

平成 25 年度 特別研究報告書

アドホックネットワークにおける
電波強度の検討

龍谷大学 理工学部 情報メディア学科
T100430 畠 俊光

指導教員 三好 力 教授

内容概要

近年、無線通信技術の発達と端末の小型化によって、多くの人が携帯端末を持つようになった。無線通信可能が端末が普及したために、どこでもアドホックネットワークの構築が容易になっている。アドホックネットワークとは無線基地局等を介さずに端末間のみでネットワークを構築することで、基地局の故障や大規模災害時のインフラが整ってない場合において有効な通信手段となる。

良質なアドホックネットワークの構築のためには均質なネットワーク、端末の電力消費、端末の移動によるネットワークの変更に伴うルーティングプロトコル等の問題をクリアしなければならない。既存のルーティングプロトコルでは、ホップ数と呼ばれる経由する端末数が少ない経路構築が基本となっている。しかし、通信状況が不安定な端末が選ばれてしまう可能性があった。本研究では、各端末間の電波強度を考慮することで安定した経路構築ができるルーティングプロトコルの研究に取り組む。

目次

第1章	はじめに	1
1.1	研究背景	1
1.2	研究の目的	1
第2章	携帯電話について	2
2.1	携帯電話の普及率	2
2.2	携帯電話の無線通信	3
第3章	アドホックネットワーク	4
3.1	アドホックネットワークとは	4
3.2	ルーティングプロトコル	5
3.2.1	Reactive(On-demand)型	5
3.2.2	Proactive(Table-driven)型	5
3.3	AODV	6
3.4	OLSR	8
3.5	問題点	11
第4章	提案手法	12
4.1	ルーティングプロトコルの問題	12
4.2	ルーティングプロトコルの検討	12
第5章	実験と評価	13
5.1	実験目的	13
5.2	実験環境	13
5.2.1	ns2 について	13
5.2.2	ns2 設定	13
5.3	実験内容	14
5.3.1	電波強度によるスループットの変化	14
5.3.2	ホップ数によるスループットの変化	15
5.4	実験結果	16
5.4.1	実験 1	16
5.4.2	実験 2	17
5.4.3	スループットの低下原因	18
5.5	考察	20
第6章	まとめ	21
	謝辞	
	参考文献	
	付録	

第1章 はじめに

1.1 研究背景

無線通信技術を利用した様々なデバイスの登場により、モバイルアクセスの時代の到達が感じられる。通信形態としてはインターネットが主流となり、インターネットモバイルアクセスは、日々の生活に欠かせないものになっている。

無線通信サービスで分かりやすい例は携帯電話サービスだと考えられる。2013年1月の時点で129,254,000件もの人が携帯電話を使用している。現在の携帯端末機能は、電話はもちろんのこと、メールやWeb検索などのパソコンでできるほとんどの機能が搭載されている。ネットワークとしてみた携帯電話は、基地局と携帯端末間の1対1のネットワークであるが、Bluetooth機能を搭載した機種種の登場により、新たなネットワーク利用の可能性が出てきている。

現在までに一般化している無線ネットワークは、そのほとんどが無線基地局などのインフラテクスチャを必要とするものである。これに対し、それらのインフラを必要としないネットワークがアドホックネットワークである。アドホックネットワークは、端末を持った人々が集まることにより自然発生的に形成される。基地局がなくても通信が可能なおことから、災害時など基地局に障害が発生した場合に活躍が期待できる。しかし、基地局を持たないネットワークということで解決しなければならない問題が多く残されている。

1.2 研究の目的

1.1節で述べたように、アドホックネットワークには技術的な課題が残っている。その課題の一つにルーティングプロトコルの問題がある。現在主流となっているルーティングプロトコルは、送信元ノードから宛先ノードまでのルートが複数ある場合に、ホップ数が最小となる経路が選択される。しかし、この手法では最適なルートが選ばれず高いスループットが得られない可能性がある。これは、アドホックネットワークの性質から電波強度が安定していない可能性があり、電波強度が低下するとスループットも低下するため、ホップ数が最小となる経路が高いスループットを得られるとは限らない。

本研究では、電波強度の低下によるスループットへの影響度、ホップ数の増加によるスループットへの影響度の2点を考慮してルーティングプロトコルを作成し、既存手法との比較を行う。

第2章 携帯電話について

2.1 携帯電話の普及率

日本の携帯電話サービスは、1979年に電電会社(現NTT)が持ち運びができる電話サービスを開始したことが始まりである。当時の技術では車のトランクに収容するような大きさであり不便な点が目立った。しかし、1991年にアナログ方式の mova、1993年にデジタル方式の mova が登場したことにより我が国で携帯電話が普及するようになった。

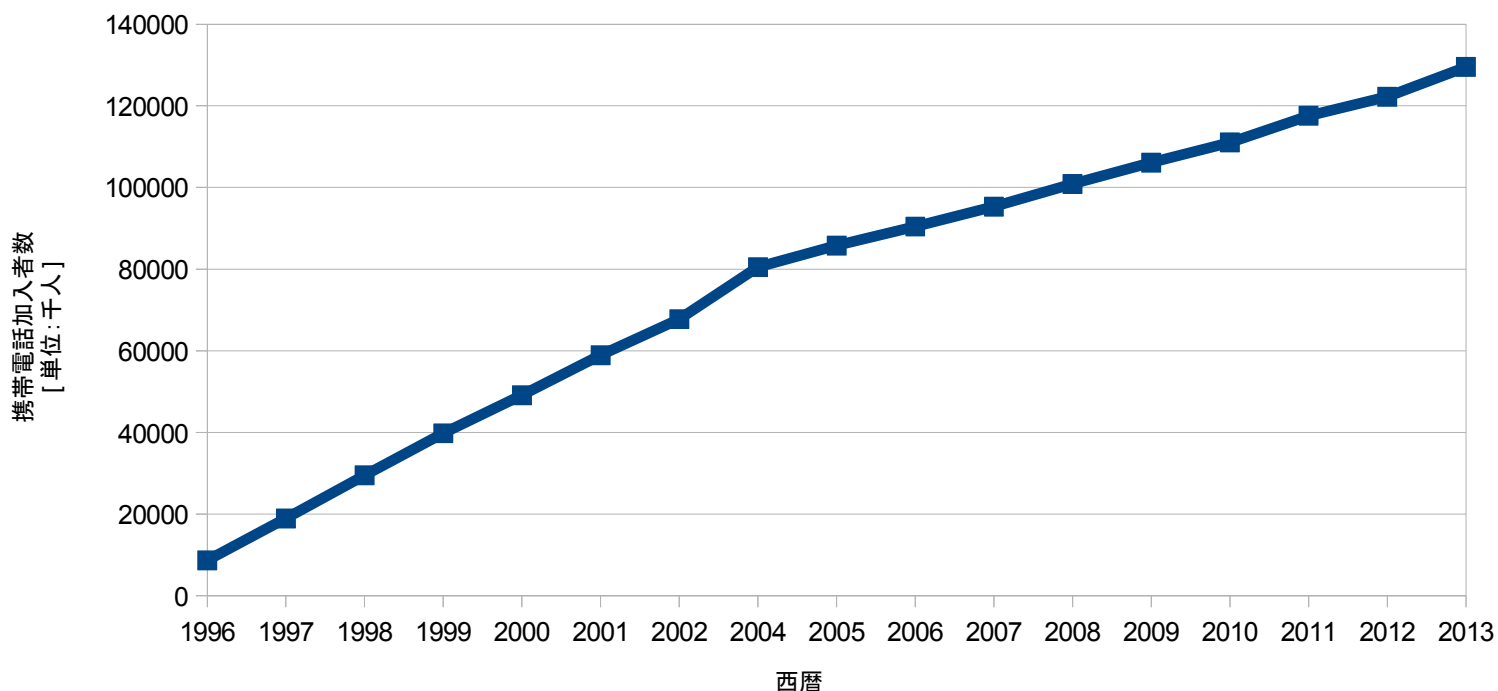


図1:わが国の携帯電話加入者数の推移

図1から分かるように、2012年には日本の総人口を上回る数の携帯加入者数があることが分かる。現在の携帯電話はパソコンでできるほとんどの機能が搭載されている。そのため、人がいれどどこでもアドホックネットワークを構築することが可能になっている。アドホックネットワークの性能を上げることができれば近くの人とは基地局を介さずにより通信しやすくなる。

2.2 携帯電話の無線通信

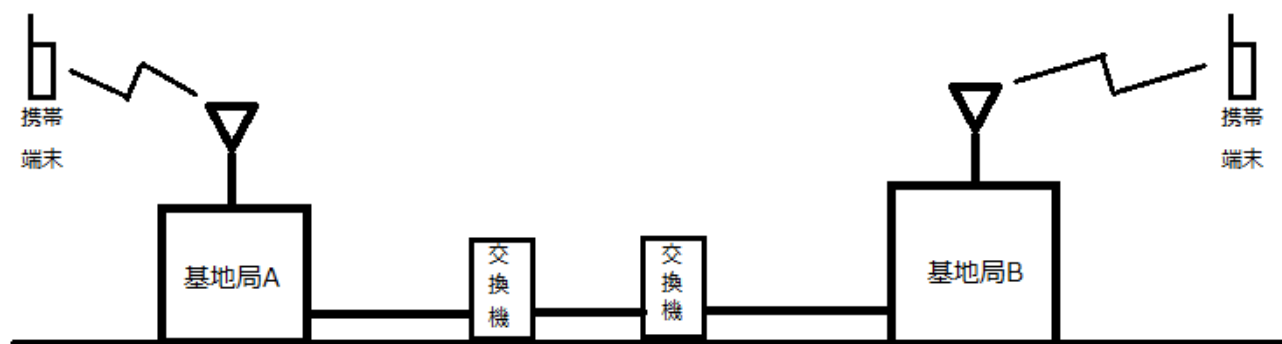


図2: 基地局の通信図

携帯電波の通信では、携帯電話同士が直接通信するのではなく、図2のように携帯電話からの電波はまず最寄の基地局に送信され、交換機や基地局を経由しながら相手の携帯電話に電波を送り届けている。基地局や交換機は有線または、無線・衛星回線につながっている。

災害時・停電時など電源に異常があった場合にそなえて、基地局や交換機は大型バッテリーを備えているが、大規模災害のときや特殊な状況下では基地局が機能せず通信不可になることがある。通信不可になる原因を以下に示す。

- 過度のトラフィック集中による輻輳によって基地局の容量オーバー
- 基地局と交換機を繋ぐ通信回線の切断
- 大型バッテリーは1日程度で空になるため、バッテリー切れの前に充電されない場合

通信回線の物理的な切断、バッテリーの充電切れは大規模災害時に起こりうることであるが、トラフィック集中による通信不可は大規模災害に限ったことではない。

トラフィック集中が起こりうる状況は大勢の人が一斉に同じ基地局に電波を送信することが原因になる。そのため以下のような状況で起こりうる。

- 災害発生時の安否確認の電話
- 年末年始の挨拶メール
- 大規模イベントで多くの人が集まり通信を行う場合

以上のような状況下では、1つの基地局に通信が普段より集中してしまい容量オーバーが起きて通信不可になってしまう場合がある。近年はスマートフォンの普及に伴い、もともとのトラフィックが増大したことで無線LANスポットを各地に設置している。

このような状況下で災害発生時の安否確認の場合、アドホックネットワークを用いることで基地局に負担をかけず、また基地局が障害を起こしていた場合にも対応できると考えられる。

第3章 アドホックネットワーク

3.1 アドホックネットワークとは

エリア内の端末が相互に通信ができるようにすると、基地局が使用できないような場合において、端末どうしの通信が有効な通信手段となる。

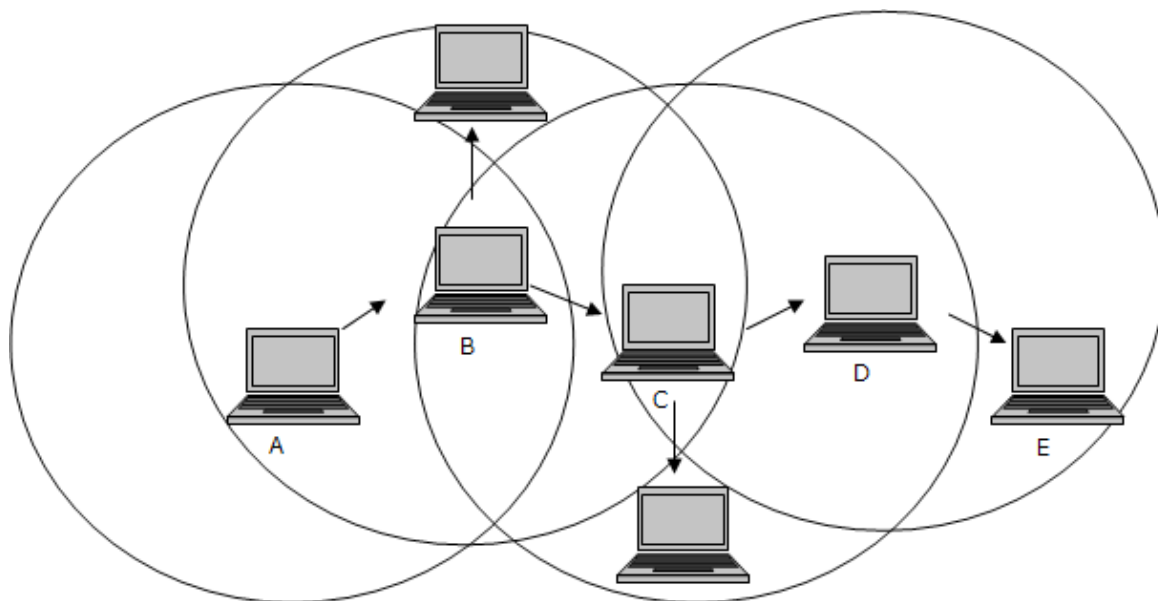


図3:アドホックネットワーク

図3に示すようにアドホックネットワークは各端末が同様の性質をもつ均質なネットワークであり、各端末は、その通信範囲内にあるほかの端末と通信を行う。図3において、左端の端末Aが右端の端末Eと通信したい場合は、周りの端末に通信経路構築を要求し、端末Aから端末Eまでの通信経路が構築される。このように、アドホックネットワークでは通信相手を探すための通信経路手続き、ルーティングプロトコルが重要になってくる。各端末は移動速度に差はあるものの、移動していると考えていくことが必要になる。

アドホックネットワークの特徴をまとめると以下のようなようになる。

- 端末は端末どうしで通信し、マルチホップ通信になる
- 端末が自律的にネットワークを構築する
- 通常の利用では、端末が移動していると考えられる
- 各端末が中継することから、端末の電力消費は大きい
- 均質なネットワークである

アドホックネットワークは端末が次々に情報を中継するマルチホップ機能をもつことから、以下に示すような技術的課題がネットワークを特徴付けるものとなっている。

- ネットワーク構築のため、ルーティングプロトコルを用いる
- マルチホップする端末が故障したときのネットワークの再構築
- アドレス解決

3.2 ルーティングプロトコル

ワイヤレスアドホックネットワークにおいて、送信元端末から宛先端末へ通信する場合、宛先端末が自分の通信エリア内にあることは稀である。したがって、宛先端末の情報となるアドレスを含むパケットを送信し、宛先端末の場所を特定してから送信を開始することになる。この送信元から宛先までの経路をつかさどる機能をルーティングプロトコルとよぶ。アドホックネットワークにおけるルーティングプロトコルは、どの端末を経由してマルチホップし、一番効率よくルートを効率できるか、というネットワークに欠かせない重要な技術となっている。ルーティングプロトコルは、大きく分けて Reactive(On-demand)型と Proactive(Table-driven)の2つになる。それぞれ以下のような特徴を持つ。

3.2.1 Reactive(On-demand)型

Reactive 型プロトコルは、通信要求が発生してから周囲の端末の存在を確かめ、経路表を作成して通信を確立する。このような手続きを踏むため、Reactive 型には二つの利点がある。

一つ目は、経路表が常に最新状態になっていることである。携帯電話などの端末が移動するようなアドホックネットワーク環境では、作成した経路表が数分後には役にたたなくなってしまうことがある。そのため、通信のつど新しく経路表を作成することは都合がよい。二つ目は、通信要求時以外ではパケットを送信しないため、端末の電力消費を低く抑えることができる。これにより、端末の稼働時間が増加し、バッテリー切れによるネットワークから端末の消失を防ぎ、ネットワーク全体が長期間使えるようになる。一方で、通信要求が起こってから経路表を作成するため、通信を開始するまでに遅延が発生してしまう。

代表的な Reactive 型プロトコルは、DSR、AODV、ABR などがある。これらのプロトコルの長所・短所は、以下のとおりである。

- 経路表の更新が不要なので、通信と無関係な制御パケットが流れることはない
- 通信時に経路を決定するので、通信開始までに遅延が発生する
- 端末の移動が頻繁で、通信頻度が比較的低いネットワークに適している

3.2.2 Proactive(Table-driven)型

Proactive 型プロトコルは、通信を行う前にあらかじめ経路表を作成する。このため、端末の移動がない限り常に通信可能状態にあり、通信要求が起こるとただちに通信を開始できるという利点がある。しかし、アドホックネットワークが想定しているモバイルネットワーク環境下では、一度作成した経路表がいつまでも有効であるとは限らず、経路表を常に更新しておく必要がある。このため、端末は定期的にパケットを送信して隣接関係の存在確認が必要となる。端末がこれを行うため、ネットワークへの通信負荷が発生し、また、端末のバッテリー消費が Reactive 型と比べて大きくなるという欠点がある。

代表的な Proactive 型プロトコルには、OLSR、DSDV、TBRPF、FSR などがある。このプロトコルの長所・短所は、以下のとおりである。

- 経路表を逐次更新するため、通信とは無関係な制御パケットが常に流れている
- 経路表を逐次更新しているため、通信開始時の遅延は少ない
- Reactive 型プロトコルに比べて、端末の電力消費が大きい
- 端末の移動が少なく、比較的通信頻度が高いネットワークに適している

3.3 AODV

AODV(Ad hoc On-demand Distance Vector)ルーティングプロトコルは,Reactive 型のルーティングプロトコルであるため,通信要求が発生してからルート構築を行う。

ルートの構築は,制御パケットを用いて次のように行われる,まず送信元端末は,RREQ(Route Request)パケットを通信範囲内に存在する端末に向けてブロードキャストする.送信元端末の通信範囲内に最終目的地である宛先端末が存在しない場合,RREQ パケットを受信した端末は,過去に同じ RREQ パケットを受信していなければ,受信した RREQ パケットをさらにブロードキャストする.これにより,RREQ パケットがネットワーク全体にフラッティングされ,いずれ宛先端末に到達する.

表 1:AODV の RREQ パケットフォーマット

Type	J	R	G	D	U	Reserved	Hop Count
RREQ ID							
Destination IP Address							
Destination Sequence Number							
Originator IP Address							
Originator Sequence Number							

表 1 の各フィールドは以下の情報を持つ。

- Type
RREQ の場合は「1」である.そのほかの場合は,以下のとおり.

Route Request(RREQ)	1
Route Reply(RREP)	2
Route Error(RERR)	3
Route-Reply Ack(RREP-ACK)	4
- J:Join(参加)フラグ.マルチキャストのための予約フィールド
- R:Repair(修復)フラグ.マルチキャストのための予約フィールド
- G:Gratuitous RREP フラグ.送信元ノードが宛先ノードと双方向の通信を実現したい場合はこの G フラグを立てて RREQ を行う
- D:Destination Only フラグ.宛先ノードのみ,この RREQ に返信が必要な場合は D フラグを立てる.
- U:Unkown シーケンス番号フラグ.宛先シーケンス番号が不明なとき,この U フラグを立てる.
- Reserved 「0」として送信.受信側は無視
- Hop Count 送信元の IP アドレスから当該ノードまでのホップする
- RREQ ID 送信元の IP アドレスとともに当該送信ノードが発生した.RREQ の識別 ID 番号
- Dst.IP Addr RREQ が発出された時の宛先 IP アドレス
- Dst.Sqn.No. 送信元から宛先へ向けて発出されたパケットの最新シーケンス番号
- Org.IP Addr RREQ を発出した送信元 IP アドレス
- Org.Sqn.No 送信元から発出された RREQ の最新シーケンス番号

表 2: AODV の RREP パケットフォーマット

Type	R	A	Reserved	Prefix Size	Hop Count
Destination IP Address					
Destination Sequence Number					
Originator IP Address					
Life Time					

表 2 の各フィールドは以下の情報をもつ

- Type RREP の場合は「2」である
- R:Repair(修復)フラグ.マルチキャストのための予約フィールド
- A:ACK(確認応答)フラグ
- Reserved 「0」として送信.受信側は無視
- Prefix Size 「0」でなければ,5bit のプレフィックス値は,同じプレフィックスをもつネクストホップが次の宛先として使用されることを規定している
- Hop Count 送信元から宛先までのホップ数
- Dst.IP Addr 通信経路が提供される差手先 IP アドレス
- Dst.Sqn.No 通信経路に与えられた宛先シーケンス番号
- Org.IP Addr 通信経路が提供される RREQ を発生した送信元 IP アドレス
- Life Time RREP を受信したノードが当該経路が使用できると考えられる時間.単位は ms

RREQ パケットを受信した宛先端末は,RREQ パケットが伝わってきた逆のルートを辿って RREP パケットを送信元端末に向けて送信する.送信元端末に RREP パケットが届いた時点で,双方向のルートが構築され通信が可能となる.また,RREQ パケット,RREP パケットを受信した各端末はパケットに基づいてルーティングテーブルを作成する.もし中継端末が新しいルートを保持していれば,宛先端末の代わりに RREP パケットを返す.

3.4 OLSR

OLSR(Optimized Link State Routing)プロトコルはテーブル駆動型,Proactive 型のルーティングプロトコルであり,常にネットワーク内のほかのノードと位置情報を交換している.したがって OLSR では通信開始前に経路が決定されており,すぐに通信を行うことができる.

OLSR の特徴の一つに,フラッティングの高効率性があげられる.ここで,フラッティングとは一斉送信を繰り返し,制御情報がエリア内のノードに伝達されることを表す.OLSR では,ノード密度が高いほど効率的なフラッティングを行うことができる.

各ノードは,隣接ノードをある一組のセットを選んで MPR とする.OLSR ではこの MPR として選ばれたノードのみがネットワークに拡散する制御トラフィックをフォワードする責任を持ち,通信制御回数を減らすことで,フラッティングを効率的に行う.MPR に選ばれたノードは,リンク状態を宣言する際に,とくに重要な役割を果たす.すべての宛先に対して最短ルートを提供する OLSR の要求は,MPR ノードがそれぞれ MPR セレクターに対してリンク状態を取得することができる.

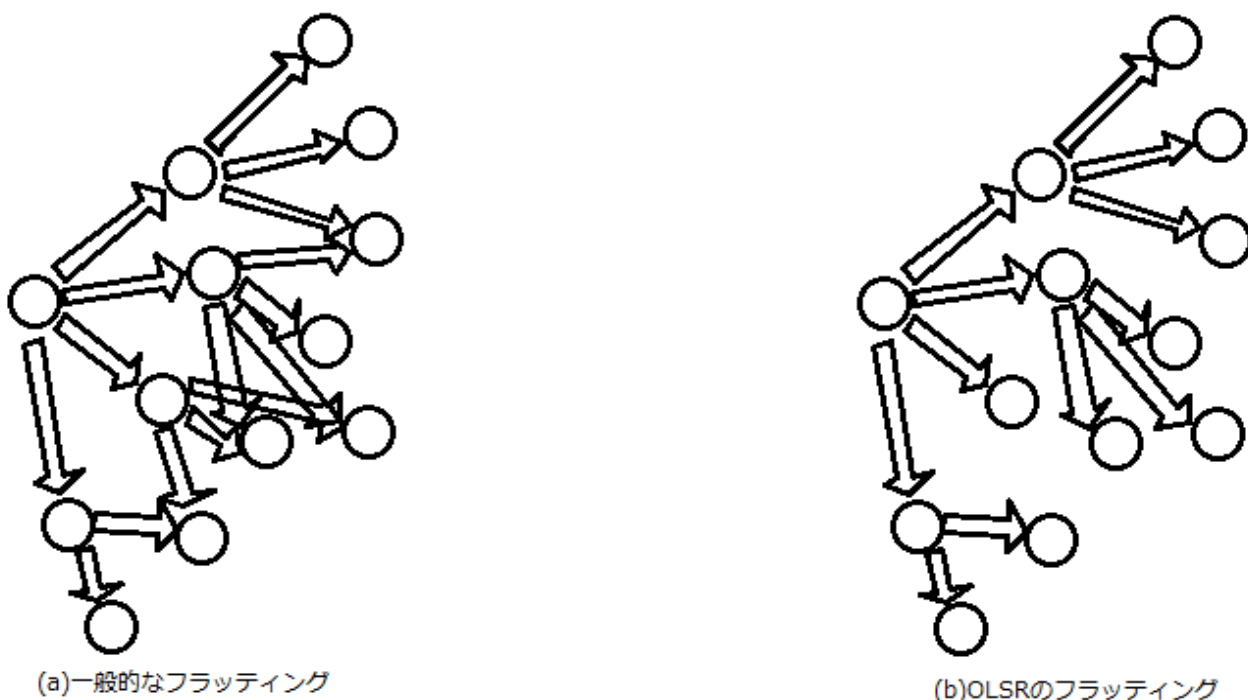


図 4:MPR の概念

図 4 では MPR の概念を示している.図 4 の(a)は,一般的なフラッティングである.左端の送信元ノードから経路に関する制御情報が送信され,各ノードに伝播していつている.しかし,一部のノードには同じ情報が複数到達していることがわかる.OLSR ではこのような重複を避けるために MPR の概念を導入している.図 3 の(b)を見てみると,グループ単位で送信が行われているため重複が避けられていることがわかる.このノードの集まりと MPR 集合と呼ぶ.

MPR 集合を使った効率的な経路探索を行うために,HELLO メッセージと Willingness が重要になっている.HELLO メッセージは,ノード間で各ノードのもつ情報を交換するメッセージであり,隣接ノードの情報,「Willingness」フィールドでノードの再送信への貢献度などの情報交換を行う.

Willingness は再送信に対する当該ノードの積極性を示すものである。以下の 5 通りが規定されており、これらを用いて MPR 集合を決定している。

WILL_NEVER	= 0
WILL_LOW	= 1
WILL_DEFAULT	= 3
WILL_HIGH	= 6
WILL_ALWAYS	= 7

このように、MPR 集合やある種の階層的な扱いを導入することは、フラッティングの際の情報の重複や、ノードの電力消費低減に役立つ。OLSR のいくつかの特徴は以下のとおり。

- 隣接するノードから MPR に選ばれたノードは、その情報は制御メッセージを通じて定期的にアナウンスする。これにより、自分を MPR に指定したノードに通信接続が可能であることを示す。
- ルートの算出にあたり、ネットワーク内のどのノードに対しても、ある特定ノードからのルートを形成するために MPR を用いる
- ノードは 1 ホップ接続する MPR を対象性をもって選び出す。つまり、双方向リンクを構成している。これは、片方向リンクに生じる様々な問題を自動的に解決してくれる

表 3:OLSR のパケットフォーマット

Packet Length		Packet Sequence Number
Message Type	Vtime	Message Size
Originator Address		
Time To Live	Hop Count	Message Sequence Number
MESSAGE		
Message Type	Vtime	Message Size
Originator Address		
Time To Live	Hop Count	Message Sequence Number
MESSAGE		

表 3 の各フィールドは以下の情報をもつ

[Packet Header]

- Packet Length パケット長
- Packet Sequence Number 新しい OLSR のパケット送信されるたびに,1ずつ加算される

[Message Header]

- Message Type 8byte が予約されている.メッセージには HELLO メッセージ,TC メッセージ,MID メッセージ,HNA メッセージの 4 種類がある
- Vtime 受信後,この情報が有効である時間 単位は s
- Message メッセージの大きさ 単位は byte
Message Type フィールドからメッセージ終了までの大きさを表す
- Org. Addr OLSR パケットを送出したノードのアドレス
- TTL メッセージ転送の最大ホップ数
転送される時に必ず 1 減らされる,1 か 0 の場合は転送されない
- Hop Count 生成元からのホップ数
最初は 0 に設定され,転送ごとに 1 ずつ加算される
- Message Seq. No Originator は情報をパケット化して送信するたびに,唯一の一連番号をシーケンス番号として付与する.これらメッセージは UDP を用い,ポート番号 698 を使って送受信される.

3.5 問題点

ほとんどのアドホックネットワークルーティングプロトコルはホップ数と呼ばれる経由する端末が一番少なくなる経路を選択する。しかし、ホップ数が最小となる経路を選択した場合には通信可能範囲ギリギリの端末が選ばれてしまった場合には通信が途切れてしまう危険性がある。また、OLSR方式ではホップ数が最小となる経路が複数発見された場合には、最初に発見したホップ数が最小となる経路が選択されてしまう。

OLSR方式のような Proactive 型のルーティングプロトコルでは、制御メッセージを一定周期毎に送信している。そのため、送信頻度が高くなれば端末の電力消費が大きくなってしまう。しかし、送信頻度が少なければ安定した経路選択ができなくなる恐れがある。

第4章 提案手法

4.1 ルーティングプロトコルの問題

上述したようにルーティングプロトコルは課題となる点がいくつかある。AODV方式、OLSR方式ではフラッティングを行った結果ホップ数が最小となる経路を選択するが、最小となる経路が複数ある場合には最初に発見した経路を選択してしまう。そのため、最小となる経路が複数ある場合には最適な経路が選択されていない可能性がある。最小となる経路の中から通信が安定している端末のみを選び経路構築ができればスループットが向上すると考えられる。

4.2 ルーティングプロトコルの検討

多くのルーティングプロトコルではホップ数が最小となる経路を選択するが、電波強度が不安定な端末はルーティング候補から外すことで、安定した電波強度のみの経路を構築する。また、ホップ数が最小となる経路が複数発見された場合にも、各端末の電波強度を考慮して選択を行えるようにする。

本論文ではデータ伝送速度を一定に、距離を変化させることで電波強度が強い状態、電波強度が弱い状態それぞれのホップ数増加による通信速度を比較する。その結果、安定した通信を行うことのできる最適な電波強度とホップ数の検討を行う。

第5章 実験と評価

5.1 実験目的

通信速度へ影響を与えると考えられる「電波強度」「ホップ数」の2点に関して検証を行い、実際にどれほどの影響があるのか検証を行う。本研究では、ns2(Network Simulator version2)を用いて実験を行う。電波強度とホップ数の関連性を検討することで、より安定した通信を行うことを目的として、2つの実験を行う。

1つ目は電波強度の変化が、どれだけ通信速度へ影響するかを検証する。電波は指向性を持たないものと仮定すると、電波強度は距離に反比例して減少していく。そのことを考慮して、端末から端末への送信時に距離を離していき、通信速度の変化を検証する。

2つ目はホップ数の変化が、どれだけ通信速度へ影響するかを検証する。上述した電波強度の検証より、安定した電波強度下で等間隔に端末を増やしていき、それによる通信速度の変化を検証する。また、同条件で電波強度が安定しない場合でも、検証を行い通信速度の変化を検証する。

5.2 実験環境

5.2.1 ns2 について

ns2は、カリフォルニア大学バークレイ校で開発されたネットワークシミュレータである。ns2では、シミュレータの結果をネットワークアニメータ(nam)によってアニメーションで表示することで、可視化することができる。

5.2.2 ns2 設定

シミュレータの主要なパラメータ部分を記載する。本研究の実験はすべて同一のパラメータで実験を行っている。

- ネットワークインターフェース
ワイヤレス
- アンテナタイプ
無指向性アンテナ
- インターフェースキューの最大値
50
- 無線機器の性能指定
 - 信号補足スレッシュホールド(dB) 10.0
 - キャリア識別スレッシュホールド(W) 1.559e-11
 - 受信電力スレッシュホールド(W) 6.76252e-10
 - 受信伝送速度(bps) 11Mb
 - データ伝送速度(bps) 11Mb
 - 送信電力(W) 0.28183815
 - 使用周波数(Hz) 9.14e+8
 - システムロス係数 1.0

5.3 実験内容

5.3.1 電波強度によるスループットの変化

電波強度の変化が、どれだけ通信速度へ影響するかを検証する。そのため、電波強度が減少することによってスループットがどのように変化するかを確認する。

ノード間の距離が離れることで電波強度が変化する特性を利用する。指向性を持たない等方正アンテナを用いて強さ P の電波を送信した場合、距離 $(D)m$ 離れた地点の電波強度は $P/(4\pi D)^2$ となる。そのため、ノード間の距離が離れるほど電波強度が小さくなると判断できる。

- 実験手順
 - ノード A とノード B の 2 つのノードを 10m 離して通信を開始する
 - ノード A から伝送速度 11Mbps でノード B へ送信
 - ノード B は実験開始から 15 秒後に秒速 0.5m でノード A から遠ざかっていく
 - 実験は 1000 秒間行い、最終ノード間距離は 510m となる
 - ノード B のスループットを計算する
 - 通信のブレを少なくするため、計算されるスループット 1000 個ずつから平均値をとる
 - 本実験を 5 回行い平均値を求める

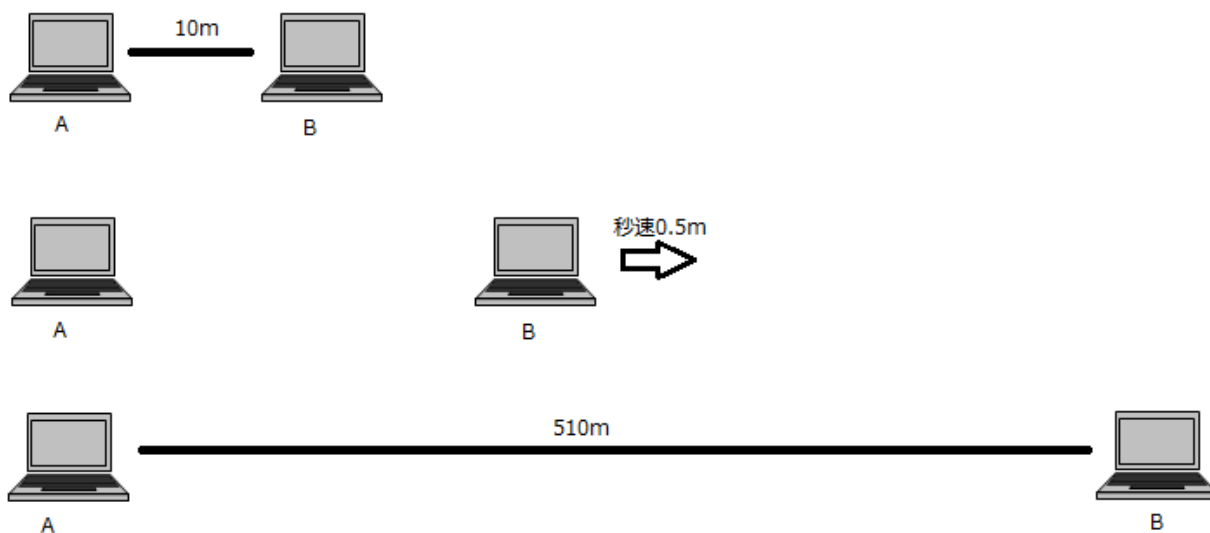


図 5: 実験 1

5.3.2 ホップ数によるスループットの変化

ホップ数の変化が、どれだけ通信速度へ影響するかを検証する。そのため、ホップ数が増加することによってスループットがどのように変化していくかを確認する。

既存ルーティングプロトコルの多くはホップ数が最小となる経路を選択する。本実験ではホップ数がスループットに与える影響を検証し、アドホックネットワークの実用通信範囲についても考える。電波強度が強い状態と、弱い状態でそれぞれ通信を行い、電波強度が変化することでホップ数への影響を検証する。

- 実験手順
 - 送信ノード A から伝送速度 11Mbps で宛先ノード B へ TCP 通信を行う
 - 経由するノードを1つずつ増やしていく
 - ノード間距離はすべて R m とする
 - R は 150m,200m,250m,300m,350m で実験を行う

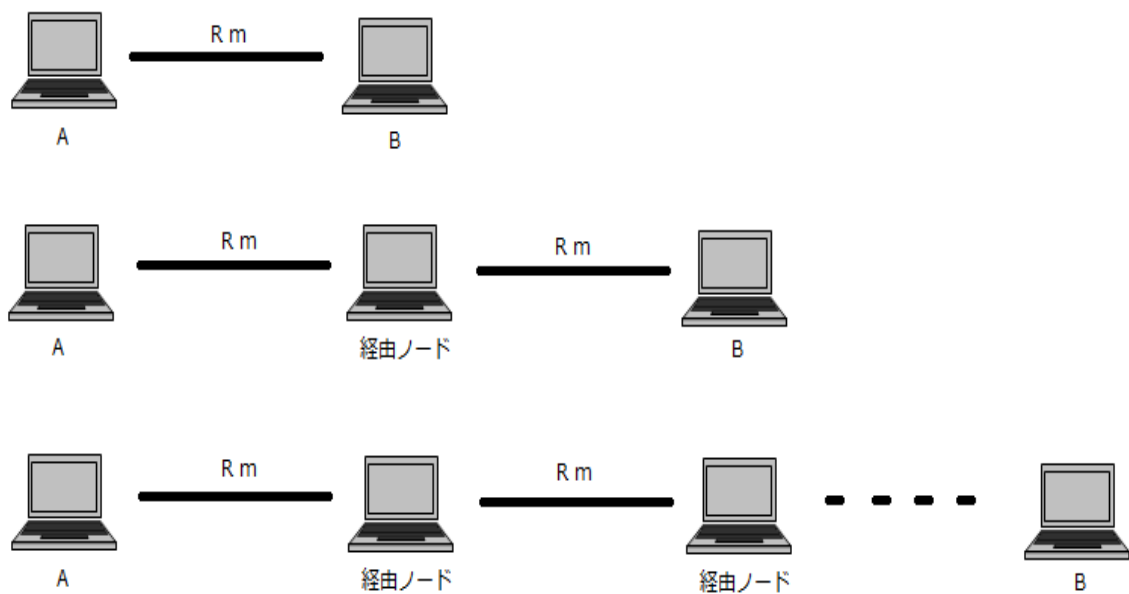


図 6: 実験 2

5.4 実験結果

5.4.1 実験 1

図 7 に電波強度が通信速度へ及ぼす影響を示す。図 7 を見ると距離が離れていくことで通信速度が遅くなっていることが分かる。

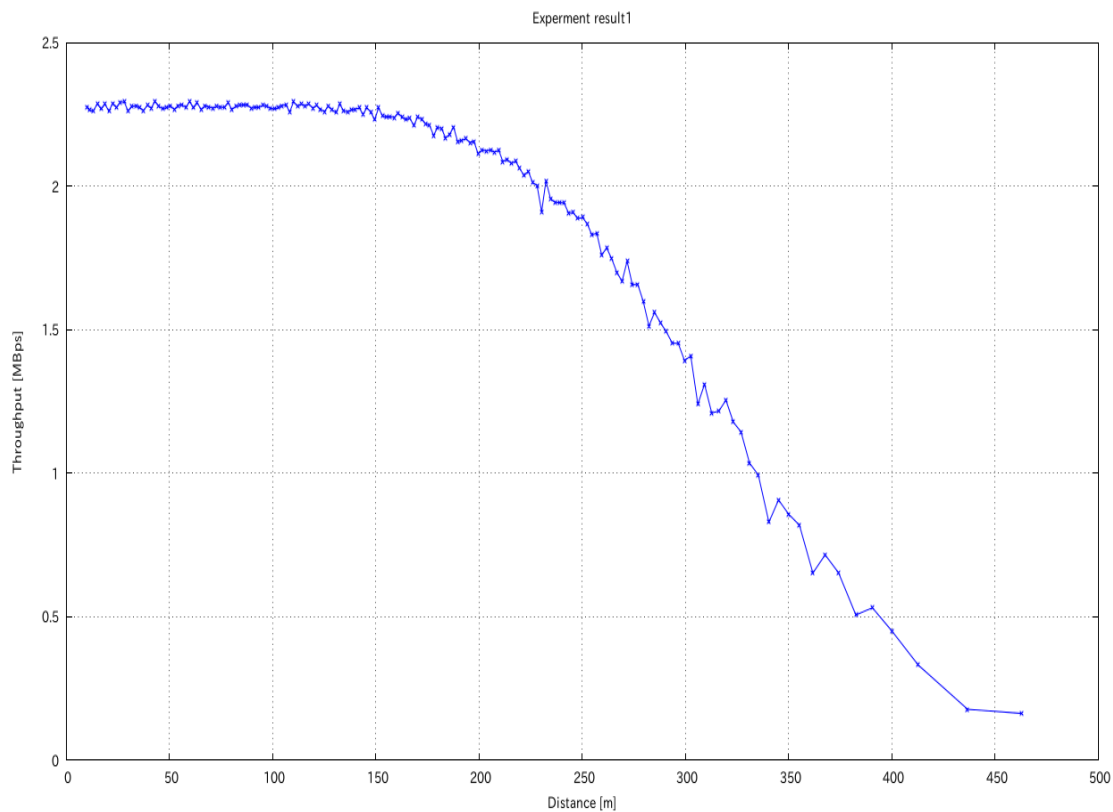


図 7: 電波強度とスループット

図 7 より,電波強度がスループットに与える影響において,通信範囲 150m 以内ならばスループットの低下をほとんどおきていないことが確認できる.10m 地点のスループットを 100%とした場合,150m 地点では 99.72%となっておりスループット低下はほとんどおきていないことが確認できる.200m 地点では 94.59%,250m 地点では 82.97%となり緩やかにスループットが低下している.250m の時点ではスループットは 1.87MBps のため十分なスループットが確保できていると思われる.250m を境目にスループットの減少が大きくなっている.300m 地点では 62.29%,350m 地点では 36.53%とスループットが急激に減少していくことが分かる.400m 地点では通信断裂が多くなっており,通信は実用的ではないと考えられる.

5.4.2 実験2

図8にホップ数の増加による通信速度への影響を示す。図8を見るとホップ数が増加することで通信速度が極端に遅くなっているのが分かる。

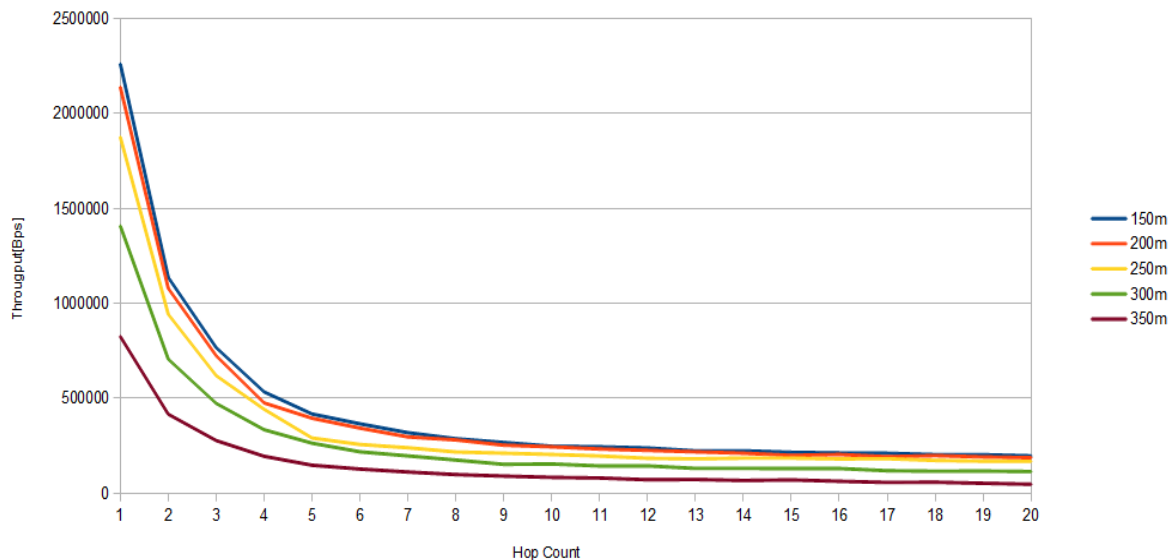


図8:ホップ数とスループット

図8よりホップ数がスループットに与える影響は非常に大きいことが分かる。電波強度が強い状態での通信ほどホップ数が増加することによるスループット減少が大きくと考えられる。全体の減少傾向としては5ホップ目までは減少が大きく、それ以降は減少が緩やかになっている。3000mなど大きく離れた場所と通信する場合には電波強度が強い状態ほどスループットが高くなる結果になった。

図9に図8の結果から得た通信距離とホップ数の関係を示す。図9を見ると頂点となる点が存在し、ホップ数最小がもっとも通信速度が速いわけではないことが分かる。

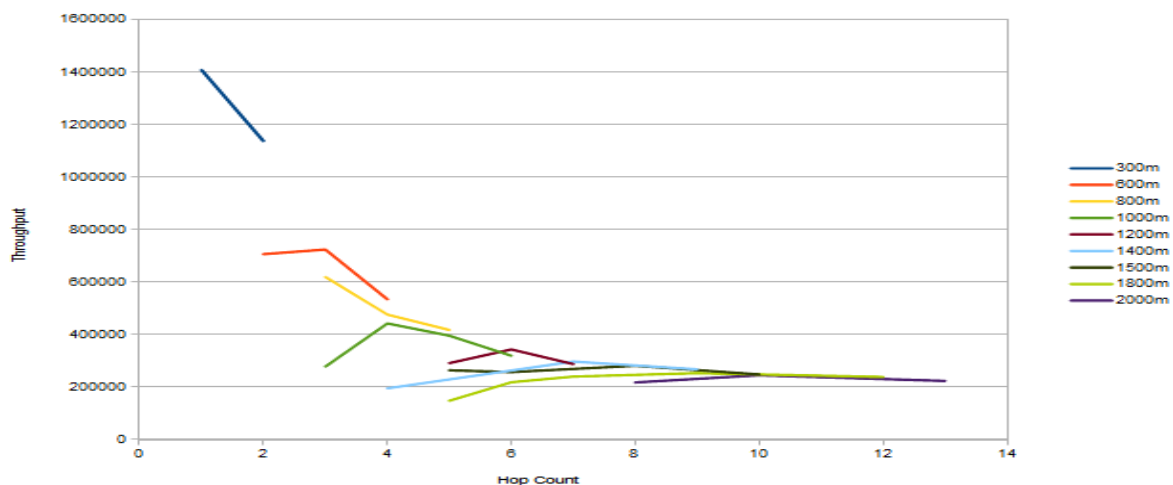


図9:距離毎のスループット

図9より、通信距離が近い場合では変化がないが、距離が離れている場合はグラフが山なりになっておりホップ数が多くても電波強度が強い方がスループットが高いことが確認できる。そのため、求める距離によって最適なホップ数は最小ホップ数でないことが確認できる。

5.4.3 スループット低下原因

図 10 から図 19 には対応するホップ数のときの通信速度とジッタを示す。通信速度が減少するときとジッタが増加するタイミングが同じであることが分かる。シーケンス番号を確認すると、図 8 では 5 ホップ目からスループットは緩やかに減少していくとしか分からないが、実際に届くパケット総数には大きな差があることが分かる。

図 10, 図 11 にはホップ数 1 のときのスループットとジッタを示す。

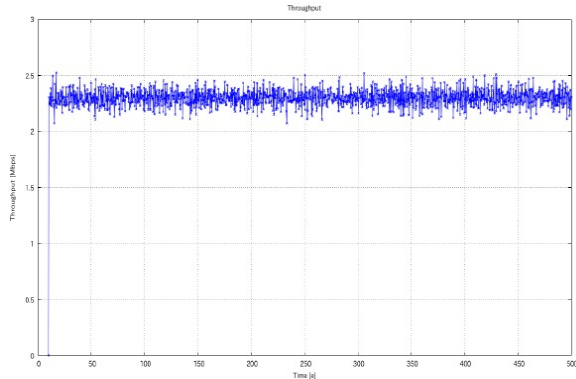


図 10: 1 ホップ目のスループット

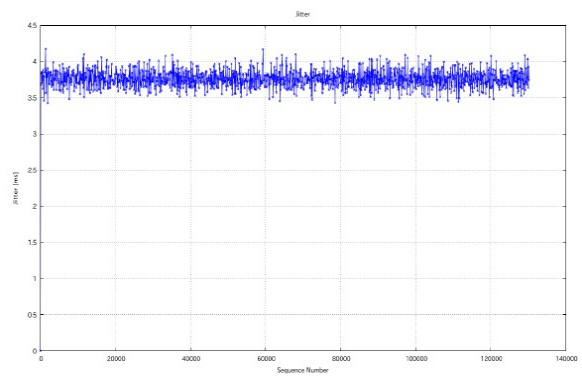


図 11: 1 ホップ目のジッタ

図 12, 図 13 にはホップ数 5 のときのスループットとジッタを示す。

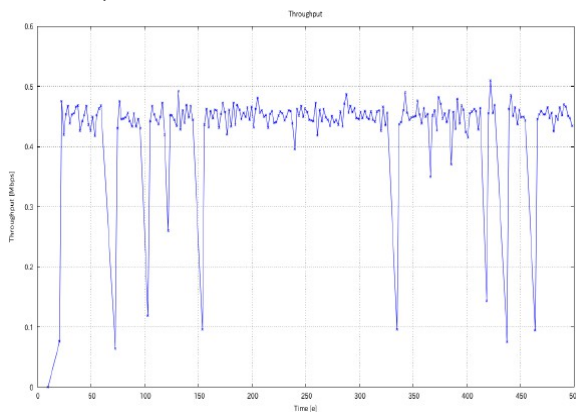


図 12: 5 ホップ目のスループット

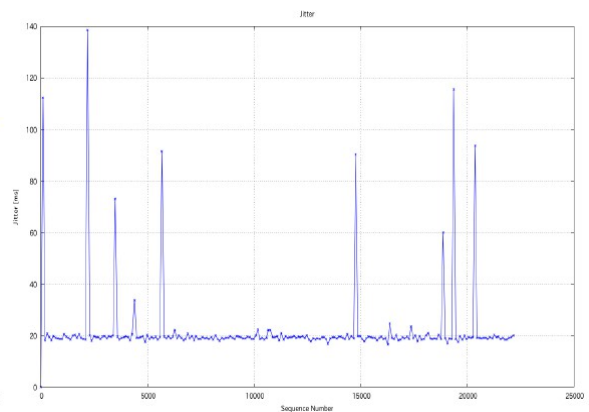


図 13: 5 ホップ目のジッタ

図 14, 図 15 にはホップ数 10 のときのスループットとジッタを示す。

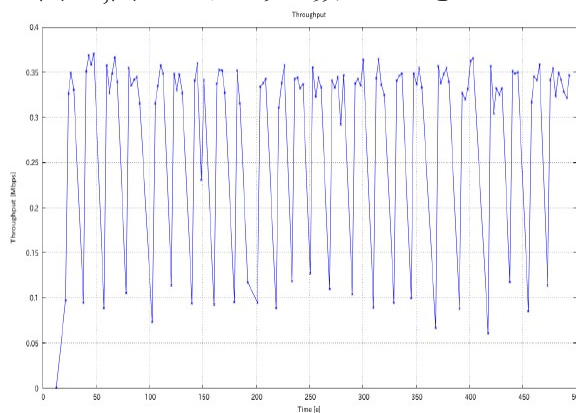


図 14: 10 ホップ目のスループット

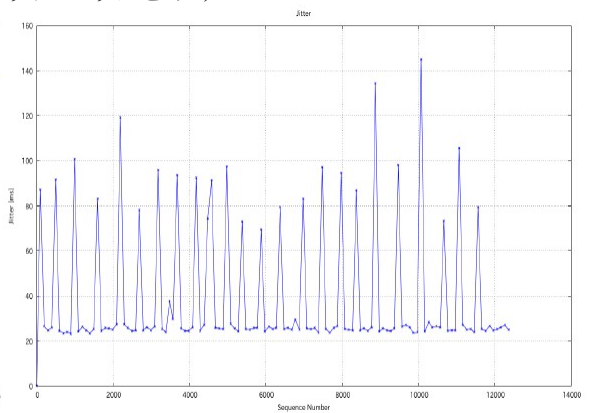


図 15: 10 ホップ目のジッタ

図 16,図 17 にはホップ数 15 のときのスループットとジッタを示す.

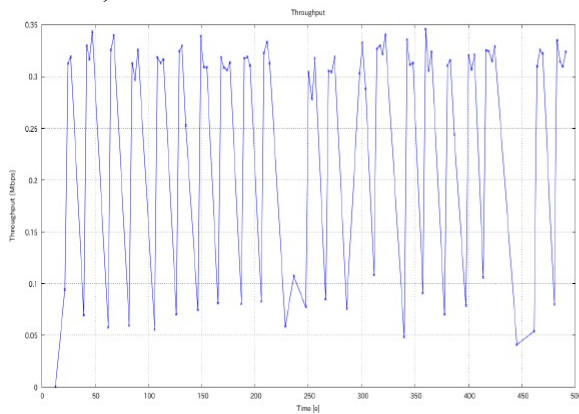


図 16:15 ホップ目のスループット

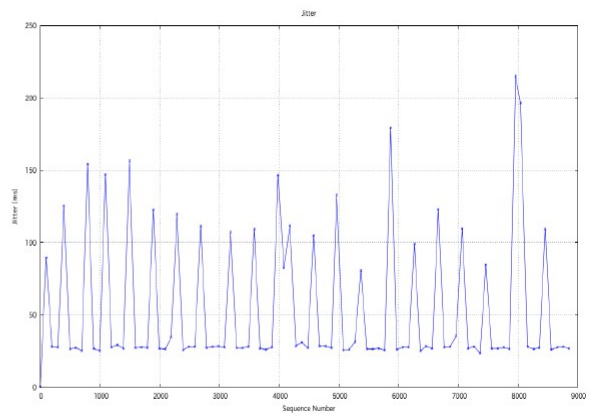


図 17:15 ホップ目のジッタ

図 18,図 19 にはホップ数 20 のときのスループットとジッタを示す.

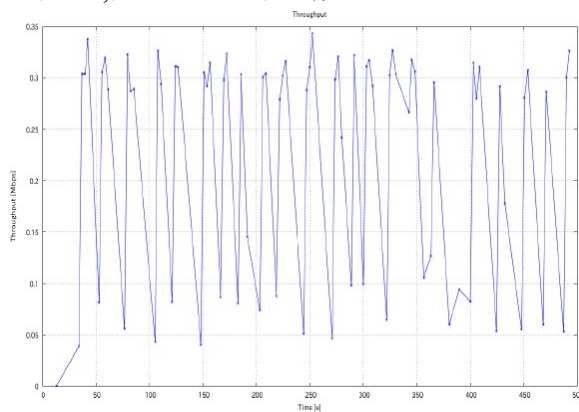


図 18:20 ホップ目のスループット

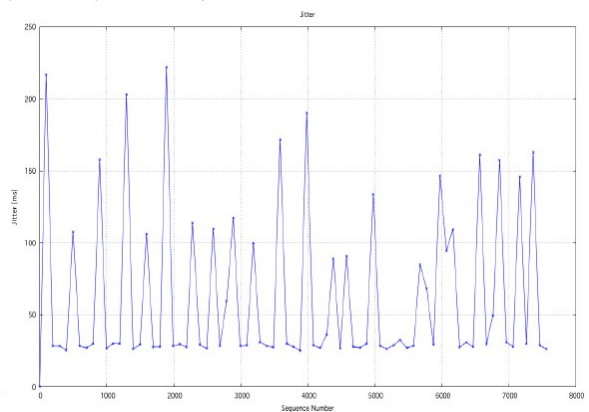


図 19:20 ホップ目のジッタ

150m 毎の各ホップ数の細かいグラフとしては図 10 から図 19 となっている.ホップ数が増加することによってスループットが低下する理由としてジッタの増加が考えられる.ジッタとは受信側の到着時間のばらつきを示したものである.送信側が 20ms 間隔でパケット送信をするが,受信側も 20ms でパケットを受信できたならばジッタは 0ms となる.しかし,ホップ数が多くなると TCP 通信の場合データ確認のやり取りが頻繁に起こってしまいズレが生じてしまう.そのため,ホップ数が増加するとスループットが減少するものと考えられる.また,シーケンス番号を参照すると,ホップ数が増加するごとにシーケンス番号の低下が見られる.グラフ上でのシーケンス番号の値は,送信側が送信したパケットが受信側にどれだけ届いているかを示している.

5.5 考察

図7より、スループットは3G回線下でも2Mbpsを記録しており、250m範囲内ならば十分実用可能範囲であると考えられる。通信範囲が400mを超えると通信断絶が非常に多くなり、まともな平均値を出すことができない結果となった。そのため、450mを超えるとほぼ通信不可状態になると考えられる。実機での電波強度の減衰については電波の電界強度の低下だけでなく、波であるために空気の干渉、人などの電波遮蔽物、自身の電波以外に飛び交う電波等の様々な要因で電波強度はシミュレータの結果よりも減衰すると考えられる。

図8より最低限の通信が可能となる0.2Mbpsならば、3000mまで通信が可能になると考えられる。ホップ数の増加によってスループットの減少は著しく、電波強度の減少具合に比べても大きいことが分かる。そのため、既存のルーティングプロトコルでもホップ数が重視されていると考えられる。

図8を確認すると6ホップ目から20ホップ目まではスループットは緩やかに減少していると思われる。しかし、図15と図19のシーケンス番号を比べると10ホップ目は12600、20ホップ目は7600と大きく減少していることが分かる。シーケンス番号は届いたパケット総数でもあるため、通信断絶は10ホップ目と20ホップ目では大きく違うと考えられる。

図9よりシミュレータの理想環境で山なりの頂点になることが多かった。通信範囲が1000mを超える場合は、ホップ数よりも電波強度が強い方がスループットが高くなる場合がある。ホップ数を算出したのは200m毎にノード間距離をとった場合ものも多くあった。山なりはホップ数が増えると傾斜が小さくなっていくため、アドホックネットワークで遠距離と通信する場合にはホップ数の増減によるスループットの変化は小さいものと考えられる。

第6章 まとめ

本論文ではアドホックネットワークにおける電波強度とホップ数によるスループットの変化を調べた。結果として、スループットに与える影響はホップ数が大きく、電波強度は一定まではスループットにあまり変化をおよぼさないとされる。しかし、与える影響が大きいからといってホップ数のみを重視するよりは、電波強度にも着目することでより高いスループットを得ることができる事が実験により判明した。

今後の課題として、ホップ数増加に伴うスループットの変化では、常にノード間距離は一定であったが、実際のアドホックネットワークではそのようなことは起こりえないため、ノード間距離が一定ではない場合も考慮する必要があると考えられる。

謝辞

本研究全般に関しまして,日々ご指導頂いた三好力教授に感謝申し上げます.また,研究を進める中で,多くの討論や助言を頂きました同三好研究室の皆様や,友人の皆様に感謝申し上げます.

参考文献

- [1] 水野 秀樹著,NS2 によるネットワークシミュレーション入門 有線からワイヤレスアドホックネットワークまで
- [2] 銭 飛著,NS2 によるネットワークシミュレーション
- [3] The Network Simulator
<http://www.isi.edu/nsnam/ns/ns-build.html>
- [4] NS Tutorial : Mobile and Wireless Simulation
http://netlab.ce.nihon-u.ac.jp/ns_manual/ns-2_Adhoc_Tutorial/ns-top_SJIS.html
- [5] NS-2 による無線ネットワークのシミュレート
http://www.net.c.dendai.ac.jp/~koki/_index.html
- [6] NS Tutorial
<http://www.isi.edu/nsnam/ns/tutorial/index.html>
- [7] UM-OLSR
<http://masimum.inf.um.es/fjrm/development/um-olsr/>
- [8] ジッタ評価
http://netlab.ce.nihon-u.ac.jp/ns_manual/page014.html

付録

```

set val(chan)
Channel/WirelessChannel ;# channel type
set val(prop)
Propagation/TwoRayGround ;# radio-
propagation model
set val(netif)
Phy/WirelessPhy ;# network
interface type

set val(mac)
Mac/802_11 ;# MAC type
set val(ifq)
Queue/DropTail/PriQueue ;# interface
queue type

set val(ll) LL ;# link
layer type
set val(ant) Antenna/OmniAntenna
;# antenna model
set val(ifqlen) 50 ;#
max packet in ifq

set val(nn) 21 ;#
number of mobilenodes
set val(rp) OLSR ;#
routing protocol

set val(x) 4100
set val(y) 600
set val(stop) 500.0
set val(tf) result.tr
set val(nf) result.nam

#
=====
=====
=====
# Main Program
#
=====
=====
=====
#
# Initialize Global Variables
#
set ns [new Simulator]
set tf [open $val(tf) w]
set nf [open $val(nf) w]

$ns trace-all $tf
$ns use-newtrace

$ns namtrace-all-wireless $nf $val(x) $val(y)

#
set topo [new Topography]

set prop [new Propagation/Shadowing]

$prop set pathlossExp_ 2.0
$prop set std_db_ 4.0
$prop set dist0_ 1.0
$prop seed <seed-type> 0
$ns node-config -propInstance $prop

$stopo load_flatgrid $val(x) $val(y)

#
# God
#
create-god $val(nn)

set chan_1_ [new $val(chan)]

# configure node

$ns node-config -adhocRouting $val(rp) \
  -llType $val(ll) \
  -macType $val(mac) \
  -ifqType $val(ifq) \
  -ifqLen $val(ifqlen) \
  -antType $val(ant) \
  -propType $val(prop) \
  -phyType $val(netif) \
  -channel $chan_1_ \
  #-channelType $val(chan)\
  -topoInstance $topo \
  -agentTrace ON \

```

```

-routerTrace ON \
-macTrace OFF \
-movementTrace OFF
#for {set i 0} {$i < $val(nm)} {incr i} {
# set node_($i) [$ns_node]
# $node_($i) random-motion 0
    ;# disable random motion
#}

Phy/WirelessPhy set CPTthresh_ 10.0
Phy/WirelessPhy set CSTthresh_ 1.559e-11
#550m
Phy/WirelessPhy set RXThresh_ 6.76252e-10
#250m
Phy/WirelessPhy set bandwidth_ 11Mb
Phy/WirelessPhy set Pt_ 0.281838
#Phy/WirelessPhy set Pt_ 0.08
Mac/802_11 set dataRate_ 11Mb
Mac/802_11 set basicRate_ 1Mb
Phy/WirelessPhy set freq_ 9.14e+8
Phy/WirelessPhy set L_ 1.0
Phy/WirelessPhy set debug_ false

for {set i 0} {$i < $val(nm)} {incr i} {
    set node_($i) [$ns_node]
    $node_($i) random-motion 0
}

#
# (x y) z=0
#
$node_(0) set X_ 50.0
$node_(0) set Y_ 200.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 200.0
$node_(1) set Y_ 200.0
$node_(1) set Z_ 0.0

$node_(2) set X_ 350.0
$node_(2) set Y_ 200.0
$node_(2) set Z_ 0.0

$node_(3) set X_ 500.0
$node_(3) set Y_ 200.0
$node_(3) set Z_ 0.0

$node_(4) set X_ 650.0
$node_(4) set Y_ 200.0
$node_(4) set Z_ 0.0

$node_(5) set X_ 800.0
$node_(5) set Y_ 200.0
$node_(5) set Z_ 0.0

$node_(6) set X_ 950.0
$node_(6) set Y_ 200.0
$node_(6) set Z_ 0.0

$node_(7) set X_ 1100.0
$node_(7) set Y_ 200.0
$node_(7) set Z_ 0.0

$node_(8) set X_ 1250.0
$node_(8) set Y_ 200.0
$node_(8) set Z_ 0.0

$node_(9) set X_ 1400.0
$node_(9) set Y_ 200.0
$node_(9) set Z_ 0.0

$node_(10) set X_ 1550.0
$node_(10) set Y_ 200.0
$node_(10) set Z_ 0.0

$node_(11) set X_ 1700.0
$node_(11) set Y_ 200.0
$node_(11) set Z_ 0.0

$node_(12) set X_ 1850.0
$node_(12) set Y_ 200.0
$node_(12) set Z_ 0.0

$node_(13) set X_ 2000.0
$node_(13) set Y_ 200.0
$node_(13) set Z_ 0.0

$node_(14) set X_ 2150.0
$node_(14) set Y_ 200.0
$node_(14) set Z_ 0.0

$node_(15) set X_ 2300.0
$node_(15) set Y_ 200.0
$node_(15) set Z_ 0.0

$node_(16) set X_ 2450.0
$node_(16) set Y_ 200.0
$node_(16) set Z_ 0.0

$node_(17) set X_ 2600.0
$node_(17) set Y_ 200.0
$node_(17) set Z_ 0.0

```

```

$node_(18) set X_ 2750.0
$node_(18) set Y_ 200.0
$node_(18) set Z_ 0.0

$node_(19) set X_ 2900.0
$node_(19) set Y_ 200.0
$node_(19) set Z_ 0.0

$node_(20) set X_ 3050.0
$node_(20) set Y_ 200.0
$node_(20) set Z_ 0.0

#Sns at 5.0 "$node_(0) setdest 350.0 200.0 0.1"

set tcp [new Agent/TCP]
$tcp set class_ 2
set sink [new Agent/TCPSink]

$ns attach-agent $node_(0) $tcp
$ns attach-agent $node_(20) $sink
$ns connect $tcp $sink

set ftp [new Application/FTP]
$ftp attach-agent $tcp

$ns at 10.0 "$ftp start"

#puts "Loading connection pattern..."
#source $val(cp)

# (NAM )
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns initial_node_pos $node_($i) 25
}

#
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns at $val(stop) "$node_($i) reset"
}

$ns at $val(stop) "stop"
$ns at $val(stop)+0.01 "puts \"NS EXITING...\"
; $ns halt"

proc stop {} {
# global ns tracefd
global ns tf nf
$ns flush-trace
close $tf
close $nf
# exec nam study3.nam &
# exit 0
}

puts "Starting Simulation..."
$ns run

```