

## DTN プロトコルを用いたスマート端末によるマルチホップネットワークの実現方法

北田真也<sup>1</sup> 佐藤剛至<sup>1</sup> 内田法彦<sup>2</sup> 平川剛<sup>3</sup> 柴田義孝<sup>3</sup>

災害時は通信基地局の破損や停電によって、通信インフラを利用したスマート端末の活用が困難となる。我々はこれまでに SDN 技術を用いて、破損や輻輳による通信インフラの障害や状態変化を検出した後、衛星通信など非常時にも有効な通信回線に切り替えることで災害時も利用できる Never Die Network(NDN)の研究[1]を行なってきた。一方、Delay Tolerant Network(DTN)の研究の発展により、災害時の劣悪な通信環境下において非常に有効的なネットワークが実現可能となってきた[2]。そこで本研究では災害時の通信障害や伝送遅延を考慮したスマート端末上で動作する DTN ベースのマルチホップネットワークの実現を目指す。

### 1. はじめに

災害時にはスマート端末を活用することで安否情報や被災地の情報を共有することが可能である。しかし災害発生後は通信基地局の破損やアクセス集中、大規模な停電により、通信インフラの活用が困難となる。

Mobile Ad hoc Network (MANET)はインフラレスな環境で展開できるネットワークである。被災者間での展開を考慮した場合、端末を持つ多くの被災者が移動するため、トポロジが変動し続ける可能性が高く、end-to-end 経路を維持するのが困難であると思われる。一方、Delay / Disruption Tolerant Network (DTN)は通信途絶や中断が生じた際にデータをストレージに蓄積し、新たなノードを発見した際に送信する手法である。近年では盛んに行われている研究であり、災害時の劣悪な通信環境下においても活用が可能となってきた。

本論文では災害時の劣悪なネットワーク環境への対応として、スマート端末による DTN ベースのマルチホップネットワークの実現研究のこれまでの経過報告と今後のアプローチを提案する。本研究におけるマルチホップネットワークはバケツリレー式にノード同士の通信を行なうことでアドホックネットワークを実現する。MANET のように動的に経路を管理しないため、常に利用できる経路が存在せず、end-to-end を実現するためにアプリケーション層およびバンドル層での実装が重要になる。本研究では既存のフラッディングベースプロトコルである Epidemic Routing [2] を基に拡張したルーティングで通信の制御を行なう。我々が拡張したプロトコルでは不要な通信を減らすことで、ノード間の通信途絶だけでなくデータの重複による情報の錯綜と省電力な動作も考慮する。また、災害発生後も電源の供給とネットワークが利用できる災害対策本部にデータサーバーを設けて、被災地からの情報収集も行なう。

### 2. 関連研究

災害時のネットワーク環境を考慮した DTN の設計では不安定なノード接続で目的先まで到達させること、不要な通信とデータ重複を減らしてバッテリーリソースを考慮することが重要となる。我々はシミュレータ上で行われている DTN の研究と実機を使用した研究を調査した。

#### 2.1 実機上での MANET システム

実機上のプロトコル開発の研究はハードウェアやOSのスペックに依存することが多く、この条件下でより効果的なプロトコルを設計する必要がある。スマート端末で形成するhybrid cellularMANET [3]の研究は筆者らが設計したマルチパスルーティングプロトコルHMANETでデータ転送を制御する。更にこの研究では既存プロトコルのHybrid

Wireless Mesh Protocol (HWMP)よりも優れた性能であることを実証した。MANET と DTN を組み合わせた研究[4]ではネットワークの状況に応じて MANET と DTN を切り替える機能をスマート端末向けに実装している。この自動で切り替えを行なうアルゴリズムはネットワークリソースの浪費を回避することを可能にする。

#### 2.2 シミュレータを使用した高性能プロトコル

シミュレータを使ったプロトコル開発はハードウェアに依存する事なく、複雑な計算や処理が可能である。3次元位置情報を用いた DTN プロトコル[5]は各ノードの位置に基づいて経路を選択するプロトコルであり、既存プロトコルである Epidemic Routing よりも到達率や遅延時間において優れた性能を示している。area of interest (AOI)と呼ばれる情報の関心度に基づいて蓄積した情報の集約を行なう DTN の研究[6]はオーバーヘッドを減らすように設計されており、Epidemic Routing よりも少ない遅延時間で転送できることを示した。また、SMART[7]は遭遇頻度に基づいてソーシャルマップを作成し、各リンクに重みをつける。Multicent[8]は複数あるノードに優先順位をつけてルーティングとパケットの転送を行なう。更に QoS にも対応しており、柔軟なネットワークの構築が可能である。

我々が実現するシステムは災害時における実機上で動作するものである。災害時は多くのノードが移動することと

1 岩手県立大学ソフトウェア情報学研究科  
Graduate school of Software and Information Science, Iwate Prefectural University Graduate School

2 福岡工業大学情報工学部情報通信工学科  
Department of Information and Communication Engineering, Faculty of Information Engineering, Fukuoka Institute of Technology

3 岩手県立大学ソフトウェア情報学部  
Faculty of Software and Information Science, Iwate Prefectural University

有限であるバッテリーリソースやデータの重複による情報の錯綜を避けることを考慮する必要がある。そこで我々はスマート端末が持つ機能に注目し、DTNへの導入を考案した。以下のセクションでこれまでの取り組み、その結果と今後の取り組みについて提案する。

### 3. システム構成

#### 3.1 システム概要

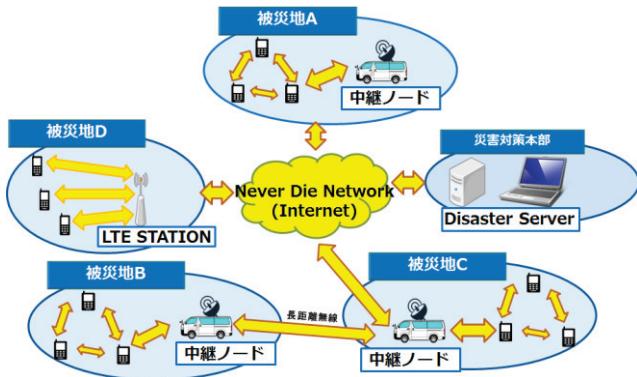


図 1. システム概要図

図 1 では本研究のシステム概要図を示す。図中の被災地 A, B, C は以下の条件を想定する。

- 各地点で LTE や 3G などの携帯通信網が利用不可能
- ノードとなるスマート端末はエリア内を移動する
- 災害発生後に中継ノードが各地点に配置される

この条件に基づいたネットワークを実現するために被災地 A, B, C では DTN ベースのアドホック通信を使用する。本研究のアドホック通信では安否情報や被災地の状況をメッセージデータや画像データとして扱う。アドホック通信で中継されるデータは中継ノードを目的先のノードとしてホップする。本研究における中継ノードは長距離無線や Never Die Network (NDN) を車載したインターネットアクセスが可能な移動中継車であり、災害対策本部や他の地点にデータを転送する。本研究で想定する災害対策本部は以下の条件を満たす。

- 災害発生後も電源の供給とインターネットアクセスが可能
- 津波や土砂崩れのような二次災害の影響を受けない
- Disaster Server が情報の取りまとめを行なう

被災地 D は災害発生後も LTE や 3G などの携帯通信網が利用できることを想定し、災害発生後も基地局を通じてデータ通信を利用できる。以上の構成より本システムではスマート端末所持者間の通信、中継ノードとの通信、中継ノードと他地点の通信を実現する。

#### 3.2 ルーティングアルゴリズム

現段階のルーティングアルゴリズムは Epidemic Routing の仕組みに準拠して実装しており、隣接する全てのノードにデータを転送する。本研究では past\_connection と呼ばれます。

るリストに接続したノードを履歴として保存し、このリストを基に接続する隣接ノードを決定して実現する。

Step 1. 隣接ノードを探索し、リストを作成

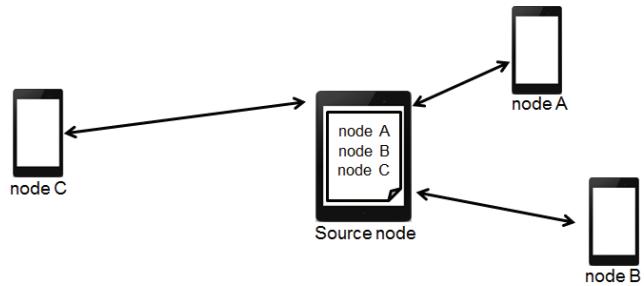


図 2. 隣接ノード探索の仕組み

最初にソースノードは隣接ノードを探索し、発見したノードの一覧を作成する。ソースノードは探索ビーコンを隣接ノードにブロードキャストし、応答を受けて取ることで隣接ノードが近くにいることを認識する。図 2 ではノード探索の例を示す。ノード A はソースノードに最も近いためソースノードの探索ビーコンに早く応じることができたとする。他のノードからの応答を受信するたびにこのリストは更新される。以下ではこのリストを Discovery's List と呼ぶ。

Step 2. ソースノードは past\_connection が空か確認する

If past\_connection is empty;

Discovery's List の先頭ノードと接続する。

Else;

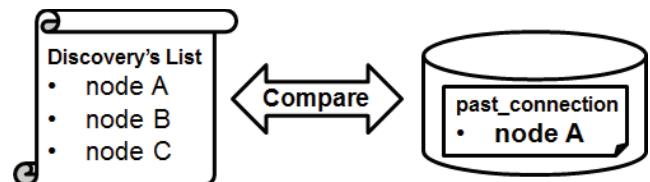


図 3. リストの比較例

Discovery's List と past\_connection を比較する。past\_connection は過去に接続したノードを履歴として保存されるリストである。このとき Discovery's List の n 番目のノードに注目する。

If Discovery\_list (n) &gt; past\_connection;

past\_connection に n 番目のノードが存在しなかつた場合、ノード n と接続し past\_connection に追加する。

Else;

ノード n が既に past\_connection に存在した場合はノード n とは接続しない。

その後 n+1, n+2... と次のノードを確認する。

図 3 では Step 2 の例を示す。最初に Discovery's List の先

頭であるノード A が past\_connection に存在するか確認する。この例ではノード A が past\_connection に存在するためノード A とは接続しない。次に Discovery's List の 2 番目であるノード B に注目して past\_connection と比較をする。ノード B は past\_connection に存在しないのでソースノードはノード B との接続を決定する。もしリストのノードが全て past\_connection に存在した場合、ソースノードは隣接ノードの探索を継続してリストを更新し続ける。このステップで IP 層までの接続が確立される。

### Step 3. TCP を使用したデータ転送

接続後の確立後、ストレージ内の全てのデータをノードに転送する。災害時の画像データはテキストデータよりも情報として有効的であるが、オーバーヘッドと転送時間の増加の原因になる。本システムでは TCP 方式を使用しているのでデータ転送中にパケットがドロップしてもデータは再送される。またデータ転送中に接続が途絶した場合は直ちにそのノードとの再接続を試みる。もしそのノードが通信範囲外に移動していた場合は他のノードと新しく接続を確立する。

各データにはシーケンス ID が振ってあり、同一のデータを受信した場合はストレージに格納せずに破棄する。

現段階のアルゴリズムは past\_connection を使用した接続管理とデータへのシーケンス ID 付与で不要な通信を減らし、バッテリー消費量とデータの重複を考慮している。

### 3.3 システムアーキテクチャ

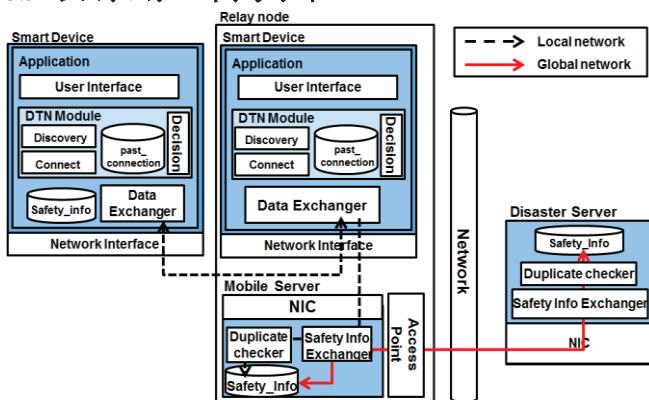


図 4. システムアーキテクチャ

図 4 では本研究で実現するシステムのアーキテクチャを示す。本研究の DTN ベースのネットワークシステムは 1 つのアプリケーションとして実現する。このアプリケーションには User Interface, Data Exchanger, DTN Module のモジュールと情報を蓄積するストレージとなる safety\_info で構成される。更に DTN Module は Discovery, Decision, Connect と接続記録を管理する past\_connection で構成される。本システムを使用するユーザーは User Interface をユーザー自身で操作し、メッセージや画像データを安否情報や被災地に

関する情報として入力する。その後隣接ノードや中継ノードを DTN Module の Discovery で探索する。次に Decision で past\_connection を参照して接続ノードを決定し、それに基づいて Connect で接続を行なう。ノードとの接続が完了した後、Data Exchanger を使って safety\_info のデータを転送する。Data Exchanger は TCP 通信を使用しており、接続相手のノードが別の情報を持っていた場合、受信することも可能である。その後、受信したデータは safety\_info に蓄積される。中継ノードではスマート端末、Mobile Server、NDN や長距離無線を使用したアクセスポイントが車載される。中継ノード内のスマート端末は Mobile Server にデータを転送する。災害時の限られた通信において情報の重複が全くない状態が望ましいが、重複率 0% を実現するのは困難である。そこで今回我々は一時的にサーバー側で重複を確認するモジュールを実装した。サーバー内の Duplicate checker は safety\_info 内の情報を参照して受信した情報の重複を確認し、重複を発見したらその受信したデータを削除する。その後、Mobile Server はインターネットアクセスを介して Disaster Server や他地点の Mobile Server にデータを転送する。Disaster Server も同様に Mobile Server から受信したデータの重複を確認する。

## 4. プロトタイプシステム

### 4.1 デバイス概要

図 5 のプロトタイプシステムの構成図ではスマート端末で構成されるマルチホップネットワークで中継したデータが Relay node に到達し、Disaster Server に転送するまでの流れを示している。本研究でノードとなるスマート端末は全て同じ機種を使用しており、現段階では最大 7 台で構成される。Relay node 内は 1 台のスマート端末と Mobile Server で構成されている。Relay node 内のスマート端末は Mobile Server へ転送するためにアクセスポイントへの接続とアドホック通信を切り替えながら動作する。またインターネット接続を利用できるように本構成では NDN をインターネットアクセス網として車載している。Disaster Server と中継ノード内の Mobile Server ではデータの重複を確認して、重複があったデータを削除する。

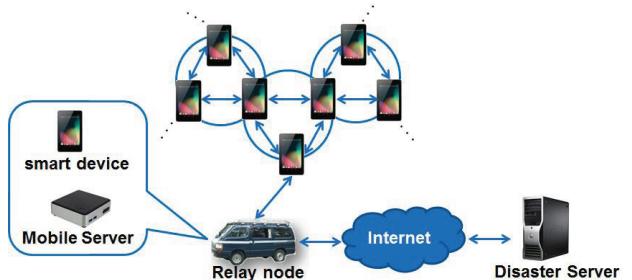


図 5. プロトタイプシステム

## システム詳細

表 1. ハードウェア構成

Smart Devices	Nexus 7 (2013) OS: Android 4.4.X CPU: APQ8064 QuadCore1.5GHz MEM: RAM 2GB Develop: Android SDK NIC: IEEE 802.11n
Mobile Server	OS: Ubuntu 12.04 CPU: Core i3 4010U MEM: RAM 4GB Develop: JDK7
Disaster Server	OS: Ubuntu 12.04 CPU: Core 2 Extreme MEM: RAM 4GB Develop: JDK7

表 1 ではプロトタイプシステムにおけるハードウェアの詳細を示す。本研究のシステムは Android 端末上で起動するものであり、Android SDK を使用して実装している。また、我々は一般的なスマート端末に内蔵される WiFi を使用してアドホック通信を実現している。データを蓄積するストレージは SQLite を用いている。図 6 は現在のスマート端末上で起動するアプリケーションの画面である。本システムの現在状況が status として表示されるようになっており、受信したデータや接続履歴はインテントを使用して閲覧可能である。

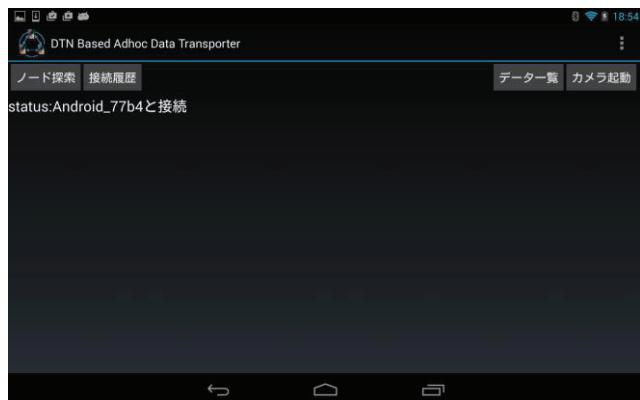


図 6. スマート端末アプリケーション

Mobile Server と Disaster Server は Ubuntu を使用したコンピュータであり、JAVA で実装した TCP ソケットプログラムと MySQL を使用してサーバーとしての機能を実現する。

## 5. 予備実験

本研究において我々はスマート端末間での予備実験を行なった。実験の目的は本研究のアプリケーションシステムでのデータ転送時間と通信可能距離の調査である。我々は実際に Android 端末を使用し、見通しを取ることが可能な屋外にて 2 種類の通信実験を実施した。

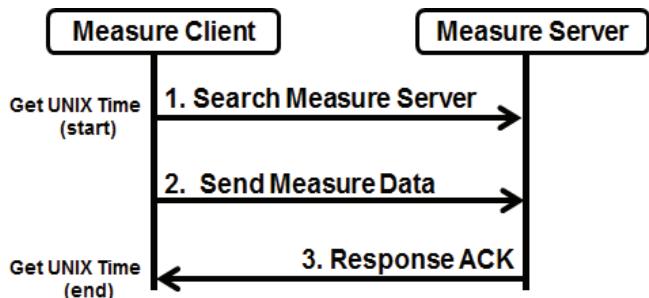


図 7. 測定ツールフロー

我々はまず実験を行なうために前述のシステムとは別にデータ転送時間を測定するツールを実装した。このツールは本研究で実装したアプリケーション仕様に基づいたものである。クライアントがサーバーに 1MB の測定用データを転送して、転送にかかった時間を測定する。プロトタイプシステムで使用する Nexus 7 に内蔵されるカメラで画像データを撮影した際のサイズが約 1MB であるため、今回は測定用のデータを 1MB として実装した。図 7 では測定ツールの仕組みをフローで示す。このフローの動作手順は以下の通りである。

1. 測定クライアント端末で測定サーバー端末の探索し、開始時刻(start)として UNIX 時刻を取得する。
2. 接続の確立後、測定データを送信する。
3. 測定サーバー端末で受信完了後、受信完了を知らせる ACK メッセージを測定クライアント端末に送信する。
4. 測定クライアント端末はこの ACK メッセージで転送が完了したことを確認し、終了時刻(end)として UNIX 時刻を再度取得する。

以上が 1 ホップ分の通信の手順になり、"end-start" で 1 ホップにおける転送時間をミリ秒単位で算出可能となる。今回作成したツールでは転送時間の測定しか行なえないため、今後パケットの到達率やデータの重複率の測定を行なう際には別途の実装が必要になる。このツールを使用して行なった実験とその結果を以下に示す。

### 5.1 通信実験 1

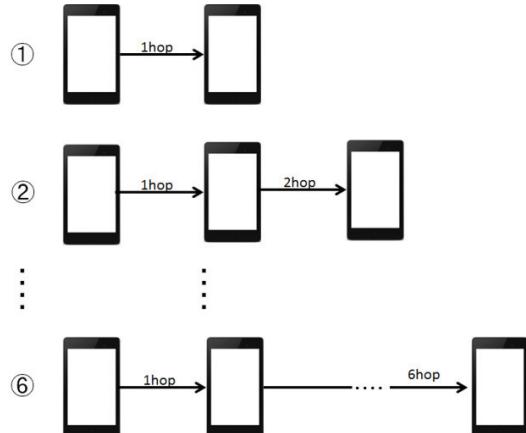


図 8. 通信実験 1 の構成図

表 2. 通信実験 1 の詳細条件

Parameter	Value
Number of hops	1~6
Distance btwn hop	1m
Data size	1MB
Test area	$150 \times 5 (\text{m}^2)$
Speed	0 m/s
Number of test	5

図 8 では通信実験 1 の構成図を示す。通信実験 1 ではホップ数を 1 ホップから 6 ホップまで増加させたときの転送時間の変化を測定した。表 2 では実験の詳細を示しており、我々はホップ間の距離を 1m 設けて、 $150 \times 5 (\text{m}^2)$  の見通しが取れる電波の干渉を受けにくい場所を Test area として選択した。またこの実験では各ノードは移動せず、位置を固定した状態で 5 回ずつ測定した。実験の結果と考察は後述する。

### 5.2 通信実験 2

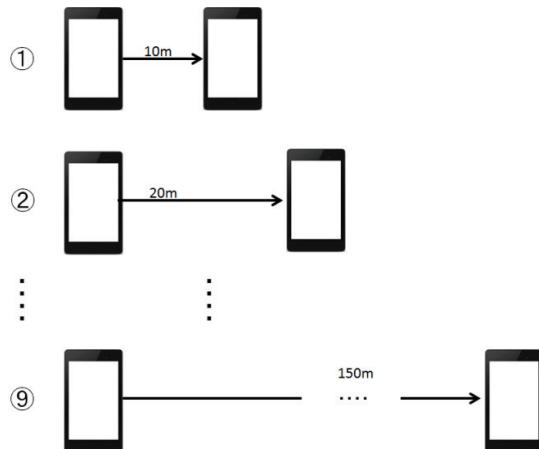


図 9. 通信実験 2 の構成図

表 3. 通信実験 2 の詳細条件

Parameter	Value
Number of hops	1
Distance btwn hop	10~150 (m)
Data size	1MB
Test area	$150 \times 5 (\text{m}^2)$
Speed	0 m/s
Number of test	5

我々は通信実験 2において、1 ホップのノード間の距離を 10m から 150m まで増加させたときの転送時間の変化を測定した。図 9 と表 3 にて通信実験 2 の詳細を示す。本実験ではホップ数を 1 で固定し、10m から 60m までは 10m ごとに測定し、それ以降は 30m ごとに測定を行なった。実験場所は通信実験 1 と同じ場所で行ない、通信実験 2 も各ノードは一切移動せず、位置を固定した状態で 5 回ずつ測定した。

### 5.3 通信実験 1 と通信実験 2 の結果

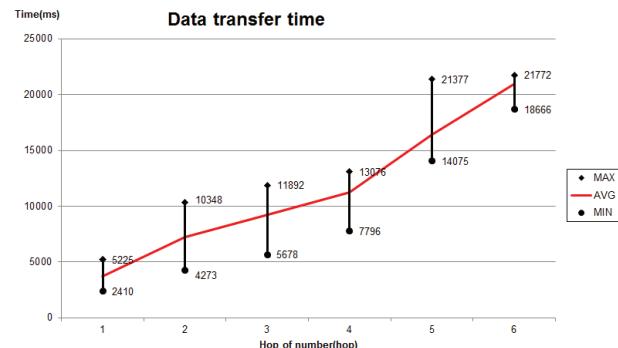


図 10. 通信実験 1 の結果

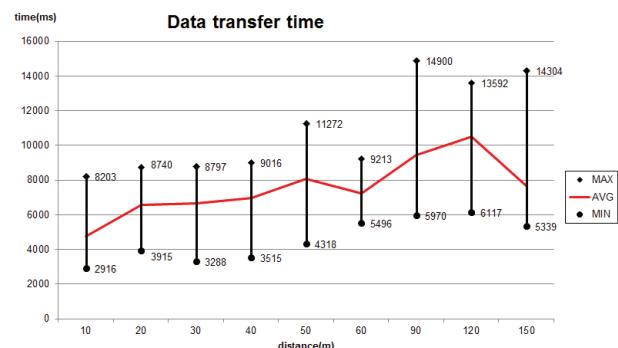


図 11. 通信実験 2 の結果

図 10 と図 11 では通信実験 1 と通信実験 2 の結果をグラフで示す。このグラフでは 5 回ずつ測定した結果の最大時間、平均時間、最小時間を記述した。通信実験 1 ではホップ数の増加に伴って転送時間がほぼ比例する形になった。この結果によって、ホップ数  $n$  のときの転送時間を見積もることが可能となる。その一方で通信実験 2 では 100m 以上離れたときも通信することが可能であり、安定した通信が可能であることが分かった。距離が長くなるほど最大時間と最小時間の間が大きくなり、我々はこの原因を距離が

長くなることでパケットロスが生じやすくなつたことと探索ビーコンの受信とその応答の時間が増加したと思われる。また、今回の実験環境では  $150 \times 5$  ( $m^2$ ) の範囲であったため、通信範囲外となる距離まで測定できなかつた。今後は更に広い Test area での実施とパケットロス率の測定及び探索ビーコンの受信とその応答の時間も考慮して測定する必要がある。

以上の実験を通じて、ノードが移動することを考慮した場合、現在のストレージ内の全てのデータを転送する方式では転送中に相手ノードが通信範囲外に移動してしまうことで、TCP 方式によって分割されたデータを途中まで転送したことが無駄になつてしまふ。そこで我々は次のセクションで今後のアプローチを説明する。

## 6. 今後のアプローチ

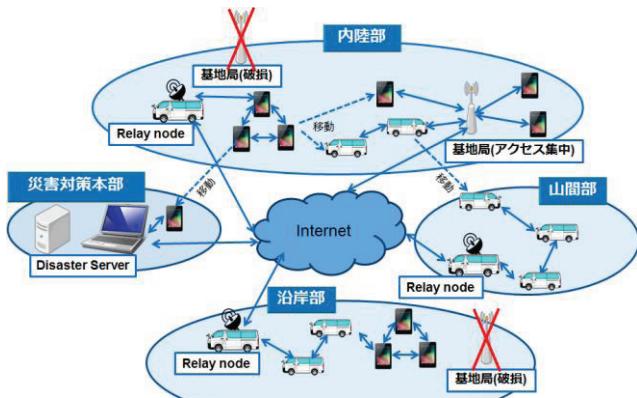


図 12. 今後のシステム構成

本研究の今後のアプローチとして図 12 で新たなシステム構成を示す。この図では被災エリアを内陸部と山間部と沿岸部とする。本構成で内陸部では災害によって携帯通信網の基地局が破損し、残った基地局にアクセスが集中している状態を示し、沿岸部では基地局が破損したことによって一切携帯通信が利用できない状態を示している。またスマート端末であるノードがそれぞれ移動することを前提とし、内陸部と沿岸部では歩いて移動するスマート端末の所持者と自動車に乗つて移動する所持者がいると想定すると、歩行者間の通信、自動車間の通信、歩行者と自動車の通信があると考えられる。沿岸部と内陸部に挟まれる山間部は自動車での移動が多くなるため自動車間の通信が主となる。本研究の今後の方針として様々な移動手段のノードが Relay node や携帯通信基地局の通信範囲内までデータを転送するシステムの実現を目指す。これまでノードの位置が固定した状態での通信は転送中の通信途絶を深く考慮していなかつたが、今後の方針においてノードがそれぞれの速度で移動することを考慮する必要がある。

そこで我々は今後ノード間の接続を確立する際に加速度センサーで取得できる移動速度を含む情報の交換を行な

って、各ノード内のストレージのデータを選択して転送することでこの問題の解決を検討している。例えば自動車間の通信の場合、どちらも移動速度が速いため歩行者間の通信と比較すると必然的に通信時間が短くなると思われる。この限られた時間内にできるだけ多くの情報を転送するために相手ノードの移動速度を知り、ノード内のストレージから小容量なデータを選択する。

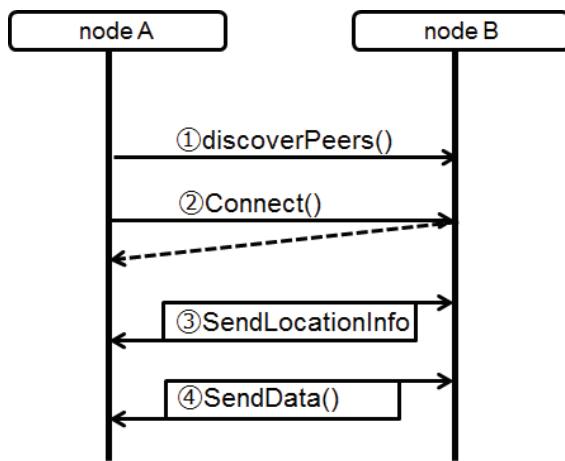


図 13. 加速度情報の交換を含めた通信フロー

図 13 ではノード間の通信フローを示す。ノードを探索して接続を確立する処理まではこれまでと同様であるが、その後ノード間で互いの移動速度を含んだ情報を交換して、より速く移動しているノードの速度に合わせてデータの選択を行なう。例えばノード A の移動速度が 40 キロでノード B の移動速度が 60 キロだった場合、ノード A はノード B が 60 キロで移動していることを考慮して転送するデータを選択する。

現段階では移動速度に基づいた転送可能なデータサイズの指標がないため、今後は自動車間による通信実験を重ねて客観的な指標の作成が必要になる。

## 7. まとめ

本論文では災害時のインフラ破損に有効なスマート端末を活用したマルチネットワークの構築手法を提案した。この提案では DTN を用いることで遅延や途絶を考慮したデータ転送を実現することを目指している。また、実際にスマート端末を使用して通信実験を行なつたことで、ホップ数の増加に伴つて転送時間がおおよそ比例することや 100m 以上離れた状態でもデータ転送が可能であることが明らかになった。現段階では接続履歴のリストと隣接ノードのリストを比較して接続を管理して不要な通信を減らすことのみを考慮したシステムであり、転送中の遅延や途絶を考慮できていない。

今後は通信実験を重ねて様々な移動速度による転送可能な容量を調査して指標を作成し、今後のアプローチに基

づいてシステムを実装して、評価を行なっていく。

## 参考文献

- 1) Goshi Sato, Noriki Uchida, Yoshitaka Shibata “Performance Evaluation of PC Router based Cognitive Wireless Network for Disaster-Resilient WANs,” Advanced Information Networking and Applications Workshops (WAINA 2014), pp. 611-616, May 2014.
- 2) Amin Vahdat, David Becker “Epidemic Routing for Partially-Connected Ad Hoc Networks,” Technical Report CS-2000-06, Department of Computer Science, Duke University, April 2000.
- 3) Chenyu Zheng, Lijun Chen, Douglas Sicker, Xinying Zeng “Hybrid Cellular-MANETs in Practice: A Microblogging System for Smart Devices in Disaster Areas,” Wireless Communications and Mobile Computing Conference (IWCMC 2014), pp. 648-653, Aug 2014
- 4) Masaya Ito, Hiroki Nishiyama, and Nei Kato “A Novel Communication Mode Selection Technique for DTN over MANET Architecture,” International Conference on Computing, Networking and Communications (ICNC 2014), Feb 2014.
- 5) Chengping Tian, Linlin Ci, Bin Cheng, Xuanya Li “A 3D Location-Based Energy Aware Routing Protocol in Delay Tolerant Networks,” Dependable, Autonomic and Secure Computing (DASC 2014), pp. 485-490, Aug 2014.
- 6) Jovilyn Therese B. Fajardo, Keiichi Yasumoto, Naoki Shibata, Weihua Sun, Minoru Ito “DTN-Based Data Aggregation for Timely Information Collection in Disaster Areas,” Wireless and Mobile Computing, Networking and Communications (WiMob 2012), pp. 333-340, Oct 2012.
- 7) Kang Chen and Haiying Shen “SMART: Lightweight Distributed Social Map Based Routing in Delay Tolerant Networks,” Network Protocols (ICNP 2012), pp. 1-10, Oct 2013
- 8) Kang Chen, Student Member, IEEE, Haiying Shen, Senior Member, IEEE, Li Yan “Multicent: A Multifunctional Incentive Scheme Adaptive to Diverse Performance Objectives for DTN Routing,” Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2013), pp. 532-540, June 2013.