

# 位置情報に基づく災害時の重要情報を優先転送する ネットワーク管理手法 ～実用化に向けた実地検証報告～

小川康一<sup>†1</sup> 吉浦紀晃<sup>†2</sup>

著者らは災害時における重要情報を優先転送する「災害 ID 付与方式」を提案している。「災害 ID 付与方式」は、位置情報に基づく制御が可能とするネットワーク管理手法である。ネットワークを流れるパケット内の IP アドレス情報から災害地の通信であると判断できた場合には、その通信に「災害 ID」を付与することである。ネットワーク機器はこの災害 ID を確認し、災害 ID が付いている場合には優先的に配送を行う。また、災害地からの通信、または災害地への通信を優先制御するために、通信が災害地からのものであるかを把握する必要がある。そのためには、IP アドレスと位置情報の対応関係の把握が必要となる。

本論文では、提案手法の有効性を検証するため、著者らの大学構内に実際の利用環境を再現した検証環境を構築した。検証環境で実験を行った結果、重要な情報が IP アドレスに関連付けられた位置情報に従い優先転送されることを確認した。

## Network Management Method for Priority Control of Important Information based on location Information during Disasters

Kohichi Ogawa<sup>†1</sup> Noriaki Yoshiura<sup>†2</sup>

The authors propose "Disaster ID Assignment Method" in that packets of important information can be transmitted at high priority in disasters. Disaster ID assignment method is a network management method that is based on location information. The method attaches IDs to packet from or to disaster areas. The IDs are used to give priority communications from or to disaster areas. Network equipment identifies communications from or to disaster areas by IDs. This paper calls the IDs "Disaster IDs". Network equipment preferentially processes the communications by the IDs. Network equipment needs to identify communications that are related with disaster areas in order to control the priority of communication from or to the disaster areas. Identification of communications for disaster areas needs to obtain location information by IP Addresses of disaster areas. In order to verify the effectiveness of this method, this paper constructed a verification environment in the campus of the authors. The results of an experiment in the verification environment confirm that the packets of important information can be transmitted faster than other packets.

### 1. はじめに

東日本大震災が発生した 2011 年 3 月以降、災害に対する人々の危機意識が高まっている。これまで著者らは、「災害 ID 付与方式」[1]と呼ばれる手法を用い、災害地に関連する通信を識別しその通信を優先する方法を提案している。

災害 ID 付与方式では、IP アドレスと位置情報のデータを収集、蓄積、共有することで、ネットワーク上の通信に含まれる IP アドレスの情報を利用して、位置情報に基づく制御が可能となる。

今回は、提案手法の具体的な利用形態を想定し、大学構内の数カ所に検証機器を設置し、提案手法に基づき実装したシステムが正しく動作することを確認した。また、問題点や今後の展開について述べる。

### 2. 本研究の位置付け

2011 年 3 月に発生した東日本大震災は災害に対する意識

を変える契機となった。今後も首都直下型地震や南海トラフ地震など、災害発生が予測されている。しかし、いつどこで地震が発生するかという予測が困難であることから、早急な対策が求められている。IT を活用した災害対策としては、安否確認や情報収集が考えられる。昨今ではスマートフォンの普及によりインターネットの利用が一般化し、災害時の安否確認の手段としてインターネットの利用が増加すると予想される。結果、ネットワークにおいて急激なトラフィック増の状態が発生し、災害地への通信帯域が確保不可能な状況が想定される。東日本大震災のような広域災害においては、的確な情報収集や収集した情報の有効活用によって、被害の状態に応じた迅速な救助や支援活動への貢献が求められている[2]。災害時には、通常の通信よりも災害地の通信について一定の品質を確保する必要がある。総務省の「大規模災害等緊急事態における通信確保の在り方に関する検討会」で議論された内容がガイドライン[3]として反映されている。災害時に緊急性の高い通話の疎通確保のための帯域制御の正当性や必要性が定義されている。本研究ではこのような背景を元に、災害時に重要情報を優先転送するネットワークの管理運用手法を提案する。

現在、ネットワーク上のどの通信が災害地に関連するも

<sup>†1</sup> 埼玉大学情報メディア基盤センター  
Information Technology Center, Saitama University

<sup>†2</sup> 埼玉大学大学院理工学研究科  
Graduate School of Science and Engineering, Saitama University

のかを明確に判断する方法は確立されていない。そこで本研究では、ネットワークを流れるパケット内の IP アドレスから災害地と判断できた場合には、その通信に目印となる「災害 ID」を付与する。パケットに災害 ID を付与することにより、ネットワーク機器は災害 ID に基づき配送の優先度を制御する。

災害地からの通信または災害地への通信を優先制御するためには、通信が災害地に関連するかを把握する必要がある。そのためには、IP アドレスと位置情報の対応関係を収集し、データベース化して利用する。

対応策を具体的に実現する方法の 1 つとして、本論文では OpenFlow[4]を用いる。OpenFlow とは、Software-Defined Networking (以下、SDN) の実現技術である。SDN はソフトウェアでネットワークの動作や振る舞いをプログラミング可能にし、ネットワーク管理の柔軟性を向上させる考え方である。OpenFlow コントローラの機能拡張によって災害地に関連する通信を判別する。

### 3. 提案手法

本章では災害時の具体的な対応策として、OpenFlow コントローラに拡張実装する機能について述べる。

#### 3.1 災害時発生時の設定情報

災害時の設定情報は、OpenFlow コントローラに予め保持する。災害時の設定情報は、災害地の通信を優先的に配送する処理を OpenFlow スイッチに行わせる。設定情報により、災害時に重要な情報を優先して配送することが可能となる。OpenFlow コントローラは、通常の設定とは別に災害時の設定情報を持つこととなる。

#### 3.2 災害情報の収集方法

OpenFlow コントローラを拡張し、災害情報を収集する機能を実装する。災害情報とは緊急地震速報や気象情報である。このように、OpenFlow コントローラによる災害情報や気象情報の収集機能によって、管理を受け持つネットワークの災害時における被災の可能性などの状況を把握できる。

現状の設計では、3.1 にあるように災害時用の設定情報を持っており、これに切り替えることで災害 ID 付与方式が有効になる利用方法を想定している。

#### 3.3 IP アドレスと位置情報の対応関係の把握

現在のネットワークの品質制御 (Quality of Service, 以下 QoS) 技術では、送信先・送信元の IP アドレスに基づき優先制御が行われる。本論文では、通信の送信元や送信先の IP アドレスから通信の優先度を決定する方法を用いる。

このとき、OpenFlow コントローラや OpenFlow 対応ネットワーク機器では、収集可能な情報は IP アドレスのみであり、送信先や送信元の IP アドレスがどのような場所であるかの判断が困難である。つまり、災害地の通信であるかを判別できない。そこで、IP アドレスと位置情報 (緯度/経

度) の対応が必要となる。この対応関係を収集し、データベース化して OpenFlow コントローラと連携させる。本論文ではこのデータベースを「IP アドレスマッピングデータベース」と呼ぶ。あわせて、データベース内のテーブルを「マッピングテーブル」と呼ぶ。

## 4. 災害時の通信品質確保の具体的手法

本章では前章の提案手法に基づき、具体的な通信の品質確保の方法について述べる。

### 4.1 災害地への重要通信の判別手法

先に述べたように、論文[1]で専用の ID として「災害 ID」の付与によって災害地からの通信を区別する方法を提案した。「災害地からの通信には災害 ID を付与する」というルールを定義したフローエントリをフローテーブルに追加する。本設定により、OpenFlow スイッチは災害 ID の読み取りが可能となる。結果、災害地向けの通信を区別し、優先制御や帯域確保を実現する。

### 4.2 災害 ID の割り当て方法

図 1 に災害時における通信制御の方法を示す。

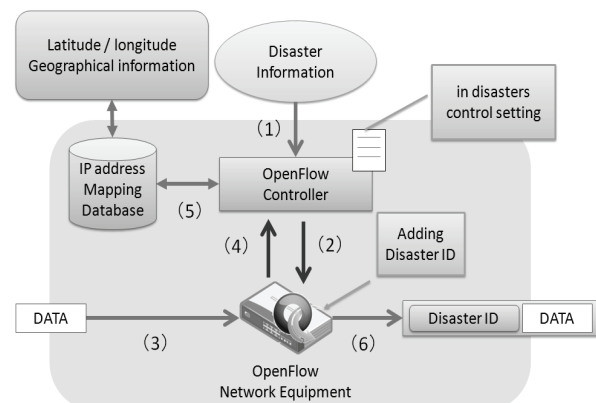


図 1 災害 ID の割り当て方法

Figure 1 Method for allocating Disaster ID.

(1) OpenFlow コントローラは災害発生の情報を受け取り、予め保持している災害時用の設定情報を読み込む。

(2) OpenFlow コントローラは OpenFlow スイッチに対して、災害 ID に基づく優先制御を実施する設定をスイッチに行う。

(3) パケットが新しく入ってきたとき、災害 ID が付与されているかのチェックを行う。付与されていない場合には次の手順 (4) に進む。災害 ID が付与されている場合には、OpenFlow スイッチで優先的に転送され、以降の手順は行われない。

(4) OpenFlow スイッチは、送信元もしくは送信先の IP アドレスを確認し、OpenFlow コントローラに問い合わせる。

(5) OpenFlow コントローラは IP アドレスマッピングデータベースで IP アドレスを照合し、当該通信の位置情報を確

認する。

(6) 災害地に対応する IP アドレスであれば、災害 ID を付与しデータを送信する。災害地の IP アドレスでない場合には、災害 ID と別の ID を付与して、他のスイッチが同じ処理を行うことを防ぐ。

## 5. 位置情報を用いた災害地向け通信の判断

本章では、位置情報を用いた災害地向け通信の判断方法について述べる。

### 5.1 IP アドレスと位置情報の対応

送信先と送信元として優先すべき場所の位置情報の正確な把握は、現在の TCP/IP プロトコルの機能のみの実現は難しい。そのため、IP アドレスと位置情報を対応させる IP アドレスマッピングデータベースが必要となる。

東日本大震災のような広範囲に渡る災害では、地域によって状況が異なる可能性があり、柔軟な対応のためには細かな地域の情報が重要となる。

### 5.2 データベースを用いた位置の特定

IP アドレスと位置情報を結びつける方法に「GeoIP」がある[5]。GeoIP は Maxmind 社が無償・有償で提供しているサービスであり、IP アドレスによる位置情報の特定が可能である。しかし、本研究では日本国内における災害地の特定を必要としており、詳細の位置情報を把握する必要がある。GeoIP では誤った情報も多く、場所の特定も国や都市のレベルでの把握にとどまることもあり、IP アドレスから正確な位置情報を把握できない。そこで、OpenFlow コントローラを拡張し、IP アドレスと位置情報の対応関係を把握する機能を持つ、IP アドレスマッピングデータベースの作成を進めている。本データベースの利用によって、IP アドレスと位置情報とを対応づけることを実現する。

図2はIPアドレスマッピングデータベースのデータ構造の概略である。IP アドレスに対応する緯度、経度を予め登録しておくことで位置状況を把握する。これらの位置情報は、OpenFlow コントローラを設置、設定する際に予め設定しておくことを想定している。このように登録された位置情報と IP アドレスのデータは各 OpenFlow コントローラが保持し、災害時以外にも組織間で共有し活用する。

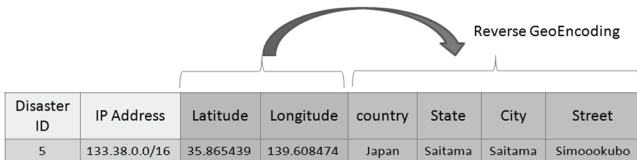


図2 IP アドレスマッピングデータベースのデータ構造  
 Figure 2 Data Structure of IP Address mapping Database.

### 5.3 位置情報の登録方法

位置情報の登録は、実際にネットワークを利用する地点

毎に行う。位置情報として緯度経度を、ネットワークの情報として利用している IP アドレスを登録する必要がある。今回はサブネット単位のネットワークに位置情報を登録できる Web アプリケーションを開発した (図3)。



図3 ネットワーク位置情報登録設定画面

Figure 3 The Screen of Network Location Information Registration.

本登録画面では、GoogleMap 上で場所を選択し、緯度経度を登録することが可能である。

## 6. データベースを用いた実際の通信

本章では、IP アドレスマッピングデータベースを用いた実際の動作について述べる。

### 6.1 データベースへの情報登録手順

本提案手法における IP アドレスマッピングデータベースへの情報登録手順は以下の通りである。

- (1) OpenFlow コントローラの IP アドレスと位置情報を IP アドレスマッピングデータベースに登録する。
- (2) OpenFlow スイッチと OpenFlow コントローラを環境に合わせ初期設定を行う
- (3) OpenFlow コントローラに災害時用の設定行う
- (4) IP アドレスマッピングデータベースに DHCP による IP 割当時の認証に用いる郵便番号を紐づけるコードと対応させたテーブルの設定を行う。
- (5) IP アドレスマッピングデータベースと OpenFlow コントローラの協調動作により、個別の IP アドレスの収集を可能にする。そして、IP 割当時に紐づく ID の情報と合わせて郵便番号コードを参照し、マッピングテーブルにデータ挿入する。

### 6.2 データベースを用いた通信

災害地への通信イメージは図4の通りである。通常、各ネットワークの OpenFlow コントローラから各ネットワークにある IP アドレスマッピングデータベースへの災害地の IP アドレスであるかどうかの問い合わせは、一度限りである。以降、災害 ID の付与された通信は OpenFlow 対応のネットワーク機器ではコントローラに問い合わせを行うこ

となく、優先して転送されるため、高速な転送が可能となる。データベースの内容については、組織間で相互にデータを交換し、各組織が持つ情報を統合する。

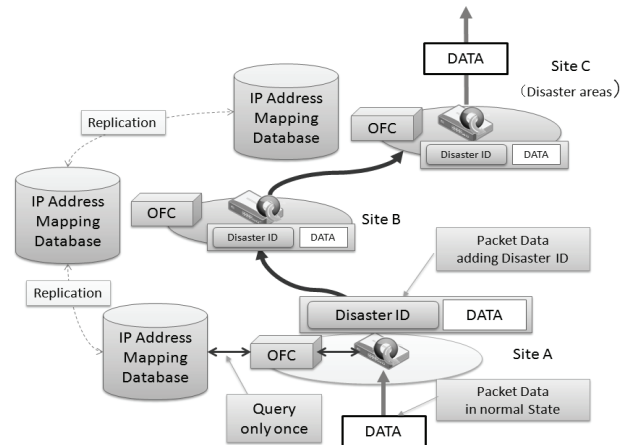


図4 災害地への通信イメージ

Figure 4 Communications to Disaster Areas.

## 7. システムの実装

本章では前章までに述べた手法に基づく具体的なシステムの実装について述べる。

### 7.1 システム基本設計

本論文で述べる手法を実際に稼働するシステムとして実現するには、OpenFlow コントローラ、OpenFlow スイッチが必要となる。合わせて、IP アドレスマッピングデータベースを実現するデータベースや災害情報を自律的に取得する機能などが必要である。システムの機能概要を図4に示す。OpenFlow コントローラでは、災害 ID の確認と付与を行う。この災害 ID の埋め込み箇所には ToS フィールドを用いる。災害 ID が無いパケットの場合には、災害地か非災害地かを IP アドレスマッピングデータベースにより判断し、災害地であった場合には ToS フィールドに高い数値を、非災害地であった場合には ToS フィールドに低い数値を割り当てる。そして、以後災害 ID が付与済みのパケットは、フローテーブルに従い転送される。

### 7.2 システム構成

システム構成を図5に示す。OpenFlow コントローラは、フレームワークの一つである Trema[6]を用いて開発した。今回は開発の容易さやソースコードの可読性を重視し、Ruby で記述している。

OpenFlow スイッチは、オープンソースで提供されている Open vSwitch[7]を用いる。現状、Open vSwitch はソフトウェアそのものに QoS を実現する機能はなく、Linux の OS の機能である Traffic Control (TC) を設定することで対応する。この設定は、Open vSwitch を制御するためのコマンドで設定可能であるが、実際の管理運用には不便であるた

め、Web インターフェースを開発した。

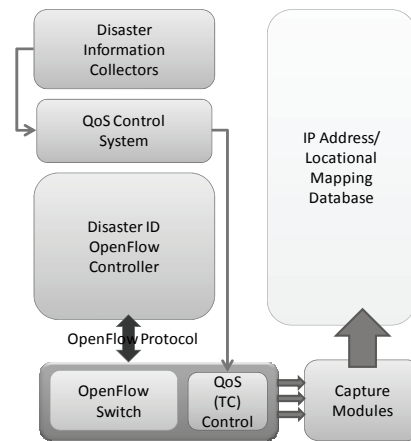


図5 システム構成の概略図

Figure 5 Schematic Diagram of the System Structure.

### 7.3 地震発生時の動作

地震が発生する際、各地の震度は地域により明確に規定されている。日本各地に設置されている地震計の設置場所は公開されている[8]。この位置情報を元に、大きな地震発生にあった箇所と緯度経度をもとに照合することで優先すべき地域かを判定する。災害時の動作ロジックは次のとおりである。

- (1) 地震速報等を発信している Twitter で、震度および発生地点を知らせるツイートを受信する
- (2) ツイート受信を元に OpenFlow スイッチを通常時運用から非常時運用に切り替える (図6)
- (3) 地震発生地点から緯度経度を確認する
- (4) IP アドレスマッピングデータベースに蓄積しているデータから、(2)で確認した緯度経度を元に、災害地の判定を行う
- (5) 災害地の判定を元に、優先すべきデータにフラグを立てる
- (6) ネットワーク転送時にデータベースのフラグを確認し、優先すべき通信に ID を付与する。

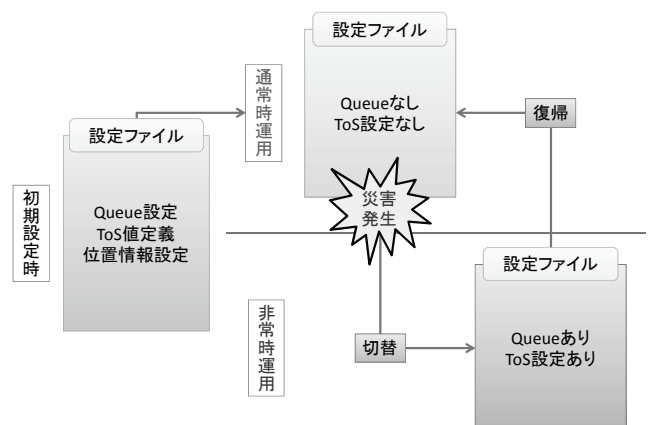


図6 状況に応じた OpenFlow コントローラの設定切替

Figure 6 Configuration of OpenFlow Controller depending on the Situation.

本システムでは、災害における優先制御の対象とする拠点の判定を行う。たとえば、判定の基準を震度3以上、半径30Km圏内に含まれる地点を制御の対象とするといった、範囲規定を行う。このとき、実際に正しく地震が発生した箇所とその周辺の箇所との距離を知る必要がある。たとえば、緯度経度間の距離は、「三平方の定理」を用いて算出することができる。しかし実用上は迅速な判定が求められるため、高速に計算を行う必要がある。MySQLは緯度経度が扱える位置情報を扱うための型として geometry 型[9]を持っており、これを用いる。

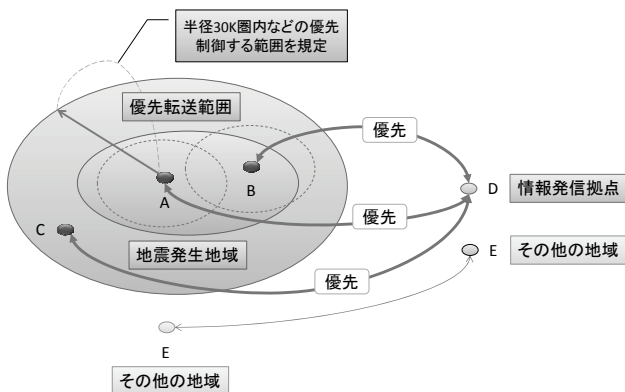


図7 優先転送範囲の判定方法

Figure 7 Method of determining the Priority Transfer Range.

図7は地震発生時の発生地点とそれ以外の地点との関係を図示したものである。この場合、地震が観測されたのは地点Aと地点Bである。仮に地震発生地域から30Km圏内の範囲を本手法における優先転送の範囲とした場合には、地点Aの緯度経度から30Km圏内に点在している地点がどこかを割り出す必要がある。たとえば、埼玉大学の緯度経度(35.864216, 139.607858)から30Km圏内の地点を検索するSQL文としては、

```
SELECT ROUND(Glength
    (GeomFromText(CONCAT('LineString(139.607858
    35.864216 ,', X(geo), ',', Y(geo), ')')
    ) * 111000 ) AS distance
```

FROM mapdb WHERE len <= 30000 ORDER BY distance  
 となる。

図7の例では、地点Cは30Km圏内に含まれるため優先転送が行われる範囲内となる。このように優先転送する範囲を決定、判定するには地点間の距離を頻繁に求めることが必要となる。そのため高速に処理できるMySQLのGeometry型を採用した。

## 8. 提案手法の有効性評価

本章では、本章では、前章で述べたシステムを用い、災害ID付与方式についての有効性の実地評価実験と実験結果について述べる。

### 8.1 評価実験の目的

7章で実装したシステムを用い、著者らが所属する大学構内に検証機器を配置した。本論文では検証機器を配置した箇所を「拠点」と呼ぶ。図8は実験の概略図である。動作および性能について実験を通して評価する。以下2点の項目を実施した。

(1)災害ID付与方式と災害ID付与方式を用いない場合、各拠点の比較

Pingを用い、2つの設定で処理にかかる時間に違いがあるかを確認する

(2)災害ID付与方式における有効性評価とパフォーマンス測定

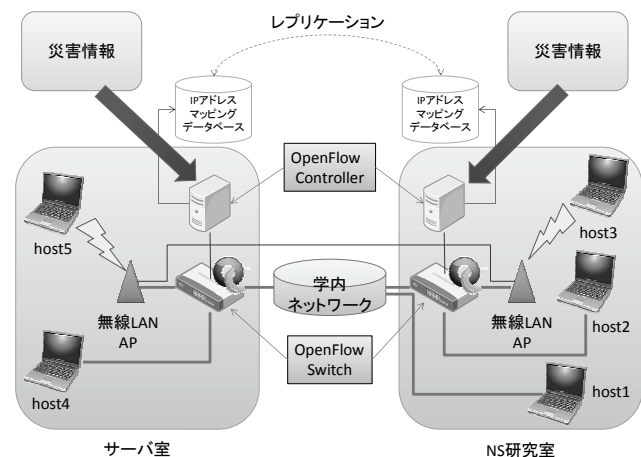


図8 評価実験の概略図

Figure 8 Schematic diagram of the Evaluation Experiment.

### 8.2 評価実験構成

OpenFlowスイッチを用いた場合に、災害IDが付与された側のクライアントで優先的な帯域を選択が可能で、制御が自動で正しく行われるかを確認する。今回は検証のために60M/bps、30M/bps、10M/bpsの3つの帯域を用意した。

学内ネットワークに接続するため、IPインターフェースを持たないVLANを2つ用意した。ひとつを通信用、もうひとつをOpenFlowのSecure Channel通信用とした。

今回の実験では、各拠点にOpenFlowスイッチ(OpenFlowコントローラを兼務)、有線LAN端末、無線LAN接続端末、無線LANアクセスポイントを用意した(図9)。無線LANアクセスポイントはBuffalo社のブロードバンドルータ(WHR-G301N)を用意し、OpenFlowプロトコルに対応するために、OpenWRT[10]をインストールし各拠点の無線LANアクセスポイントとした。OpenWRTは、ルータなどの組み込み機器向けのLinuxディストリビューションであ

る。

各拠点には 1 台のサーバと 1 台の無線 LAN 接続端末を用意し、OpenFlow スイッチを通して接続し、OpenFlow コントローラが災害時に位置情報に基づき正しく優先制御が機能するかを確認する。



図 9 拠点に設置した検証機器

Figure 9 Verification Equipment placed on the Experiment Field of Campus.

OpenFlow スイッチは、Linux サーバに Open vSwitch をインストールし、ネットワークインターフェースを 2 つ用意し、実験用のスイッチとした。また、IP アドレスから災害地かの判断を行うための IP アドレスマッピングデータベースとして MySQL を用いた。実験に用いたスイッチは iperf[11]による評価の結果、検証環境で約 94Mbps のスループットを得た。本環境での無線 LAN のスループットは IEEE802.11n で平均 50Mbps であった。

今回の実験では、検証拠点が隣接しており狭い範囲に限られるため、そのまま緯度経度を用いた実験は難しい。そこで、正しい場所ではなく仮の場所として登録し実験することにした。地震情報は、実際の情報を得るまで待つことが難いため、テストで用意した Twitter アカウントを用意し、地震情報を人為的に発信し、情報を得ることとした。

### 8.3 評価実験結果

#### (1) Ping 送信の実験結果

サーバ室側から研究室の各端末へ Ping を用いた ICMP パケットの送信を行った。結果は表 1 のとおりである。Open vSwitch を単純な Layer2 スイッチとして実装した場合、端末間の転送が 1.8ms であるのに対し、災害 ID 付与方式では 1.9~2.45ms 程度であった。また、無線 LAN 経由では、単純な Layer2 スイッチの場合は 1.8ms で、災害 ID 付与方式では 1.9~2.45ms 程度であった。

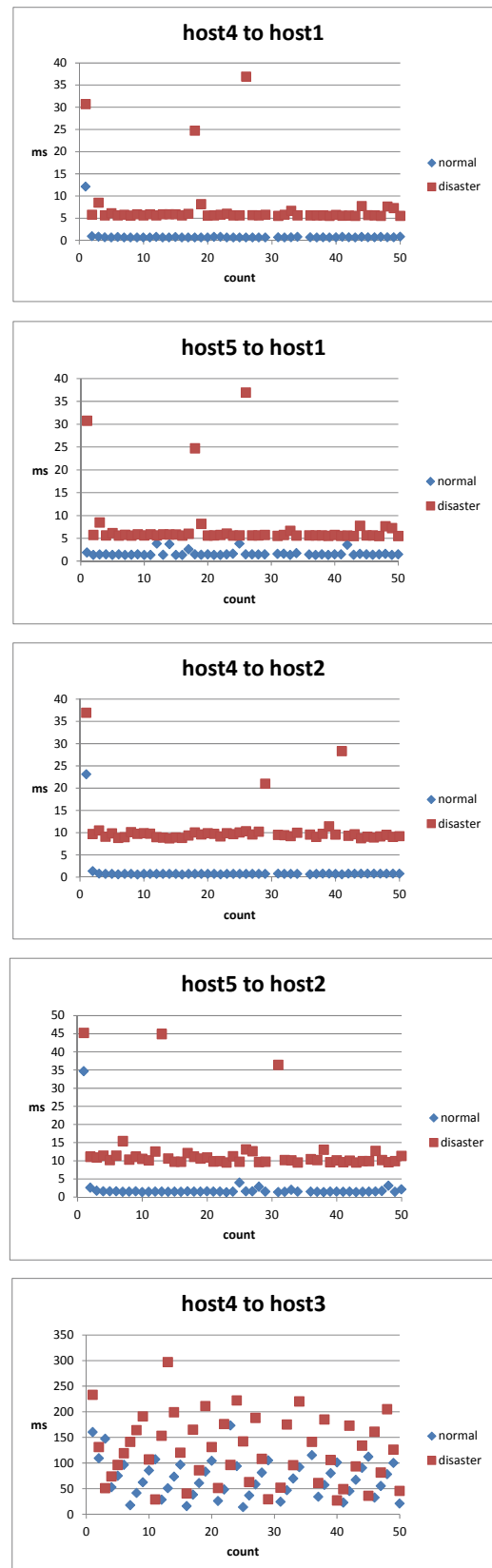
災害 ID 付与方式による遅延の原因は、SQL への問い合わせ処理が原因と考えられる。

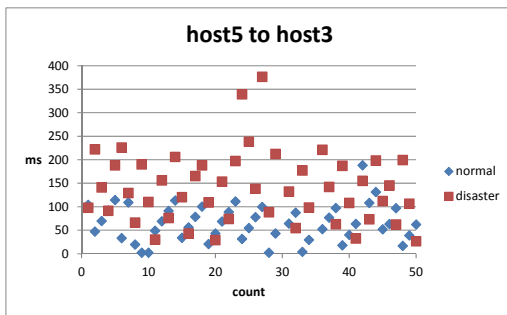
実験結果から、災害 ID 付与方式の場合には、Ping の応答速度に平均して 4.4~5.4 倍の時間がかかっていることがわかった。研究室に設置した OpenFlow スイッチを経由しない host1 と OpenFlow スイッチを経由した host2 の結果を比較しても、大きな差が見られないことがわかった。無線 LAN の端末への結果にはばらつきがあり、不安定さが見受

けられた。災害 ID 付与方式の場合には、平均して 1.8~2.2 倍の時間がかかっていることがわかった。

図 10 各端末への ICMP パケット送信の結果

Table 10 The Results of ICMP Packets Transmission to Terminals.





(2)iperf による実験結果

iperf を用い、様々な条件下にある送信元 IP アドレスから、送信元が災害地である場合と、非災害地である場合について 3 回ずつパケットの転送を実施した (表 1)。

送信元 IP アドレスが非災害地である場合には、災害地向けの通信は低帯域が利用され、送信 IP アドレスまたは受信先 IP アドレスが災害地である場合には広帯域が利用されることが確認できた。

また、2 台のスイッチを経由しても、通常時と遜色ない転送量を得ることが確認できた。

表 1 災害 ID 付与方式のパフォーマンス測定結果

Table 1 Performance Measurement Results of Disaster ID Assignment Method.

送信元	host1		host2		host3	
	通常時 (Mbits/sec)	災害時 非優先 (Mbits/sec)	通常時 (Mbits/sec)	災害時 優先 (Mbits/sec)	通常時 (Mbits/sec)	災害時 優先 (Mbits/sec)
host4	88.5	8.81	84.5	58.5	49.7	23.6
	88.5	8.94	84.5	58.4	51.0	26.4
	88.6	8.99	84.5	58.5	50.0	26.0
host5	39.1	9.19	40.2	36.1	42.6	24.6
	42.4	8.8	40.2	35	42.6	26.4
	34.4	9.14	39.7	37.6	42.6	22.4

## 9. 課題

本章では、本研究の課題と今後の展開について述べる。

### 9.1 位置情報の精度と必要範囲の粒度の調整

本研究の提案手法では IP アドレスと位置情報の対応関係が重要である。データベースによる対応では、データベースの破損や、問い合わせの速度がデータベースソフトウェアに依存する問題がある。データの更新滞った場合、災害地の IP アドレス情報が古い状態であるため、災害地への通信が優先的に転送されない可能性がある。今後、実装した IP アドレスマッピングデータベースの評価を継続し、性能向上を検討したい。

### 9.2 パフォーマンスの問題

今回、実施した検証が小規模であったため、多数のデータを用いての検証が不十分である。本提案手法は、データの問い合わせ方法やデータベースの性能に依存してしまう可能性がある。今回は時間の都合で間に合わなかったが、データベースへのデータの登録数が多い場合や、データが乱雑に登録されている場合など、状況によりネットワークの転送にどのように影響するか検討が必要である。

## 9.3 IP アドレス収集方法と個別端末の問題

今回までの実装では、個別の IP アドレスの収集方法については対応していない。携帯キャリアで導入されているラージスケール NAT やキャリアグレード NAT といった、網内でプライベートアドレスを用い、限定されたグローバル IP アドレスを共有するような利用形態では、IP アドレスと位置情報の対応関係の把握が困難となる。このような場合でも端末を特定し、位置情報との関係性を把握できる工夫が必要である。

## 10. まとめ

本論文では、OpenFlow により実装したシステムを用い、ネットワークを持つ拠点の位置情報やネットワーク情報の登録から実際の優先配送に至るまでの一連の動作についてフィールドテストにより、提案手法の実現可能性を確認した。

今後は提案手法を応用し、大学や企業などの限られた場所において、位置情報を活用した災害対策基盤の構築について検討を行っていく予定である。

## 参考文献

- 1) 小川 康一, 吉浦 紀晃, “OpenFlow を用いた災害時の通信品質の向上 ~ 位置情報に基づく災害 ID 付与方式の提案 ~,” 信学技報, vol.113, no.94, IA2013-5, pp.25-30, 2013 年 6 月.
- 2) “大震災からの教訓と ICT の役割”, 平成 24 年版情報通信白書”, pp.255-265, 総務省.
- 3) “大規模災害等緊急事態における通信確保の在り方について”, [http://www.soumu.go.jp/main\\_content/000141084.pdf](http://www.soumu.go.jp/main_content/000141084.pdf), 総務省大規模災害等緊急事態における通信確保の在り方に関する検討会, 2011 年 11 月 27 日.
- 4) Nick McKnown et al., “OpenFlow: enabling innovation in campus networks,” ACM SIGCOMM Computer Communication Review, vol.38, issue 2, pp.69-74, April 2008.
- 5) GeoIP, <http://dev.maxmind.com/geoip/>
- 6) Trema, <http://trema.github.io/trema/>
- 7) Open vSwitch, <http://openvswitch.org/>
- 8) 気象庁震度観測点一覧表 (平成 26 年 4 月 8 日現在), <http://www.data.jma.go.jp/svd/eqev/data/kyoshin/jma-shindo.html>
- 9) MySQL 5.1 リファレンスマニュアル, “16.2.2. Class Geometry”, <http://dev.mysql.com/doc/refman/5.1-olh/ja/geometry-property-function.html>
- 10) OpenWRT, <https://openwrt.org/>
- 11) iperf, <http://sourceforge.net/projects/iperf/>