。

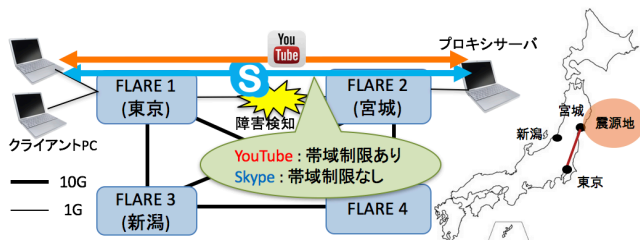


図 4: 実験ネットワーク

4.2 性能評価

図 5 に帯域制御前と帯域制御後の Skype のテレビ電話におけるパケットロス率を示す。リンク帯域を 500, 400, 300Kbps に制限した場合を見ると、帯域制御前に比べて制御後の Skype のパケットロス率が下がっていることがわかる。またリンク帯域を 500, 400Kbps に制限した場合で Skype のテレビ電話を行うと、帯域制限前にはテレビ電話が利用不可能な状態であったのに対して、帯域制限後は画像や音声に乱れがあるもののテレビ電話が可能状態になる。実験結果より、提案システムによりアプリケーション毎にネットワーク制御が可能となり、輻輳が発生した際に重要度が高いアプリケーションを優先するような制御が可能となることが確認できる。

スイッチ間のリンク帯域	帯域制御前の Skype のパケットロス率 (%)	帯域制御後の Skype のパケットロス率 (%)
1Mbps	0	0
500Kbps	0.01	0
400Kbps	7.1	3.1
300Kbps	7.5	7.3

図 5: 帯域制御前と帯域制御後の Skype のパケットロス率

5. まとめと今後の課題

本稿では、大規模災害時のネットワーク輻輳が発生した際に必要となる、ソーシャル情報に基づき通信障害を迅速に検知すると共にトラヒックの種類を判定し優先度が高いアプリケーションを優先するようなネットワーク制御システムを提案をした。また、実インターネットアプリケーションを用いた実験を行うことにより、提案システムの有効性を示した。

今後の課題として、実際に起こりうる輻輳状態に近いネットワークで本システムの実験を行うことや経路切替と帯域制限を組み合わせたアプリケーション毎のネットワーク制御を考える。

謝辞

本研究は一部、総務省戦略的情報通信研究開発推進事業 (SCOPE) 先進的通信アプリケーション開発推進型研究開発および JST CREST JPMJCR1503 の支援を受けたものである。

参考文献

[1] A. Nakao, Software-defined data plane enhancing sdn and nfv. Special Section on Quality of Diversifying Communication Networks and Services, IEICE Transactions on Communications, E98-B(1):12-19, 2015