

## 修士論文

**SDN と位置情報を利用し柔軟なグループ形成と効率的な配信を可能にしたジオキャスト手法に関する研究**

**A Study of Forming Flexible Groups and Efficient Geocasting using SDN and Locational Information.**

平成 25 年 2 月 6 日提出  
指導教員 関谷 勇司 准教授

東京大学大学院  
工学系研究科 電気系工学専攻  
37-116496 吉田 賢太郎

# 概要

インターネットの普及にともない、ネットワークを介した通信が社会インフラとして扱われ始めている。これに伴い、従来はアプリケーションが通信を行うためのインフラとして利用されていたインターネットが、さらに専門的な通信を行うためのインフラとして利用され始めている。特に動画配信に関しては、地デジ再配信のみならず、世界規模で様々な動画配信サービスが存在し、その動画配信に用いられる技術も様々である。さらに、これらの動画配信技術に加え、地域を限定して情報を配信するためのジオキャストも注目されている。

ジオキャストを行うにあたっては、グループ形成の柔軟性と配信精度の正確性が求められる。現在のインターネットでは、アプリケーションを用いたジオキャストを実現したり、IP マルチキャストのグループ形成を動的に行うことでジオキャストを実現している。しかし、IP アドレスが ID とロケータの二つの役割を同時に担っているため IP アドレスからノードの地理的な位置情報を知ることが難しい。そのため IP アドレスをグループ形成に用いていたのでは、IP ネットワーク上で特定の地理的な場所の範囲内のホストにのみ高精度に情報を送るということが難しくなる。アプリケーションを用いたジオキャストでは、いつフリーズしたり電源を切られるのかわからないコンピュータ上のマルチキャスト・クライアントが配信を担うため、安定的な配信を行うことが実現しにくい。IP マルチキャストはジオキャストのようにたびたびグループを変えて情報を配信するような通信には適さない。

そこで本研究では、Software Defined Network (SDN) を利用することによって、アプリケーションマルチキャストや IP マルチキャストでは実現不可能であった柔軟なグループ形成と効率的な配信を可能にするジオキャスト手法について提案し、規模性、グループ形成の柔軟性、配信の効率性について評価を行った。

## Abstract

With the spread of Internet, communication via networks has begun to be treated as a social infrastructure. Accordingly, Internet, which has been used as an infrastructure for applications, is beginning to be used as infrastructure for more specialized communication. In particular regarding video distribution, various video delivery services such as terrestrial digital rebroadcasting are present on a global scale and various technologies for video distribution exist. Furthermore, geocast to deliver information to a limited region is watched with interest in addition to video delivery technology.

Geocast requires accuracy of the delivery precision and flexibility of group formation. In the current Internet, geocast has been achieved using the application or dynamically forming groups with IP multicast. However, it is difficult to know the geographic location of the node from the IP address because the IP address plays the role of both the locator and the ID at the same time. This is why geocast using IP multicast has low accuracy. Because multicast client on a computer that does not know when to freeze or cut the power is responsible for the delivery in the geocast using the application, it is difficult to achieve a stable distribution. IP multicast is not suitable for communication in which group is changed frequently.

We propose a geocasting method utilizing the Software Defined Network (SDN) in order to enable the efficient delivery and flexible group formation that are not feasible with application layer multicast and IP multicast. We evaluated the scalability, flexibility of group formation and the efficiency of delivery.

# 目次

概要 .....	1
目次 .....	3
第 1 章 序論 .....	5
1-1 はじめに .....	5
1-2 論文の構成 .....	6
第 2 章 関連研究 .....	7
2-1 IP マルチキャスト .....	7
2-1-1 受信者指定型マルチキャスト .....	7
2-1-2 送信者指定型マルチキャスト .....	8
2-2 地理的な位置情報の推測 .....	8
2-2-1 Whois データベースの利用 .....	8
2-2-2 DNS を拡張する方式 .....	9
2-2-3 Ping を利用して測定した RTT から位置情報を推測 .....	9
2-2-4 GeoIP の利用 .....	11
2-3 既存手法の問題点 .....	11
第 3 章 問題解決のアプローチ .....	13
3-1 問題解決のアプローチ .....	13
3-2 OpenFlow .....	13
3-3 緊急地震速報 .....	21
第 4 章 提案手法の設計と実装 .....	24
4-1 提案手法の概要 .....	24
4-2 OpenFlow スイッチの位置情報取得 .....	24
4-3 メッセージフォーマット .....	25
4-4 ジオキャストする範囲の決定 .....	26
4-5 範囲内の OpenFlow スイッチにつながっている各ホストへのメッセージパケット送出 .....	26
第 5 章 評価 .....	28
5-1 研究室規模での動作検証 .....	28
5-2 規模性の評価 .....	29
5-2-1 実験の目的とシナリオ作成 .....	29
5-2-2 環境構築 .....	32
5-2-3 OpenFlow スイッチの総数が 10 台のときの実験結果 .....	32
5-2-4 OpenFlow スイッチの総数が 46 台のときの実験結果 .....	34
5-2-5 OpenFlow スイッチの総数が 100 台のときの実験結果 .....	36
5-2-6 OpenFlow スイッチの総数が 145 台のときの実験結果 .....	38
5-2-7 OpenFlow スイッチの総数が 199 台のときの実験結果 .....	40
5-3 規模性を評価する実験の結果の考察 .....	42
5-3-1 伝送遅延が送信者・受信者間の遅延に与える影響 .....	42
5-3-2 OpenFlow スイッチの台数が増えるときの遅延の増大の仕方 .....	43

5－3－3	送信者が送るメッセージの時間間隔が小さいときの遅延.....	46
第6章	考察.....	47
6-1	グループ形成の柔軟性の評価.....	47
6-2	配信の効率性の評価.....	47
第7章	おわりに.....	49
7-1	結論.....	49
7-2	今後の課題.....	49
謝辞	.....	51
参考文献	.....	52
発表文献	.....	53
付録	.....	54

# 第 1 章 序論

## 1-1 はじめに

インターネットの普及にともない、ネットワークを介した通信が社会インフラとして扱われ始めている。これに伴い、従来はアプリケーションが通信を行うためのインフラとして利用されていたインターネットが、さらに専門的な通信を行うためのインフラとして利用され始めている。具体的には、地デジ放送の再配信や、工業用機器の制御、警備用センサーや監視カメラのデータ送信や制御に利用され始めている。特に動画配信に関しては、地デジ再配信のみならず、世界規模で様々な動画配信サービスが存在し、その動画配信に用いられる技術も様々である。例えば、KeyHoleTV と呼ばれる動画配信サービスでは、Peer to Peer (P2P) と呼ばれる技術が動画配信に用いられており、YouTube やニコニコ動画、Ustream といった動画配信サービスでは、Contents Delivery Network (CDN) と呼ばれる技術が用いられている。また、地デジ放送の再配信では、IP マルチキャストという技術が用いられている。これらの技術に関しては、第 2 章にて詳しく述べる。

さらに、これらの動画配信技術に加え、地域を限定して情報を配信するためのジオキャストという技術も注目されている。近くにいる友達の検索、地理情報を利用した広告、車の運転者に事故情報を伝えるサービスなどに利用されている。IP アドレスでホストの位置を推測して聴取地域を限定している IP サイマルラジオ Radiko、緊急地震速報として使われる携帯電話のエリアメールもジオキャストである。ジオキャストを行うにあたっては、グループ形成の柔軟性と配信精度の正確性が求められる。

現在のインターネットでは、アプリケーションを用いたジオキャストを実現したり、IP マルチキャストのグループ形成を動的に行うことでジオキャストを実現している。しかし、現在の IP ネットワークにはノードを識別するための ID とノードの位置を示すロケータが一致しているという問題がある。地理的な位置情報を指標としてマルチキャストグループを作成しようとしたとき、IP アドレスが ID とロケータの二つの役割を同時に担っているため IP アドレスからノードの地理的な位置情報を知ることが難しい。そのため IP アドレスを用いていたのでは、IP ネットワーク上で特定の地理的な場所の範囲内のホストにのみ高精度に情報を送ることが難しくなる。

アプリケーションを用いたジオキャストでは、同時送信するパケットの中継作業をコンピュータにインストールしたマルチキャスト・クライアントが担当する。実際にジオキャストを行う際には何秒までに何台に対して配信しなければならないかという要件が存在するが、アプリケーションを用いたジオキャストでは、いつフリーズしたり電源を切られるのかわからないコンピュータ上のマルチキャスト・クライアントが配信を担うため、安定的な配信を行うことが難しい。

IP マルチキャストには受信者指定型マルチキャストと送信者指定型マルチキャストの 2 種類がある。受信者指定型マルチキャストでは、送信者が形成したグループに受信者が加入するという形でグループが形成される。送信者指定型マルチキャストでは送信者が受信者の IP アドレ

スをリスト化することによってグループを形成する。ジオキャストのように、情報を配信する地域をたびたび変えるような通信ではその都度マルチキャストグループを変更する必要がある。受信者指定型マルチキャストを用いてジオキャストを行う場合はグループを変えるたびに受信者に新たにグループに加入してもらう必要があり、送信者指定型マルチキャストを用いてジオキャストを行う場合はグループを変えるたびに送信者が情報を配信する地域に存在する受信者の IP アドレスをリスト化する作業が必要となる。このように、IP マルチキャストはジオキャストのようにたびたびグループを変えて情報を配信するような通信には適さない。

そこで本研究では Software Defined Network (SDN) を利用することによって、アプリケーションマルチキャストや IP マルチキャストでは実現不可能であった柔軟なグループ形成と効率的な配信を可能にするジオキャスト手法について提案し、評価を行う。

提案手法ではネットワーク側でグループ管理が行われるため、送信者は情報を配信したい地域を指定して情報を送信すればよい。ネットワーク側でグループ管理を行うことで、動的にグループを形成することができ、グループ形成に柔軟性を持たせることができる。

また、提案手法は IP アドレスではなく緯度・経度・階数という地理的な位置情報を用いて経路制御を行うため特定の地理的な場所の範囲内のホストにのみ高精度に情報を送ることができる。

## 1-2 論文の構成

本論文は 6 つの章から構成されている。以下に各章の構成を示す。

### 第 1 章

序論として、背景および研究の動機、目的を説明し、本論文の構成について述べる。

### 第 2 章

受信者指定型マルチキャスト、送信者指定型マルチキャスト、IP アドレスから位置情報を推測するための技術について概説する。また、それらの手法の問題点について述べる。

### 第 3 章

既存手法の問題を受けてどのように問題点を解決するのかを述べる。

### 第 4 章

本論文で提案する SDN と位置情報を利用した柔軟なグループ形成と効率的な配信を可能にしたジオキャスト手法について説明する。

### 第 5 章

本論文の提案手法について、規模性、グループ形成の柔軟性、配信の効率性について評価を行う。

### 第 6 章

本論文のまとめを行い、今後の課題について述べる。

## 第2章 関連研究

### 2-1 IP マルチキャスト

IP マルチキャストは、特定のグループ全員にデータを一齐配信するための通信方式である。ネットワークを構成するルータは、特定の端末が送信するマルチキャストパケットを複製して、複数のノードに配送する。IP マルチキャストには受信者指定型マルチキャストと送信者指定型マルチキャストの2つがある。

#### 2-1-1 受信者指定型マルチキャスト

マルチキャストパケットを送信する端末はソース、それを受信する端末はリスナーと呼ばれる。マルチキャストパケットはグループという単位で区別される。グループの識別子としてマルチキャストアドレスが定義されている。IPv4 では 224.0.0.0/4 の範囲が、IPv6 では ff00::/8 の範囲がマルチキャストアドレスとして割り当てられる。

図 2.1 の例では、232.0.0.1 と 232.0.0.2 という2つのグループのパケットが流れている。左側2つのリスナーは 232.0.0.1 というグループのパケットを受信しており、右側2つのリスナーは 232.0.0.2 というグループのパケットを受信している。中央のリスナーは両方のグループのパケットを受信している。

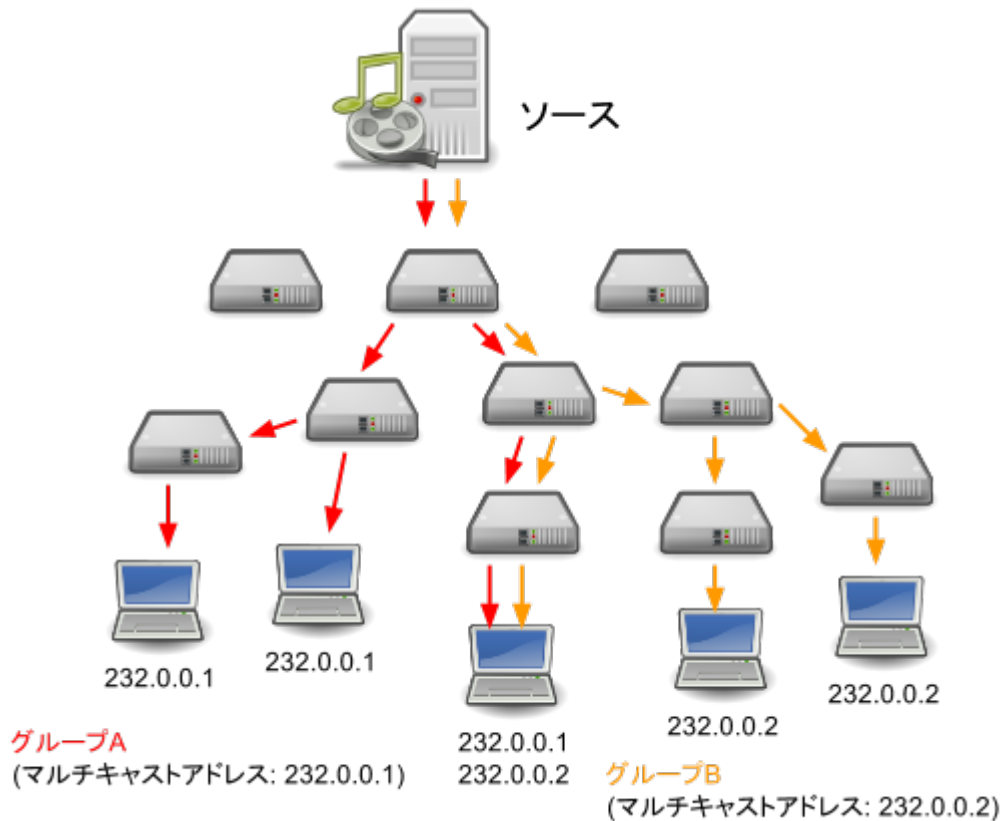


図 2.1 IP マルチキャストの例



パケットの IP ヘッダの宛先アドレスには、グループに対応するマルチキャストアドレスを格納する。ルータは、このマルチキャストアドレスを見て、パケットの転送先を判断する。ルータはグループごとに編成された経路表を持っているので、その経路表にしたがってパケットを配送する。経路表は、通常、PIM-SM、PIM-DM、DVMRP などのルーティングプロトコルによって自動的に生成される。なお、マルチキャストネットワークの配送経路は樹状になるので、配送経路の全体を表して配送木(Distribution Tree)と呼ばれる。

マルチキャストグループへの参加者を管理するプロトコルとして、IPv4 では IGMP(Internet Group Management Protocol)が、IPv6 では MLD(Multicast Listener Discovery)が利用されている。IGMP や MLD などのマルチキャストグループ管理プロトコルの目的は、リスナーがマルチキャストネットワークに対して、受信したいパケットのグループとソースを通知することである。

ルータはリスナーに対してクエリというメッセージを送信する。クエリを受信したリスナーは、ルータに対してレポートというメッセージを返す。レポートの中には、リスナーが受信したいパケットのグループとソースの情報が含まれる。レポートを受信したルータはその情報をルーティングに反映する。

## 2-1-2 送信者指定型マルチキャスト

送信者指定型マルチキャストとして XCAST6[7]が提案されている。XCAST は、受信者指定型マルチキャストにおいてマルチキャストグループの識別子として用いられていたマルチキャストアドレスを利用する代わりに、データの送信者が受信者のリストを管理し、パケットヘッダに到達すべきユニキャストアドレスのリストを明記することで宛先を指定する配送方式である。

送信者から受信者への経路上の XCAST ルータは、ヘッダの解析を行い、リストのすべての宛先についてユニキャストのルーティングテーブルを参照し、次の宛先のグループ形成を行う。そして、次のホップごとに適切なヘッダの書き換えを行い、パケットの転送を行う。宛先が残り 1 つとなった段階で、XCAST パケットは通常 のユニキャストパケットに変換され、宛先に対してユニキャストされる。また、経路上のルータが XCAST に対応していない場合には、XCAST パケットは通常 のユニキャストパケットとして扱われ、リストの最初の宛先に転送される。そして、XCAST ルータが最初のあて先で折り返して次のルータに転送される。

## 2-2 地理的な位置情報の推測

送信者指定型マルチキャストを用いてジオキャストを行う場合、受信者の IP アドレスのリストを作成する必要がある。そのときにどの IP アドレスがどういう位置情報を持っているのかを推測することになる。ここでは、IP アドレスから位置情報を推測する方法と地理的な位置情報から IP アドレスを推測する方法を説明する。

### 2-2-1 Whois データベースの利用

Whois とは、IP アドレスやドメイン名の登録者などに関する情報を、インターネットユーザーが誰でも参照できるサービスである。このサービスは、主に以下の目的でレジストリやレジストラが提供している。

- 1 ネットワークの安定的運用を実施する上で、技術的な問題発生の際の連絡のために必要な情報を提供する。
- 2 ドメイン名の申請・届け出時に、同一ドメイン名や類似ドメイン名の存在を確認するために必要な情報を提供する。
- 3 ドメイン名と商標などに関するトラブルの自律的な解決のために必要な情報を提供する。連絡を取るために登録してある電子メールアドレス、電話番号、郵便番号などから IP アドレスを割り当てている組織の住所を推測することができる。この方法を利用したさまざまなサービスが提供されている。この方法での誤差は数 km から数十 km となっている。

## 2-2-2 DNS を拡張する方式

IP ネットワークにおいてジオキャストを行う手法として DNS を拡張する方式がある[9]。この方式は従来の DNS サーバを拡張し、DNS エントリに地理的な位置情報を追加する。この目的には“.geo”というドメイン名がトップレベルドメインとして用いられる。第二レベルドメインは州、第三レベルドメインは国、第四レベルは地理的な位置のポリゴンを表す。このように拡張された DNS によって名前解決がされたあとは、ユニキャストでノードにメッセージが届けられる。もしくはマルチキャストグループにノードが属することによってマルチキャストを利用してメッセージが届けられる。

## 2-2-3 Ping を利用して測定した RTT から位置情報を推測

ICMP エコーメッセージを送るためのよく知られているコマンドラインツールとして Ping がある。Ping を利用することによってエコー応答が届くまでのラウンドトリップタイム(RTT)を測ることができる。ある固定した 2 地点間の RTT は次の二つの理由によって変動する。

- 1 データは一定でない速度でルータによって処理される。
- 2 ルータは異なる経路にパケットを送出する可能性がある。

RTT は変動するものの、ある条件が揃ったとき RTT は最小となる。ルータとエンドホストの処理時間とネットワークの物理層をパケットが流れる時間が最も小さくなったときである。この最小の RTT の値は、パケットがネットワークを流れた距離を指す指標として用いることができる。しかし、ルータとエンドホストの処理時間とネットワークの物理層をパケットが流れる時間が最も小さくなるような状態を作るとは難しいので、複数回 RTT を測定してその中から最小のものをとって近似する。この値を用いて位置を推定する方法を二つ紹介する。

最初の方法は、あるホストが他の固定された複数のホストと同様のネットワーク遅延を持つ場合は、その同様のネットワーク遅延を持つホストに近い場所に存在するという考えに基づく。

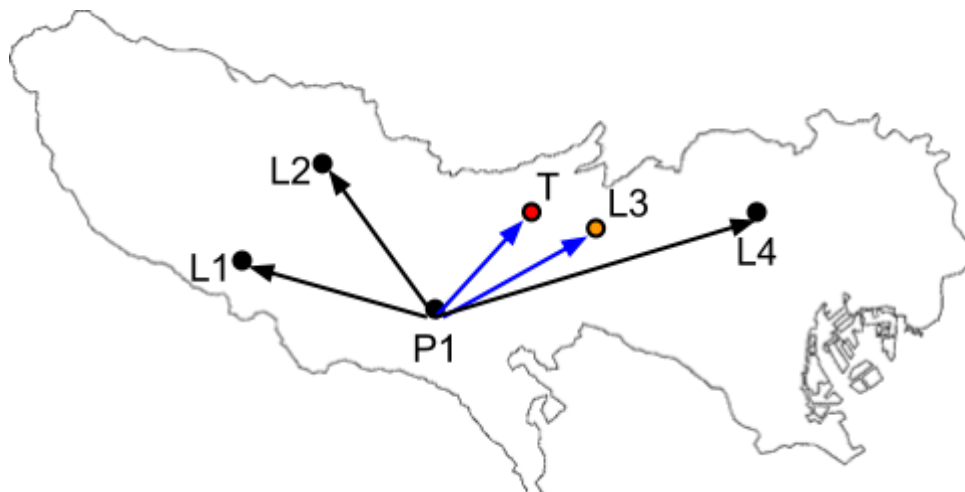


図 2.2 RTT を用いて位置を推測する方法 1

プローブマシンと呼ばれるホスト P1、同じく位置情報のわかっている複数のホスト L1、L2、L3...があるとする。L1、L2、L3...はランドマークと呼ばれる。P1 を用いてそれぞれのランドマークとの間の最小の RTT を計算し、記録する。位置情報がわからない新しいターゲットを T とする。P1 を用いて T との間の最小の RTT を計算して先ほど記録しておいた P1 とランドマーク間の最小の RTT と比較する。図 2.4 のように最も値が近かったランドマーク L3 に T は最も近いと判断する。この基本的なアプローチはプローブマシンを P1、P2、P3、...、Pn と複数にすることによって一般化される。最終的にこの方法によって、ターゲットに最も近いランドマークはどれなのかということがわかる。

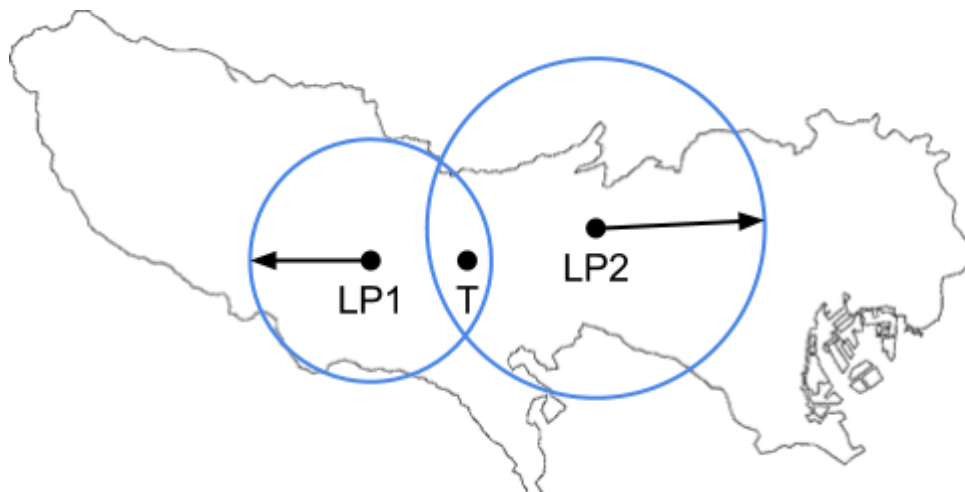


図 2.3 RTT を用いて位置を推測する方法 2

二つ目の方法はランドマークを用いて離散的に空間を区切らないという点で一つ目の方法と異なる。プローブマシンのように ping を送ることのできる LP1、LP2、...があるとする。LP1 から始めて、ターゲットホスト T との間の最小の RTT を計算し距離の制約へと変換する。例えば、T が LP1 から 200km 以内に存在すると推測できたとする。これは LP1 を中心とする半径 682km の円の内部に T が存在することを意味する。このプロセスを、LP2 を用いても行い、新たな距離の制約を得る。LP2 からは 500km 以内に存在することがわかったとする。T が存在する可能性のある範囲は図 2.4 のように二つの円の重なる部分になる。このように、さらに距離の制約

を増やしていくことによって T が存在すると考えられる範囲は小さくなっていく。Gueye ら [12]による実験によると西ヨーロッパでは 41 個、アメリカでは 94 個の制約を用いることによってターゲットの位置を推測することができる。誤差の中央値は西ヨーロッパでは 25km、アメリカでは 100km であった。

## 2-2-4 GeoIP の利用

GeoIP[5]は、MaxMind 社が提供する IP アドレスからどの国、地方、都市、郵便番号、市外局番に所在するかという位置情報を教えるサービスである。MaxMind 社によると、位置情報の正確さは国レベルで 99.8%、アメリカの州レベルで 90%、アメリカの都市レベルで 83%である。

## 2-3 既存手法の問題点

まず、既存手法の問題点をそれぞれの方法について列挙する。

### IP マルチキャスト

- 1 グループ形成に柔軟性がない。受信者指定型マルチキャストでは新しくマルチキャストグループを作った際には、受信者がそのマルチキャストグループをリッスンする必要があるので、送信者がどんどん新しいグループを作って送るというのには向かない。送信者指定型マルチキャストも新しいグループを作成する度に新しい受信者のリストを作り直して送る必要がある。このようにグループ管理が複雑になってしまう。

### Whois データベースの利用

- 1 位置情報の精度が悪い。ホストの IP アドレスが登録されている組織の近くにターゲットとなるホストが存在するとは限らない。この方法を用いるとアドレスのブロックが大きくても一つの場合へとマッピングされてしまう。
- 2 Whois データベースの情報は登録者が行うので、間違ったデータが登録される可能性がある。
- 3 情報が古い可能性がある。Whois データベースには最後に更新された日付もたいてい記録されているが、この日付が古すぎると情報が間違っている可能性がある。

### DNS を拡張する方式

- 1 ドメインネームを登録しているホストにしか使えないため、IP アドレスをもらってネットワークに接続しているだけのホストの位置情報を知ることができない。

### Ping を利用して測定した RTT から位置を推測

- 1 すべてのターゲットホストが ICMP エコーメッセージを返してくるとは限らない。実際、ping を返さないように設定されているホストが増加している。
- 2 最小の RTT を集めるために何度も Ping をターゲットホストに実行するので、攻撃に見えてしまう。例えば 40 個ほどの距離の制約を得るためにそれぞれ 10、15 回 ping を実行するとすると合計で 400 回から 600 回 Ping を行うことになる。
- 3 ターゲットホストのネットワークへのアクセスがダイヤルアップなどの遅延の大きい接続方法で行われている場合は位置情報の精度が悪くなってしまう。

- 4 ターゲットホストがエコー応答を返すのを意図的に遅らせることによって、RTT を本来の値よりも大きくして位置情報の精度を悪くすることができる。

#### GeoIP の利用

- 1 もともと用途が広告提供、詐欺スクリーニング、ウェブ分析、ファイアウォール機能の一部、スパムからのユーザ保護、フィッシング対策であるため都市レベルまでの位置情報の提供しかされていない。

このように位置情報を推測する方法は様々なものがあるが、高い位置情報の精度と粒度を持ち、幅広いホストに対して位置情報を推測することができる方法は存在しない。

## 第3章 問題解決のアプローチ

### 3-1 問題解決のアプローチ

本研究では、マルチキャストにおけるグループ作成の指標として地理的な位置情報に絞り込む。受信者を地理的な位置情報によって受信者を限定してマルチキャストを行うことは、ジオキャストと呼ばれている。ジオキャストを行う際の限定する位置情報の範囲をジオキャストエリアと呼ぶ。

IP ネットワーク上でジオキャストを実現するためには、2 点の解決しなければならない問題がある。一点目はマルチキャストグループへの受信者の加入・離脱によってグループ管理が複雑になる点である。二点目は IP ネットワークにおいて IP アドレスが ID とロケータの二つの役割を担っているために位置情報でグループを形成するときの精度が低いという点である。

本論文ではこの問題に対処するため、グループ管理の複雑さの軽減と地理的な位置情報によるグループ形成の精度を高めて効率的な配信を可能にすることを目的とし、それを実現するための手段として SDN を利用する手法を提案する。さらにこの手法を実現するために OpenFlow を利用し、OpenFlow スイッチに記述した位置情報をグループ作成と経路制御に用いる手法の提案を行う。本論文ではグループ形成の柔軟性と配信の効率性を必要とする通信の例として緊急地震速報を例に挙げる。

### 3-2 OpenFlow

ここで、提案手法で用いた OpenFlow の概要を述べる。OpenFlow は、ソフトウェアによってネットワークの構成を行うための新しい標準である。2009 年 12 月 31 日にスタンフォード大学が中心となってバージョン 1.0 が策定された。2012 年 4 月 19 日に策定されたバージョン 1.3 では IPv6 環境での運用が可能になった。本研究ではバージョン 1.0 を用いた。

2011 年 3 月 21 日には、グーグル、マイクロソフト、Yahoo!、ドイツテレコム、ベライゾン、Facebook などが中心となって Open Networking Foundation が結成された。その後 OpenFlow は Open Networking Foundation が中心となって、標準化と普及が推進されている。

OpenFlow の最大の特徴は、これまで 1 台のネットワーク機器の中に組み込まれていた、ネットワークの経路制御機能とデータ転送機能を分離したことにある。従来のスイッチは、1 台のスイッチの中にパケット伝送機能、パケット制御機能、設定機能が実装されている (図 3.1)。一方、OpenFlow ではパケット制御機能と設定機能がスイッチから分離され OpenFlow コントローラに集約されるため、複数のスイッチの一元管理が可能になる (図 3.2)。

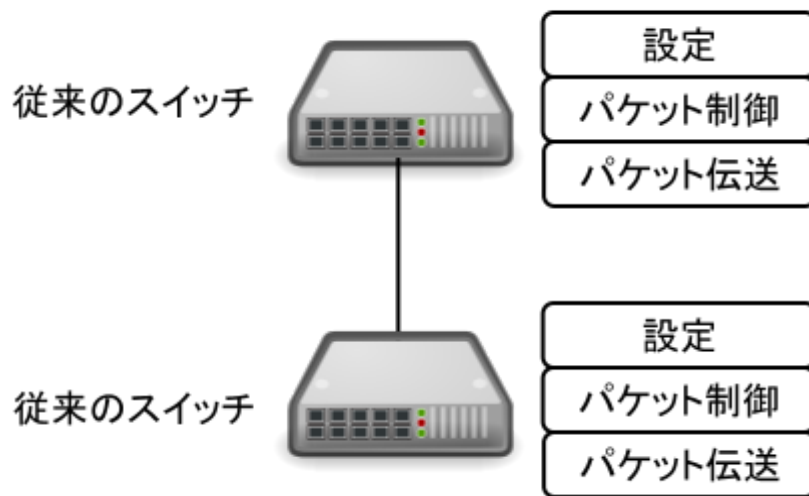


図 3.1 従来のスイッチのスタック構成

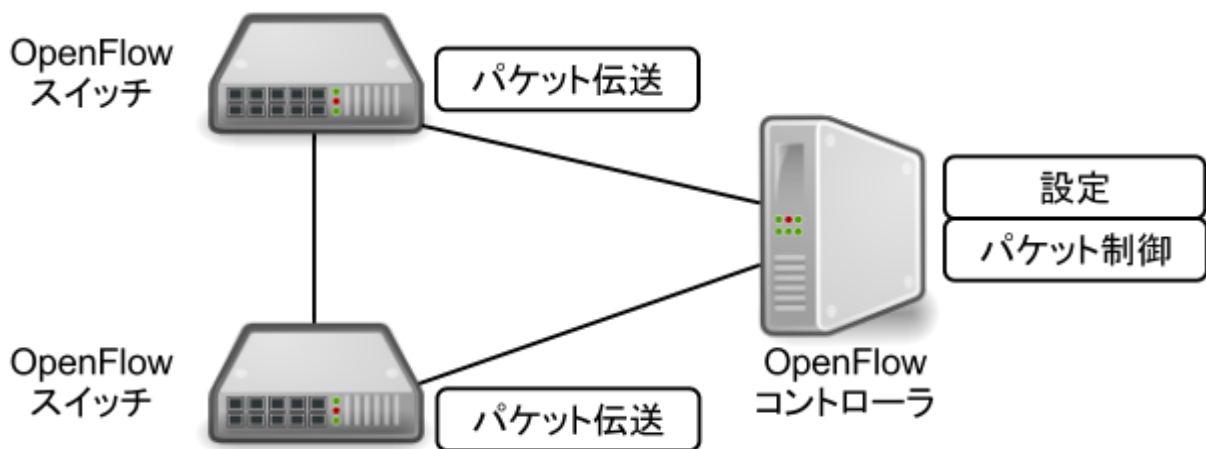


図 3.2 OpenFlow のスタック構成

OpenFlow は、経路制御を司る OpenFlow コントローラと、データ転送機能を備える OpenFlow スイッチ、そして OpenFlow コントローラと OpenFlow スイッチがコミュニケーションをするための OpenFlow プロトコルからなる。

一般に OpenFlow コントローラはソフトウェアで実装され、OpenFlow スイッチはネットワーク機器としてハードウェアで実現されるが、OpenFlow スイッチはソフトウェアで実装することも可能である。代表的なものに Open vSwitch がある。

OpenFlow では、OpenFlow コントローラが「この条件の packets に対してはこのように動作する」という、処理対象となる packets の条件とアクションが含まれるルール群をフローテーブルとして定義し、これを OpenFlow スイッチに書き込む(図 3.3)。

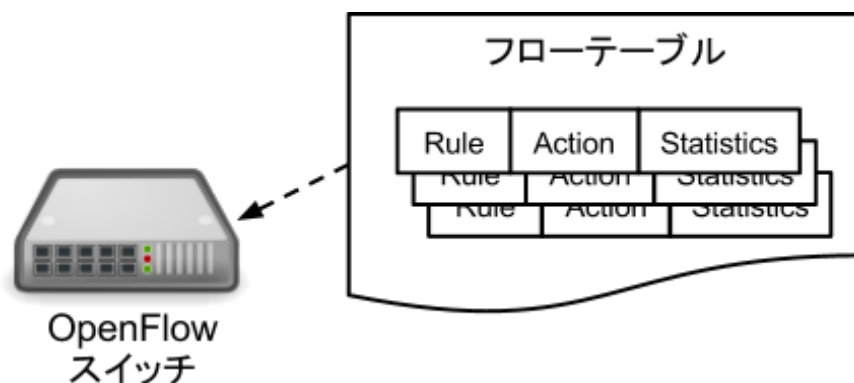


図 3.3 フローテーブル

条件には、ポート番号、通信元イーサネットアドレスや送信先 IP アドレス、VLAN ID、MPLS ラベルなどさまざまなフィールドとカウンタによる統計情報が使われる。そして条件に該当するパケットに対するアクションとして、特定のスイッチポートから出力する、フィールドを書き換える、ドロップする、などが指定できる。

例えば「行き先物理アドレスが XXXX ならば、スイッチの 3 番ポートから出力する」「TCP の 25 番ポートのパケットが来たら、行き先 IP アドレスを YYYY に書き換える」などの定義が可能である。例えばレイヤ 2 の情報を基にしたルールならば従来のスイッチのように振るまい、レイヤ 3 の情報を基にしたルールならばルータのように振るまえる。また、特定のパケットをドロップすればファイアウォールのようにもなり、従来のネットワーク機器ではできなかったようなレイヤをまたがった振るまいも定義できる。

OpenFlow スイッチは通常、OpenFlow スイッチ内に書き込まれたフローテーブルに従ってパケットを処理するため、OpenFlow コントローラとの通信を行うことなく高速にパケットを処理していく。ただし、フローテーブルのルールが見つからないパケットが来た場合には、OpenFlow コントローラへの問い合わせが発生する。

OpenFlow バージョン 1.0 において、パケットの条件には 12 種類の設定が可能である。この 12 種類の条件のことを OpenFlow では 12 タプルと呼ぶ。表に 12 タプルの一覧を示す。アクションにはパケットをどのように処理するかを設定することができる。表 3.1 に指定可能な代表的な値を示す。

表 3.1 ヘッダフィールドに指定可能な条件の一覧

No	ヘッダフィールドに指定可能な条件
1	OpenFlow スイッチの物理ポート
2	宛先 MAC アドレス
3	送信元 MAC アドレス
4	Ether タイプ
5	VLAN ID



6	VLAN プライオリティ
7	宛先 IP アドレス
8	送信元 IP アドレス
9	IP プロトコル種別
10	IP ToS 値
11	宛先 L4 ポート番号
12	送信元 L4 ポート番号

パケットのどこを書き換え、どこに転送するかは、すべて OpenFlow コントローラが決定する。OpenFlow スイッチは OpenFlow コントローラから指示を受けて動作する。OpenFlow スイッチが独自の判断で動作することは基本的にない。未知のパケットを受信した場合、OpenFlow スイッチは OpenFlow コントローラにパケットの制御方法について問い合わせをする。OpenFlow には 3 種類のパケット制御方法がある。まずは、その 3 種類の制御方式についてそれぞれ説明する。

制御方式 1 は、OpenFlow の最も基本的な制御パターンである。この制御方式は次のように動作する。

- 1 サーバ 1 はサーバ 2 に対してパケットを送信する。
- 2 OpenFlow スイッチはサーバ 1 からパケットを受け取り、受信したパケットに対応する制御ルールを自身が保持していないか確認する。
- 3 受信したパケットに対応する制御ルールを保持している場合、OpenFlow スイッチは制御ルールに従ってパケットを書き換え、転送を行う。
- 4 OpenFlow スイッチから転送されたパケットがサーバ 2 に届く。

制御方式 1 では、パケットの任意の個所を書き換えることはできない。書き換え可能な個所は、OpenFlow バージョン 1.0 の場合はヘッダフィールドに設定可能な 12 種類になる。

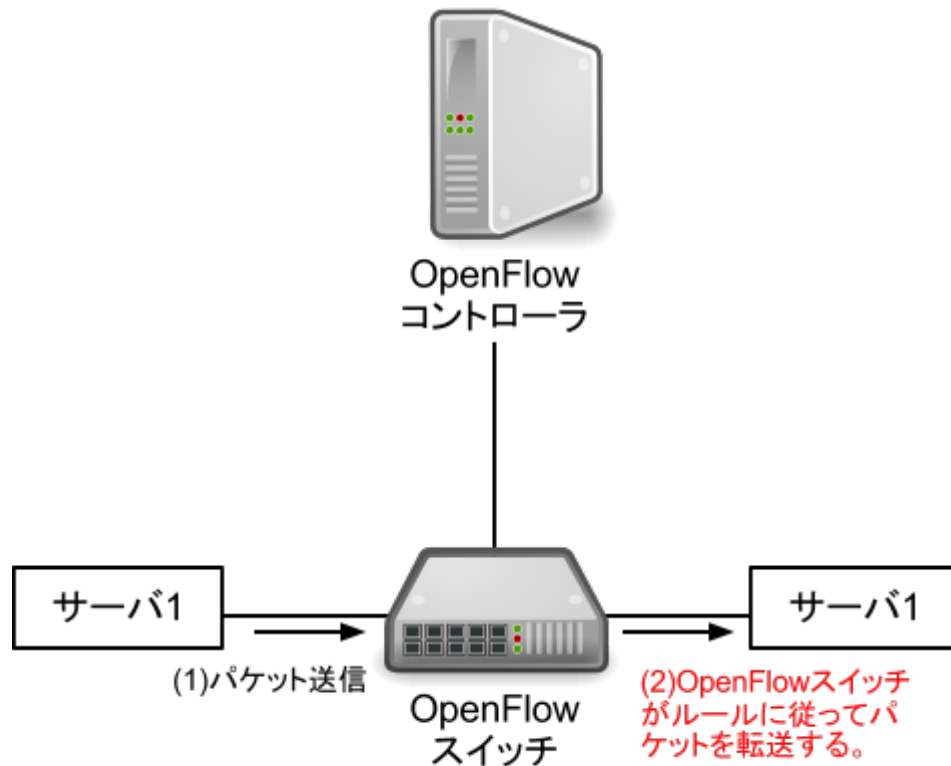


図 3.4 パケット制御方式 1

制御方式 2 は、OpenFlow を用いてネットワークを構築した直後によく発生するパターンである。この制御方式は以下のように動作する。

- 1 サーバ 1 はサーバ 2 に対してパケットを送信する。
- 2 OpenFlow スイッチはサーバ 1 からパケットを受け取り、受信したパケットに対応する制御ルールを自身が保持していないか確認する。
- 3 受信したパケットに対応する制御ルールを保持していない場合、OpenFlow スイッチは受信したパケットをバッファに一時保存し、OpenFlow コントローラにパケットの制御方法を問い合わせる。
- 4 OpenFlow コントローラは、OpenFlow スイッチが受信したパケットに関する情報を受け、パケットの制御方法を決定する。
- 5 OpenFlow コントローラは、OpenFlow スイッチにパケットの制御方法を指示する。同時に、OpenFlow スイッチに制御ルールを書き込む。
- 6 OpenFlow スイッチは OpenFlow コントローラの指示に従い、OpenFlow スイッチのバッファに一時保存されているパケットを書き換え、サーバ 2 にパケットを転送する。

上記の処理を行うと、OpenFlow スイッチに制御ルールが書き込まれる。その状態で再びサーバ 1 から同様のパケットを受信すると、今度は OpenFlow スイッチに制御ルールが書き込まれているため、OpenFlow スイッチは OpenFlow コントローラに問い合わせることなく、制御方式 1 と同じ要領でパケットを転送する。制御方式 2 では OpenFlow

スイッチでパケットを書き換えるため、パケットを書き換えることができる個所は制御方式 1 と同様に 12 種類になる。

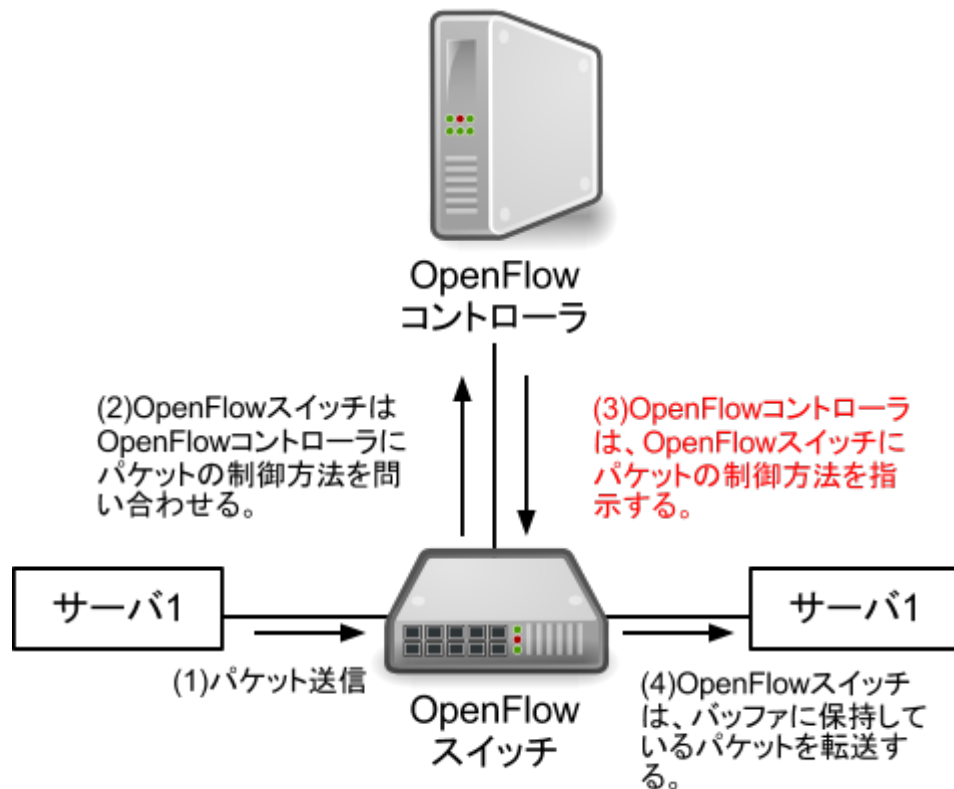


図 3.5 パケット制御方式 2

制御方式 3 は制御方式 2 の応用になる。この制御方式は次のように動作する。

- 1 サーバ 1 はサーバ 2 に対してパケットを送信する。
- 2 OpenFlow スイッチはサーバ 1 からパケットを受け取り、受信したパケットに対応する制御ルールを自身が保持していないか確認する。
- 3 受信したパケットに対応する制御ルールを保持していない場合、OpenFlow スイッチは受信したパケットをバッファに一時保存し、OpenFlow コントローラにパケットの制御方法を問い合わせる。
- 4 OpenFlow コントローラは、OpenFlow スイッチが受信したパケットに関する情報を受け、パケットの制御方法を決定する。
- 5 OpenFlow コントローラは必要に応じて OpenFlow スイッチから受信したパケットを書き換え、書き換えたパケットを転送するよう OpenFlow スイッチに指示する。書き換えられたパケットに OpenFlow ヘッダを付加したパケットが OpenFlow コントローラから OpenFlow スイッチに送信される。
- 6 OpenFlow スイッチは OpenFlow コントローラの指示に従い、OpenFlow コントローラから受信したパケットをサーバ 2 へ転送する。ただし、厳密にはパケットの転送先は OpenFlow コントローラが決定する。

制御方式 3 は OpenFlow スイッチに制御ルールを書き込まない。よって、サーバ 1 から同様のパケットを受信した場合、OpenFlow スイッチはパケットの制御方法がわからないため、前回と同様に OpenFlow コントローラにパケットの制御方法を問い合わせる。また、制御方式 3 はサーバ側でパケットの書き換えを行う。よって、この制御方式ではパケットのあらゆる場所を書き換え可能である。さらに、場合によってはパケットを新規に生成し、サーバ 1 から送信されたパケットとはまったく異なるパケットをサーバ 2 に届けることもできる。

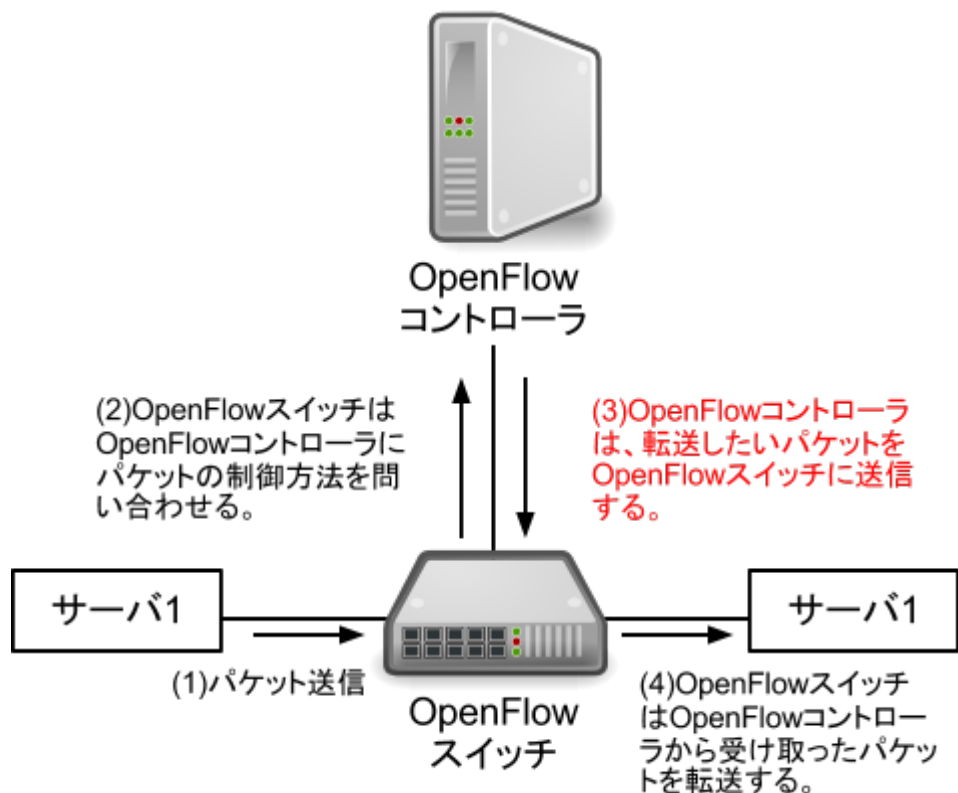


図 3.6 パケット制御方式 3

OpenFlow プロトコルに従ってやり取りされるメッセージを列挙すると以下の表 3.2 のようになる。

表 3.2 OpenFlow プロトコルに従ってやり取りされるメッセージ

メッセージ	利用用途
Hello	初期接続時のネゴシエーション
Echo Request/Reply	OpenFlow スイッチの死活監視、および応答
Vendor	ベンダが独自定義
Features Request/Reply	OpenFlow スイッチの機能・特徴の問い合わせ、および応答
Set Config Request	OpenFlow スイッチのコンフィグレーションの設定
Get Config Request/Reply	OpenFlow スイッチのコンフィグレーションの取得、および応答
Flow Mod	OpenFlow スイッチのフローの登録・変更・削除
Port Mod	OpenFlow スイッチの物理ポートの状態変更
Queue Get Config	OpenFlow スイッチの Queue のコンフィグレーションの取得、およ

Request/Reply	び応答
Statics Request/Reply	OpenFlow スイッチの統計情報の取得、および応答
Packet Out	OpenFlow スイッチから Packet を送出させる
Barrier Request/Reply	OpenFlow スイッチに送ったこれまでのメッセージに対する処理が完了したかを確認、およびその応答
Packet In	OpenFlow スイッチがパケットを受信したことを OpenFlow コントローラに通知
Flow Removed	OpenFlow スイッチ上のフローが有効時間切れになったことを OpenFlow コントローラに通知
Port Status	OpenFlow スイッチのポートの状態が変化したことを OpenFlow コントローラに通知
Error	エラーがあったことを通知する

ここではそのうち利用頻度の高いもの、OpenFlow スイッチのふるまいに大きな影響を与えるものについて詳しく説明していく。

Flow Mod メッセージは OpenFlow スイッチで記述されているフローを書き換えたい場合に OpenFlow コントローラから OpenFlow スイッチに対して送信するメッセージである。新しく OpenFlow スイッチにフローを追加する場合もこの Flow Mod メッセージが用いられる。

OpenFlow スイッチのフローテーブルのフローを修正、追加する際にはフローのデータのうちヘッダフィールドおよびアクションを指定する。つまり、マッチングに用いるためのヘッダフィールドおよび、ヘッダフィールドが一致した際にどのように処理するかを記述したアクションをセットにして、OpenFlow コントローラからスイッチに送信する。

Packet Out メッセージは OpenFlow スイッチからパケットを送出させるために、OpenFlow コントローラから OpenFlow スイッチに対して送信するメッセージである。この際、OpenFlow スイッチのどのポートからどのようなパケットを送出させるかについては OpenFlow コントローラにて自由に指定することができる。例えば、OpenFlow コントローラにて LLDP パケットを生成し OpenFlow スイッチの全ポートからパケットを送出するといったことが可能になる。

Packet In メッセージは OpenFlow スイッチから OpenFlow コントローラに対して、パケットを受信したことを通知するメッセージである。例えば、OpenFlow スイッチは、どのフローのヘッダフィールドにも一致しない未知のパケットを受け取った時に、このメッセージを用いて OpenFlow コントローラに対してそのパケットを通知する。OpenFlow コントローラは通知の内容を解析し、ふるまいを決定する。一般的には、OpenFlow コントローラ側ではそのパケットの内容を判断したうえで、OpenFlow スイッチに対して Flow-Mod や Packet-Out といったメッセージを送信する。

Statics Request メッセージは OpenFlow スイッチに置いて管理されている統計情報の取得をするためのメッセージである。統計情報としては、フロー単位・ポート単位などの単位でパケット数・バイト数などの情報を取得することができる。また、OpenFlow スイッチのフローテーブルで管理されているフロー数などもこのメッセージを送ることで取得することが可能である。Statics Reply メッセージで OpenFlow コントローラに通知される。

Features Request メッセージは OpenFlow スイッチのデータパス ID、システム記述、バッファのサイズを取得するためのメッセージである。データパス ID とは OpenFlow スイッチを識別するための 64 ビットの ID である。Features Reply メッセージでこれらの情報は OpenFlow コントローラに通知される。

OpenFlow コントローラと OpenFlow スイッチがつながるまでの流れを図 3.7 に示す。

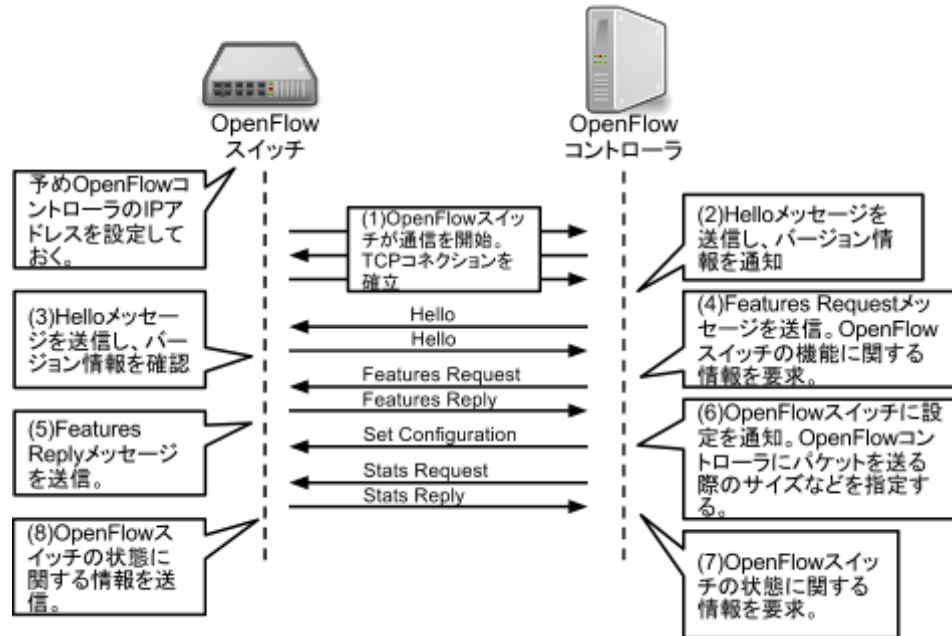


図 3.7 OpenFlow コントローラと OpenFlow スイッチがつながるまでの流れ

OpenFlow コントローラを作るフレームワークとして NOX[8]、POX[4]、Trema[3]などが開発されている。NOX は C++/Python、POX は Python、Trema は C/Ruby で記述する。

仮想スイッチとして Open vSwitch[2]が開発されている。この Open vSwitch を OpenFlow スイッチとしても利用することができる。基本的な構成要素は、Open vSwitch デーモン(ovs-vswitchd)、構成情報を管理するデータベースサーバ(ovsdb-server)、パケットの転送処理を行うカーネルモジュール(openvswitch\_mod.ko)の三つである。Open vSwitch 設定の流れは次のようになっている。

1. ovs-vsctl コマンドが ovsdb-server に設定情報を送信する。
2. ovsdb-server が ovs-vswitchd に設定情報を送信する。
3. ovs-vswitchd が openvswitch\_mod を設定する。
4. ovs-vswitchd が ovsdb-server に設定結果を送信する。
5. ovsdb-server が ovsdb に設定情報、設定結果を登録する。
6. ovsdb-server が ovs-vsctl コマンドに設定結果を送信する。

### 3-3 緊急地震速報

本論文では、提案手法のアプリケーションの例として緊急地震速報を挙げる。気象庁のウェブページによると緊急地震速報の仕組みは次のように述べられている[6]。以下、この内容を適宜要約し、引用して紹介する。

緊急地震速報は、地震の発生直後に、各地での強い揺れの到達時刻や震度を予想し、可能な限り素早く知らせる情報のことである。強い揺れの前に、自らの身を守ったり、列車のスピードを落としたり、あるいは工場等で機械制御を行うなどの活用がなされている。

地震が発生すると、震源からは揺れが波となって地面を伝わっていく（地震波）。地震波には P 波（Primary「最初の」の頭文字）と S 波（Secondary「二番目の」の頭文字）があり、P 波の方が S 波より速く伝わる性質がある。一方、強い揺れによる被害をもたらすのは主に後から伝わってくる S 波である。このため、地震波の伝わる速度の差を利用して、先に伝わる P 波を検知した段階で S 波が伝わってくる前に危険が迫っていることを知らせることが可能になる。

緊急地震速報には、全国約 235 箇所（平成 24 年 3 月現在）の地震計に加え、防災科学技術研究所の高感度地震観測網（全国約 850 箇所）を利用している。多くの地震計のデータを活用することで、地震が起きたことを素早くとらえることができる。

発表する内容は、地震が発生した場所や、震度 4 以上の揺れが予測された地域名称などである。

具体的な予測震度の値は、 $\pm 1$  程度の誤差を伴うものであること、及び、できるだけ続報は避けたいことから発表せず、「強い揺れ」と表現することとされている。震度 4 以上と予測された地域まで含めて発表するのは、震度を予測する際の誤差のため実際には 5 弱である可能性があることと、震源域の断層運動の進行により、しばらく後に 5 弱となる可能性があるというふたつの理由による。

猶予時間については、気象庁から発表する対象地域の最小単位が、都道府県を 3～4 つに分割した程度の広がりを持ち、その中でも場所によってかなり異なるものであるため、発表されない。

平成 18 年 8 月 1 日より先行的に活用できる分野について提供している緊急地震速報は、機器制御などの高度な利用者向けとして、平成 19 年 10 月 1 日以降も、引き続き提供している。また、各家庭用の端末などで、高度利用者向けの緊急地震速報（予報）を受信し、受信地点の予測震度や主要動到達予想時刻などを表示する等にも利用されている。

高度利用者向けの緊急地震速報（予報）の内容・発表条件については次の通りである。

1. 高度利用者向けの緊急地震速報（予報）の内容：地震の発生時刻、地震の発生場所（震源）の推定値、地震の規模（マグニチュード）の推定値、予測される最大震度が震度 3 以下のときは、予測される揺れの大きさの最大（最大予測震度）、予測される最大震度が震度 4 以上のときは、地域名に加えて震度 5 弱以上と予測される地域の揺れの大きさ（震度）の予測値（予測震度）、その地域への大きな揺れ（主要動）の到達時刻の予測値（主要動到達予測時刻）
2. 高度利用者向けの緊急地震速報（予報）の発信条件：気象庁の多機能型地震計設置のいずれかの観測点において、P 波または S 波の振幅が 100 ガル以上となった場合。地震計で観測された地震波を解析した結果、震源・マグニチュード・各地の予測震度が求まり、そのマグニチュードが 3.5 以上、または最大予測震度が 3 以上である場合。

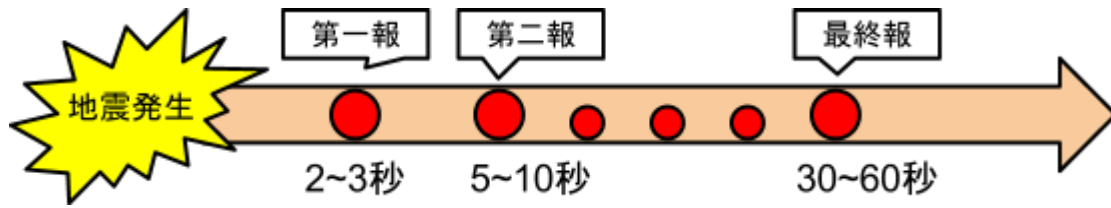


図 3.8 高度利用者向け地震速報

緊急地震速報（予報）が従来の地震情報と異なる点はその迅速性である。気象庁は緊急地震速報（予報）として下図のように地震を検知してから数秒～1 分程度の間数回(5～10 回程度)発表する。第 1 報は迅速性を優先し、その後提供する情報の精度は徐々に高くなっていく。ほぼ精度が安定したと考えられる時点で最終報を発表し、その地震に対する緊急地震速報の提供を終了する。

以上が、緊急地震速報の仕組みの紹介である。本論文ではグループ形成の柔軟性と配信の効率性を必要とするジオキャストとして緊急地震速報を想定し提案手法の評価を行う。特に 5 章 2 節の提案手法の規模性の評価では、OpenFlow スイッチを日本全国に展開して緊急地震速報を送信者が送信するという想定で評価を行う。詳細は 5 章で述べる。



## 第 4 章 提案手法の設計と実装

### 4-1 提案手法の概要

提案手法を構成する要素は送信者、受信者、OpenFlow コントローラ、OpenFlow スイッチの 4 つである。提案手法の目的は、柔軟性のあるグループ形成と効率的な配信を可能にするマルチキャスト手法の実現である。

提案手法は以下のように動作する。

- 1 送信者は、予め決められた特定のマルチキャストアドレス、ポート番号宛てに UDP でメッセージを送る。提案手法用の制御ヘッダとメッセージは UDP のペイロード部に入る。
- 2 送信者がつながっている OpenFlow スイッチにメッセージパケットが届くと、未知のパケットなのでその OpenFlow スイッチは OpenFlow コントローラに制御方法を問い合わせる。このときにパケットの情報も OpenFlow コントローラに送られる。
- 3 OpenFlow コントローラは制御ヘッダの情報を元にジオキャストする範囲を決定する。ジオキャストする範囲が決まったあと、その範囲に含まれる OpenFlow スイッチのデータパス ID を求める。
- 4 3 で求めたデータパス ID を持つ OpenFlow スイッチにつながっているホストすべてにメッセージパケットを送る。
- 5 受信者は予め決められているマルチキャストアドレスをリッスンしておき、メッセージパケットが届いたら UDP ペイロード部から制御ヘッダをのぞきメッセージを表示する。

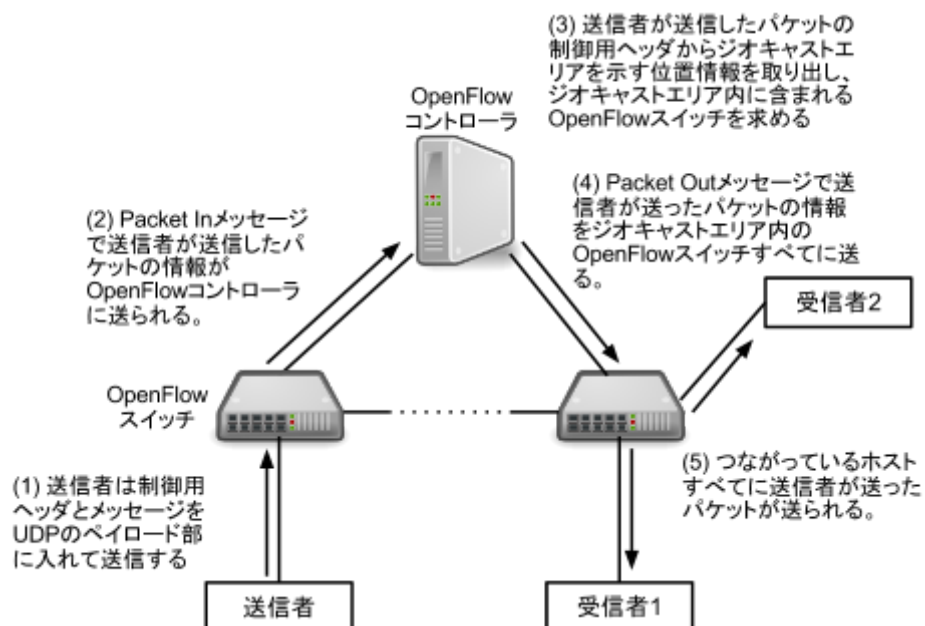


図 4.1 提案機構の動作

### 4-2 OpenFlow スイッチの位置情報取得

本研究では Open vSwitch[2]を OpenFlow スイッチとして用いる。Open vSwitch には、システム記述部という管理者が自由に情報を文字列で記述できる設定項目がある。提案手法は、既存の OpenFlow プロトコルに影響を与えずに OpenFlow スイッチの位置情報を伝達することを目的とし、位置情報をシステム記述部に埋め込むことにした。実装は、Open vSwitch の制御プログラムである ovs-vswitchd を起動する際に、緯度や経度、この OpenFlow スイッチが置かれた建物の階数といった位置情報をシステムの環境変数から読み取り、システム記述部に格納する。OpenFlow プロトコルには Features Request、Features Reply というやりとりが存在する。このやりとりは OpenFlow コントローラと OpenFlow スイッチ間で定期的に行われ、OpenFlow スイッチのデータパス ID、システム記述、バッファのサイズが OpenFlow コントローラに伝えられる。すべての OpenFlow スイッチのシステム記述が OpenFlow コントローラに伝えられるので、OpenFlow コントローラは全 OpenFlow スイッチの位置情報を把握することができる(図 4.2)。

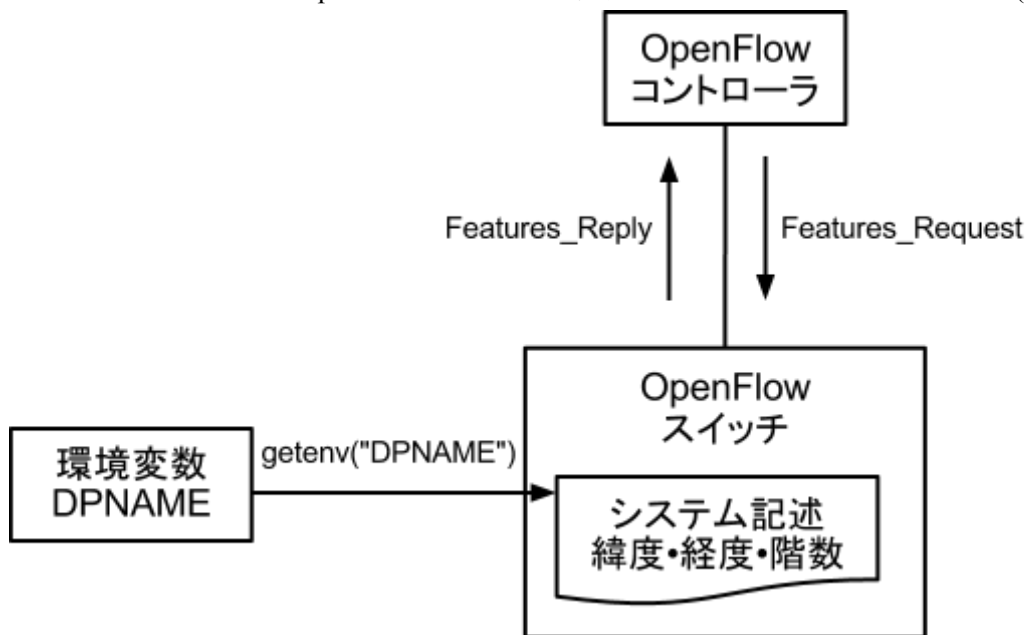


図 4.2 OpenFlow スイッチの位置情報取得

また、本研究では Trema [3] を OpenFlow コントローラとして用いる。Trema には様々なアプリケーションが含まれているが、中でも show\_description というツールは Feature Request を送信することにより全ての OpenFlow スイッチに設定された情報を得る機能を持つ。本研究の実装は、この show\_description を用いて、OpenFlow スイッチの位置情報を抽出する。

### 4-3 メッセージフォーマット

メッセージフレームの構成は図 4.3 のようになっている。先頭から順にジオキャストであることを示すマジックコード、グループ名、緯度低、緯度高、経度低、経度高、階数低、階数高、メッセージが入る。緯度低から階数高までのフィールドはジオキャストする範囲を指定するために用いる。マジックコードとグループ名は 8 バイト、範囲を指定するために用いるフィールドはそれぞれ 4 バイト、メッセージは可変長である。このメッセージフレームを UDP のペイロード部に入れ、IP アドレス 224.107.120.123、ポート 65000 宛てに送信する。

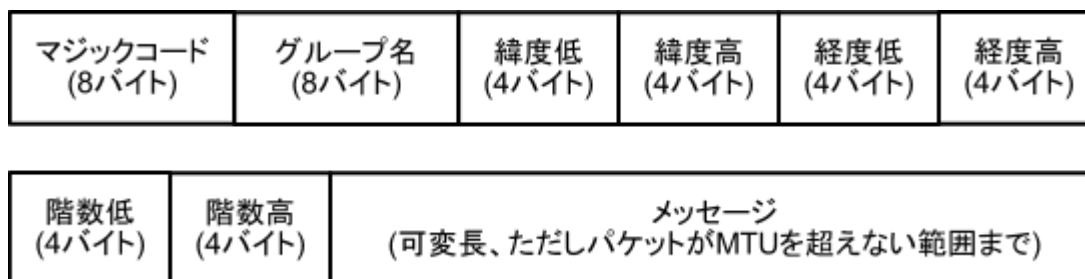


図 4.3 メッセージフォーマット

## 4-4 ジオキャストする範囲の決定

OpenFlow スイッチはデータパス ID によって識別されるのでホストに対してパケットを送出する OpenFlow スイッチの決定はジオキャストする範囲に含まれる OpenFlow スイッチのデータパス ID の集合を求めることによって行う。

提案機構はメッセージフレーム内のグループ名により二つの動作に分かれる。

- 1 グループ名が GPS00000
- 2 グループ名がそれ以外

1 の場合、メッセージフレームの緯度低～緯度高、経度低～経度高、階数低～階数高の範囲がジオキャストする範囲となる。階数を指定するときに階数低の値が階数高の値よりも大きければ全フロアが指定されることになる。階数低と階数高の値が同じときにはその階だけが指定されることになる。各 OpenFlow スイッチの位置情報はすべて OpenFlow コントローラが把握しているのでジオキャストする範囲に含まれるデータパス ID の集合を求める。

2 の場合、予め OpenFlow コントローラに登録されているグループ名に対応するデータパス ID の集合を求める。

## 4-5 範囲内の OpenFlow スイッチにつながっている各ホストへのメッセージパケット送出

データパス ID の集合を求めた後に、その集合に属する OpenFlow スイッチの受信者がつながっているポートすべてからメッセージパケットを送出する。このとき、OpenFlow スイッチ間リンクとメッセージパケットが最初に送られてきたポートにはメッセージパケットを送出しない。図 4.4 のように、OpenFlow スイッチのポートにつながっている機器が、OpenFlow スイッチなのかホストなのかは、LLDP パケットを送出してくるかどうかで判断されている。LLDP パケットを送ってくる機器が OpenFlow スイッチ、LLDP パケットを送ってこない機器がホストであると判断される。

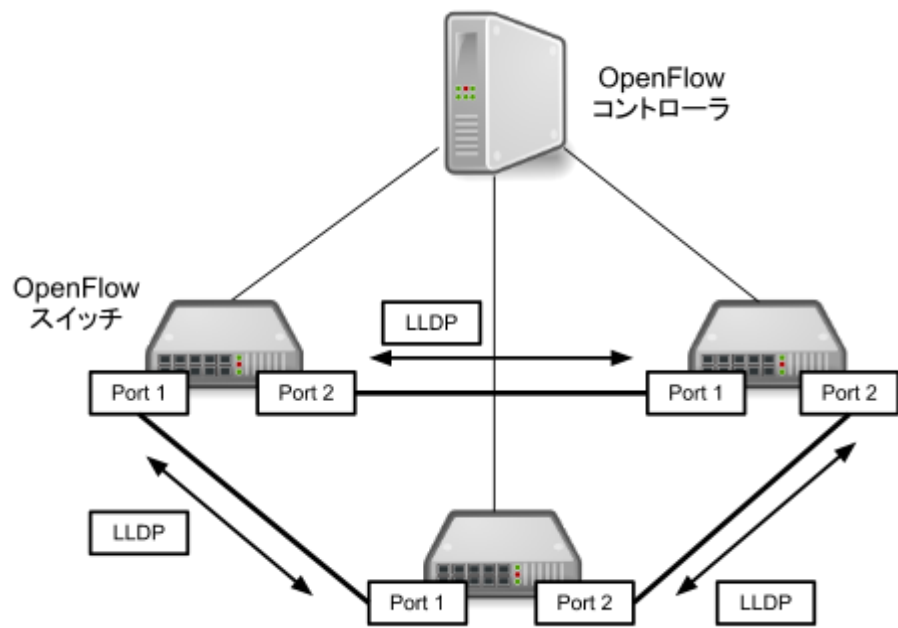


図 4.4 トポロジー情報の把握

## 第5章 評価

### 5-1 研究室規模での動作検証

OpenFlow コントローラ 1 台、OpenFlow スイッチ 5 台、送信者 1 台、受信者 2 台を用意し図 5.1 のようなネットワークを構成して OpenFlow コントローラが単一である手法の動作を検証した。

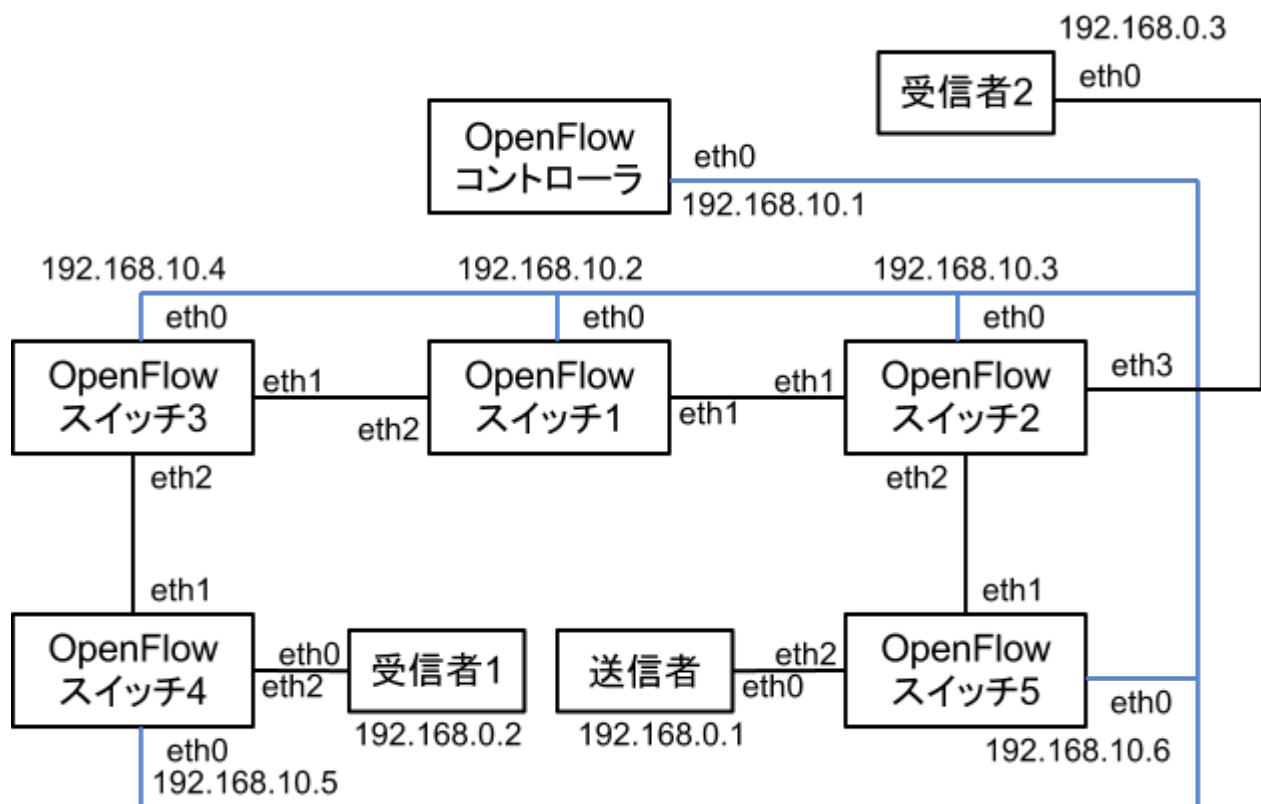


図 5.1 ネットワーク構成図

受信者はマルチキャストアドレス 224.107.120.123 をリッスンし、提案機構の制御用のヘッダである先頭の 40 バイトを除いた文字列を表示する。緯度/経度/階数の位置情報は図 5.1 のように設定した。

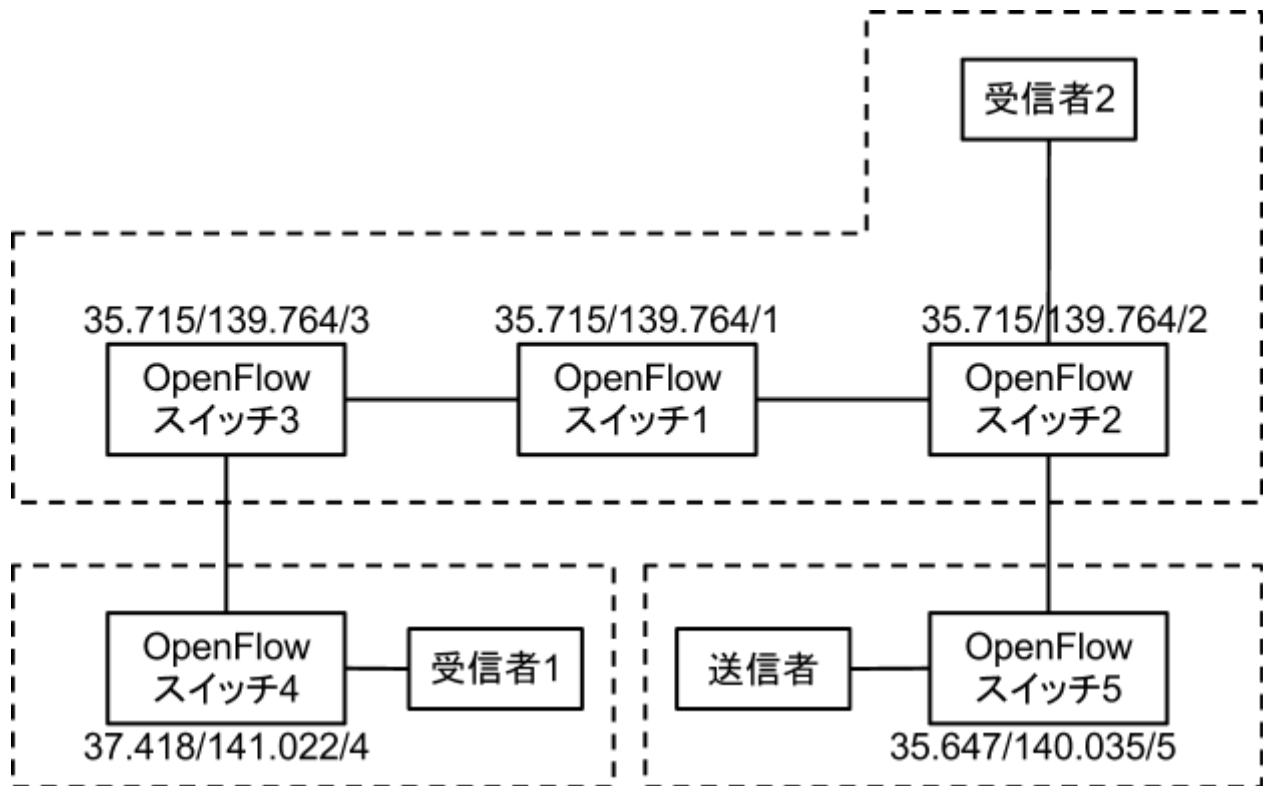


図 5.2 位置情報の割当

動作検証を行うために行ったシナリオは次の三つである。いずれの場合もグループ名には GPS00000 と入れ、可変長のメッセージ部分には「緊急地震速報です。」と入れた。

緯度低から階数高までのフィールドを順に 36.0000、38.0000、140.0000、142.0000、2、1 とする。この範囲に入るのは OpenFlow スイッチ 4 のみなので受信者 1 にだけメッセージが届くことになる。

緯度低から階数高までのフィールドを順に 35.0000、36.0000、139.0000、140.0000、2、2 とする。この範囲に入るのは OpenFlow スイッチ 2 のみなので受信者 2 にだけメッセージが届くことになる。

緯度低から階数高までのフィールドを順に 35.0000、38.0000、139.0000、142.0000、1、5 とする。この範囲にはすべての OpenFlow スイッチが含まれるので受信者 1 と受信者 2 の両方にメッセージが届くことになる。

実験を行ったところ三つの場合すべて想定したとおりの結果が得られた。

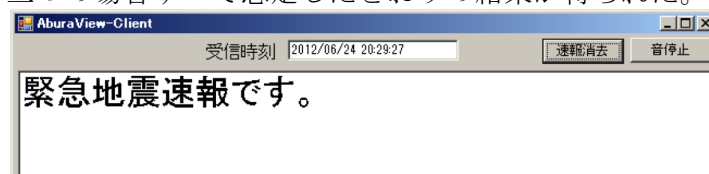


図 5.3 メッセージ表示画面

## 5-2 規模性の評価

### 5-2-1 実験の目的とシナリオ作成

1 台の OpenFlow コントローラにつながる OpenFlow スイッチの台数を増やしながら遅延を測定することで規模性を評価する。提案手法では経路制御するための処理をすべて OpenFlow コントローラが行っている。OpenFlow スイッチの台数が増えたときに遅延が線形に増大するのか、指数関数的に増大するのかを明らかにする必要がある。合わせて、距離が大きく離れた地域に OpenFlow スイッチを展開したときに伝送遅延が送信者・受信者間の遅延に対してどのような影響を与えるのかも明らかにする。想定するアプリケーションは次の通りである。

緊急地震速報のように、地理的な範囲を指定して、その範囲に含まれる端末にのみ情報を届ける。OpenFlow コントローラと送信者は東京にあり、OpenFlow スイッチは各都道府県に展開される。

これを元に次のような実験のシナリオを作成した。

1. 送信者、OpenFlow コントローラ、OpenFlow スイッチを仮想マシンで用意する。
2. OpenFlow スイッチは日本全国に均等に展開されるとする。実験では、北海道（札幌）、東北（仙台）、関東（東京）、中部（名古屋）、近畿（大阪）、中国（広島）、四国（高知）、九州（福岡）、沖縄（那覇）に均等にあるとする。
3. 地理的に離れた場所によって生じる伝送遅延は、各 OpenFlow スイッチ上で OpenFlow コントローラと通信するのに用いるネットワークデバイスに `tc` コマンドで遅延を生じさせることによって実現する。
4. OpenFlow スイッチの台数は、10、46、100、145、199 台のように段階的に増やしていく。
5. OpenFlow コントローラ、送信者、送信者のつながる OpenFlow スイッチは 1 台ずつ気象庁にあるとする。送信者はそれぞれの地方宛てに地理的な範囲を指定してメッセージを送り、その遅延を測定する。送信者側が送るパケットの間隔は、1 秒、3 秒、5 秒、10 秒の 4 つの場合について測定する。遅延の測定はそれぞれの条件で、10 回ずつ測定する。
6. 遅延の測定の仕方は、送信者と受信者側で `tcpdump` を用いてパケットをキャプチャし、受信時刻と送信時刻の差分を求めることで行う。ただし、この実験では受信者が直接つながる OpenFlow スイッチから受信者にパケットが届くまでの時間は十分小さい。そこで、受信者が直接つながる OpenFlow スイッチの受信者がつながるポートにおいてキャプチャして時刻を記録し、それを受信時刻とみなす。

`tc` コマンドで生じさせる遅延の値を決めるために、東大浅野キャンパスに存在するサーバからそれぞれの地方にあるサーバに 100 回 `ping` を打ち RTT を測った。その結果を表 5.1 に示す。

表 5.1 各地方の RTT 測定結果

場所	応答したサーバ	最小値 (ms)	平均 (ms)	最大値 (ms)	分散 (ms <sup>2</sup> )	平均/2 (ms)
北海道	nameserv.sys.hokudai.ac.jp	15.106	15.611	22.918	0.814	7.8055
宮城	narugo.net.tohoku.ac.jp	12.301	12.695	12.946	0.157	6.3475
愛知	nameserv.gw.nagoya-u.ac.jp	9.702	9.959	10.786	0.155	4.9795
大阪	www.city.osaka.lg.jp	15.115	15.345	15.912	0.175	7.6725
広島	www.pref.hiroshima.lg.jp	16.755	17.208	18.235	0.267	8.604

高知	www.kochi-marugoto.com	20.795	21.16	25.033	0.443	10.58
福岡	www.kyushu-u.ac.jp	21.32	21.53	21.804	0.161	10.765
沖縄	cc3.cc.u-ryukyu.ac.jp	38.018	38.371	40.856	0.416	19.1855

OpenFlow スイッチに設定した緯度・経度・階数と伝送遅延を表 5.2 に示す。RTT は往復にかかる時間を表しているため、RTT の半分にあたる値を伝送遅延として用いる。

表 5.2 OpenFlow スイッチに設定した緯度・経度・階数と伝送遅延の値

場所	緯度	経度	階数	伝送遅延
北海道	43.064615	141.346807	1	7.8055
宮城	38.268837	141.346807	1	6.3475
東京	35.68952	139.691704	1	0
愛知	35.180188	136.906565	1	4.9795
大阪	34.686556	135.519433	1	7.6725
広島	34.396585	132.459508	1	8.604
高知	33.559704	133.53108	1	10.58
福岡	33.606784	130.418314	1	10.765
沖縄	26.212401	127.680932	1	19.1855

実験において送信者は北海道、宮城、東京、愛知、大阪、広島、高知、福岡、沖縄の 9 地方の中の 1 地方にのみジオキャストエリアを限定してメッセージを送信する。その際に設定した緯度・経度・階数は表 5.3 の通りである。

表 5.3 送信者がメッセージ送信時に指定する緯度・経度・階数

ジオキャストエリア	緯度低	緯度高	経度低	経度高	階数低	階数高
北海道	43	44	141	142	1	1
宮城	38	39	140	141	1	1
東京	35	36	139	140	1	1
愛知	35	36	136	137	1	1
大阪	34	35	135	136	1	1
広島	34	35	132	133	1	1
高知	33	34	132	133	1	1
福岡	33	34	130	131	1	1
沖縄	26	27	127	128	1	1



## 5-2-2 環境構築

本論文では送信者、OpenFlow コントローラ、OpenFlow スイッチを仮想マシンで用意する。送信者 1 台、OpenFlow コントローラ 1 台を 1 台の物理マシン上に起動させる。各地方に展開される OpenFlow スイッチについては各地方につき 1 台物理マシンを割り当てる。ネットワークの配線図を図 5.4 に示す。

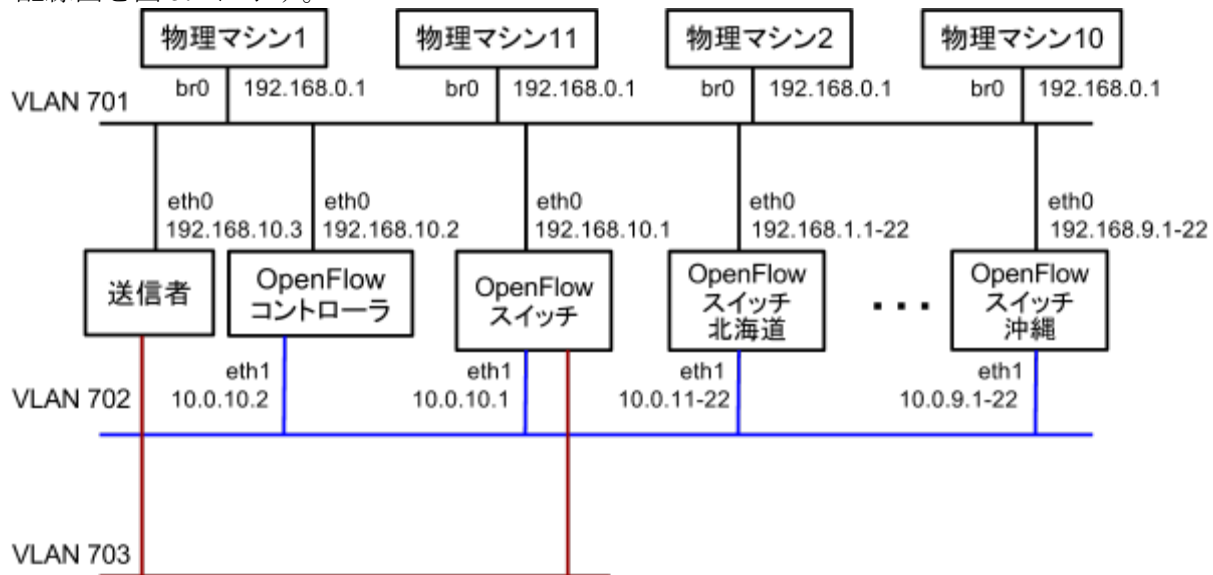


図 5.4 ネットワークの配線図

VLAN の番号の割当は表 5.4 の通りに行った。

表 5.4 VLAN 番号の割当

VLAN 番号	用途
701	実験中の仮想マシン制御
702	OpenFlow コントローラと OpenFlow スイッチ間の通信
703	送信者と送信者が直接つながっている OpenFlow スイッチの間の通信

## 5-2-3 OpenFlow スイッチの総数が 10 台のときの実験結果

送信者 1 台、OpenFlow コントローラ 1 台、送信者がつながる OpenFlow スイッチ 1 台、各地方に展開される OpenFlow スイッチを各 1 台ずつ用意する。各地方 1 台ずつ計 9 台と送信者がつながる OpenFlow スイッチ 1 台で合わせて 10 台となる。仮想マシンとそのネットワークの構成は図 5.5 の通りである。

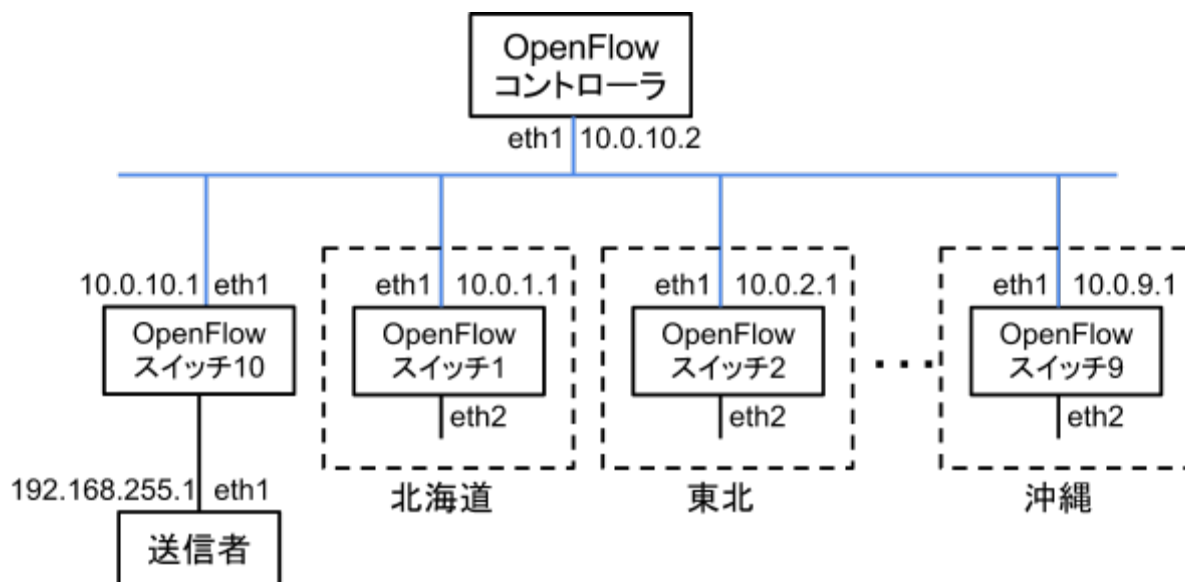


図 5.5 OpenFlow スイッチの総数が 10 台のときの構成図

送信者がパケットの間隔 1、3、5、10 秒で送信したときの結果をそれぞれ以下に示す。

表 5.5 OpenFlow スイッチの総数が 10 台で送信者が間隔を 1 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	109	8
東北	110	4
関東	103	4
中部	111	8
近畿	111	6
中国	113	13
四国	113	7
九州	114	5
沖縄	126	12

表 5.6 OpenFlow スイッチの総数が 10 台で送信者が間隔を 3 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	106	2
東北	109	1
関東	100	1
中部	109	1
近畿	111	1
中国	109	1
四国	114	1
九州	113	1
沖縄	121	1

表 5.7 OpenFlow スイッチの総数が 10 台で送信者が間隔を 5 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
-----------	--------	----------

北海道	128	18
東北	134	23
関東	130	23
中部	129	18
近畿	134	18
中国	132	24
四国	134	14
九州	137	18
沖縄	143	16

表 5.8 OpenFlow スイッチの総数が 10 台で送信者が間隔を 10 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	109	2
東北	109	2
関東	101	2
中部	108	2
近畿	108	2
中国	111	3
四国	110	2
九州	112	2
沖縄	121	2

#### 5-2-4 OpenFlow スイッチの総数が 46 台のときの実験結果

送信者 1 台、OpenFlow コントローラ 1 台、送信者がつながる OpenFlow スイッチ 1 台、各地方に展開される OpenFlow スイッチを各 5 台ずつ用意する。各地方 5 台ずつ計 45 台と送信者がつながる OpenFlow スイッチ 1 台で合わせて 46 台となる。仮想マシンとそのネットワークの構成は図 5.6 の通りである。

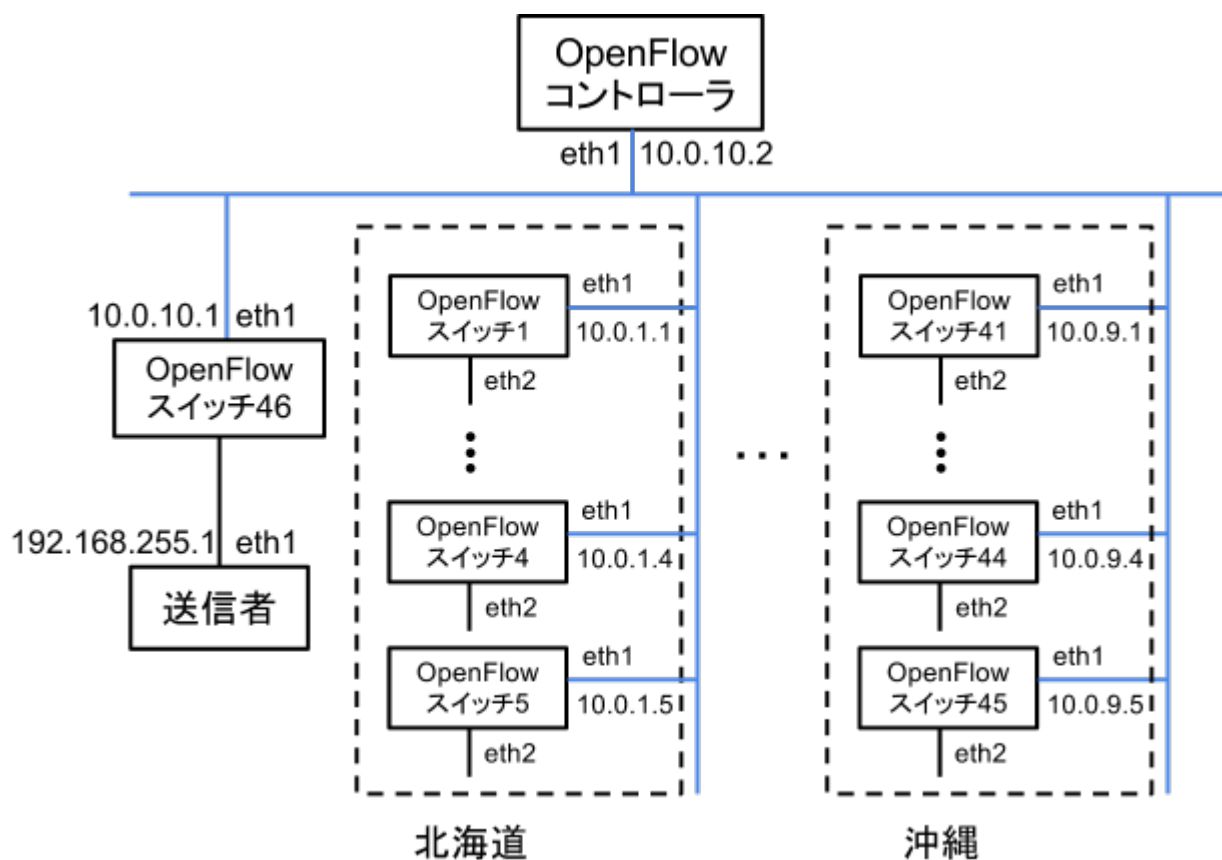


図 5.6 OpenFlow スイッチの総数が 46 台のときの構成図

送信者がパケットの間隔 1、3、5、10 秒で送信したときの結果をそれぞれ以下に示す。

表 5.9 OpenFlow スイッチの総数が 46 台で送信者が間隔を 1 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	131	5
東北	130	6
関東	126	4
中部	134	12
近畿	133	5
中国	136	6
四国	137	6
九州	137	6
沖縄	147	6

表 5.10 OpenFlow スイッチの総数が 46 台で送信者が間隔を 3 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	126	3
東北	131	1
関東	121	4
中部	129	5

近畿	133	2
中国	134	2
四国	137	1
九州	133	1
沖縄	145	1

表 5.11 OpenFlow スイッチの総数が 46 台で送信者が間隔を 5 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	154	21
東北	148	14
関東	143	12
中部	154	19
近畿	155	24
中国	160	30
四国	154	12
九州	157	22
沖縄	163	13

表 5.12 OpenFlow スイッチの総数が 46 台で送信者が間隔を 10 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	127	3
東北	131	3
関東	131	11
中部	130	2
近畿	133	2
中国	133	3
四国	137	4
九州	137	3
沖縄	144	3

## 5-2-5 OpenFlow スイッチの総数が 100 台のときの実験結果

送信者 1 台、OpenFlow コントローラ 1 台、送信者がつながる OpenFlow スイッチ 1 台、各地方に展開される OpenFlow スイッチを各 11 台ずつ用意する。各地方 11 台ずつ計 99 台と送信者がつながる OpenFlow スイッチ 1 台で合わせて 100 台となる。仮想マシンとそのネットワークの構成は図 5.7 の通りである。

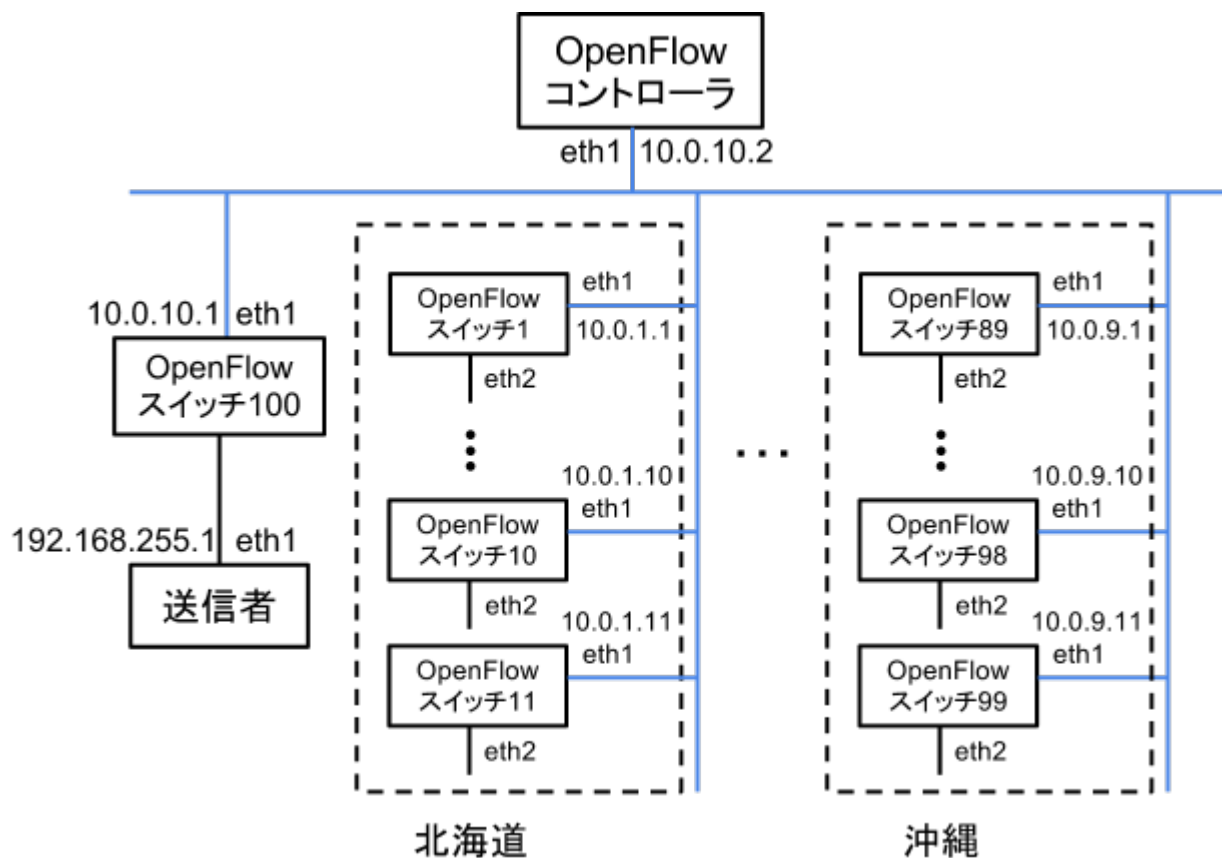


図 5.7 OpenFlow スイッチの総数が 100 台のときの構成図

送信者がパケットの間隔 1、3、5、10 秒で送信したときの結果をそれぞれ以下に示す。

表 5.13 OpenFlow スイッチの総数が 100 台で送信者が間隔を 1 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	169	17
東北	173	17
関東	158	15
中部	164	6
近畿	169	12
中国	162	6
四国	173	8
九州	178	12
沖縄	180	15

表 5.14 OpenFlow スイッチの総数が 100 台で送信者が間隔を 3 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	168	7
東北	168	5
関東	157	4
中部	165	6

近畿	161	5
中国	162	2
四国	169	2
九州	171	5
沖縄	178	2

表 5.15 OpenFlow スイッチの総数が 100 台で送信者が間隔を 5 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	200	22
東北	188	17
関東	181	12
中部	181	9
近畿	189	16
中国	180	10
四国	197	22
九州	185	22
沖縄	211	33

表 5.16 OpenFlow スイッチの総数が 100 台で送信者が間隔を 10 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	179	31
東北	163	5
関東	159	4
中部	186	40
近畿	180	18
中国	173	4
四国	185	25
九州	168	4
沖縄	180	5

## 5-2-6 OpenFlow スイッチの総数が 145 台のときの実験結果

送信者 1 台、OpenFlow コントローラ 1 台、送信者がつながる OpenFlow スイッチ 1 台、各地方に展開される OpenFlow スイッチを各 16 台ずつ用意する。各地方 16 台ずつ計 144 台と送信者がつながる OpenFlow スイッチ 1 台で合わせて 145 台となる。仮想マシンとそのネットワークの構成は図 5.8 の通りである。

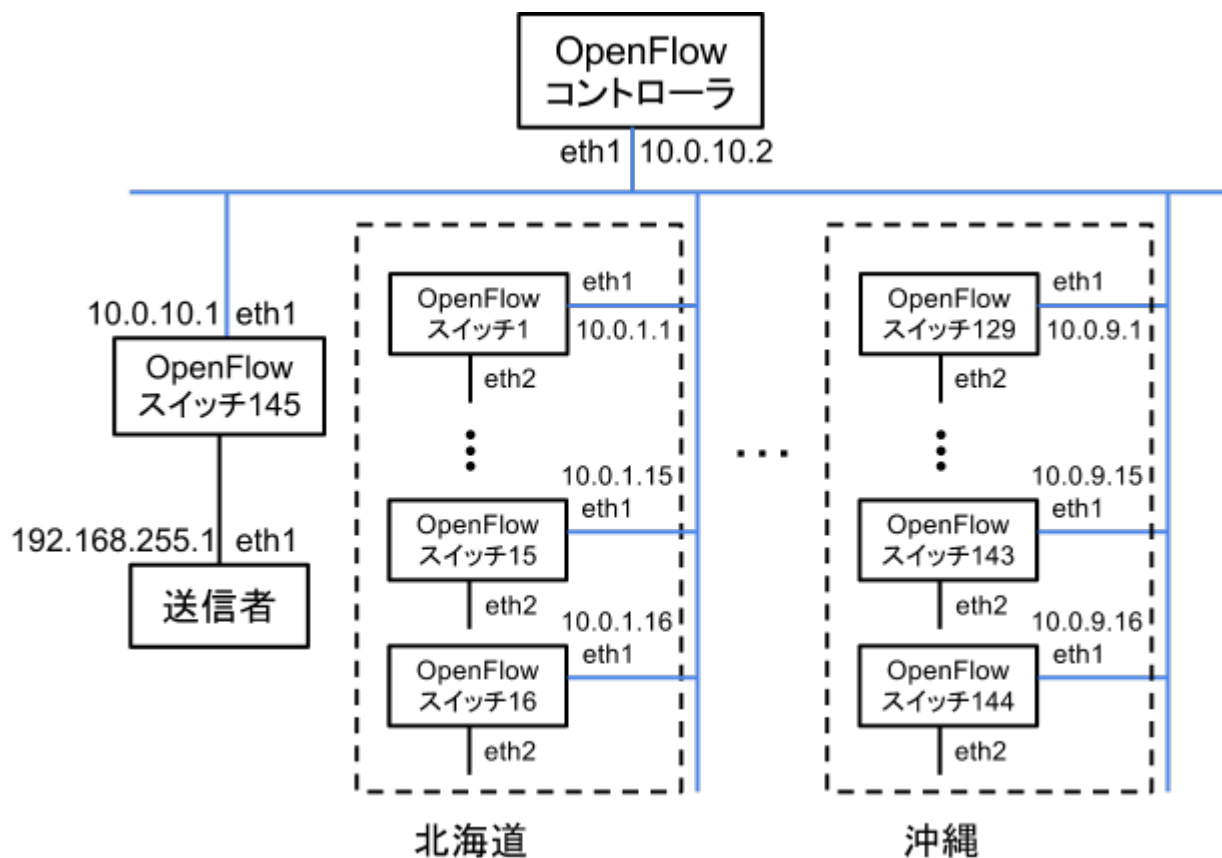


図 5.8 OpenFlow スイッチの総数が 145 台のときの構成図

送信者がパケットの間隔 1、3、5、10 秒で送信したときの結果をそれぞれ以下に示す。

表 5.17 OpenFlow スイッチの総数が 145 台で送信者が間隔を 1 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	219	2
東北	216	6
関東	220	4
中部	215	7
近畿	223	18
中国	226	22
四国	224	6
九州	227	11
沖縄	229	6

表 5.18 OpenFlow スイッチの総数が 145 台で送信者が間隔を 3 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	220	5
東北	220	9
関東	219	5
中部	226	19



近畿	221	5
中国	223	3
四国	222	4
九州	226	5
沖縄	233	2

表 5.19 OpenFlow スイッチの総数が 145 台で送信者が間隔を 5 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	256	24
東北	239	27
関東	249	16
中部	234	14
近畿	254	20
中国	252	21
四国	230	9
九州	255	28
沖縄	267	18

表 5.20 OpenFlow スイッチの総数が 145 台で送信者が間隔を 10 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	231	23
東北	221	5
関東	225	17
中部	221	23
近畿	220	6
中国	246	27
四国	230	23
九州	221	6
沖縄	244	5

## 5-2-7 OpenFlow スイッチの総数が 199 台のときの実験結果

送信者 1 台、OpenFlow コントローラ 1 台、送信者がつながる OpenFlow スイッチ 1 台、各地方に展開される OpenFlow スイッチを各 22 台ずつ用意する。各地方 22 台ずつ計 198 台と送信者がつながる OpenFlow スイッチ 1 台で合わせて 199 台となる。仮想マシンとそのネットワークの構成は図 5.9 の通りである。

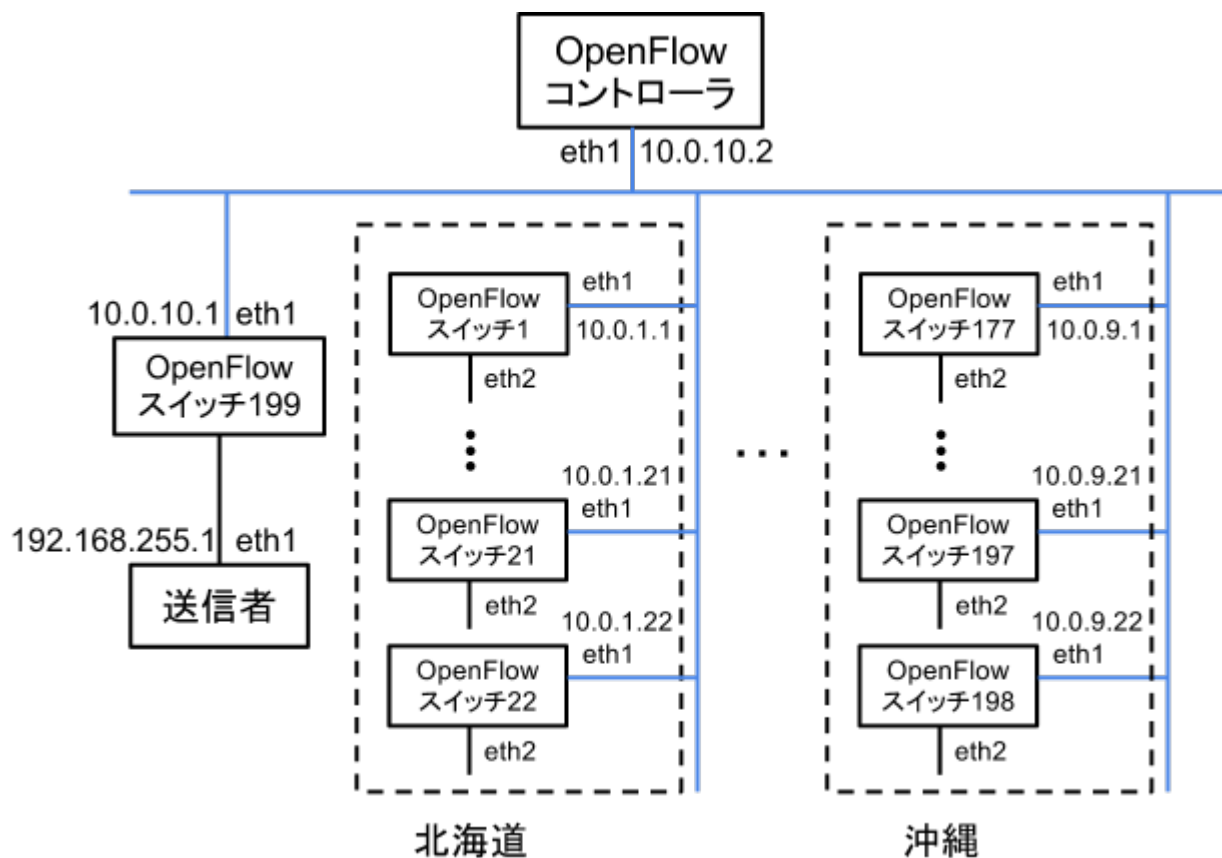


図 5.9 OpenFlow スイッチの総数が 199 台のときの構成図

送信者がパケットの間隔 1、3、5、10 秒で送信したときの結果をそれぞれ以下に示す。

表 5.21 OpenFlow スイッチの総数が 199 台で送信者が間隔を 1 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	2044	708
東北	2216	856
関東	2215	845
中部	2218	855
近畿	2201	842
中国	2183	841
四国	2253	852
九州	2185	844
沖縄	2214	838

表 5.22 OpenFlow スイッチの総数が 199 台で送信者が間隔を 3 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	1191	19
東北	1185	14
関東	1175	14
中部	1182	16

近畿	1181	5
中国	1188	8
四国	1192	8
九州	1193	17
沖縄	1193	5

表 5.23 OpenFlow スイッチの総数が 199 台で送信者が間隔を 5 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	1208	48
東北	1207	17
関東	1202	23
中部	1209	22
近畿	1190	9
中国	1212	14
四国	1210	24
九州	1215	26
沖縄	1207	16

表 5.24 OpenFlow スイッチの総数が 199 台で送信者が間隔を 10 秒空けてメッセージを送信するときの遅延測定結果

ジオキャストエリア	平均[ms]	標準偏差[ms]
北海道	1202	31
東北	1190	13
関東	1183	19
中部	1201	18
近畿	1190	9
中国	1188	7
四国	1192	8
九州	1198	16
沖縄	1218	25

## 5-3 規模性を評価する実験の結果の考察

### 5-3-1 伝送遅延が送信者・受信者間の遅延に与える影響

提案手法の経路制御は地理的な位置情報に基づいている。提案手法が実際に使われる状況を考えると、OpenFlow コントローラから地理的に大きく離れた場所に OpenFlow スイッチが展開されることが予想される。その際、OpenFlow コントローラと OpenFlow スイッチ間の距離に基づく伝送遅延の送信者・受信者間の遅延に対する影響が小さい必要がある。OpenFlow コントローラと OpenFlow スイッチ間の距離に基づく伝送遅延が送信者・受信者間の遅延にどのような影響を与えるのかを明らかにするためにそれぞれの地方をジオキャストエリアにしたときの遅延から関東をジオキャストエリアにしたときの遅延を引いた値と tc コマンドを用いて加えた伝送遅延を比較する。表 5.25 にそれぞれの地方をジオキャストエリアにしたときの遅延から関東を

ジオキャストエリアにしたときの遅延を引いた値と tc コマンドを用いて加えた伝送遅延を折れ線グラフにして示す。送信者がメッセージを送る間隔が 10 秒であるときの実験結果を用いた。それぞれの地方をジオキャストエリアにしたときの遅延から関東をジオキャストエリアにしたときの遅延を引いた値の単位はミリ秒である。

表 5.25 それぞれの地方をジオキャストエリアにしたときの遅延から関東をジオキャストエリアにしたときの遅延を引いた値と tc コマンドを用いて加えた伝送遅延

ジオキャストエリア	10 台	46 台	100 台	145 台	199 台	伝送遅延
北海道	8	-4	20	6	19	7.8055
東北	8	0	4	-4	7	6.3475
関東	0	0	0	0	0	0
中部	7	-1	27	-4	18	4.9795
近畿	7	2	21	-5	7	7.6725
中国	10	2	14	21	5	8.604
四国	9	6	26	5	9	10.58
九州	11	6	9	-4	15	10.765
沖縄	20	13	21	19	35	19.1855

OpenFlow スイッチの総数が 10 台、46 台のときは OpenFlow コントローラのメッセージを処理する負荷が小さいためほぼ伝送遅延と同じような値を示している。このことから、OpenFlow コントローラと OpenFlow スイッチ間の距離に基づく伝送遅延の送信者・受信者間の遅延に対する影響は片側伝送遅延で済んでいると考えられる。

TCP はスリーウェイハンドシェイクを行ってコネクションを確立したあとにデータをやり取りする。データのやり取りでは Ack を返す必要がある。UDP ではコネクションを特に確立せずにデータを送信する。データを受け取った側は Ack を返さない。

OpenFlow コントローラと OpenFlow スイッチ間の距離に基づく伝送遅延の送信者・受信者間の遅延に対する影響が片側伝送遅延で済む理由は、OpenFlow プロトコルは TCP を用いているものの、OpenFlow コントローラと OpenFlow スイッチのコネクションがつながったままなので一度 OpenFlow コントローラが Packet Out メッセージによって送信者のメッセージデータを OpenFlow スイッチに送ってしまえば済むからだと考えられる。提案手法で送信者がメッセージを送るときは UDP を用いているので受信者が一度メッセージを受け取ってしまえばそれ以降のやり取りは存在しない。

### 5-3-2 OpenFlow スイッチの台数が増えるときの遅延の増大の仕方

表 5.5 から表 5.24 までの実験結果を見ると、OpenFlow スイッチの台数が増えるときのそれぞれの地方における遅延の増大の仕方は同様の傾向を示しているので、ここでは沖縄のみに注目する。OpenFlow スイッチの台数と沖縄における遅延の関係を表す散布図を図 5.10 に示す。エラーバーの誤差範囲としては 95%の信頼区間を用いた。

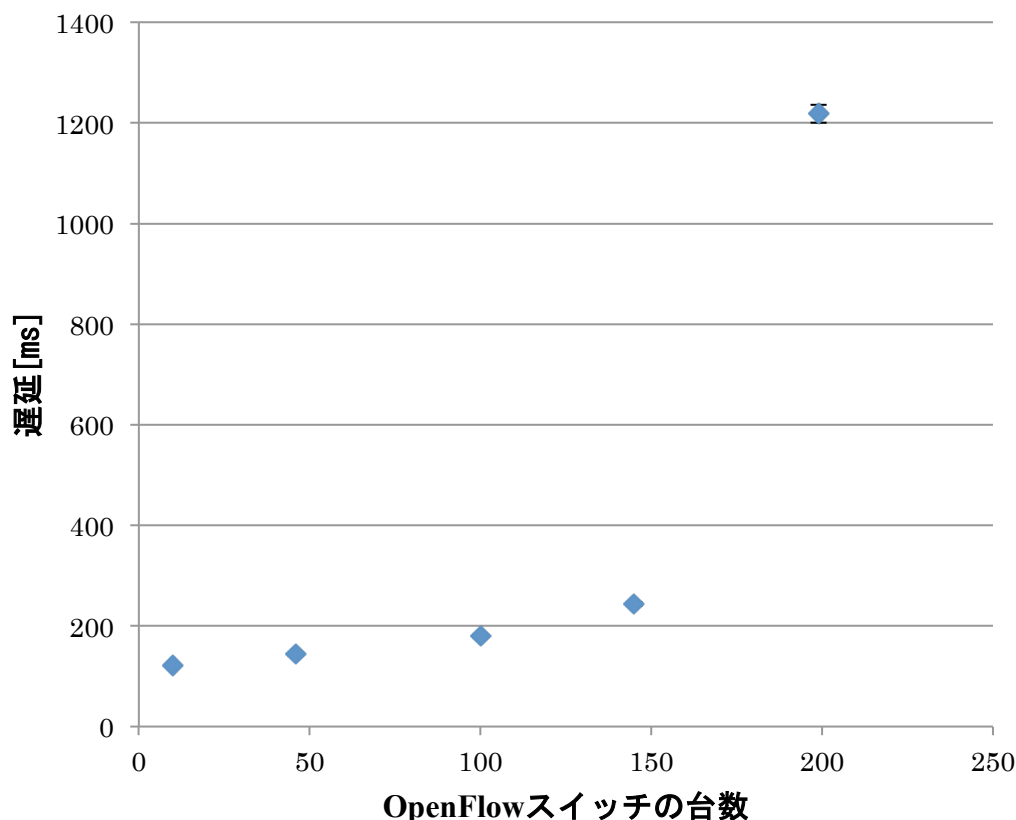


図 5.10 OpenFlow スwitchの台数と沖縄における遅延の関係

図 5.10 を見ると、OpenFlow スwitchの総数が 150 台のときから 200 台のときにかけて遅延が急激に大きくなっている。

送信者が十分な時間間隔を空けてメッセージを送信する状況下で、提案手法の遅延を主に構成する要素としては、伝送遅延、全 OpenFlow スwitchに対して Features Request を送りその返答である Featureas Reply を処理して位置情報を得るのにかかる時間、ジオキャストエリアに含まれる OpenFlow スwitchを検索するのにかかる時間の三つが考えられる。

伝送遅延は OpenFlow コントローラと OpenFlow スwitchの間の距離に基づいている。5-3-1 で考察した通り、送信者・受信者間の遅延に与える影響は片側遅延のみで済むので、OpenFlow の台数の増加に対して伝送遅延は変化しない。

ジオキャストエリアに含まれる OpenFlow スwitchの検索は線形探索で行っている。線形探索による検索を行うのにかかる時間は、検索する対象の数の増加に対して線形に増加する。したがって、ジオキャストエリアに含まれる OpenFlow スwitchを検索するのにかかる時間は OpenFlow スwitchの総数に対して線形に増加する。

全 OpenFlow スwitchに対して Features Request を送りその返答である Featureas Reply を処理して位置情報を得るのにかかる時間が OpenFlow スwitchの総数の増加に対してどのように変化するかを明らかにするために、提案手法の該当する部分では Trema のアプリケーションとして提供されている show\_description を用いているので show\_description の実行時間を測定した。

その結果を表 5.26 と図 5.11 に示す。図 5.11 の実行時間の誤差範囲は 95%の信頼区間としている。

表 5.26 show\_description の実行時間の測定結果

OpenFlow スイッチの台数	実行時間[ms]	標準偏差[ms]
10	91	1
46	123	2
100	222	18
145	274	10
199	1237	11

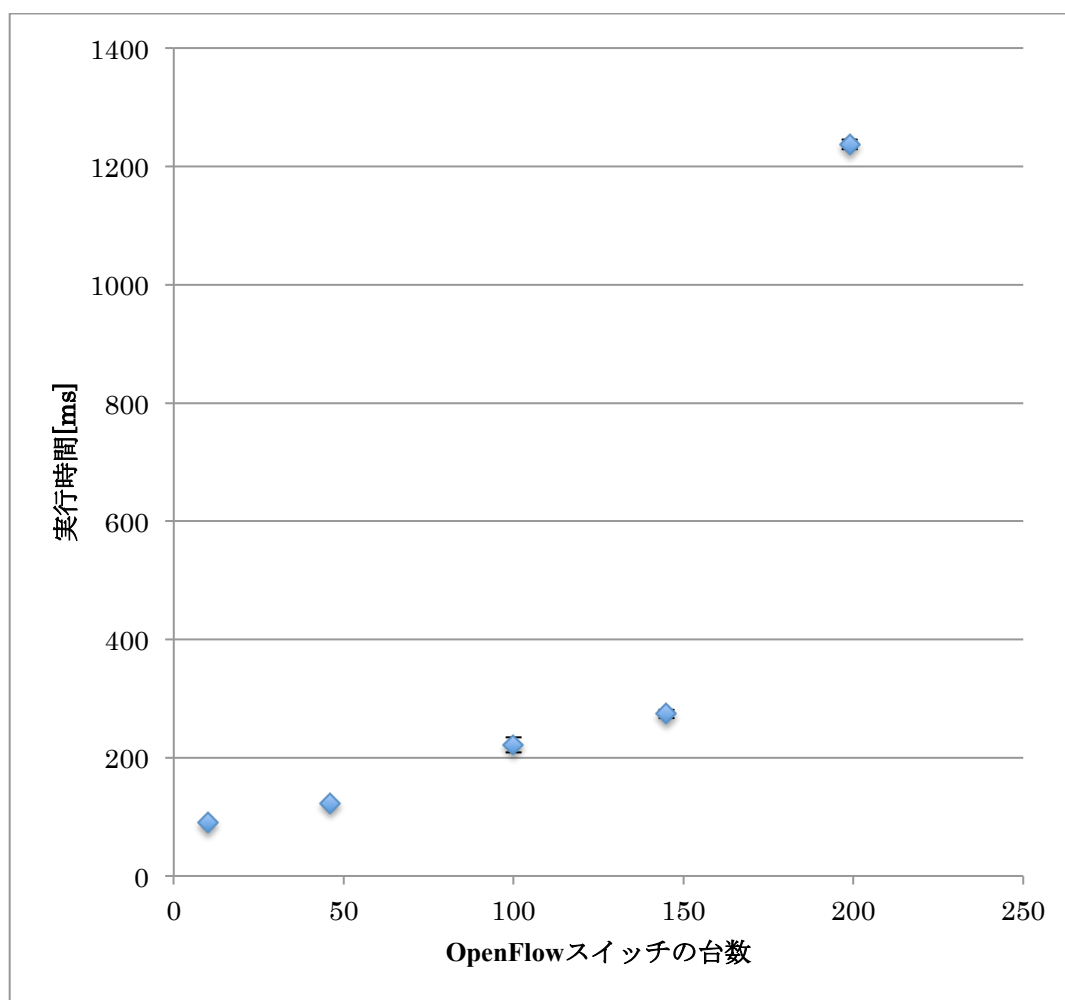


図 5.11 OpenFlow スイッチの台数と show\_description の実行時間の関係

図 5.11 を見ると、実行時間が OpenFlow スイッチの台数の増加に対してメッセージの遅延と同様の変化を示している。表 5.26 の実行時間の値から、show\_description の実行時間は送信者・受信者間の遅延に対してほぼ 9 割以上の大きな割合を占めている。

show\_description の実行時間がある OpenFlow スイッチの台数を境にして急激に増大する原因としては、OpenFlow コントローラ側における多数の Features Replay による衝突が考えられる。

4 章 4 節で述べたように提案手法は予め形成しておいたグループにジオキャストを行うこともできる。予め形成しておいたグループにジオキャストを行う際は、show\_description は実行されない。show\_description の実行時間は送信者・受信者間の遅延に対してほぼ 9 割以上の大きな割合を占めているので、送信者・受信者間の遅延を大幅に削減できる可能性がある。

### 5-3-3 送信者が送るメッセージの時間間隔が小さいときの遅延

提案手法の遅延は送信者が送るメッセージの時間間隔からも影響を受ける。OpenFlow スイッチの総数が 199 台のときの遅延測定結果を見ると、送信者がメッセージを送る時間の間隔が 1 秒のときだけ遅延測定値の平均が大きくばらつきも大きい。ジオキャストエリアを沖縄とし OpenFlow スイッチの総数が 199 台で送信者が間隔を 1 秒空けてメッセージを送信するときの 10 回の遅延測定結果を散布図として図 5.12 に示す。

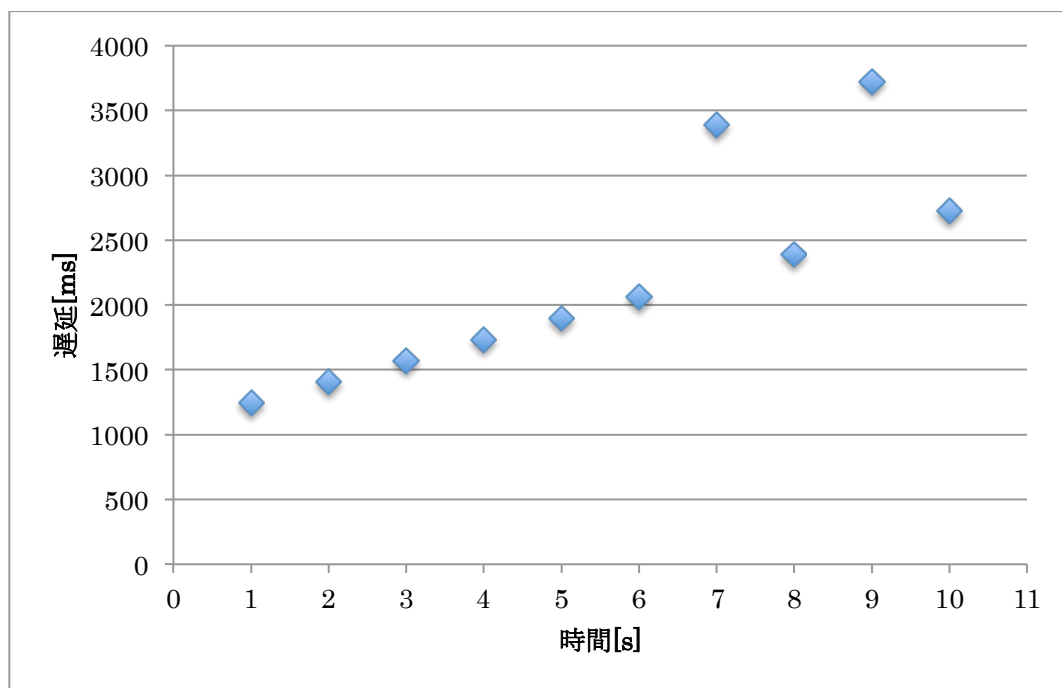


図 5.12 OpenFlow スイッチの総数が 199 台で送信者が間隔を 1 秒空けてメッセージを送信するときの 10 回の遅延測定結果

図 5.12 をみると、7 個目と 9 個目をのぞくメッセージは約 150ms ずつ大きくなっており、線形に増加している。7 個目ではいったん約 1300ms 増加し、8 個目では約 1000ms 減少している。9 個目でも同様にいったん大きく増加し、10 個目では大きく減少している。

メッセージを送る毎にメッセージの遅延が増加する原因は、OpenFlow コントローラが前のメッセージの処理を終える前に送信者が次のメッセージを送ることで、次のメッセージの遅延に前のメッセージの残りの処理時間が上乗せされているからだと考えられる。

## 第6章 考察

### 6-1 グループ形成の柔軟性の評価

受信者指定型マルチキャスト、送信者指定型マルチキャストのそれぞれを用いたジオキャストと提案手法のグループ形成の柔軟性について定性的な評価を行う。

受信者指定型マルチキャストを用いてジオキャストを行う場合、ジオキャストエリア1つに対してマルチキャストグループ1つを割り当てることになる。さらに、ジオキャストエリアに含まれる受信者にその該当するグループに加入してもらう必要がある。別のジオキャストエリアにジオキャストするときにはジオキャストエリアが少し変化しただけでもマルチキャストグループをもう1つ増やし、受信者に新たにそのグループに加入してもらうことになる。

送信者指定型マルチキャストを用いてジオキャストを行う場合、送信者がジオキャストエリアに含まれる受信者のIPアドレスのリストを作成する必要がある。送信者はリストを作成するために予めIPアドレスと地理的な位置情報の組み合わせを求めておくことになる。別のジオキャストエリアにジオキャストする際には、IPアドレスと地理的な位置情報の組み合わせから受信者のIPアドレスを求めて、送信者指定型マルチキャストの受信者リストを作成する作業を繰り返すことになる。

提案手法を用いてジオキャストを行う場合、送信者はジオキャストエリアを緯度・経度・階数の情報で指定するだけでよい。ジオキャストエリアを変える際には緯度・経度・階数で変化する部分を変えてメッセージを送信する。グループの形成は動的に OpenFlow コントローラが行う。このように SDN 技術を用いることにより、従来のジオキャスト手法と比較して柔軟なグループ形成が行えるのではないかと考えられる。

### 6-2 配信の効率性の評価

受信者指定型マルチキャスト、送信者指定型マルチキャストのそれぞれを用いたジオキャストと提案手法の配信の効率性について定性的な評価を行う。

受信者指定型マルチキャストを用いてジオキャストを行う場合、送信者がジオキャストエリアを決めた後にマルチキャストグループを作成し、受信者にそのマルチキャストグループに加入してもらう必要がある。ジオキャストエリアに含まれる受信者を送信者が把握し、その受信者にマルチキャストアドレスを周知して該当するマルチキャストグループに加入してもらうのは困難である。

送信者指定型マルチキャストを用いてジオキャストを行う場合、受信者のリストを作成する前にジオキャストエリアに含まれる受信者のIPアドレスを知る必要がある。2章で述べたように、IPアドレスから地理的な位置情報を調べても数十kmの精度でしかわからない。IPアドレスを指標としてグループを形成しジオキャストを行うと数十kmよりも小さい粒度のジオキャストエリアであった場合、誤差が大きいのでジオキャストエリアに含まれないホストにも多数の packets が配信されることが予想される。結果として無駄の多い配信となってしまう。



提案手法はジオキャストエリア内の OpenFlow スイッチの受信者がつながっているポートのみからメッセージを送信する仕組みになっている。ジオキャストエリア外のホストにもメッセージが送信されるかは OpenFlow スイッチとホストがどのようにつながっているかに依る。

OpenFlow スイッチから近くに存在すれば高精度にジオキャストすることが可能であり、OpenFlow スイッチから遠く離れた場所に存在するようであれば精度は低くなり無駄が増える。提案手法を用いるアプリケーションに応じて予めどれくらいの精度が必要なかを決定し、その精度の範囲内に OpenFlow スイッチとホストの距離が収まるように、OpenFlow スイッチとホストの接続を行えば効率的な配信を行うことができる。

## 第7章 おわりに

### 7-1 結論

本論文では、IP ネットワークにおいて、柔軟なグループ形成と効率的な配信を行うジオキャスト手法の実現を目標とし、それを実現するための方法として、SDN を利用することを検討した。さらに、提案手法のグループ形成の柔軟性の評価と規模性の評価を行った。規模性の評価については、柔軟なグループ形成と効率的な配信を行うジオキャスト手法を必要とする例として緊急地震速報を挙げ、OpenFlow が日本全国に展開されるという想定で仮想環境を用いて遅延を測定する実験を行った。

規模性を評価する実験の結果、提案手法の遅延はある OpenFlow スイッチの台数を境に急激に増大することが示された。伝送遅延の送信者・受信者間の遅延に対する影響は片側遅延のみで済むということがわかった。また、OpenFlow コントローラが前のメッセージをすべて処理してしまう前に次のメッセージを送信者が送ると次のメッセージの遅延が大きくなってしまったこともわかった。提案手法を用いる際には、その用途に応じて、送信者側がどれくらいのトラヒックで送信するかを予め適切に決めておく必要がある。

グループ形成の柔軟性については、受信者指定型マルチキャスト、送信者指定型マルチキャストのそれぞれを用いたジオキャストと提案手法を比較して定性的な評価を行った。SDN 技術を用いることにより、従来のジオキャスト手法と比較して柔軟なグループ形成が行えると考えられることを述べた。

配信の効率性についても、受信者指定型マルチキャスト、送信者指定型マルチキャストのそれぞれを用いたジオキャストと提案手法を比較して定性的な評価を行った。受信者指定型マルチキャストではジオキャストエリアに含まれる受信者を送信者が把握することが困難なために、送信者指定型マルチキャストでは IP アドレスから地理的な位置情報を調べても数十 km の精度でしかわからないために、配信の精度が低くなるのに対して、提案手法では OpenFlow スイッチと受信者の距離を配信が求める精度に収まるようにすれば配信の精度が高くなり配信の効率が高くなることを述べた。

### 7-2 今後の課題

今後は上記で挙げられた課題に加え、以下の課題についても取り組んでいく必要がある。

#### 1. OpenFlow コントローラの階層化

規模性の評価実験で、ある台数を境に急激に遅延が増大することがわかった。この急激な遅延の増大を防ぐ方法として OpenFlow コントローラの階層化が挙げられる。

例えば OpenFlow コントローラの階層化を2段階に行うときは次のようにして実現することができる。1段階目の OpenFlow コントローラを OpenFlow コントローラ 1、2段階目の OpenFlow コントローラを下位の OpenFlow コントローラと呼ぶ。下位の OpenFlow コントローラは複数存在することができ、それぞれが担当のエリアを持つ。予め下位のエリアのうち代表となる OpenFlow スイッチ 1 台を選び、その OpenFlow スイッチと OpenFlow コ

ントローラ 1 はつながっているようにしておく。ただし、このリンクは OpenFlow スイッチと OpenFlow コントローラ間のリンクではなく、単なるリンクとする。下位エリアの代表となる OpenFlow スイッチはそのエリアを担当する下位の OpenFlow コントローラ 1 台と OpenFlow コントローラ・OpenFlow スイッチ間の接続をする。

まず、送信者がメッセージを送る。メッセージが送信者のつながる OpenFlow スイッチに届く。その OpenFlow スイッチから Packet In メッセージで OpenFlow コントローラ 1 へメッセージのデータが届く。OpenFlow コントローラ 1 はジオキャストエリアに含まれるエリアを担当する下位のエリアを割り出す。OpenFlow コントローラ 1 はジオキャストエリアに含まれる下位エリアの代表となる OpenFlow スイッチにジオキャストパケットを送る。下位の OpenFlow スイッチは Packet In メッセージで OpenFlow コントローラ 2 へメッセージのデータを送る。下位の OpenFlow コントローラはジオキャストエリア内に含まれる OpenFlow スイッチを検索してホストがつながっているポートそれぞれに Packet Out メッセージを送る。ジオキャストエリア内の OpenFlow スイッチはメッセージを指示通りポートから出す。ジオキャストエリア内のホストがメッセージを受け取る。

このように階層化を行うことによって OpenFlow コントローラの負荷を分散させることができる。

## 2. 受信者の増加による遅延の増加

受信者がつながっているポートからメッセージを送出するようにジオキャストエリア内の OpenFlow スイッチに OpenFlow コントローラが指示する必要があるが、受信者が増加すると指示を送る回数が増えるのでそのぶん線形に処理時間が増えると予想される。この処理時間の影響を明らかにする必要がある。

## 謝辞

本研究を進めるにあたり、多くの方々のご協力をいただきました。

関谷勇司准教授には、数々のアイデアや有益なアドバイスをいただきました。ご心配・ご迷惑をお掛けしてしまったこともしばしばでしたが、その都度温かい目で見えていただいたことに心から感謝しております。

また、宮本大輔助教には、日頃の打ち合わせに加え、オープンルータコンペティションや評価実験でのアドバイスなど大変お世話になりました。平日・休日問わず研究に対する相談に乗っていただき誠に感謝しております。

そして、同期に感謝いたします。この2年間、互いに切磋琢磨し、苦しみや喜びを分かち合いながらも楽しく過ごすことができたことを嬉しく思います。また、2年間の研究室生活を楽しいものにしてくださったCNLのメンバーの皆様にはとても感謝しております。

最後に、これまでの成長を見届け勉学に打ち込める環境を与えてくれた、家族、特に精一杯応援してくれていた亡き父に感謝いたします。ありがとうございました。

## 参考文献

- [1] OpenFlow Consortium. OpenFlow website. <http://openflowswitch.org/>.
- [2] Open vSwitch information: <http://openvswitch.org>
- [3] Trema: <http://trema.github.com/trema/>
- [4] POX, Python-based OpenFlow Controller: <http://www.noxrepo.org/pox/about-pox/>.
- [5] MaxMind LLC, GeoIP, <http://www.maxmind.com/geoip/>
- [6] 気象庁 Homepage: [http://www.seisvol.kishou.go.jp/eq/EEW/kaisetsu/\(2013年2月現在\)](http://www.seisvol.kishou.go.jp/eq/EEW/kaisetsu/(2013年2月現在))
- [7] R. Boivie et al., Explicit Multicast (Xcast) Basic Specification, IETF Internet-Draft, <http://www.ietf.org/internet-drafts/draft-ooms-xcast-basic-spec-01.txt>, 2001.
- [8] Gude, N., Koponen, T., Pettit, J., Pfaff, B., Casado, M., McKeown, N., and Shenker, S. NOX: towards an operating system for networks. *SIGCOMM Computer Communication Review* 38, 3 (2008), 105–110.
- [9] T. Imielinski and J. Navas, "GPS-Based Addressing and Routing," In *IETF, Network Working Group, Request for Comments, RFC 2009*, Nov. 1996.
- [10] J. C. Navas and T. Imielinski, "Geocast-geographic addressing and routing," In *Mobile computing and networking*, pp. 66 - 76, 1997.
- [11] C. Maihöfer, "A Survey of Geocast Routing Protocols," In *IEEE Communications Surveys & Tutorials*, vol. 6, no. 2, pp. 32-42, 2004.
- [12] GUEYE, B., ZIVIANI, A., CROVELLA, M., AND FDIDA, S, "Constraint-based geolocation of Internet hosts," In *Proceedings of the Internet Measurement Conference*, pp.288–293, 2004.
- [13] J.C. Navas and T. Imielinski, "Geocast – geographic addressing and routing," In *Proc. of ACM/IEEE MOBICOM '97*, 1997.
- [14] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, 38(2):69–74, 2008.

## 発表文献

- [15] Kentaro Yoshida, Daisuke Miyamoto, Yuji Sekiya, "Design and Evaluation of Area Limited Multicast with OpenFlow," In Proceedings of the 3rd Annual International Conference on Network Technologies & Communications (NTC), October 2012.
- [16] Kentaro Yoshida, Daisuke Miyamoto, Yuji Sekiya, "Area Limited Multicast with OpenFlow," In AsiaFI School, Workshop, and Meetings (poster session), August 2012.
- [17] 吉田賢太郎, 宮本 大輔, 関谷 勇司. "OpenFlow を用いたエリア限定マルチキャスト", 信学技報 112(134) (IN2012-34) pp.7-12, 2012 年 7 月

## 付録

表 1 OpenFlow スイッチの総数が 10 台で送信者が 1 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	108	107	103	112	109	112	111	116	126
2 回目	105	110	104	109	112	146	114	111	122
3 回目	109	107	102	104	109	108	111	110	120
4 回目	106	111	100	108	109	106	109	113	125
5 回目	105	109	104	107	111	110	113	111	121
6 回目	130	120	113	111	129	106	131	127	161
7 回目	106	111	099	131	108	122	109	113	124
8 回目	111	109	103	106	111	108	113	114	121
9 回目	107	105	102	110	109	106	112	114	119
10 回目	106	110	105	108	107	104	109	114	122

表 2 OpenFlow スイッチの総数が 10 台で送信者が 3 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	106	112	102	111	113	110	116	115	123
2 回目	104	109	101	109	113	109	114	113	120
3 回目	103	109	101	110	111	109	113	112	120
4 回目	104	109	100	110	111	109	113	113	122
5 回目	103	109	100	109	111	109	113	113	122
6 回目	109	110	98	106	112	109	113	113	122
7 回目	109	109	101	109	110	109	113	113	121
8 回目	108	108	99	109	111	108	113	113	121
9 回目	108	108	99	109	111	108	114	113	121
10 回目	108	107	100	108	110	108	113	113	121

表 3 OpenFlow スイッチの総数が 10 台で送信者が 5 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	107	106	103	105	109	110	114	112	124
2 回目	115	124	118	107	129	128	130	126	148
3 回目	124	123	121	121	124	122	124	131	171
4 回目	127	186	144	156	159	151	128	166	136

5 回目	131	121	182	126	118	126	151	134	135
6 回目	115	129	117	125	125	130	131	127	144
7 回目	139	136	123	161	129	193	125	156	133
8 回目	172	128	151	123	157	113	154	119	127
9 回目	123	132	129	127	137	117	129	136	142
10 回目	127	160	110	140	157	131	157	160	170

表 4 OpenFlow スイッチの総数が 10 台で送信者が 10 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	107	111	99	106	107	117	110	112	124
2 回目	109	113	102	109	111	110	109	115	121
3 回目	107	109	100	106	109	112	113	112	119
4 回目	112	107	99	105	106	110	110	110	124
5 回目	110	111	101	167	110	108	108	157	121
6 回目	108	109	100	106	108	111	112	113	119
7 回目	111	107	103	110	106	110	110	111	123
8 回目	109	110	101	108	168	108	107	114	121
9 回目	108	108	153	162	107	113	111	162	118
10 回目	110	107	103	110	111	111	109	110	122

表 5 OpenFlow スイッチの総数が 46 台で送信者が 1 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	136	126	129	132	137	133	141	137	150
2 回目	135	122	124	125	127	135	135	134	145
3 回目	132	128	121	133	133	135	130	139	143
4 回目	128	123	126	165	129	131	135	136	145
5 回目	124	128	124	126	137	136	133	132	143
6 回目	130	141	134	132	143	148	152	152	163
7 回目	139	135	126	128	129	128	135	134	146
8 回目	125	130	124	132	131	137	133	136	143
9 回目	130	128	121	143	134	142	138	137	148
10 回目	136	135	127	127	128	131	136	133	144

表 6 OpenFlow スイッチの総数が 46 台で送信者が 3 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
--	-----	----	----	----	----	----	----	----	----



1 回目	130	133	120	137	136	137	138	134	147
2 回目	124	132	119	127	134	134	139	133	145
3 回目	125	132	119	125	135	133	137	133	145
4 回目	124	132	119	126	133	134	138	133	144
5 回目	125	131	118	125	134	134	137	134	145
6 回目	124	131	119	125	134	131	136	132	144
7 回目	124	131	119	126	134	133	137	133	145
8 回目	125	131	125	132	131	135	138	131	144
9 回目	123	130	126	133	131	133	135	133	144
10 回目	131	130	129	136	131	131	138	133	143

表 7 OpenFlow スイッチの総数が 46 台で送信者が 5 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	124	136	125	130	131	134	136	135	140
2 回目	150	128	144	151	149	144	150	157	155
3 回目	187	151	135	146	140	159	154	150	160
4 回目	134	148	149	174	158	180	170	178	187
5 回目	150	175	131	143	209	207	153	149	165
6 回目	173	144	134	145	141	213	143	173	158
7 回目	142	135	150	195	145	151	147	203	173
8 回目	136	162	142	141	184	130	151	139	174
9 回目	164	153	159	157	140	134	179	137	158
10 回目	181	146	163	159	154	146	156	144	165

表 8 OpenFlow スイッチの総数が 46 台で送信者が 10 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	126	138	128	128	134	131	145	138	146
2 回目	122	129	138	132	136	134	139	135	148
3 回目	128	133	128	130	133	132	137	141	145
4 回目	125	131	128	128	130	131	133	137	141
5 回目	133	129	187	133	136	137	141	140	215
6 回目	129	134	120	130	194	134	137	142	142
7 回目	126	131	125	126	132	130	134	137	139
8 回目	128	126	155	133	136	135	137	131	146
9 回目	130	133	182	130	133	132	135	139	194
10 回目	126	129	125	128	130	139	133	135	148

表 9 OpenFlow スイッチの総数が 100 台で送信者が 1 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	169	188	197	163	200	172	185	204	177
2 回目	215	164	149	169	168	157	179	177	173
3 回目	160	162	155	161	170	162	164	173	172
4 回目	164	165	152	163	160	160	171	179	222
5 回目	163	162	149	169	166	156	168	175	174
6 回目	161	217	168	178	174	175	185	194	171
7 回目	161	174	151	160	162	157	170	178	177
8 回目	161	162	154	158	167	163	165	166	178
9 回目	168	167	153	164	163	162	169	169	181
10 回目	165	166	150	158	160	158	170	167	177

表 10 OpenFlow スイッチの総数が 100 台で送信者が 3 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	183	176	157	181	171	167	173	180	179
2 回目	173	170	157	163	170	162	169	170	178
3 回目	163	163	157	162	161	166	168	170	178
4 回目	163	171	161	164	158	162	173	179	176
5 回目	160	162	154	164	157	161	168	167	179
6 回目	160	171	149	160	159	161	170	169	177
7 回目	170	166	158	164	157	162	168	168	179
8 回目	173	169	156	172	159	160	168	169	175
9 回目	163	162	159	165	158	162	170	169	178
10 回目	171	167	164	160	159	161	168	170	183

表 11 OpenFlow スイッチの総数が 100 台で送信者が 5 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	243	202	161	166	163	163	172	176	195
2 回目	168	217	175	171	211	180	189	191	195
3 回目	195	175	171	181	182	195	195	185	235
4 回目	213	159	174	184	184	165	192	189	292
5 回目	189	181	190	173	179	176	233	210	187
6 回目	205	206	203	180	190	182	241	173	188

7 回目	175	184	189	190	207	190	185	177	195
8 回目	201	193	176	192	177	188	189	191	186
9 回目	221	182	190	190	212	180	192	169	211
10 回目	188	186	183	184	182	185	184	193	227

表 12 OpenFlow スイッチの総数が 100 台で送信者が 10 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	163	159	164	166	179	177	240	175	186
2 回目	169	165	161	161	169	172	171	164	183
3 回目	164	161	156	229	166	168	179	170	177
4 回目	167	157	155	156	173	172	168	166	184
5 回目	225	164	158	228	218	173	170	162	173
6 回目	163	160	157	160	172	178	180	170	175
7 回目	160	169	162	165	181	179	219	166	188
8 回目	166	173	166	164	170	171	186	166	181
9 回目	249	158	156	267	205	178	175	168	174
10 回目	160	168	152	166	165	167	166	172	176

表 13 OpenFlow スイッチの総数が 145 台で送信者が 1 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	217	226	222	228	220	254	234	246	226
2 回目	220	209	226	210	265	221	229	228	225
3 回目	219	216	220	210	216	218	223	224	232
4 回目	221	220	220	226	208	214	219	228	228
5 回目	219	211	215	209	215	225	221	218	222
6 回目	220	226	224	219	214	276	236	246	243
7 回目	223	212	215	213	246	213	220	220	226
8 回目	216	217	224	211	217	222	217	217	228
9 回目	228	217	221	206	214	207	224	224	234
10 回目	228	208	215	213	217	212	222	220	224

表 14 OpenFlow スイッチの総数が 145 台で送信者が 3 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	229	207	222	278	224	225	221	237	230
2 回目	218	210	224	225	217	228	221	227	233

3 回目	219	232	219	215	223	225	222	228	233
4 回目	215	211	228	217	217	220	232	223	233
5 回目	229	220	213	217	233	223	220	224	234
6 回目	217	219	220	221	219	221	221	223	232
7 回目	217	220	213	224	216	224	221	222	232
8 回目	220	220	218	214	221	221	220	220	232
9 回目	218	231	216	227	222	220	220	226	231
10 回目	220	228	217	222	221	225	220	228	238

表 15 OpenFlow スイッチの総数が 145 台で送信者が 5 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	250	212	270	217	261	237	222	243	284
2 回目	248	226	272	236	244	249	236	246	262
3 回目	247	221	239	253	272	257	238	239	291
4 回目	275	226	240	231	292	237	242	277	150
5 回目	304	234	246	223	244	237	236	232	176
6 回目	226	297	246	237	234	241	225	226	158
7 回目	235	280	227	239	239	241	231	238	161
8 回目	233	229	241	260	278	238	218	303	291
9 回目	273	233	271	223	242	290	215	299	254
10 回目	271	229	238	219	238	291	235	243	239

表 16 OpenFlow スイッチの総数が 145 台で送信者が 10 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	216	227	226	214	224	241	218	222	256
2 回目	221	214	220	207	226	227	218	225	242
3 回目	221	218	218	211	219	249	218	215	244
4 回目	269	213	222	210	226	231	213	216	238
5 回目	230	223	226	268	221	299	279	217	244
6 回目	218	220	218	214	216	235	219	232	243
7 回目	226	228	218	209	227	230	220	228	242
8 回目	215	225	211	206	208	232	233	214	245
9 回目	277	219	271	262	213	293	265	221	239
10 回目	216	225	220	207	222	223	219	216	245

表 17 OpenFlow スイッチの総数が 199 台で送信者が 1 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの

送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	1181	1206	1231	1222	1216	1200	1255	1197	1244
2 回目	1343	1390	1396	1384	1388	1371	1419	1368	1407
3 回目	1513	1561	1564	1547	1553	1533	1587	1542	1570
4 回目	1675	1729	1729	1710	1718	1699	1774	1705	1734
5 回目	1836	1893	1900	1918	1882	1874	1947	1868	1898
6 回目	1999	2064	2063	2079	2064	2038	2110	2033	2062
7 回目	2187	3418	3403	3415	3385	3365	3439	3372	3390
8 回目	3515	2417	2403	2415	2386	2365	2439	2372	2390
9 回目	2514	3743	3730	3744	3711	3693	3780	3699	3725
10 回目	2679	2743	2730	2744	2711	2694	2781	2699	2725

表 18 OpenFlow スイッチの総数が 199 台で送信者が 3 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	1209	1222	1180	1226	1190	1205	1202	1232	1202
2 回目	1176	1191	1171	1171	1180	1195	1190	1215	1190
3 回目	1185	1178	1215	1177	1180	1194	1189	1193	1200
4 回目	1175	1181	1171	1179	1178	1187	1180	1182	1196
5 回目	1181	1180	1168	1178	1180	1181	1208	1182	1190
6 回目	1189	1180	1170	1175	1178	1184	1186	1181	1190
7 回目	1182	1179	1169	1179	1178	1182	1187	1183	1190
8 回目	1234	1179	1171	1178	1177	1181	1191	1191	1189
9 回目	1174	1180	1169	1176	1178	1181	1191	1191	1189
10 回目	1201	1177	1171	1176	1188	1186	1198	1180	1189

表 19 OpenFlow スイッチの総数が 199 台で送信者が 5 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	1343	1179	1224	1179	1177	1189	1196	1192	1189
2 回目	1195	1201	1176	1197	1179	1201	1203	1199	1193
3 回目	1191	1199	1216	1199	1194	1195	1217	1191	1198
4 回目	1205	1198	1192	1191	1188	1209	1218	1193	1247
5 回目	1184	1202	1188	1221	1192	1211	1189	1228	1203
6 回目	1188	1218	1191	1224	1204	1213	1229	1255	1204
7 回目	1190	1245	1185	1188	1184	1231	1268	1263	1210
8 回目	1182	1202	1192	1208	1192	1224	1199	1205	1215

9 回目	1199	1214	1205	1243	1191	1217	1191	1221	1207
10 回目	1201	1211	1253	1236	1200	1232	1192	1199	1207

表 20 OpenFlow スイッチの総数が 199 台で送信者が 10 秒の時間間隔を空けて  
ジオキャストエリアをそれぞれの地方に限定してメッセージを送信したときの  
送信者・受信者間の遅延測定結果[ms]

	北海道	東北	関東	中部	近畿	中国	四国	九州	沖縄
1 回目	1257	1183	1226	1221	1186	1189	1192	1195	1229
2 回目	1188	1192	1171	1192	1194	1190	1187	1191	1199
3 回目	1194	1189	1176	1189	1182	1190	1184	1189	1205
4 回目	1180	1185	1178	1181	1180	1177	1181	1182	1201
5 回目	1244	1222	1171	1223	1194	1191	1191	1201	1268
6 回目	1179	1199	1167	1200	1208	1189	1209	1199	1203
7 回目	1181	1182	1209	1218	1190	1194	1190	1215	1201
8 回目	1175	1179	1181	1224	1194	1180	1197	1235	1206
9 回目	1238	1184	1176	1186	1179	1180	1197	1192	1255
10 回目	1202	1181	1174	1180	1191	1197	1192	1181	1214