

JGN-X 上の FLARE によるソーシャル情報に基づく SDN 経路制御実験

平久 紬[†] 柳田 晴香[†] 丸 千尋[†] 中尾 彰宏^{††} 山本 周^{††}

山口 実靖^{†††} 小口 正人[†]

[†] お茶の水女子大学

〒 112-8610 東京都文京区大塚 2-1-1

^{††} 東京大学

〒 113-8654 東京都文京区本郷 7-3-1

^{†††} 工学院大学

〒 163-8677 東京都新宿区西新宿 1-24-2

E-mail: [†]{g1220526,g1120541,g1120536,oguchi}@is.ocha.ac.jp, ^{††}{nakao,shu}@iii.u-tokyo.ac.jp,

^{†††}sane@cc.kogakuin.ac.jp

あらまし 現在のインターネットは、モバイル端末の高機能化やクラウドコンピューティングの発達により、多種多様なアプリケーションのトラフィックが混在しているため、大地震などの緊急災害時のネットワーク輻輳では、すべてトラフィックが同一に扱われ緊急情報に中々アクセスできないという問題が生じる。そこで、トラフィック種別を判定し、アプリケーションごとに制御を行う方式と、Twitter などのソーシャル・データから実社会の急激な状況変化を検出し、自動でユーザが必要とするアプリケーション情報を安定に提供できる仕組みが必要と考える。本研究では、自動制御を実現するプラットフォームとして、広域ネットワークテストベッド JGN-X の各拠点間を VLAN 結合したネットワーク仮想化ノード FLARE を使用し、SDN による自動経路制御実験を行う。これにより、ソーシャル情報に基づく SDN 経路制御手法の有効性を示す。

キーワード SDN, OpenFlow, DPN, FLARE, JGN-X, VLAN

SDN path control experiment based on social information by FLARE on JGN-X

Tsumugi TAIRAKU[†], Haruka YANAGIDA[†], Chihiro MARU[†], Akihiro NAKAO^{††}, Shu
YAMAMOTO^{††}, Saneyasu YAMAGUCHI^{†††}, and Masato OGUCHI[†]

[†] Ochanomizu University

2-1-1 Otsuka, Bunkyo-ku, Tokyo, 112-8610 JAPAN

^{††} Tokyo University

7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8654 JAPAN

^{†††} Kogakuin University

1-24-2 Nishi-Shinjuku, Shinjuku-ku, Tokyo, 163-8677 JAPAN

E-mail: [†]{g1220526,g1120541,g1120536,oguchi}@is.ocha.ac.jp, ^{††}{nakao,shu}@iii.u-tokyo.ac.jp,

^{†††}sane@cc.kogakuin.ac.jp

1. はじめに

近年、モバイル端末の高機能化やクラウドコンピューティングの発達により、現在のインターネットは、多種多様なアプリ

ケーションのトラフィックが混在している。このようなネットワークでは、トラフィックは実社会の状況により大きく変動する。このことから、大地震などの災害時のネットワーク輻輳では、ユーザからのアクセスが集中し、ビッグデータが発生し、

また、すべてのトラフィックが同一に扱われることから、ユーザが必要とする緊急情報に中々アクセスできないという問題が生じる。また、通常ではネットワークの状態はセンサを使って管理されているが、2011年3月11日に東日本大震災が発生した際には、トラフィック量は通常の数倍以上にも増えた[1]。また、経路制御に必要なトラフィック状態に関する情報が膨大であり、センサのみでネットワーク全体の状態を迅速に把握することは困難であった。そのため、長い間、ネットワーク障害により、電話が繋がらないなどといった問題が生じた。その一方で、地震発生から1時間以内に、東京からだけで毎分1,200件以上のツイートが投稿され、地震に関するリアルタイムでの情報のやりとりが非常に活発に行われたことが示されている[2]。このことから、Twitter[3]などのソーシャル情報は、災害時に有用なものであると言える。そこで、トラフィックの種別を判定して、アプリケーションごとに制御を行う方式と、Twitterなどのソーシャル・データから実社会の急激な状況変化を検出し、自動でユーザが必要とするアプリケーション情報を安定に提供できる仕組みの実現が必要と考えられる。

本研究では、自動制御を実現するプラットフォームとして研究開発テストベッドネットワーク JGN-X の各拠点間を VLAN 結合したネットワーク仮想化ノード FLARE スイッチを使用し、Twitter などのソーシャル情報に基づく SDN によるトラフィックの自動経路制御実証実験を行い、その結果より本システムの有効性を示す。

本稿の構成は以下の通りである。2. 章で関連研究について述べ、3. 章でソーシャル情報に基づく経路制御システム概要について紹介する。4. 章では本研究の実験環境を、5. 章では JGN-X における実験ネットワークについて説明する。6. 章では仮想 Mininet 上の実験を行い、7. 章では JGN-X 上の FLARE スイッチで実証実験を行い、本システムの動作を確認する。最後に、8. 章で本稿をまとめる。

2. 関連研究

SDN や OpenFlow [4] の技術を用いた、トラフィックの自動制御は既に実現している [5][6][7][8]。ネットワークのトラフィック量の変動に応じて、ネットワークの構成などを動的に制御している。通常のトラフィック操作のみで制御ネットワークをスイッチ間接続によって構成されるデータネットワーク上に自動構築することを可能としている。また、Twitter 上で発信している緊急地震速報や Yahoo!ニュースの地震カテゴリの記事による情報を基にしたトラフィック制御も存在する [9][10]。これらの研究は、アプリケーションの種別を判定し、電話などの情報通信を行うアプリケーションは重要と判断し優先的にトラフィックを流し、YouTube やゲームなどの娯楽目的のアプリケーションは災害時には重要ではないと判断し、トラフィックを後回しにするような、アプリケーション毎の経路制御を行っていない。そのため、ユーザが緊急災害時に必要とするアプリケーション情報を安定に提供することを目標としている本研究とは異なる。

そこで、本研究ではアプリケーションの種別を判断し、ア

プリケーション毎に経路を切り替えることが可能な FLARE スイッチ [11][12][13][14] を使用し、実験を行っていく。

3. ソーシャル情報に基づく経路制御システム概要

災害時におけるソーシャル・データに基づいた、トラフィックの自動経路切り替えシステムの概要は以下の通りである。図1はシステムの概要図を示している。(1)-(4)の流れで、自動で経路設定を行っていく。

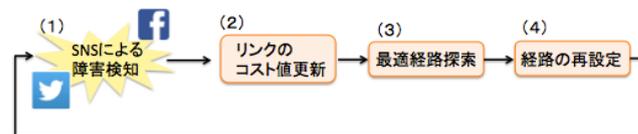


図1: 自動経路制御システムの概要

(1) Twitter による障害検知 [15]

リアルタイムにツイートを監視し、通信障害がどこで起きているかを検知する。

通信障害に関係あるキーワードを含むツイートを取得し、取得したツイートを同じ地名名詞が含まれるツイートごとにまとめる。機械学習を用い、ツイートに出現する地名が「その地域で障害が起きている」かを判断し、地名名詞ごとにツイート数を数える。

(2) スイッチ間のコスト値の更新

スイッチ間のリンクのコスト値を、60秒間隔で更新する。

リンクのコスト値の初期値を1とし、通信障害に関係のあるツイート中にスイッチと対応させた地名を含むツイートが20件以上あったら、+1する。

(3) 最適経路探索

グラフ上の2頂点間の最短経路を求めるアルゴリズムである、ダイクストラ法を用い、コスト値が最小になるような最適経路探索を行い、トラフィックの経路を決定する。

(4) 経路の再設定

OpenFlow の REST-API を使用し、あらかじめ経路を記述したシェルスクリプトを用意しておく。(3)で決定された経路に、コントローラから自動で再設定を行い、経路を切り替える。

4. 実験環境

本研究では、国立研究開発法人情報通信研究機構 (NICT) が運用する研究用ネットワークである JGN-X [16] を使用する。JGN-X の拠点および大学に設置した DPN の FLARE ノードを使用し、ソーシャル情報による SDN 切替実験を実施した。

DPN (Deeply Programmable Network) は、SDN (Software Defined Network) を発展させたものである。OpenFlow 等の従来の SDN との違いは、ネットワークの制御を担うコントロールプレーンだけでなく、パケットを転送するデータプレーンのプログラムも可能であるということである。

DPN のコンセプトを実装したスイッチに、東京大学大学院情報学環中尾研究室によって開発された、FLARE スイッチと呼ばれるものが存在する。FLARE は物理的なスイッチ上にスライスと呼ばれる仮想的なスイッチを複数個設置可能で、Slicer スライスでパケットを各スライスに割り当てる仕組みになっている。このスライスは互いに独立している。FLARE はネットワーク仮想化により、複数のネットワーク機能をプログラミングし動作させることができる。よって、それぞれのアプリケーションに対し、適切な仮想化されたネットワークに割り当て、制御できる。そこで、この FLARE を使用することにより、SDN では不可能であった、緊急時に特定のアプリケーションのトラフィックを優先的に流すような、きめこまやかな制御が可能となる。

FLARE では、データプレーンもプログラム可能であり、このデータプレーン機能は Click module router [17] と呼ばれるソフトウェアルータによって実現されている。Click は C++ で実装されており、Element と呼ばれるモジュールを組み合わせることにより、スイッチやルータ等のネットワーク機器を実現することが可能である。また、Element 自体を開発することも可能である。

5. 実験ネットワーク

本研究で使用する JGN-X 上の FLARE は図 2 のように全国 8 か所に 9 個設置されており、それぞれのノードは VLAN で接続されている。JGN-X 上の FLARE 間のノードでは、タグ VLAN が使用されている。タグ VLAN では、イーサネットフレームにタグをつけて VLAN を識別しており、1 つの通信ポートで複数の異なる VLAN を通信させることが可能となっている。図 2 に、それぞれのノード間のリンクにおける VLAN ID を示している。

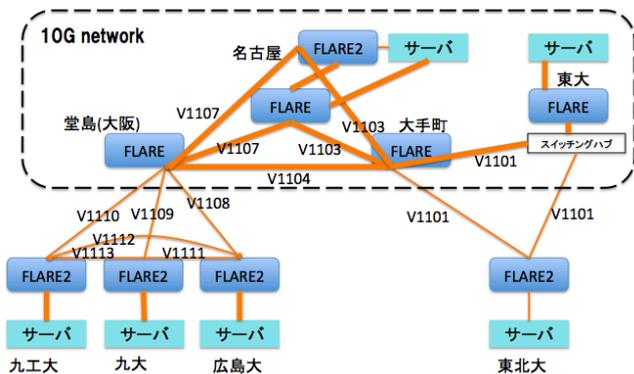


図 2: JGN-X 上の FLARE

実験を行うにあたり、各ノードにおいて FLARE スイッチを VLAN 対応の形で構成した。FLARE は OpenFlow 機能をサポートしており、FLARE の OpenFlow は Click で実装されているため、VLAN 機能をスイッチ外で処理することが可能である。図 3 は名古屋の FLARE スイッチである。

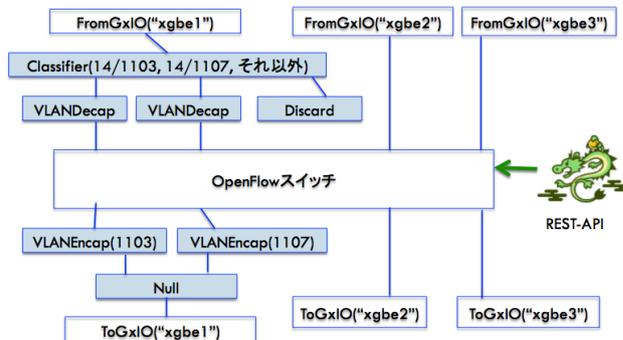


図 3: FLARE ノード (名古屋)

xgbe1 のポートは、大手町、堂島に繋がっているため、流れてくるパケットを OpenFlow スイッチに流す前に VLANDecap で VLAN のタグを外し、また、OpenFlow スイッチから出てきたパケットには堂島や大手町に流すため、VLAN のタグをつける構成になっている。また、xgbe2 や xgbe3 は、それぞれ名古屋のもう一つの FLARE スイッチとサーバに繋がっているため、VLAN のタグをつけたり外したりする必要はない。

このようなスイッチを作成することにより、VLAN で接続されていることを考えずに、REST-API を使用した経路設定をコントローラから指示することが可能である。

また、図 4 のように、コントローラ側で収集したソーシャル情報を解析し、あるアプリケーションはルート 1 の経路を、またあるアプリケーションはルート 2 の経路を流れるように、アプリケーション毎の経路制御を行うことを考える。そのため、新たにスライスを作り、各 FLARE ノードのスライスに OpenFlow 機能を同様に実装した。

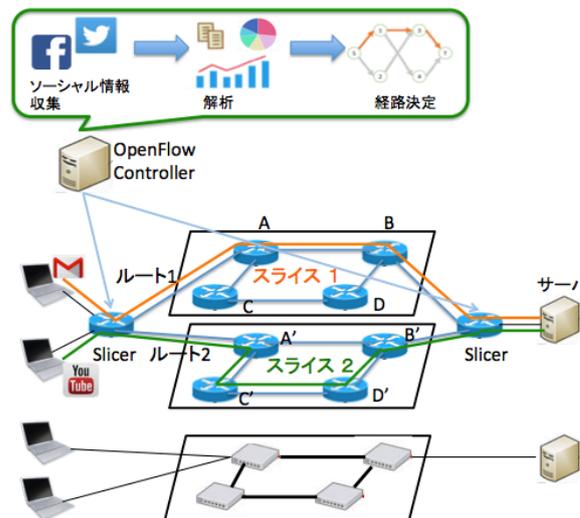


図 4: アプリケーションごとの経路制御

また、Slicer スライスに OpenFlow スイッチを実装することにより、コントローラから Slicer スライスにスライスを切り替えるように指示を与えることによって、スライス間の切り

替えを行う。図5のように FLARE ノードの Slicer スライスに OpenFlow スイッチを使用し、OpenFlow コントローラから指示を与えられるように構成した。

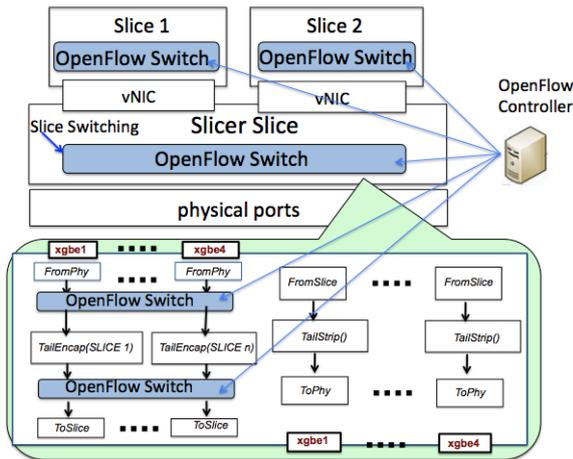


図 5: Slicer スライスに OpenFlow スイッチを使用した FLARE ノード

ここで、Slicer スライスに流れてきたパケットが、どのスライスに流れるのかを判定するために、GRE トンネルを使用した。GRE トンネルは、あるトラフィックを別のプロトコルでカプセル化して伝送する。GRE トンネルを経由し通信が行われる場合、GRE ヘッダとともに新しい IP ヘッダが付加されてパケットが転送される。図6に図5の Slicer スライスの内部を示す。これは、九工大の例である。サーバに接続されている xgb1 のポートから入ってくるパケットは、スライス ID のタグを追加する前に、GRE トンネルのタグを追加した。また、JGN-X のネットワークに繋がっている gbe3 から xgbe1 へ出力するパケットは、GRE を外す必要がある。GRE トンネルは OpenFlow 機能でサポートされていないので、GRE トンネルを使用したスイッチは、FLARE ならではである。このような形にすることで、スライス間の切り替えが可能となり、アプリケーション毎の経路制御も可能となる。

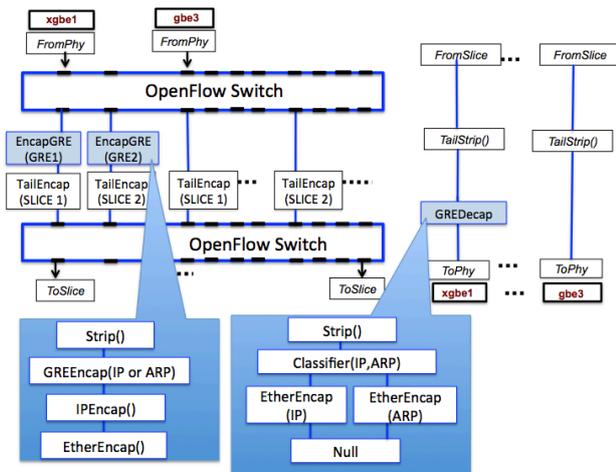


図 6: Slicer スライス (九工大)

6. 仮想環境 Mininet 上の実験

6.1 実験概要

JGN-X 上で実証実験を行う前に、OpenFlow のスイッチ、コントローラ、通信ノードを全てエミュレートした仮想環境である Mininet [18] 上で実験を行った。実験環境は以下の通りである。

表 1: 開発環境

OS	ubuntu14.04 64bit
フレームワーク	Mininet 2.2.1
コントローラ	Ryu-manager 3.23
スイッチ	Open vSwitch 2.0.2

JGN-X 上の FLARE スイッチと同様のトポロジを作成し、ソーシャル情報に基づく経路制御システムの実験を行った。このうち、図7のように九大、東大、名古屋、東北大にそれぞれ対応するスイッチを使用して実験を行った。東日本大震災時の 2011 年 3 月 11 日 14 時から 15 時の実際のツイートに基づき、東北大から東大への通信を想定し実験を行った。

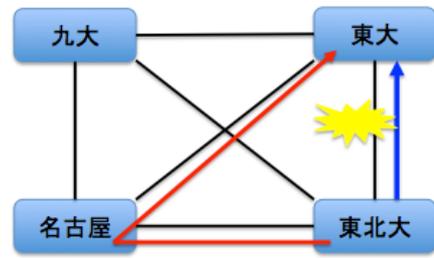


図 7: Mininet 上の実験で使用したノード

6.2 実験結果

各スイッチ上で tcpdump を使用し、スイッチに流れるパケットの情報を確認した。名古屋に対応するスイッチに流れるパケット情報を tcpdump で確認した結果を図4に示す。赤い線より上は障害検知前に流れていたパケットを、下は障害検知後に流れていたパケットを示している。

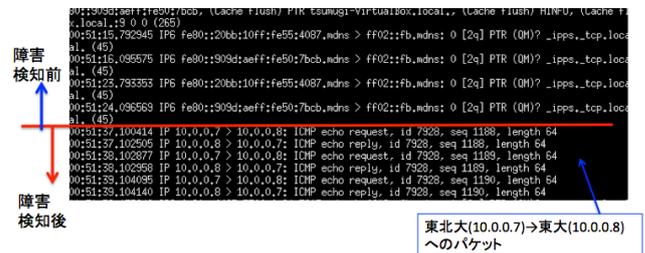


図 8: 名古屋のスイッチを通るパケット

名古屋に対応するスイッチでは、通信障害を検知した後は東北大に対応するスイッチ (10.0.0.7) から東大に対応するスイッチ (10.0.0.8) へのパケットが流れていることが分かった。同様に全てのスイッチ上に流れるパケットを確認したところ、ツイート

によって、東大に対応するスイッチと東北大に対応するスイッチの間で障害を検知し、東北大から東大へ流れるパケットは、名古屋を経由する経路に切り替わったことを確認した。

この結果より、仮想環境 Mininet 上で、ソーシャルメディアの情報からトラフィックの経路を切り替えることに成功した。

7. JGN-X 上の実験

7.1 実験概要

JGN-X での実験では、仮想環境 Mininet と同様に東日本大震災時の 2011 年 3 月 11 日 14 時から 15 時の実際のツイートを基に、東北大から東大への通信を想定する。実験構成の都合上、東北大-東大サーバ間の障害経路切替として、アクセスするサーバを東大のサーバのデータをレプリケーションしている名古屋に設置したサーバにアクセスする経路切替シナリオで実験を行った。図 9 はシナリオを図示したものである。1 のルートから 2 のルートに経路が切り替わるシナリオである。

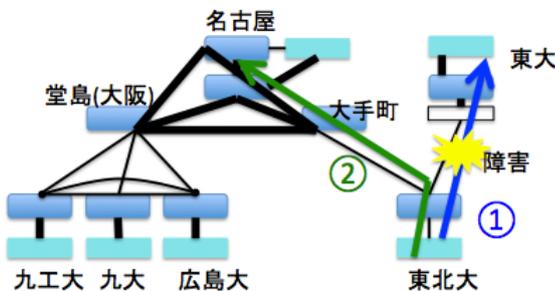


図 9: JGN-X における実験のシナリオ

7.2 実験結果

図 10 に実験の結果を示す。2700 秒付近で経路の切り替えを行っている。経路が切り替わったことを Ping の応答遅延時間を測定することにより確認した。

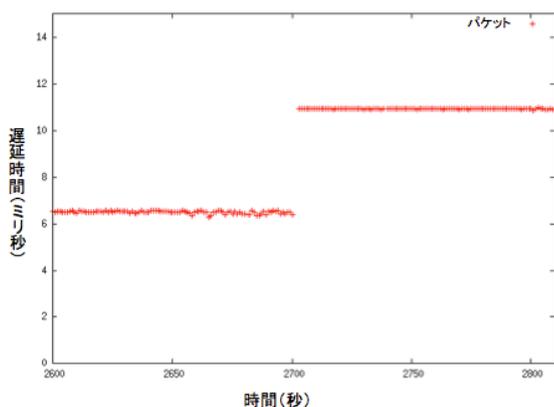


図 10: 実験結果 (遅延時間)

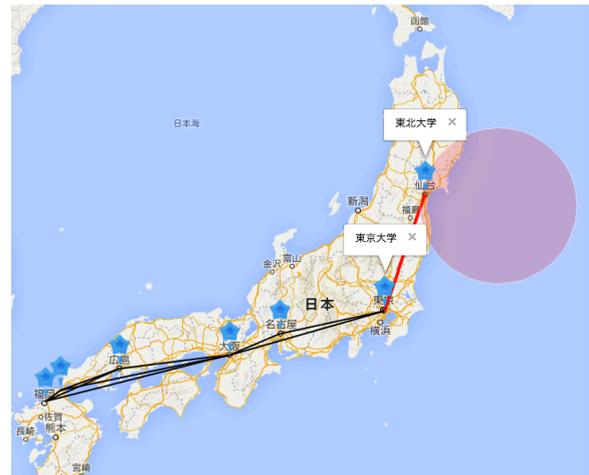
最初は東北大と東大間の経路を通る通信を行っていたが、東北大と東大の間で障害を検知した後、東北大、大手町、名古屋の経路を通り通信が行われたことが、Ping の応答遅延時間から

分かる。現時点では、システムが障害を検知してから、約 3 秒ほどでパケットが流れる経路が切り替わることを確認した。

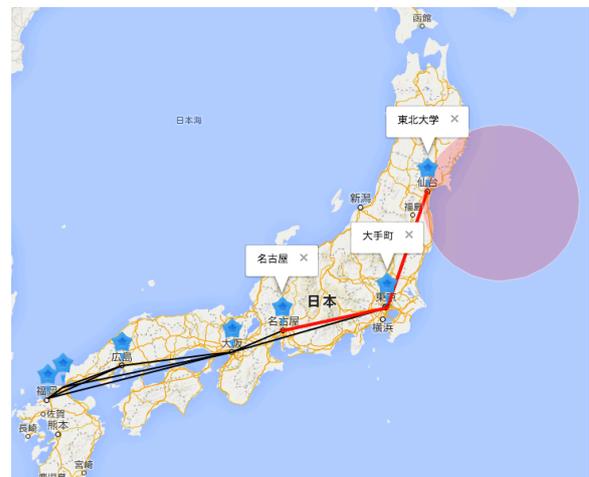
この結果より、ソーシャル・データとして Twitter の情報からネットワークの経路を自動で切り替える実験に成功し、本システムが有効であると言える。

7.3 実験結果の可視化

経路が自動で切り替わったことを確認するため、ネットワークの経路の可視化を行った。実証実験での障害検知前と障害検知後のネットワークを図 11 に示す。



(a) 障害検知前



(b) 障害検知後

図 11: 実験結果の可視化

(a) は障害検知前を、(b) は障害検知後のネットワークを示したものである。それぞれ、図 2 の JGN-X 上の FLARE を Google Map 上に表示している。青いピンは各 FLARE スイッチ、黒い線は FLARE スイッチ間のリンク、赤い円は震源地を表している。赤い線がパケットが実際に流れた経路を表している。(a) 障害検知前は東北大から東大へパケットが流れており、(b) 障害検知後は東北大、大手町、名古屋の経路を通っている。可視化結果からも、ネットワークの経路が切り替わったことを確認した、

8. まとめと今後の課題

VLAN で接続されている JGN-X 上の各ノードにおいて FLARE スイッチを VLAN 対応の形で構成し、実験環境を整えた。これにより、VLAN で接続されていないスイッチを使用している場合と同じ経路設定を記述し、実行可能となった。また、スライス間の切り替えを行うため、各 FLARE ノードの Slicer スライスに OpenFlow スイッチを入れた。また、GRE トンネルを使用し、複数あるスライスを識別し、スライス間の切り替えを可能にした。そこで、実際に 2011 年 3 月に起こった東日本大震災時のツイートを基に、ネットワークの経路を自動的に切り替える実験を仮想環境 Mininet 上でを行い、各スイッチ上で流れるパケットの情報を tcpdump で確認したところ、本システムが正常に動作し経路が切り変わったことを確認した。また、研究開発用テストベッド環境である JGN-X 上でも同様に本システムの実証実験を行った。JGN-X では Ping の応答遅延時間を測定することによって、本システムが正常に動作し経路が切り変わったことを確認した。また、パケットが流れる経路を Google Map 上に可視化した結果からも、本システムにより経路が切り変わったことを確認した。その結果、ソーシャル・データを基にした自動的な経路切り替え実験に成功し、本システムが有効であることを示した。

今後の課題としては、実際に異なるアプリケーションのトラフィックを流し、アプリケーションの種類を判別し、アプリケーション毎に経路を切り替える実験を行っていく。また、2011 年 3 月に起こった東日本大震災時のツイートだけでなく、他の災害時におけるツイートを基に本システムの実証実験を行っていく。

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によるものである。

文 献

- [1] 総務省：「東日本大震災における通信の被災・輻輳状況」,
http://www.soumu.go.jp/main_content/000136962.pdf
- [2] 吉次由美：「東日本大震災に見る大災害時のソーシャルメディアの役割」,
https://www.nhk.or.jp/bunken/summary/research/report/2011_07/20110702.pdf
- [3] Twitter, <http://twitter.com/>
- [4] OpenFlow コンソーシアム：OpenFlow 仕様,
Open Networking Foundation: <http://www.openflow.org/050819.pdf>,
2005 年 8 月 19 日
- [5] 飯島明夫：「OpenFlow/SDN のキャリアネットワークへの適用について」 電子情報通信学会技術研究報告. NS, ネットワークシステム 112.231 (2012): 85-87.
- [6] NEC：「OpenFlow の活用により災害時における移動通信ネットワークの動的なソフトウェア制御を実現」,
<http://jpn.nec.com/rd/innovation/feature/2013/11-openflow.html?コンセンサス>, 2013 年 11-12 月号
- [7] 橋本直樹, 廣津登志夫：「冗長経路を利用した OpenFlow 制御ネットワークの自動構築」,
日本ソフトウェア科学会 第 15 回インターネットテクノロジーワークショップ
- [8] 橋本直樹, 廣津登志夫 「OpenFlow ネットワークにおける制御ネットワークの自動構築」,
日本ソフトウェア科学会 第 14 回インターネットテクノロジーワークショップ
- [9] 原瑠理子, 長谷川友香, 小口正人：「モニタリング情報に基づく OpenFlow を用いたネットワークトラフィック制御モデル」,
DEIM2014, C9-6, 2014 年 3 月
- [10] 高橋裕, 秋山友理愛, 神津智樹, 山口実靖：「バーストの負荷変動を考慮した OpenFlow を用いた動的資源割り当て (SDN/OpenFlow)」,
電子情報通信学会技術研究報告. NS, ネットワークシステム 113.472 (2014): 225-229.
- [11] Akihiro Nakao, "FLARE: Open Deeply Programmable Network Node Architecture",
Stanford Univ. Networking Seminar, October 2012.
http://netseminar.stanford.edu/10_18_12.html
- [12] 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.
- [13] Akihiro Nakao, "Software-Defined Data Plane Enhancing SDN and NFV", Special Section on Quality of Diversifying Communication Networks and Services, IEICE Transactions on Communications, 2015 Jan
- [14] Akihiro Nakao, Ping Du, "Application and Device Specific Slicing for MVNO",
2014 International Science and Technology Conference (Modern Networking Technologies) (MoNeTeC), 2014, October
- [15] Chihiro Maru, Miki Enoki, Akihiro Nakao, Shu Yamamoto, Saneyasu Yamaguchi, and Masato Oguchi, "Network Failure Detection System for Traffic Control using Social Information in Large-Scale Disasters",
ITU Kaleidoscope Conference 2015: Trust in the Information Society, S5.3, 9-11 December 2015.
- [16] NICT, Network Testbed JGN-X
<http://www.jgn.nict.go.jp>
- [17] UCLA.edu, The Click Module Router Project
<http://read.cs.ucla.edu/click/click>
- [18] Mininet An Instant Virtual Network on your Laptop (or other PC),
<http://mininet.org/>