

# 災害時に向けたネットワーク適応型 コミュニケーションシステムの提案

松本 尚幸<sup>1</sup> 島田 秀輝<sup>2</sup> 佐藤 健哉<sup>1</sup>

概要：近年のスマートフォンやタブレット端末の普及とインターネットにおけるソーシャル・ネットワーキング・サービスの発展により，誰もが容易に他者とコミュニケーションをすることができる。しかし，自然災害により通信インフラが破壊された地域では，コミュニケーション手段を掲示板などに頼るしか無い状況になってしまう。これに対して，携帯電話会社は災害用伝言サービスを提供しているが，小さいコミュニティでの利用に適していない点や，リアルタイム性に欠けているといった問題点が存在する。本研究は災害時でのモバイル端末を用いたコミュニケーションシステムを提案する。災害時における無線 LAN 環境での利用を想定し，多くの人が所持しているモバイル端末を用いる。サーバにはアプリケーションのインストールを必要とするが，クライアントは端末に備わっている Web ブラウザを用いることで，事前の準備を必要とせず容易にシステムを利用することができる。本システムはインターネット接続の有無に適応可能であり，グローバルサーバと連携したサービスを提供可能となっている。また，本システムをモバイル端末を用いた実機環境にて遅延時間を計測，評価し，有効性を示す。

## A Proposal of Network Adaptive Communication System for Disaster Areas

NAOYUKI MATSUMOTO<sup>1</sup> HIDEKI SHIMADA<sup>2</sup> KENYA SATO<sup>1</sup>

### 1. はじめに

#### 1.1 背景

近年，スマートフォンやタブレット端末の普及により通話やビデオチャットだけでなく，インターネットを利用したコミュニケーションツールやソーシャル・ネットワーキング・サービス (SNS) の利用が活発となっている。これらのサービスを利用することにより家族や友人とのコミュニケーションや不特定多数の人に対しての情報発信を容易にすることができるため，モバイル端末を利用したコミュニケーションは日常生活において他者とのコミュニケーション手段として有効である。しかし，地震や異常気象などの自然災害により被害を受けた地域では，住民は避難所での生活を強いられる事となることがある。こういった被災地では通信インフラが破壊され，住民間，家族間でのコミュニケーション手段を掲示板等の手段に頼るしか無い状況と

なる。

災害によって通信インフラが破壊された場合にはインターネット回線を利用したコミュニケーション手段の利用が不可能となる。このような不足の事態に備えて国内の携帯電話会社各社は災害用伝言サービス [1] を提供している。この災害用伝言サービスは固定電話やスマートフォンを用いることで伝言やメッセージを残すことができるサービスである。各キャリアが共通の伝言サービスを提供している他，一部のキャリアでは独自のサービスを提供している。これらのサービスは災害時に発生する輻輳を回避する目的で作成されているため，緊急時における目的の人物の安否確認においては有効と言える。また，携帯会社が契約者に対して提供しているサービスであるため信頼性が高い。契約を結んでいる利用者はこれらのサービスを利用ことができ，家族や知人の安否確認や伝言を残すことができる。

#### 1.2 目的

本研究は災害時でのモバイル端末を用いたコミュニケー

<sup>1</sup> 同志社大学大学院理工学研究科情報工学専攻

<sup>2</sup> 同志社大学研究開発推進機構

シオンシステムを提案することを目的としている。災害時での利用を想定するため導入に要するコストを最小限に抑える必要がある。そのため、提案システムでは無線 LAN を構築する機器を必要とするが、サーバとクライアントには、今日、多くの人が日常的に所持しているモバイル端末を主に用いる。クライアントでは、事前の準備を必要とせず、モバイル端末に標準的に搭載されている Web ブラウザを用いてサーバにアクセスする。サーバは LAN 内のクライアントからのリクエストに応える役割を持つ。サーバの役割を持つ端末はサーバの機能を持つためのアプリケーションの準備を必要とする。また、本システムはインターネットへの接続の有無に適応可能であり、インターネットへの接続がある場合は本システムにアクセスするすべての端末と通信が可能である。インターネットへの接続が無い場合はサーバ端末が存在する LAN 内でのみ通信が可能となる。通信相手となる端末が同一 LAN 内に存在しない場合はサーバ端末に通信内容が一時保存され、再度インターネットへ接続された時に目的の端末へと通信が開始される。本システムを用いることでインターネット回線の有無に適応可能であり、災害時における LAN 内に存在する端末同士の効率的、かつリアルタイムなコミュニケーションシステムを実現する。

## 2. 既存サービスの問題点

1.1 節で述べた、災害用伝言サービスは安否確認には有効な手段である。しかし、このサービスは伝言やメッセージを残すためには、残したい対象の人物の電話番号を知っている必要があるため、目的の人物は電話番号の知っている家族や友人、知人に限られてしまうといった問題点が存在する。また、これらのサービスは大規模かつ広範囲の利用が想定され、特定の人物に対してのコミュニケーションで利用される。それ故、被災地における避難所等、小規模のコミュニティでの利用に適しておらず、また、リアルタイムのコミュニケーションには利用できない。

被災地の避難所では住民の安否を確認するために掲示板を設置し、張り紙を貼ることで伝言を残す。安否確認を目的として掲示板を利用するならば有効的と言えるが、コミュニケーションの手段としての掲示板の利用は非常に非効率的である。このため、インターネットが利用できない場での効率的なコミュニケーション手段が必要である。

## 3. 提案システム

### 3.1 構成

本研究は被災地でのモバイル端末を用いたコミュニケーション手段を提案することを目的としている。本提案システムの概略図を図 1 に示す。

今日、多くの人が日常的に所持しているスマートフォンと標準的に搭載されている Web ブラウザを用いて、LAN

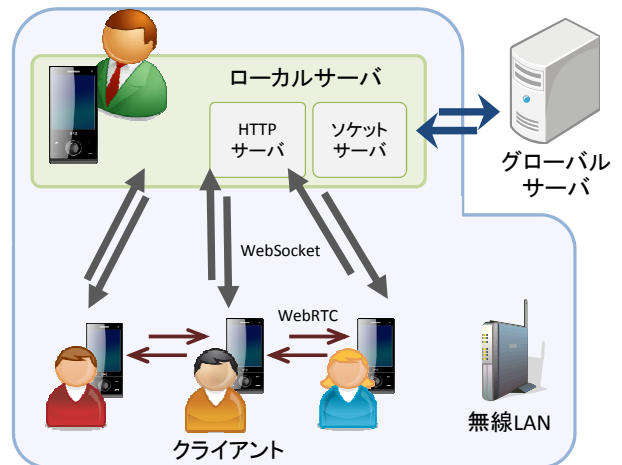


図 1 提案システム概略図

内に設置されたローカルサーバにアクセスすることで誰もが容易にコミュニケーションをとることができる。ローカルサーバには Web アプリケーションを提供する役割とクライアントからの通信を処理する役割を持つ。本システム内の通信はブラウザ内の WebSocket 通信 [2] と WebRTC 通信 [3] によって行われる。サーバ・クライアント間は WebSocket 通信のみで通信が行われ、クライアントが送受信するメッセージの中継、WebRTC 通信の開始に必要なメッセージをクライアントにブロードキャストするために利用する。クライアント間の通信は WebRTC 通信によって動作し、サーバを介さず通信することでリアルタイムな通信が可能となる。

### 3.2 ローカルサーバ

クライアント同士が通信をしようことを可能にするためには、無線 LAN 内にクライアント同士を仲介する役割と、Web サーバとしてクライアントからのアクセスに応え、クライアントからのメッセージを他のクライアントへと転送する役割を持つローカルサーバの存在が必要となる。また、クライアントからのアクセスがあった際には、システム内のクライアントを識別するための ID を割り振る。クライアントのシステムへの参加・離脱はシステムに参加している全てのクライアントに通知することができるため、ユーザはシステムに誰が参加しているか知ることができる。クライアントが不在の場合でも一時的にメッセージを保管することもできる。

### 3.3 クライアント

本研究では、システムを Web アプリケーションとして利用するため、システムに参加するクライアントは Web ブラウザを利用可能である必要がある。Web アプリケーションは JavaScript で記述されており、HTML5 の機能の一部

表 1 実装環境

ローカルサーバ 1	機器名称	Asus Nexus7
	OS	Android 4.4.2
	CPU	Qualcomm Snapdragon S4 Pro 1.5GHz(Quad Core)
	RAM	2GB
ローカルサーバ 2	機器名称	FUJITSU F-05D
	OS	Android4.0.4
	CPU	OMAP4430 1.2GHz (Dual Core)
	RAM	1GB
グローバルサーバ	OS	Windows7
	CPU	Intel Core 2 Quad 2.66GHz
	RAM	2GB
無線規格	IEEE 802.11g	
クライアント	Android 端末 Windows7 Windows8	
ブラウザ	Chrome	

として策定されている WebSocket 通信と WebRTC 通信を利用することでサーバ・クライアント間の相互の Push 通信とクライアント同士の P2P 通信が可能としている。

### 3.4 グローバルサーバ

グローバルネットワーク上にサーバを設置することで、ローカルサーバが複数存在する場合でもクライアントを検索することが可能となる。また、ローカルサーバの敷居を超えて、別のローカルサーバに参加しているクライアントに向けてメッセージを送信する場合でもグローバルサーバを経由して送信することが可能となる。目的のクライアントが不在の場合でも、一時的にサーバにメッセージを保管することで次回参加した場合にメッセージを伝達することができる。これによって提案システムに遅延耐性を持たせる事ができる。

## 4. 実装

### 4.1 実装環境

本研究ではローカルサーバ並びに、グローバルサーバは Java 言語統合開発環境である Eclipse を用いて実装した。ローカルサーバに関しては Android アプリケーションとして作成する必要があるため AndroidSDK も用いた。ローカルサーバが提供する Web アプリケーションは HTML, JavaScript により記述した。実装において使用した機器を表 1 に示す。

### 4.2 ローカルサーバ

ローカルサーバとして動作する Android 端末は事前に無線 LAN に参加していることと、本研究で作成した An-

droid アプリケーションをインストールしている必要がある。ローカルサーバの起動は端末内にインストールされた Android アプリケーションを起動するだけでよい。ローカルサーバの実装画面を図 2 に示す。

ローカルサーバが起動がされると自身の端末が持っている IP アドレスとローカルサーバが取得したポート番号が LAN 内におけるローカルサーバの URL として表示される。同時に、サーバとしてクライアントからのアクセスを受け付けることが可能となる。クライアントからのアクセスを受け取った際には、リクエスト内容からストレージ内から対象のファイルを読み込み、読み込んだ内容をレスポンスとしてクライアントへと送信する。ローカルサーバは HTTP サーバとして動作すると同時に、ソケットサーバとしても動作し、クライアントに対しては WebSocket 通信を行うことで双方向の通信が可能となっている。ローカルサーバはクライアントの識別のためにそれぞれに ID を割り当てる役割と、クライアントからのメッセージを受け取り、それを他のクライアントへと伝える役割を持つ。WebSocket 通信による通信内容は Web アプリケーションが扱える形式で記述する必要があるため、JSON 形式に整形して通信を行う。

グローバルサーバに対しては、自身が管理しているクライアントに変更があった際に通信をすることで、グローバルサーバがクライアントの変更を察知し対応することを可能としている。

ローカルサーバアプリケーションの画面には HTTP サーバ、ソケットサーバのログ情報が表示される。表示される項目は以下の 4 つとなる。

- HTTP リクエスト
- クライアントの参加・離脱
- WebSocket 通信内容
- グローバルサーバとの通信内容

### 4.3 Web アプリケーション

クライアントとして本システムを利用するためには使用する端末内に WebSocket 通信に対応した Web ブラウザがインストールされている必要がある。本システムにおける、クライアントとローカルサーバの通信手順を示したシーケンス図を図 3 に示す。

#### (1) ローカルサーバへのアクセス

クライアントは Web ブラウザを用いてローカルサーバが示した URL にアクセスする。

#### (2) Web アプリケーションの実行

クライアントはローカルサーバが示した URL にアクセスすることで HTML ファイルを取得する。次に記述されている内容から JavaScript, その他のファイルを取得する。

#### (3) システムへの参加

```

AndroidWebSocketServer
192.168.24.52:33401 Go
css/style.css
js/dateformatter.js
js/app2.js
/192.168.2.100:36160 が参加
index.html
/192.168.2.102:51003 が退室
js/jquery.js
js/dateformatter.js
css/style.css
js/app2.js
/192.168.2.102:51008 が参加
index.html
/192.168.2.101:50056 が退室
js/jquery.js
js/dateformatter.js
css/style.css
js/app2.js
/192.168.2.101:50064 が参加
js/jquery-2.0.2.min.map
/192.168.2.101:50064 が参加
message : /192.168.2.100:36160:
{"type":"wschat","from":"192.168.2.100:36160","to":"global","date":1399547326910,"data":{"user":"192.168.2.100:36160","text":"全体へのメッセージ"}}
message : /192.168.2.102:51008:
{"type":"rename","from":"192.168.2.102:51008","to":"global","date":1399547422731,"data":"Android"}
message : /192.168.2.101:50064:
{"type":"rename","from":"192.168.2.101:50064","to":"global","date":1399547440750,"data":"WindowsPC"}
message : /192.168.2.101:50064:
{"type":"rename","from":"192.168.2.101:50064","to":"global","date":1399547479572,"data":"192.168.2.101:50064"}
message : /192.168.2.101:50064:
{"type":"rename","from":"192.168.2.101:50064","to":"global","date":1399547493699,"data":"WindowsPC"}
message : /192.168.2.101:50064:
{"type":"wschat","from":"192.168.2.101:50064","to":"192.168.2.100:36160","date":1399547544131,"data":{"user":"WindowsPC","text":"宛先を指定したメッセージ"}}

```

図 2 ローカルサーバ実装画面

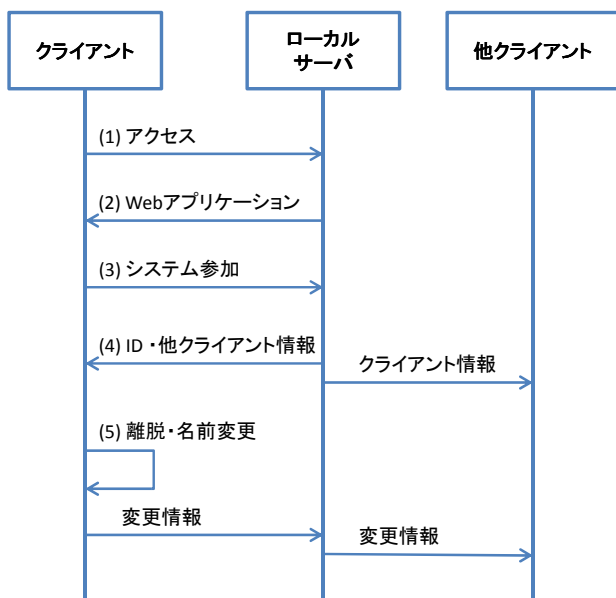


図 3 Web アプリケーション通信手順

JavaScript ファイルを実行することにより、ローカルサーバとの WebSocket 通信が開始され、本システム

への参加が完了し、Web アプリケーションとして利用することができる。

(4) 他クライアント情報の取得

システムへの参加が完了するとローカルサーバからシステムに参加している全てのクライアントの情報が伝えられ、Web ブラウザ上の UI に表示される。また、自身がシステムに参加した旨が、他の全てのクライアントに送信される。

(5) クライアント情報の変更

クライアントの参加、離脱、名前変更が発生した際に、ローカルサーバから通信を逐次受け取ることでリアルタイムにシステムに参加しているクライアントの情報が受け取ることができ、UI 上でも確認することができる。

クライアントが送受信、ローカルサーバが送信するメッセージは以下のように構成されている。

- メッセージタイプ
- 送信元
- 宛先
- 日時
- メッセージ内容

WebSocket によるメッセージはすべてローカルサーバから送信されてくるため、メッセージが何を含んでいるかを識別する必要がある。メッセージタイプによりメッセージ内容の形式も異なる。

Web アプリケーションの UI は図 4 のようになっており、画面上部に自身の名前とメッセージ内容、宛先を指定する入力フォームがある。その下部に現在システムに参加しているクライアントの名前が一覧表示される。自身の名前が変更されると変更情報が送信され、他のクライアントの Web ブラウザ上の表示名も逐次更新される。以降の領域は受信したメッセージが表示される領域となる。

クライアント間のメッセージ内容は送信元クライアントの名前とテキストで構成される、受信したメッセージから順に表示されるが、宛先にクライアント ID が指定されている場合は対象のクライアントのみがメッセージを読むことができる。

また、WebRTC 通信に対応している Web ブラウザを用いることでローカルサーバを経由しない、クライアント間の一対一通信が可能となる。

4.4 グローバルサーバ

グローバルサーバはローカルサーバからのソケット通信を受け付け、クライアントの情報を管理する。グローバルサーバの実行画面を図 5 に示す。ローカルサーバはクライアントの変更があった際にクライアントの ID、参加もしくは離脱の情報を送信する。クライアントの情報を受け取ったグローバルサーバはローカルサーバの IP アドレスと紐

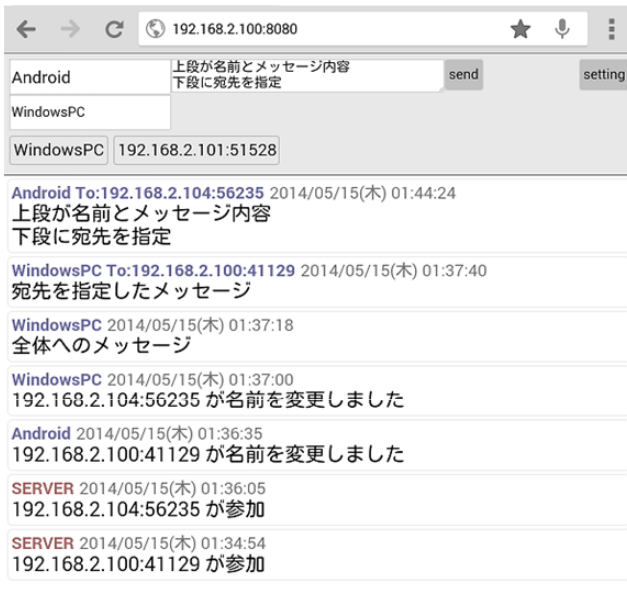


図 4 Web アプリケーション実装画面

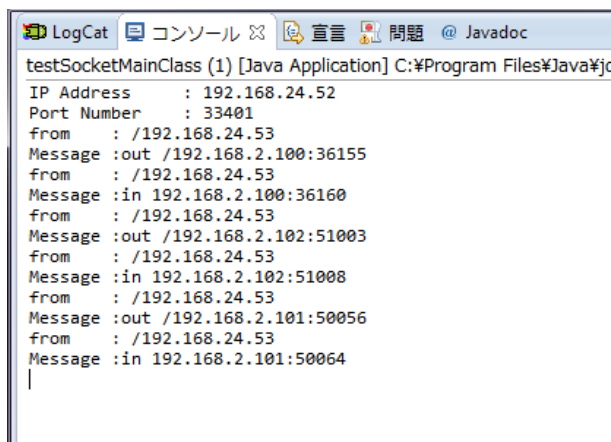


図 5 グローバルサーバ実行画面

付けて管理する。

## 5. 評価

### 5.1 評価環境とクライアント ID の割り当て

実装した本システムを Android 端末上で動作させ、システムに参加している端末数とメッセージの送信量を様々な変化させた通信にて発生する遅延時間、通信量を計測した。評価する環境については、Asus Nexus7 と FUJITSU F-05D 上にローカルサーバを動作させ、クライアントには Android 端末、Windows PC 及び Google Chrome を用いた。様々な端末数を考慮した計測を行うために、クライアントを識別するための ID には、端末の IP アドレスとブラウザが用いたポート番号を合わせることで ID とした。これにより計測において、同一端末であっても別端末として認識することを可能とした。

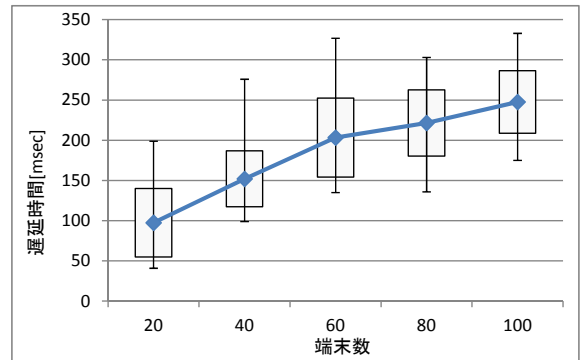


図 6 送信量 129Byte, 各端末数における遅延時間 (Nexus7)

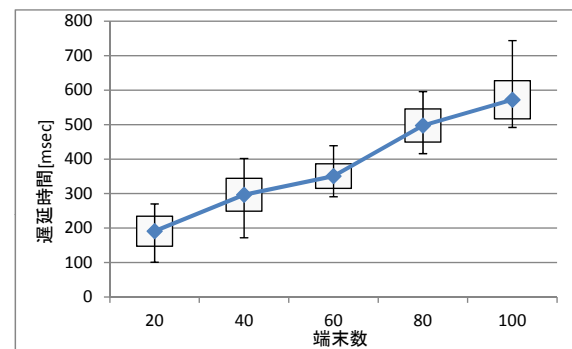


図 7 送信量 1000Byte, 各端末数における遅延時間 (Nexus7)

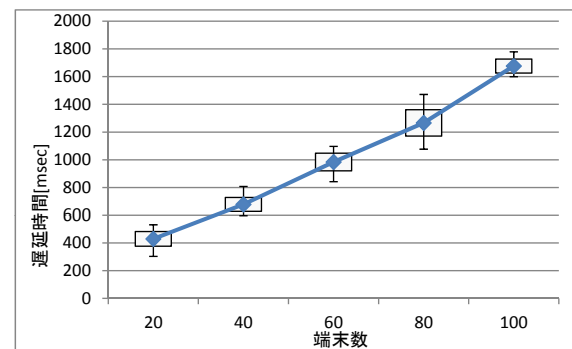


図 8 送信量 5000Byte, 各端末数における遅延時間 (Nexus7)

### 5.2 端末数と送信量による遅延時間の評価

利用をする端末数を 20 から 100 の 5 段階とし、送信したメッセージがローカルサーバを経由し、受信が完了するまでの遅延時間の計測を行った。また、それぞれの端末数において 1 メッセージあたりの送信量を 129 (最小), 1000, 5000Byte に変化させた際の遅延時間についても計測を行った。計測結果の平均値, 最低値, 最高値, 標準偏差を図 6, 図 7, 図 8, 図 9 に示す。また, 1 メッセージあたりのローカルサーバの送信 Byte 数を表 2 に示す。

## 6. 考察

図 6, 図 7, 図 8 から端末数の増加に伴って通信に要する時間が増加しているのが確認でき, 表 2 からは送信 Byte 数も端末数に比例して増加していることが確認できる。Nexus7 での計測において, メッセージ送信量が 129Byte



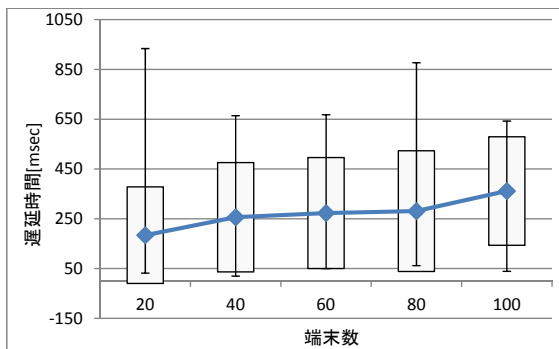


図 9 送信量 129Byte, 各端末数における遅延時間 (F-05D)

表 2 1 メッセージあたりの総送信量 [Byte]

端末数	1 メッセージの送信量		
	129	1000	5000
20	2793	21672	104560
40	5453	41280	209120
60	8117	61920	313680
80	10640	82560	418240
100	13300	103200	522800

と 1000Byte では端末台数が 100 であっても遅延時間は 1 秒未満となり、最大で 744 ミリ秒であった。これに対してメッセージ送信量が 5000Byte では遅延時間が 1600 ミリ秒にまで達し、1000Byte と比較して約 3 倍の遅延時間が発生している。これはローカルサーバがメッセージを全ての端末に送信する際に、端末 1 台ずつに通信を行っているため、端末数が増加すると CPU の処理能力の限界に達してしまい、端末の増加率以上に通信時間が必要となったためである。また、実装上端末の情報は配列に格納されており、先に処理される端末と、後に処理される端末が存在する。そのため、端末数が多い場合の計測では、端末によってメッセージの受信に要する時間に大きく差があり、その差が数百ミリ秒に及ぶことも確認することができた。

以上から、メッセージ送信量が遅延時間に大きく影響していることがわかった。今回の実装ではテキストデータのみが送信可能であるが、絵文字などの画像データをメッセージとともに送信する場合は送信量が急激に増加することが考えられる。また、送信するデータはテキストデータである必要があるため画像データをテキストデータに変換をする必要があり、より CPU を消費することとなる。今後、画像データなどを扱う場合にはシステムのパフォーマンスに影響しないかを検討する必要がある。

今回、計測により確認できた点として、ローカルサーバに掛かる負荷と処理能力不足が挙げられる。これはクライアントが参加、離脱をした際のクライアントへの通知の頻度を下げることで軽減することが可能である。また、ローカルサーバとなる端末の処理能力に応じて参加するクライアントの数に制限を設けることで CPU の専有を回避することができる。

## 7. 関連研究

藤川らの研究 [4] では、災害時に利用される災害時専用システムが自治体の専門の職員の手により運営され、地域住民が直接利用できない問題点を指摘し、平常時と災害時によって利用形態を切り替える自律的被災情報提供システムを開発している。これは本研究と同様に、平常時と災害時により、利用形態を切り替え比較的小規模のコミュニティでのコミュニケーションツールとして利用できる。本研究と異なる点として、平常時に利用する場合ではコミュニケーションツールとしてだけでなく、自治体や行政とも連携をしての情報伝達、回覧板としての利用、イベント情報の共有ができることが挙げられる。提案システムでは、平常時、災害時ともにコミュニケーションツールとしての利用のみであり、自治体との連携は想定していない。しかし、本システムではモバイル端末と専用アプリケーションで構成されており、容易にシステムの導入をすることができる。

小山らの研究 [5] では、災害時における Twitter の利用の有用性を指摘し、Twitter を利用した災害時における DTN に基づいたメッセージ中継法を提案している。この研究はオフラインでも利用可能な Twitter クライアントの開発と Android 端末のすれ違い通信を可能とするアプリケーションである MONAC を利用し、ユーザ同士がオフライン時に保管していたメッセージを受け渡しをして、インターネットにアクセスが可能になった時点で Twitter や災害時サービスにアクセスする手法をとっている。また、すれ違い時には、アプリケーションの起動時間やユーザ同士の遭遇履歴などを利用して中継経路の設定を行っている。こちらの研究は Android アプリケーションを利用するため導入が容易であるが、利用者全員がアプリケーションをインストールしている必要がある。本研究と異なる点として、提案システムでは無線 LAN 環境が必要ではあるが、ローカルサーバのみが Android アプリケーションをインストールする必要があり、クライアントは Web ブラウザから利用する。

## 8. おわりに

本研究では、平常時と災害時のインターネット接続の有無に適応可能なコミュニケーションシステムを提案した。提案システムでは低コスト、かつ誰もが容易に利用できるためにローカルサーバには Android 端末、合わせて Android アプリケーションを用いる。システムを利用するクライアントは Web ブラウザを利用するため、事前の準備を必要とせず利用できる。平常時の利用ではグローバルサーバと連携した広範囲の利用が可能であり、災害時にはインターネット接続を必要とせず、ローカルエリア内で機能することができる。評価に関しては、2 台の Android 端末を用いて通信にて発生する遅延時間を計測し

た．端末数が 100 台，1 メッセージ 1000 文字を想定した計測においても，平均遅延時間が 1 秒未満となることが確認され，リアルタイムなコミュニケーションが可能であることが確認できた．しかし，本研究は現段階では，コミュニティ内だけの利用を想定しており，SNS や行政機関の情報と連携したシステムの提供を想定していない．また DTN の観点から，ローカルサーバ内に保存したメッセージを他端末へ受け渡しをすることでグローバルサーバにアクセスできる可能性が高まることが考えられる．今後は他の SNS や行政機関の情報との連携，他のローカルサーバ間の連携を検討していく予定である．

## 参考文献

- [1] 総務省 | 安全・信頼性の向上 | 災害用伝言サービス，  
[http://www.soumu.go.jp/menu\\_seisaku/ictseisaku/net\\_anzen/hijyo/dengon.html](http://www.soumu.go.jp/menu_seisaku/ictseisaku/net_anzen/hijyo/dengon.html)，(参照 2014.05.13)
- [2] IETF "The WebSocket protocol"，  
<http://tools.ietf.org/html/draft-ietf-hybi-thewebsocketprotocol-17>，(参照 2014.05.13)
- [3] W3C, "WebRTC 1.0: Real-time Communication Between Browsers"，  
<http://www.w3.org/TR/webrtc/>，(参照 2014.05.13)
- [4] 藤川昌浩，亀川誠，松本佳昭，吉木大司，森信彰，松野浩嗣，" 災害発生時に防災システムの効果を最大限に高めるための地域コミュニティシステムの開発"，情報処理学会第 74 回全国大会，Vol.2012，No.1，pp45-47(2012)
- [5] 小山由，水本旭洋，今津眞也，安本慶一，" 大規模災害時の安否確認システムと広域無線網利用可能エリアへの DTN に基づいたメッセージ中継法" 電子情報通信学会技術研究報告，Vol.112，No.44，pp171-177(2012)