

モバイルアドホックネットワークにおけるモバイルエージェントの
利用に関する研究

神奈川工科大学大学院 工学研究科

情報工学専攻 博士学位論文

藤川 丈自

あらまし

現在の無線通信技術は基地局やアクセスポイントなどの固定された通信基盤を必要とするものが多く、基盤が存在しない場合や、大規模な災害などにより物理的な通信網途絶が発生した場合、ネットワークの稼働率の大幅な低下が想定される。そのような際、固定の通信基盤を用いず、ネットワークを構成することができるアドホックネットワークの利用が有効とされている。アドホックネットワークでは、電波が届かず直接通信できないノード間の通信を、途中に存在するノードが中継ノードとして機能することで実現する。

アドホックネットワークの技術課題の一つとしてルーチングが挙げられ、多くのプロトコルが提案されている。ルーチングの性能を表す指標としてスループット、パケット到達率、制御パケット量等が挙げられる。本論文ではこれらの性能を向上させることを目的とし、分散処理技術の一つであるモバイルエージェント (MA) を利用したルーチング手法を提案する。本論文で提案する方式ではノード情報管理と経路構築の機能を持った MA のプログラムを特定のノードが実行することで、ルーチングを実現する。さらに、この技術を応用してコンテンツ指向型ネットワークにおける MA を用いたコンテンツ取得手法を提案する。コンテンツ指向型ネットワークとは、インターネットなどで広く用いられている IP に基づくホスト指向型ネットワークに代わり、送受信データに着目して設計された次世代ネットワークのアーキテクチャのことである。本論文では目的に合わせた 3 種類のルーチング手法と 1 種類のコンテンツ取得手法を提案して評価する。ルーチング手法としては、第一に MA に全ノードの位置情報を管理させる方式、第二に第一の方式を基本としてスループット向上を目的にマルチパスを作成するように経路計算を改良した方式、第三に屋内等の GPS が使用できない環境を考慮し、位置情報を利用する代わりに隣接ノード情報を利用した方式を提案する。コンテンツ取得手法としてはネットワーク負荷の削減を目的に MA に全ノードの位置情報とコンテンツリストを管理させる方式を提案する。

第 1 章は緒論であり、研究の背景や目的について簡単に述べる。第 2 章では MA に全ノードの位置情報を管理させる方式について述べる。現在主流のリアクティブルーチングプロトコルの多くは、経路構築時に制御パケットのフラッディングを必要としており、ネットワーク負荷の増加が課題となっている。これに対して、経路構築時のフラッディングをなくし、制御パケット量を削減することを目的に MA が各ノードの位置情報を一元的に管

理し、その情報から経路を作成する方式を提案した。計算機シミュレーションによる性能評価の結果より、既存の方式に比べ制御パケットを大幅に削減するとともにパケット到達率が向上することを示す。

第3章ではスループットの向上を目的とした MA 利用型マルチパスルーチングを提案する。これは2章の方式を基本として経路計算を改良しマルチパスを算出させるようにした方式である。この方式は位置情報を元に通信可能距離を考慮して経路間の干渉を抑えたマルチパスを作成することで、よりスループットの高い経路を作成することが可能である。計算機シミュレーションによる性能評価の結果より、既存の方式に比べスループットが大きく向上することを示す。

第4章では隣接ノード情報を利用した MA ルーチングを提案する。2章、3章で提案した方式では位置情報に基づき経路を計算するため、各ノードが自身の位置情報を取得する必要がある。しかし、位置情報の取得には一般的に GPS を利用することが想定されているため、屋内での利用が困難である。また、GPS の測定誤差によるプロトコルの性能低下等の問題も生じる。そこで、1章の方式を基本に位置情報の代わりに隣接ノード情報を用いて同様なルーチングを実現する方式を提案する。また、計算機シミュレーションによる性能評価の結果より、パケット到達率において位置情報を利用する方式よりも少し劣るものの AODV, OLSR よりは高い性能であることを示す。また、平均経路構築遅延と消費電力においても提案方式が他の方式に比べ同等かそれ以上の性能であることを示す。これらの結果から総合的に判断して、本方式が性能を維持しつつ位置情報を利用することによるデメリットや制限を除くことができるということを示す。

第5章ではコンテンツ指向型ネットワークにおける MA を用いたコンテンツ取得手法を提案する。既存のコンテンツ取得手法ではコンテンツを要求するためのメッセージの送信にフラッディングを用いる。そのため、要求したコンテンツを複数のノードがキャッシュしていた場合、要求パケットを受信した複数のノードがコンテンツを送信するため重複したコンテンツ送信が発生するという問題がある。これはネットワークの負荷を増加させる。これに対して、1章で提案する MA を用いたルーチング方式をコンテンツ指向型 MANET に応用することでネットワーク負荷を削減する手法を提案する。また、計算機シミュレーションによる性能評価の結果より、既存の方式に比べコンテンツ取得率、コンテ

ンツ伝送遅延が向上することを示す。第6章は結論であり、本研究で得られた成果を総括する。

目次

1. 緒論	1
1.1 研究の背景.....	1
1.2 アドホックネットワーク	3
1.3 ルーティングプロトコル.....	4
1.4 コンテンツ指向型ネットワーク	6
1.5 Mobile Agent (MA)	7
1.6 本研究の位置付け.....	8
参考文献	10
2. モバイルエージェントを用いた位置情報利用型ルーティング	12
2.1 はじめに.....	12
2.2 位置情報利用型ルーティングプロトコル.....	14
2.2.1 フォワーディングプロトコル.....	15
2.2.2 位置情報取得プロトコル.....	18
2.3 提案方式	23
2.3.1 提案方式におけるモバイルエージェントの動作.....	23
2.3.2 位置情報更新.....	25
2.3.3 経路情報取得.....	27
2.3.4 経路の算出.....	27
2.3.5 経路構築.....	29
2.4 シミュレーションによる性能評価.....	30
2.5 シミュレーション結果・考察.....	32
2.6 2章のまとめ.....	38
参考文献	39
3. モバイルエージェントを用いた経路干渉考慮型マルチパスルーティング	40
3.1 はじめに	40
3.2 マルチパスルーティングプロトコル.....	42

3.2.1	EECA.....	42
3.2.2	経路干渉考慮型マルチパスルーチング.....	43
3.3	提案方式.....	45
3.3.1	経路算出.....	45
3.3.2	データフォワーディング.....	47
3.3.3	効率的なパスの割り当て.....	47
3.3.4	現在位置情報の推測.....	48
3.4	シミュレーションによる性能評価.....	49
3.5	シミュレーション結果・考察.....	52
3.6	3章のまとめ.....	61
	参考文献.....	62
4.	モバイルエージェントを用いた隣接ノード情報利用型ルーチング.....	63
4.1	はじめに.....	63
4.2	提案方式.....	65
4.2.1	位置情報を利用しない Mobile Agent (MA)の運用.....	65
4.2.2	隣接ノード情報更新.....	66
4.2.3	経路情報取得.....	67
4.3	シミュレーションによる性能評価.....	68
4.4	シミュレーション結果・考察.....	70
4.5	4章のまとめ.....	78
	参考文献.....	79
5.	コンテンツ指向型 MANET におけるモバイルエージェントを利用したコンテンツ取得手法.....	80
5.1	はじめに.....	80
5.2	REMIF.....	82
5.3	提案方式.....	83
5.3.1	位置・コンテンツ情報更新.....	83
5.3.2	コンテンツ要求.....	83
5.3.3	コンテンツ送信・キャッシング.....	85

5.4 シミュレーションによる性能評価.....	86
5.5 シミュレーション結果と考察.....	89
5.6 5章のまとめ.....	94
参考文献	95
6. 結論	96

第 1 章

1. 緒論

1.1 研究の背景

近年、スマートフォンの普及やハードウェアの高性能化により、無線通信の技術が飛躍的に発展している。しかし、現在の無線通信技術は基地局やアクセスポイントなどの通信インフラストラクチャを必要とするものが多く、インフラストラクチャがない場合、ネットワークに接続することができないという欠点がある。そこで、インフラストラクチャを用いず、ネットワークを構成することができるアドホックネットワークが注目されている。アドホックネットワークでは、あて先端末に電波が直接届かず通信できない場合、途中に存在する端末がパケットを中継することでデータの送受信を行う。これをマルチホップ通信という。アドホックネットワークではこのマルチホップ通信を用いることで、インフラストラクチャを使わず端末を持ち寄るだけで自律的にネットワークを構成することが可能である。

アドホックネットワークは、端末の移動を考慮しない単純なアドホックネットワーク、移動端末により構成されるモバイルアドホックネットワーク (Mobile Ad Hoc Network : MANET)、自動車の特性や動きに特化した車車間 MANET (Vehicular ad hoc network : VANET)、センサからの情報を収集することを目的とした無線センサネットワーク (Wireless sensor network) などに分類される。そして、それぞれの特徴を生かした、運転支援、防災・防犯・災害時対策、環境・エネルギー対策、高齢者介護・医療、農業等の分野での応用が期待されている。

現在では、無線センサネットワークによる環境モニタリング等については実用化が始まっている。また、スマートフォンやゲーム機、PC などでもシングルホップ(1 対向)による通信に限られる場合が多いが、アドホック通信が使用されている。なお、本論文では端末の移動を考慮した MANET を研究対象とする。

アドホックネットワークに関する研究は広範囲で行われており、有線ネットワークやインフラストラクチャを使用した無線 LAN 環境では存在しない課題を解決するために様々な検討がなされている。特にルーチングが大きな課題となっており研究が行われている。アドホックネットワークにおいて、通信範囲外の端末と通信する際、マルチホップ通信による中継を行うが、このような機能をルーチングと呼ぶ。有線ネットワークでいえばルータの機能である。基本的にアドホックネットワークで動作する端末はルータの機能を備えている。この端末を本論文ではノードと呼ぶこととする。多くのアドホックネットワークではインフラストラクチャを利用せず、基本的には各ノードの機能が対等であるため、ノード自身が自律分散的に中継ノードを選択する機能が必要である。したがって、従来の有線ネットワークと違い、ノードの移動特性や消費電力、無線帯域などネットワークの特徴を考慮し、最適化したルーチングプロトコルが必要になる。代表的なルーチングプロトコルとしては、AODV[1], DSR[2], OLSR[3] などが提案されている。現在でもこれらの改良を含む多くのルーチングプロトコルが提案されている。また、ネットワークを構成する端末にスマートフォンやノート型 PC などを想定した場合、端末の電力に限りがある。そこで、消費電力を削減することを目的とし、経路を確立するまでの制御メッセージを少なくする手法の提案やその実装評価なども行われている。

また、アドホックネットワークはインフラストラクチャを使用せず自律分散的に動作するという特性上、ネットワークに存在するノードの情報が一元的に管理されず、基本的には必要に応じて各ノードそれぞれから情報を取得する、もしくはそれぞれが自身の情報を広告する必要がある。そのため、大きなコストが生じる。例えば、上述した代表的なルーチングプロトコルではフラッディングを用いて各ノードがネットワーク全体に制御パケットを送信する必要がある。そこで、本論文では分散処理技術である **Mobile Agent (MA)** に着目し、アドホックネットワークの柔軟性を保ちつつネットワーク内の全ノードの情報を一元的に管理するルーチング手法を検討する。さらに、これらの技術を応用して送受信データに着目して設計された次世代ネットワークのアーキテクチャであるコンテンツ指向型ネットワークにおける MA を用いたコンテンツ取得手法についても検討する。

本章の以降では、本研究で扱われる技術の概要について簡潔に説明する。

1.2 アドホックネットワーク

アドホックネットワークは、現在用いられている通信ネットワークと違い通信基地局などのインフラストラクチャを用いなくとも、周りの無線ノードと自律的に構築できるネットワークであり、柔軟性が高く、設置の容易なシステムを構成することができる。もともと米国の軍用に開発された技術であるが一般の通信に用いても大きなメリットがあるため現在では研究機関や企業などで研究が盛んに行われている。自身の通信範囲外のノードにデータを送るときには、中継可能なノードを探し、そのノードを経由して送信先ノードにデータを送る(マルチホップ通信)。図 1.1 にアドホックネットワークの構成例を示す。アドホックネットワークでは、ある程度広い範囲で構成された場合や遮蔽物がノード間にある場合は直接送信先ノードと通信を行うことはできないため一般的にマルチホップ通信が不可欠な要素である。

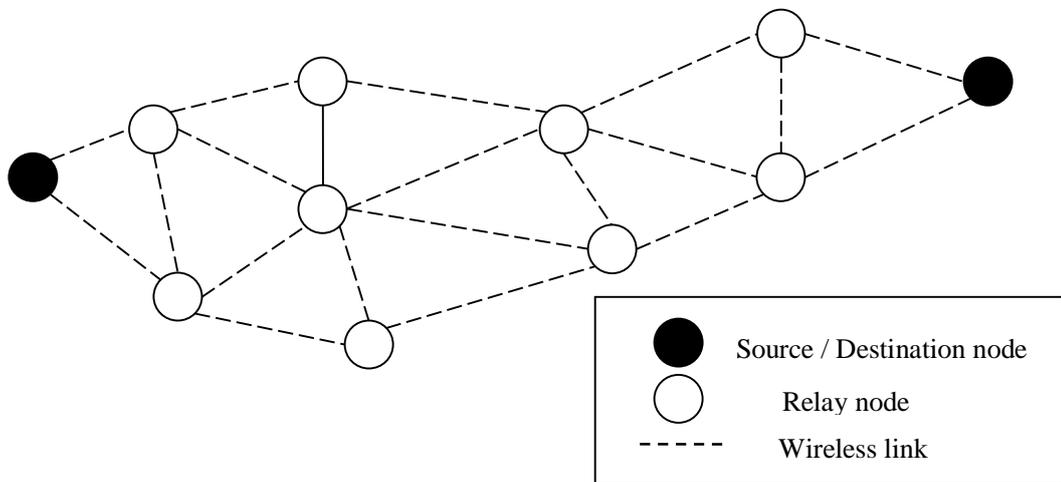


図 1.1 Ad hoc network

1.3 ルーチングプロトコル

ネットワークにおいてノードどうしが通信を行う際，通信するノード間の経路が必要となる．ルーチングはネットワークにおいてパケットを生成したノード（始点）からそのパケットのあて先ノード（終点）までの経路を選択し，パケットを配送する仕組みである．インターネットでは，RIP (routing information protocol), OSPF (open shortest path first)などが知られている．これらのルーチングプロトコルをアドホックネットワークで利用することは可能であるが，ノード移動によるリンク接続状態の急激・頻繁な変化への対応，無線帯域の効率利用などが課題となる．そこでアドホックネットワークの特性を考慮したルーチングプロトコルの最適化が必要になる．アドホックネットワークのルーチングプロトコルは経路情報の生成タイミングの観点から大きくプロアクティブ型とリアクティブ型とに分類される．プロアクティブ型のルーチングプロトコルは，前述のインターネットのルーチングプロトコルをベースとしてアドホックネットワークへの最適化を行う観点から開発されたもので，ノード間で周期的な制御メッセージの交換を行うことにより，各ノードが全ての終点への経路情報を常時保持する．通信要求時にはあて先への経路があらかじめ分かっているため，即時に通信できるという特徴を持つ．また，常時経路が確保されているため，通信タイムアウト，送信データロスが少ない利点があることから，頻繁に通信を行うネットワークには向いている．しかし，常時経路を確保するために制御メッセージの送信を頻繁に行うため，ネットワーク負荷や消費電力の面で欠点がある．なお，本論文における負荷とはデータの送受信より端末に生じる負担のこととする．代表的なものにOLSR，TBRPF[4]がある．リアクティブ型のルーチングプロトコルでは各ノードは通信要求が生じたときに経路情報を獲得・保持する．このため通信トラフィックが少ないときには制御メッセージのオーバーヘッドが比較的少ない方式である．しかし，経路構築を行う際に多少の時間がかかるため，通信が開始されるまでに時間がかかるという欠点がある．ノードの移動を考慮に入れたアドホックネットワークではプロアクティブ型に比べ，リアクティブ型のルーチングプロトコルが有効とされている．代表的なものにAODV，DSRがある．また，制御メッセージを減らし，オーバーヘッドを削減するための方法の一つとして位置情報を用いたルーチングプロトコルも提案されている．位置情報を用いたルーチングプロトコルはフラッドイングを必要とせずオーバーヘッドが小さいという利点がある．さらに経路決定のプロセスが簡単であり，単純

性と拡張性に優れている。位置情報を用いたルーチングには、位置情報を用いて実際に経路を決定しパケットを進めるフォワーディングプロトコル、経路を決定するために不可欠なネットワーク内のノードの位置情報を共有・取得するプロトコルが必要である。代表的なプロトコルとしては、位置情報を共有・取得するプロトコルには DREAM[5]や OCTOPUS[6]、フォワーディングプロトコルに GIDER[5]や COMPASS[7]などが挙げられる。さらに、パケット転送効率の向上を目的としたマルチパスルーチングプロコルの提案も行われている。先に述べたルーチングプロトコルでは、常に各方式で適切であると判断された単一の経路によりパケットが転送されるが、マルチパスルーチングでは複数の経路を同時に利用することでパケット転送効率の向上を達成する。代表的なプロトコルとして EECA[8], SMR[9], AODVM [10], 経路干渉考慮型マルチパスルーチング[11]などがある。

1.4 コンテンツ指向型ネットワーク

1.3 で想定しているネットワークはインターネットなどでも広く用いられている IP に基づくホスト指向型ネットワークだが、それに対して送受信データに着目して設計された次世代ネットワークのアーキテクチャであるコンテンツ指向型ネットワーク (ICN; Information-Centric Network) [12]が注目を集めている。これは、近年インターネットの利用形態がホスト間の通信ではなく、情報の配信・流通システムへ変化してきている流れを受けてのものである。コンテンツ指向型ネットワークの基本的なアイデアは特定のホストでなくコンテンツ名によってネットワークが機能する点にある。また、ネットワーク内キャッシングによる効率的なデータ取得が行えるという特徴を有する。この研究領域での有名な研究プロジェクトとしては NDN[13], CCN[14]等が挙げられる。

ICN の典型的な動作の具体例としては、まず、コンテンツを取得したいノードは Interest と呼ばれるリクエストパケットをネットワーク全体にブロードキャストする。このリクエストを受信したノードが、もし要求されているコンテンツを持っている場合は要求ノードへコンテンツを送信する。持っていない場合は、Interest の転送を継続する。

これらの研究は有線環境でのものが主であったが、近年では、MANET 等の無線環境に適した研究も行われている。また、MANET を構成するノードはスマートフォン等の小型の移動体無線端末を想定する場合が多いため、利用できる電力が限られている場合が多い。しかし、多くのコンテンツ指向型 MANET のコンテンツ取得手法では Interest パケットの送信にフラッドイングを用いることから、制御パケットの負荷が大きい傾向にある。これらの問題を解決するため、いくつかの手法が提案されている。TOP-CCN[15] では MPR 集合を用いたパケットのフラッドイングにより、制御パケットを抑制している。E-CHANET[16] ではカウンタベースのパケット転送抑制機構を用いている。REMIF[17] では Interest パケットの周囲の送信状況を用いて重複送信をチェックすることで制御パケットを削減している。

1.5 Mobile Agent (MA)

Mobile Agent (MA)とは、ネットワークに接続されたノードを移動しながら様々な処理をするプログラム（エージェント）であり、分散処理技術の一つである。もともと有線ネットワークでの分散システムとして開発されたが[18][19]、アドホックネットワークに応用するための研究[20][21] や、アドホックネットワークにおいて MA を効率よく運用する手法の提案[22]や解析[23]もおこなわれている。MA が行う処理としては、ノードからの情報収集と解析、あるいはノードへの情報配信などがある。

1.6 本研究の位置付け

ルーチングプロトコルはアドホックネットワークの性能に大きな影響を与える重要な要素の一つである。本論文で述べる研究ではルーチングに MA を用いることで効率的なルーチングを実現し、ネットワーク性能を向上させることを目的とする。また、このルーチング技術を次世代のネットワークアーキテクチャとして注目されているコンテンツ指向型ネットワークに応用した手法についても述べる。本論文では、3種類の MA を用いたルーチングとコンテンツ取得手法を提案し優位性を示す。

第一に、MA に全ノードの位置情報を管理させる方式について述べる。現在主流のリアクティブルーチングプロトコルの多くは、経路構築時に制御パケットのフラッディングを必要としており、ネットワーク負荷の増加が課題となっている。これに対して、経路構築時のフラッディングをなくし、制御パケット量を削減することを目的に MA が各ノードの位置情報を一元的に管理し、その情報から経路を作成する方式を提案する。計算機シミュレーションによる性能評価の結果より、既存の方式に比べ制御パケットを大幅に削減するとともにパケット到達率が向上することを示す。

第二に、スループットの向上を目的とした MA 利用型マルチパスルーチングを提案する。これは 1 つ目の方式を基本として経路計算を改良しマルチパスを算出させるようにした方式である。この方式は位置情報を元に通信可能距離を考慮して経路間の干渉を抑えたマルチパスを作成することで、よりスループットの高い経路を作成することが可能である。計算機シミュレーションによる性能評価の結果より、既存の方式に比べスループットが大きく向上することを示す。

第三に、隣接ノード情報を利用した MA 利用型ルーチングを提案する。上記 2 方式では位置情報に基づき経路を計算するため、各ノードが自身の位置情報を取得する必要がある。しかし、位置情報の取得には一般的に GPS を利用することが想定されているため、屋内での利用が困難である。また、GPS の測定誤差によるプロトコルの性能低下等の問題も生じる。そこで、1 つ目の方式を基本に位置情報の代わりに隣接ノード情報を用いて同様なルーチングを実現する方式を提案する。また、計算機シミュレーションによる性能評価の結果より、パケット到達率において位置情報を利用する方式よりも少し劣るものの AODV, OLSR よりは高い性能であることを示す。また、平均経路構築遅延と消費電力においても提案方式

が他の方式に比べ同等かそれ以上の性能であることを示す。これらの結果から総合的に判断して、本方式が性能を維持しつつ位置情報を利用することによるデメリットや制限を除くことができるということを示す。

最後にコンテンツ指向型ネットワークにおける MA を用いたコンテンツ取得手法を提案する。既存のコンテンツ取得手法ではコンテンツを要求するためのメッセージの送信にフラッドイングを用いる。そのため、要求したコンテンツを複数のノードが保持していた場合、要求パケットを受信した複数のノードがコンテンツを送信するため重複したコンテンツ送信が発生するという問題がある。これはネットワークの負荷を増加させる。これに対して、2章の方式をコンテンツ指向型 MANET のコンテンツ取得手法に応用することでネットワーク負荷を削減する手法を提案する。計算機シミュレーションによる性能評価の結果より、既存の方式に比べコンテンツ取得率、消費電力、コンテンツ伝送遅延が向上することを示す。

参考文献

- [1] C. Perkins, E. Belding-Royer, S. Das " Ad hoc on- demand distance vector routing, " RFC3561, pp. 4089-4095, 2003.
- [2] IETF MANET Working Group, " The dynamic source routing protocol for mobile ad hoc networks (DSR), " draft-ietf-manet-der-10. txt, Internet-Draft, Jul. 2004.
- [3] T.Clausen, "Optimized Link State Routing Protocol" IETF, <https://www.ietf.org/rfc/rfc3626.txt>, Oct. 2003.
- [4] Mobile Ad-Hoc Networks Working Group, "Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)", draft-ietf-manet-tbrpf-08.txt, Internet-Draft, October. 2003.
- [5] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, "A distance Routing Effect Algorithm for Mobility (DREAM)," Proceedings of the International Conference on Mobile Computing and Networking, pp.76-84, 1998.
- [6]R. Melamed, I. Keidar, and Y. Barel, "Octopus: A Fault-Tolerant and Efficient Ad-Hoc Routing Protocol," Proceedings of The 24th IEEE International Conference on Reliable Distributed Systems, pp.39-49, 2005.
- [7] J. Urrutia, "Two Problems on Discrete and Computational Geometry," Proceedings of Japan Conference on Discrete and Computational Geometry, pp42-52, 1999.
- [8] Z. Wang, E. Bulut and B. K. Szymanski "Energy Efficient Collision Aware Multipath Routing for Wireless Sensor Networks," ICC09 International Conference on Communication, pp. 1-5, Jan. 2009.
- [9] S. J. Lee, M. Gerla , "Split Multipath Routing with Maximally Disjoint Paths in Ad hoc Networks," IEEE International Conference on Communications, pp. 3201-3205, 2001.
- [10] Y. Ge, G Wang, Q. Zhang and M Guo, "Multipath Routing with Reliable Nodes in Large-Scale Mobile Ad-Hoc Networks," IEICE transactions on information and systems 92(9), pp. 1675-1682, 2009-9-1.
- [11] 小松 辰成, 塩川 茂樹, "経路間の干渉を考慮したマルチパスルーティングにおけるスループットの改善" 信学技報, AN2011-55, pp. 1-6, 2012 年 1 月.
- [12] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking." Communications Magazine, vol. 50, no. 7, pp. 26-36, 2012.
- [13] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named data networking." ACM SIGCOMM Computer Communication Review, vol. 44, no. 3, pp. 66-73, 2014

- [14] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content." In Proceedings of the 5th international conference on Emerging networking experiments and technologies, ACM, 2009, pp. 1-12.
- [15] J. Kim, D. Shin, and Y.B. Ko, "TOP-CCN: Topology aware Content Centric Networking for Mobile Ad Hoc Networks", 19th IEEE International Conference on Networks(ICON) 2013
- [16] M. Amadeo, A. Molinaro, and G. Ruggieri, "E-CHANET: Routing, forwarding and transport in Information-Centric multihop wireless networks." Computer Communications, vol. 36, no. 7, pp. 792-803, 2013.
- [17] R. A. Rehman, T. D. Hieu, H. Bae, S. Mah, and B. Kim "Robust and Efficient Multipath Interest Forwarding for NDN-based MANETs", Wireless and mobile Networking Conference, July.2016
- [18] M. S. Greenberg, J. C. Byington and D. G. Harper, "Mobile agents and security," Communications Magazine, IEEE, Vol. 36, Issue 7, pp76–85, July. 1998.
- [19] K. Mohammadi, and H. Hamidi, "Evaluation of fault-tolerant mobile agents in distributed systems," The First IEEE and IFIP International Conference in Central Asia, Sept. 2005.
- [20] A. F. Farhan, D. Zulkhairi and M. T. Hatim, "Mobile agent intrusion detection system for Mobile Ad Hoc Networks: A non-overlapping zone approach," Internet, 2008. ICI 2008. 4th IEEE/IFIP International Conference, pp.1–5, Sept. 2008.
- [21] R. T. Meier, J. Dunkel, Y. Kakuda and T. Ohta, "Mobile Agents for Service Discovery in Ad Hoc Networks," Advanced Information Networking and Applications, 2008(AINA 2008). Pp.114-221, March. 2008.
- [22] L. Wanlong, L. Dayou and Z. Hui, "Fault-Tolerance Mechanism of Mobile Agent In Mobile Ad Hoc Networks," Wireless Communications, Networking and Mobile Computing, 2008(WiCOM '08), pp.1–4, Oct. 2008.
- [23] Shigeki Shiokawa, "Performance Analysis for Use of Mobile Agent in Wireless Multihop Networks", IEEE International Conference on Ubiquitous and Future Networks (ICUFN' 16), pp.827-832, July 2016

第 2 章

2. モバイルエージェントを用いた位置情報 利用型ルーチング

2.1 はじめに

近年、無線通信技術の急速な発展に伴い、アドホックネットワークに関する研究が盛んに行われている。アドホックネットワークには特有の検討課題がいくつかある。中でも送信元ノードと送信先ノード間の経路を決定するルーチングは重要であり、多くのプロトコルが提案されている。アドホックネットワークのルーチングプロトコルは経路情報の生成タイミングの観点から大きくプロアクティブ型とリアクティブ型とに分類されるが、ノードの移動を考慮に入れたアドホックネットワークではプロアクティブ型に比べ、リアクティブ型のルーチングプロトコルが有効とされている。現在主流である AODV[1], DSR[2]などのリアクティブルーチングプロトコルの多くは、経路構築時に制御パケットのフラッディングを必要としており、ネットワーク負荷の増加が課題となっている。

これに対し経路構築時のフラッディングをなくすために位置情報を利用したルーチングプロトコルが注目されている。その背景には、GPS の精度向上や GPS 付のスマートフォンの普及など、アドホックネットワーク内の全ノードが位置情報を利用できるという仮定が現実的になったことが挙げられる。位置情報を用いたルーチングプロトコルは、従来のリアクティブルーチングよりも経路構築時の制御メッセージによる負荷が小さいことに加え、経路決定のプロセスが簡単であり、単純性と拡張性にも優れているという利点がある。このプロトコルの大きな特徴は、各ノードが、パケットに含まれた送信先の位置情報に基づき次の中継ノード（次ホップノード）を逐次的に選択していくことである。

次ホップノード選択方式としては、GEDIR[3], COMPASS[4]が提案されている。これらのプロトコルでは、隣接ノードと送信先ノードの位置に基づいて次ホップノードを選択する。動作の詳細については 2.3.1 で説明する。GEDIR や COMPASS を利用する場合、各送

信ノードは送信側先ノードや隣接ノードの位置情報を必要とする。従って、ネットワーク内のノード位置情報の取得を行う機能が必要となる。隣接ノード位置情報の取得には一般的に HELLO パケット交換等の単純な方法が用いられる。しかし、隣接ノード以外が送信側先ノードであり、かつその位置情報が未知の場合には、HELLO パケット交換以外の手段を用いて取得する必要がある。これまで DREAM[3]や OCTOPUS[5]等のプロトコルが提案されている。詳細については 2.2.2 で説明する。

以上に挙げた位置情報利用型ルーチングは、既存のリアクティブルーチングと比較して経路構築要求時の制御パケット量は削減できるものの、位置情報管理に伴う制御パケットによる負荷が依然課題として残っている。そこで本章では、モバイルエージェント (MA) を用いて位置情報を一元的に管理し経路構築を行うルーチングプロトコルを提案し、制御パケットの削減を図る。

提案方式では、位置情報管理と経路構築の機能を持った MA のプログラムを特定のエリア内に存在するノードが実行することで、ルーチングを実現する。ノードの位置情報更新は、予め設定した閾値以上に移動したノードのみが MA に向けてパケットを送信することで行う。MA を利用することによるオーバーヘッドは増加するが、位置情報の広告は必要なくなるため、プロトコル全体としての制御パケット量を従来方式より少なくすることが可能である。提案方式と既存方式の一つである OCTOPUS を比較対象とし計算機シミュレーションにより性能評価を行う。その結果を用いて、経路構築成功率と送受信制御パケット量にける提案方式の優位性を示す。

2.2 位置情報利用型ルーチングプロトコル

ノード移動を考慮したアドホックネットワークで有効とされている AODV, DSR 等のリアクティブ型ルーチングプロトコルは経路探索要求制御メッセージを全てのノードに転送するフラッディングを用いることから、経路検出に大きい通信オーバーヘッドと時間オーバーヘッドを要する。これに対し、制御メッセージを減らし、オーバーヘッドを削減するための方法の一つとして位置情報を用いたルーチングプロトコルが提案されている。この背景には、GPS の精度の向上やスマートフォンの普及により、無線ノードによる位置情報取得が容易になってきていることも関係している。これらのプロトコルでは、各中継ノードが、自身と全ての隣接ノードの位置及び送信先ノードの位置に基づいてデータメッセージの転送先ノード（以降次ホップノードと呼ぶ）を選択することで経路を決定する。位置情報を用いたルーチングプロトコルはフラッディングを必要としないためオーバーヘッドが小さいという利点がある。さらに経路決定のプロセスが簡単であり、単純性と拡張性に優れている。

位置情報利用型ルーチングには、位置情報を用いて実際に経路を決定しパケットを進めるフォワーディングプロトコル、経路を決定するために不可欠なネットワーク内のノードの位置情報を共有・取得するプロトコルが必要である。本節では具体的なプロトコルを例にとり、これらの概要を述べる。

2.2.1 フォワーディングプロトコル

位置情報を用いて実際に経路を決定し、パケットを進めるプロトコルのことをフォワーディングプロトコルと呼ぶ。フォワーディングプロトコルでは隣接ノード、送信先ノード等の位置情報を用いて、次ホップノードを選び、転送を繰り返すことで経路を決定する。代表的な手法に送信先ノードとの距離をメトリックとした GEDIR と角度をメトリックとした COMPASS がある。以下にそれぞれの概要を述べる。

GEDIR

GEDIR では隣接ノードの中から最もあて先ノードに近いノードを次ホップノードに選ぶ。GEDIR の動作を図 2.1 に示す。送信先ノードを D 、次ホップノードを選ぶノードを I とし、 k 個ある隣接ノードを $N_1(i) (1 \leq i \leq k)$ とすると次ホップノード $I+1$ は $N_1(i)$ と D の間の距離が最少となるノードである。このプロセスを繰り返しながら転送することでノード D へデータを届けることが可能である。GEDIR で検出された送信元ノードから送信先ノードまでの経路のホップ数は最短経路にほぼ等しく、平均的にはフラッディングを用いたルーチングプロトコルによる検出経路よりも短いことが分かっている。

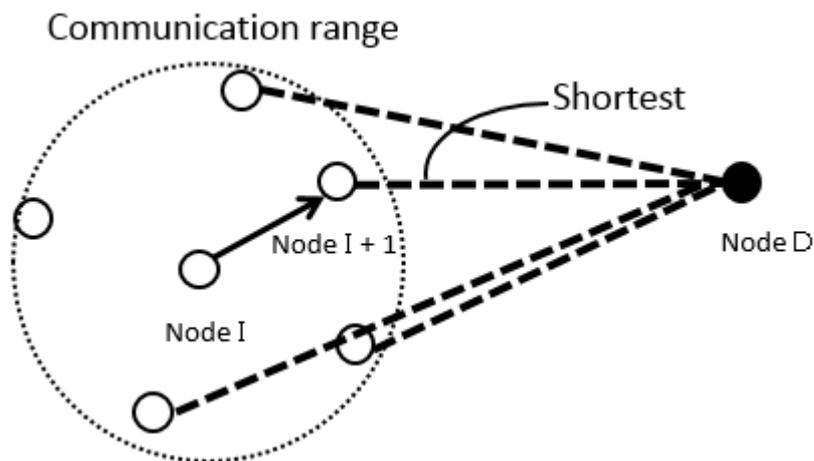


図 2.1 Forwarding in GEDIR

COMPASS

COMPASS では隣接ノードの中から送信先ノードへの角度が最も小さいノードを次ホップノードに選ぶ。COMPASS の動作を図 2.2 に示す。送信先ノードを D 、次ホップノードを選ぶノードを I とし、 k 個ある隣接ノードを $M_1(i)$ ($1 \leq i \leq k$) とすると次ホップノード $I+1$ は I から D と $M_1(i)$ を見込む角が最少となるノードである。このプロセスを繰り返しながら転送することでノード D へデータを届けることが可能である。COMPASS では GEDIR のように送信先ノードとの距離が単調減少することが保証されていないことから、検出された送信元ノードから送信先ノードまでの経路は、そのホップ数が GEDIR で検出される経路より大きくなるのが一般的である。

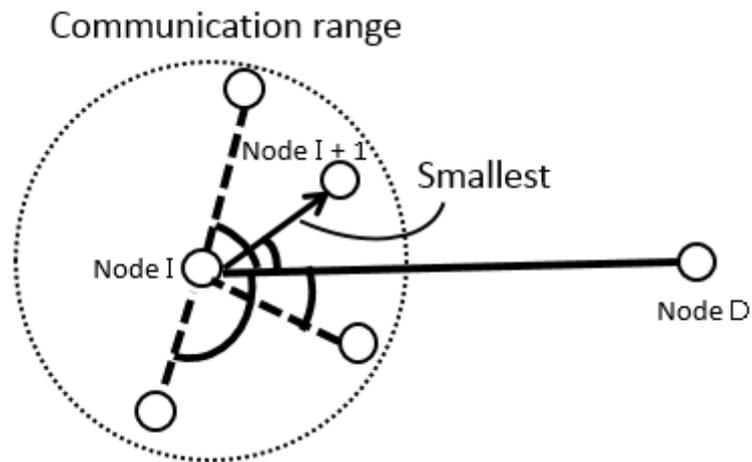


図 2.2 Forwarding in COMPASS

GEDIR と COMPASS では送信元ノードから送信先ノードまでの経路が存在するにもかかわらず経路を検出できない、デッドエンドという現象(図 2.3)が発生する可能性がある。GEDIR と COMPASS を比べると、GEDIR は経路のホップ数が少ないがデッドエンドの発生確率が高く、COMPASS は経路のホップ数が多いがデッドエンドの発生確率が低いことが分かっている。

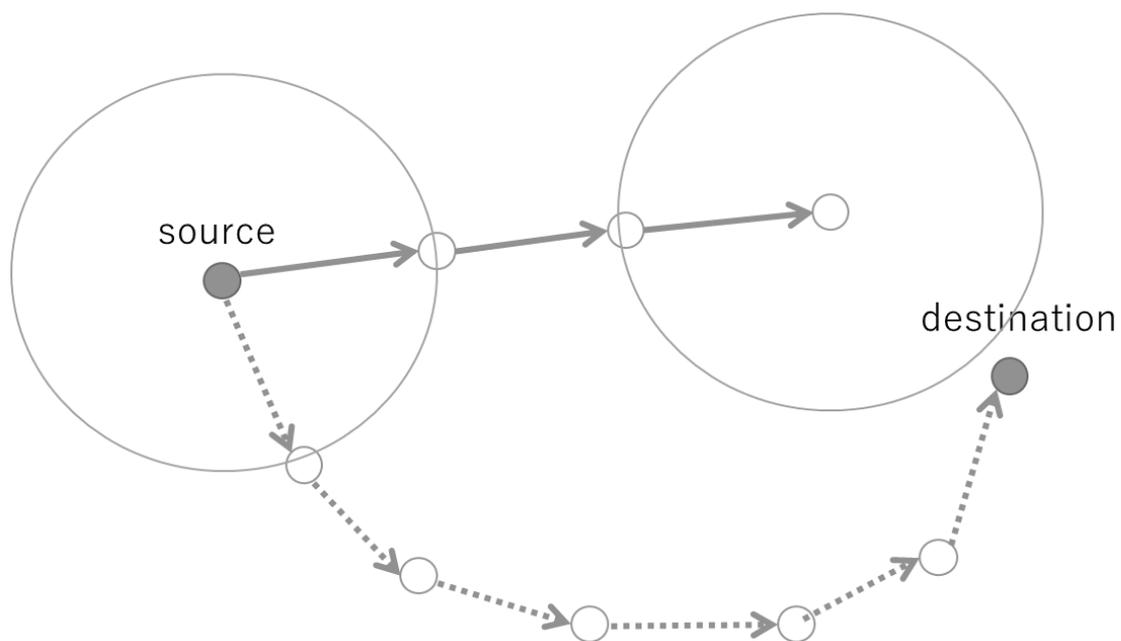


図 2.3 Dead end

2.2.2 位置情報取得プロトコル

ノードが移動するアドホックネットワークでは、位置情報は常に変化しているため、位置情報を用いてルーチングを行う場合、できるだけ正確な位置情報を把握している必要がある。先に説明したフォワーディングプロトコルではノードが全ての隣接ノードの位置情報を取得しなければならない。これを実現する方法として、HELLO パケットを用いて各ノードが自身の位置情報を全ての隣接ノードに広告することが考えられる。広告の手法として各ノードが定期的に HELLO パケットをブロードキャストするものがある。これにより無線信号到達範囲内にあるノードに通知することができるため、自身を次ホップノードとする可能性のある全てのノードがその位置情報を知ることができる。この方法では各ノードは常に隣接ノードの位置情報を把握しているため、経路構築の遅延は小さくなるが、通信要求の有無にかかわらず、定期的に HELLO パケットを送信する必要があるためトラフィックが大きくなる。

これに対して、配送要求があった際にのみ HELLO パケットを送信しそれを受信したノードに位置情報を載せた REPLY を返信してもらうことで、隣接ノードの位置情報を取得する手法がある。この手法では常時隣接ノードの位置情報を把握しているわけでないため、定期的に広告する方法に比べ経路構築の遅延が大きくなるが、配送要求がない時には HELLO パケットの送信をする必要がなくトラフィックが比較的少なくて済む。

実際の通信にはこれら HELLO パケットの交換で得られる隣接ノードの位置情報の他に送信先ノードの位置情報も取得する必要がある。そのため、様々な位置情報共有・取得プロトコルが提案されている。代表例として以下では DREAM と OCTOPUS について説明する。

DREAM

DREAM では各ノードが全てのノードの位置情報を保持する完全分散型手法を用いている。動作例を図 2.4 に示す。各ノードは予め自身の位置座標を保存しておき、自身が移動した際に移動後の位置座標と保存してある位置座標の相対距離を計算し、それがあらかじめ定められた閾値 r 以上に変化した場合に自身の位置情報をネットワーク全体に広告する。つまり、アドホックネットワークの存在領域に定められた固定座標系に対して閾値以上の移動を行うことを位置情報広告のトリガとしている。広告後、保存していた位置座標を広告後の位置座標に更新する。また、広告を受信したノードは広告に含まれる位置情報を各々の位置情報テーブルに登録し、保持する。これにより、各々のノードはネットワークに所属する全てのノードの位置情報を得ることができる。この方式では全てのノードが全てのノードに対して位置情報の広告を行うため、トラフィックの増大が問題点としてある。また、ノードの移動速度が大きければ大きいほど広告の頻度が増しトラフィックが大きくなる。

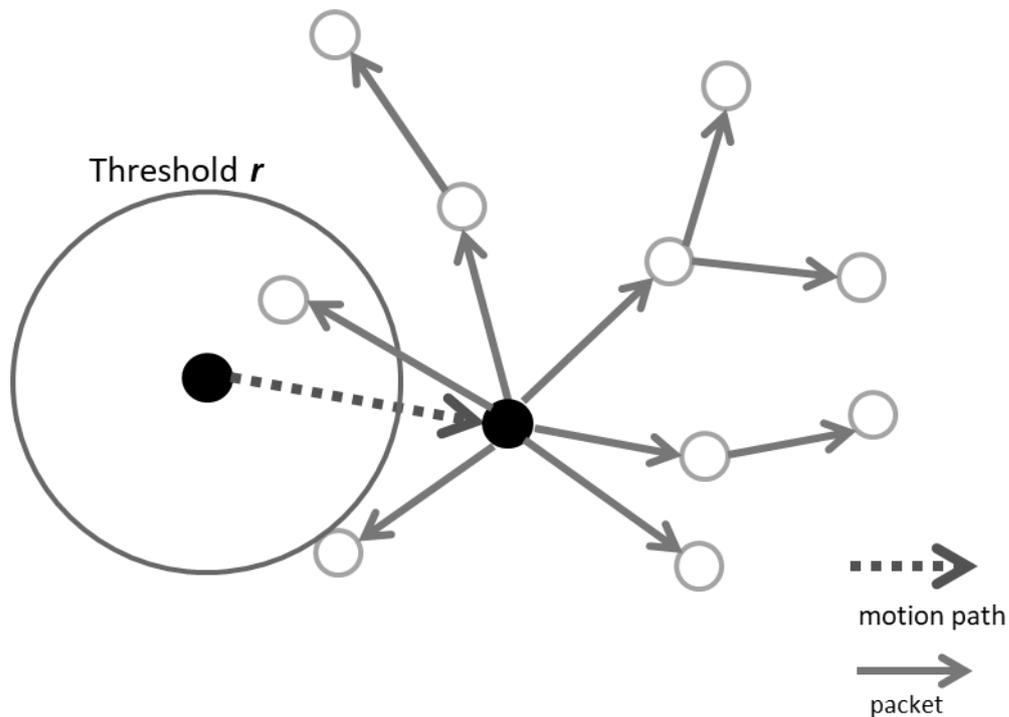


図 2.4 DREAM 動作例

OCTOPUS

OCTOPUS は、経度/緯度線を用いて図 2.5 のような仮想格子(Strip) を形成し、位置情報の広告範囲を特定の仮想格子内に限定することで広告のための制御パケットを削減する方式である。図 2.5 は 36 個の仮想格子が形成されていることを表しており、仮想格子幅はノードの通信可能距離と同程度に設定される。OCTOPUS 位置情報の共有だけでなく、フォワーディングについての動作も定義しており、位置情報更新、送信あて先ノード位置取得、フォワーディングの 3 つのプロトコルから成り立つ。なお、OCTOPUS はフォワーディングには 2.2.1 で説明した GEDIR を採用している。

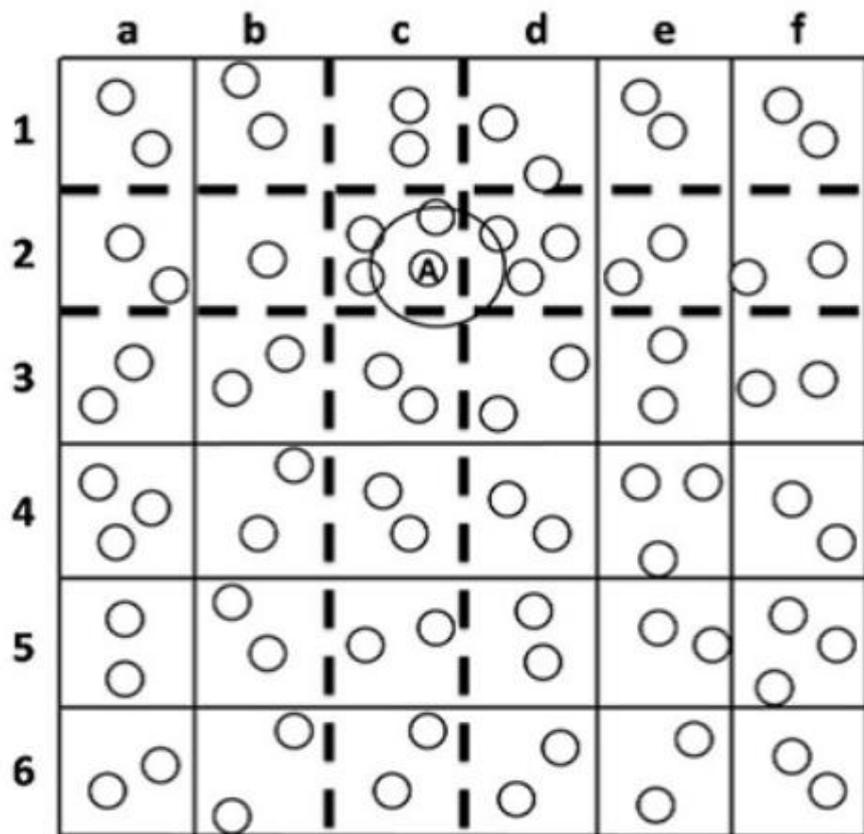


図 2.5 Strip in OCTOPUS

- ・位置情報更新

OCTOPUS ではネットワーク内の全ノードが、隣接ノード位置を保持する隣接ノードテーブルと所属仮想格子内のノード位置を保持する仮想格子テーブルを装備している。ここで所属仮想格子は、自身が存在する仮想格子と同じ経度線あるいは緯度線で構成される仮想格子の集合として定義される。図 2.5 を例にとると、ノード A の所属仮想格子は破線で挟まれたエリア a2, b2, c2, d2, e2, f2, c1, c3, c4, c5, c6 である。

隣接ノードテーブルは隣接ノードと定期的に交換する HELLO パケットに書き込まれた位置情報により更新される。仮想格子テーブルの更新には仮想格子更新パケットが用いられる。仮想格子更新パケットはネットワークエリア東（西/南/北）端である各仮想格子内で最も東（西/南/北）に存在するノードから対応する西（東/北/南）端の仮想格子内で最も西（東/北/南）に存在するノードへ定期的に送信されるパケットである。仮想格子更新パケットには送信ノードが保持している仮想格子テーブル情報が格納されている。これを受信したノードは、パケットに書き込まれたテーブル情報と自身の隣接ノードテーブルおよび仮想格子テーブルを比較することで、互いのテーブル情報を最新の状態に更新する。そしてもしもこのノードが中継ノードとして指定されていた場合には、パケットの進行方向に存在するノードのうち所属仮想格子内で最も自身から遠いものを次ホップノードと指定してブロードキャスト転送する。なお仮想格子幅が通信可能距離と同程度に設定されているため、全てのノードは仮想格子更新パケットを受信することができる。従って、全てのノードは自身の所属仮想格子に存在するノードの位置を把握できる。

図 2.6 に仮想格子更新パケットの送信例を示す。図では最南端仮想格子の一つである c6 の南端にいるノード A が対応する北端仮想格子のノード B にパケットを送信している。そして c1 から c6 までの全てのノードがパケットを受信する。

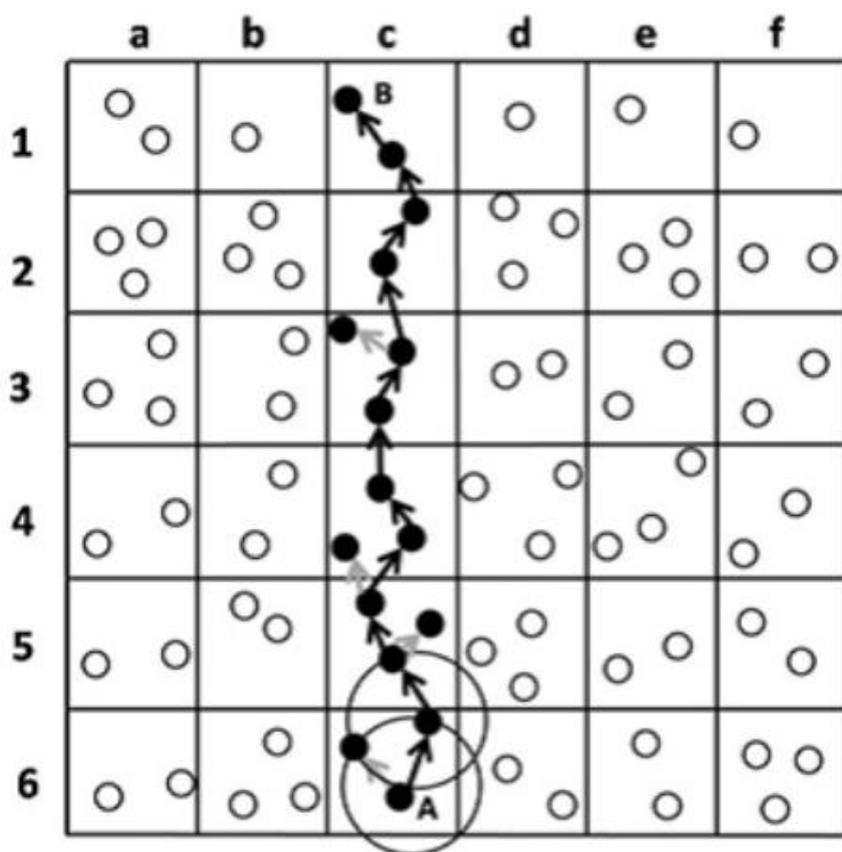


図 2.6 Sending of strip update packet.

・送信あて先ノード位置取得

データ送信の要求を持つノード（送信元ノード）は送信あて先ノードの位置が不明の場合、次の手順でその情報を取得する。まず、所属仮想格子の南北もしくは東西方向のどちらかに QUERY パケットを送信し、探索タイムアウトをセットする。QUERY パケットを受信したノードは自身の隣接テーブルおよび仮想格子テーブルを参照し、要求されたあて先ノードの位置情報を保持しているなら REPLY パケットに書き込み送信元ノードへ返信する。あて先ノードの情報を保持せず、かつ中継ノードに指定されている場合には QUERY パケットを転送する。転送方法は仮想格子更新パケットと同様である。送信元ノードが探索タイムアウト経過時点で REPLY パケットを受信しなかった場合は、東西あるいは南北方向に切り替え同様の動作を行う。以上の処理を位置情報取得成功まで繰り返す。

2.3 提案方式

OCTOPUS では、仮想格子更新パケットの到達範囲を所属仮想格子内に限定することで位置情報管理のための制御パケット量の削減を図っている。しかしながら、位置情報を含んだ HELLO パケットと同様に、定期的な送信が必要があることから、まだネットワークにかかる負荷は大きいと考えられる。

提案方式では MA に全ノードの位置情報の管理と経路の構築を行わせることで制御パケット量のさらなる削減を図る。提案方式は位置情報更新、経路情報取得、経路構築の3つのプロトコルから成り立つ。初めに提案方式における MA の役割について述べ、次いで各プロトコルの詳細を述べる。

2.3.1 提案方式におけるモバイルエージェントの動作

提案方式では図 2.7 のように MA が中心座標と半径により定義される円形エリアに存在するノードによって起動されていることを前提とし、ネットワーク内の全ノードはこの存在エリアを把握しているものと仮定する。また、MA 存在エリアの半径は1ホップ通信可能距離の半分に設定する。これにより MA を保持するノード(MA ホストノード)の ID や位置を知らなくても、把握している中心位置に向けてフォワーディングを行うことで、パケットを MA ホストノードへ届けることが可能になる。なお、MA と MA ホストノードとは厳密には異なるが、以降では両者を特に区別せず全て MA と呼ぶことにする。MA はネットワークに所属する全ノードの位置情報を記録するテーブルを保持している。そして、2.3.2 で述べる位置更新パケットを受信すると、対応するノードの位置情報を更新する。また、2.3.3 で述べる経路要求パケットを受信すると、テーブルの位置情報を用いて適当な経路を算出し返信する。

MA ノードが移動することで存在エリアから外れた場合、存在エリア内のノードで新たに MA を起動させるために、位置情報テーブルを含んだ MA をそのノードへ送信する必要がある。本論文ではこれを MA の移動と呼ぶ。MA の移動は以下の手順で行われる。

MA ノードが隣接ノードに MA 移動要求パケットをブロードキャストし、これを受信したノードは自身の位置情報を返信する。MA ノードは一定時間内に受信した返信パケットの

情報から、MA 存在エリアの中心に最も近いノードを MA の移動先として選択する。実際に MA を移動させるときには、MA の動作に必要なプログラムコードと MA が保持する位置情報テーブルを TCP 通信により送信する。通信の失敗による MA の消滅を防ぐために、MA に必要なパケットを全て送信し対応する ACK を完全に受信した時点で MA の移動が完了したと判断し、送信ノードにおける MA の起動を停止させる。もしも存在エリアにより近いノードがない場合は、該当するノードが現れるまで MA を保持する。なお本研究では、MA が存在エリアから外れることによるサービス停止は考慮しているが、MA ノード自体が無くなるなどの原因による MA の消滅は考慮しないものとする。

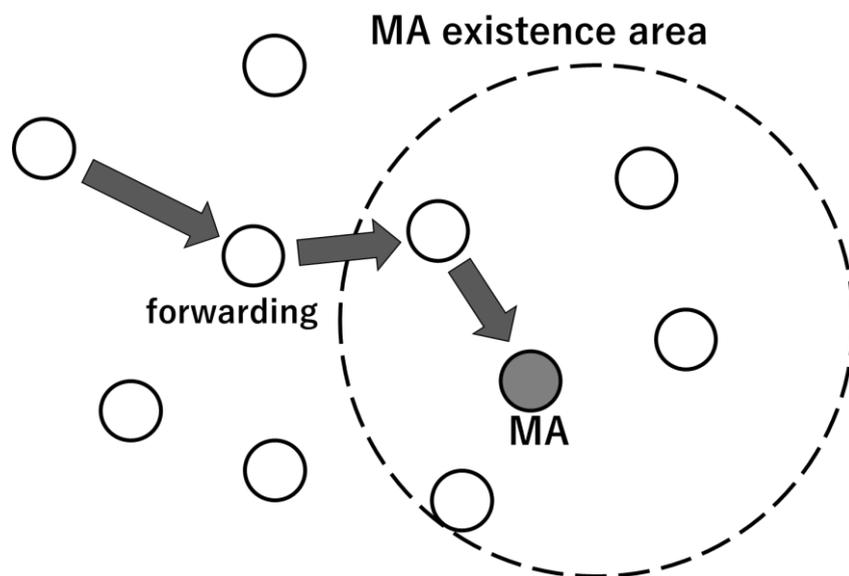


図 2.7 MA existence area and forwarding to MA

2.3.2 位置情報更新

位置情報の更新パケット送信はノードの位置変更量をトリガとして行われる。ただし、DREAMのように全ノードへ広告するのではなく、MAのみに送信することで制御パケット量の削減を実現する。

各ノードはある地点をあらかじめ基準点として定め、そこからの移動距離 d を保持する。位置情報の更新は d が閾値 r 以上になったとき行われる。位置情報更新を行うノードは、更新パケット送信後に現在の位置を新たな基準点とし d を 0 に初期化する。この様に各ノードが一定距離を移動するたびに MA へ更新パケットの送信を行う。

更新パケット送信には OCTOPUS におけるフォワーディングと同様に、距離をメトリックとしたフォワーディングアルゴリズムを用いる。つまり隣接ノードの中から最も MA 存在エリアの中心へ近いノードを次ホップノードとして選ぶことで MA への転送を実現する。ただし、提案方式は隣接ノードテーブルを装備しないため、次ホップノードを選択するタイミングで隣接ノードの位置情報を取得する必要がある。以下に具体的な手順を示す。

更新パケット送信ノードは、まずそのノード自身から MA 存在エリアの中心までの距離を含めた QUERY パケットをブロードキャストし、隣接応答タイムアウトをセットする。QUERY パケット受信ノードは、自身と MA 存在エリアの中心との距離を計算し、QUERY に含まれる距離より短い場合のみ、そのノードの位置情報を含めた REPLY パケットを返信する。これにより、次ホップノードの候補にならないノードからの無駄な返信をなくす。隣接応答タイムアウトが経過した QUERY パケット送信ノードは、受信した REPLY パケットの情報から最も MA 存在エリアに近いノードを次ホップノードに選び更新パケットを送信する。この動作を繰り返すことで MA へ更新パケットを届ける。図 2.8 にノード I が次ホップノードであるノード I+1 を選択する動作の具体例を示す。ノード I を中心とした円が QUERY パケットのブロードキャスト送信、矢印が REPLY パケット送信を表す。

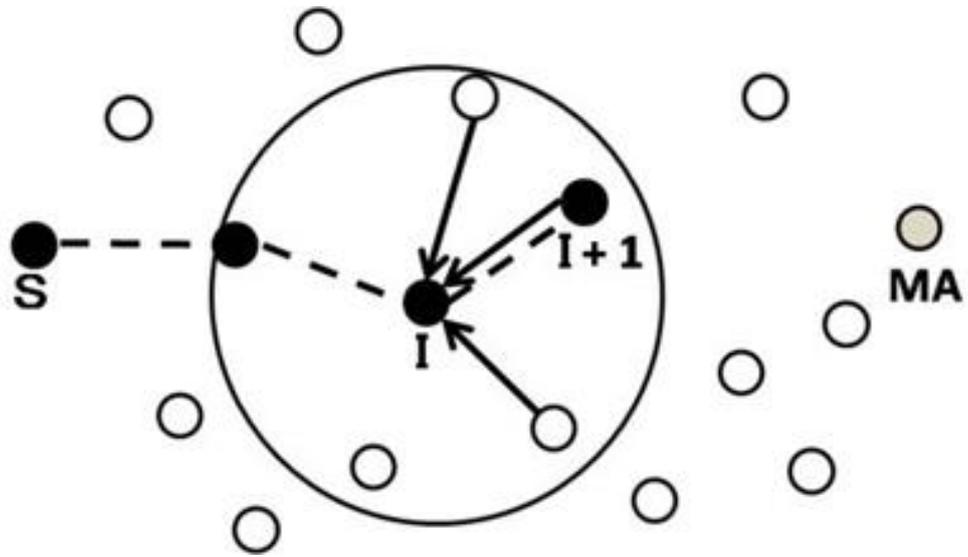


图 2.8 Next hop node selection in the proposed protocol.

2.3.3 経路情報取得

提案方式では MA が全ノードの位置情報を保持しているため、データ送信要求ノードは OCTOPUS のように送信あて先ノードの位置情報だけではなく、あて先ノードまでの完全な経路情報も取得する。

あるノードにおいて通信要求が発生した際、あて先への経路情報を保持していない場合に経路情報取得を行う。まず、通信要求ノードは経路要求パケットを位置情報更新パケットと同じ手順で MA に送信する。ただし中継ノードがデータ送信要求の対象となるあて先ノードであった場合はその時点で経路を確定し、そのノードの位置情報を書き込み経路応答パケットとして返信する。MA が経路要求パケットを受け取った場合は保持しているノード情報から適当な経路を算出し、その経路情報と送信先ノードの位置情報を書き込んだ経路応答パケットを返信する。いずれの場合も、返信には経路要求パケット送信に利用された経路を利用する。

2.3.4 経路の算出

OCTOPUS をはじめ位置情報利用型ルーチングプロトコルの多くは送信先ノードの位置情報のみを利用したフォワーディングによりデータ送信を行うが、提案方式では基本的に MA が算出した経路を利用する。そこで MA による経路算出アルゴリズムを、図 2.9 を用いながら説明する。

図 2.9 は、MA が保持している位置情報を基にノードを配置したものである。まず経路要求ノード S から送信先ノード D へ直線を引き、その直線上で経路要求ノードから $R-r$ の地点をマークする（図のひし形）。ここで R は通信可能距離であり r は 2.3.2 で述べた位置情報更新のための閾値である。次に経路要求ノード S とマークした地点を中心とする半径 $R-r$ の 2 つの円を設け、重なった部分に存在するノードの中から最もマーク地点に近いノードを次ホップノード候補 M_1 として決定する。次にノード M_1 とノード D を用いて同様の処理を行い次の候補ノード M_2 を求める。この処理を再帰的に行い、ある候補ノードの隣接ノードにノード D が含まれた時点で経路を確定させる。なお、円の重なりにノードが存在せず候補ノード M_x を求められない場合には、まず円の半径を R まで拡大して候補を探す。それ

でも見つからない場合には $M_{(x-1)}$ を決める処理に戻り、このノードの次にマーク地点に近いノードを新たな $M_{(x-1)}$ としてノード M_x を求めなおす。二つの円が重なった部分に存在するノードのみを次ホップノードの候補としているのは再帰処理の演算量を少なくするためである。したがって、ノード D までの経路を確定できない場合がある。その際にはノード D の位置情報のみを経路応答パケットに含めて送信する。また、 MA が保持している位置情報は更新閾値 r 以下の誤差を持っている。そこで円の半径を $R-r$ に設定することで、情報誤差による算出経路の有効性への影響を抑えている。

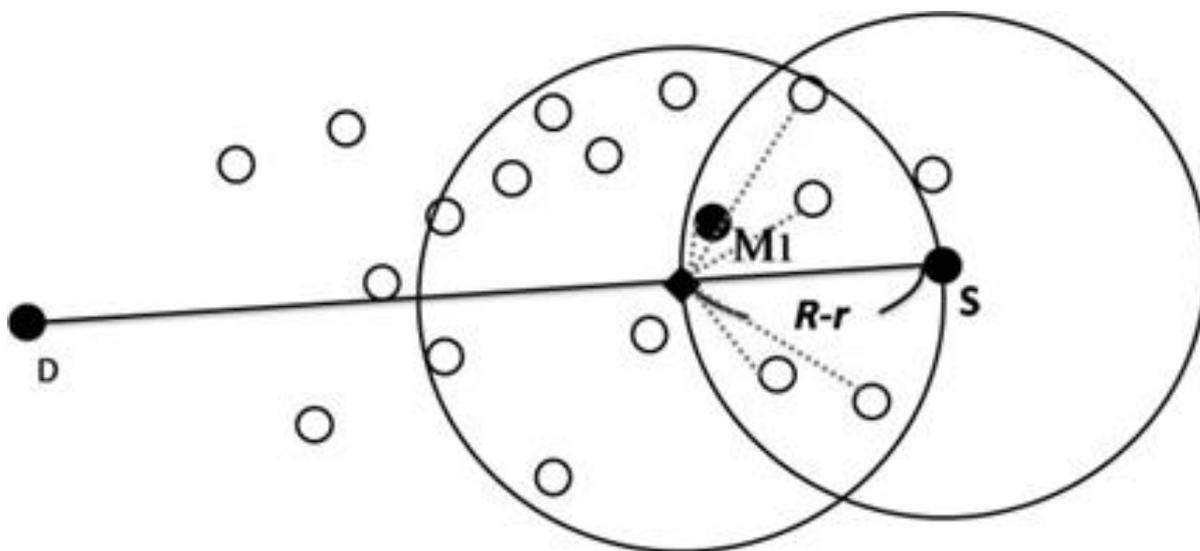


図 2.9 Route calculation by MA.

2.3.5 経路構築

データ送信要求ノードは MA から経路応答パケットを受け取っても、すぐにはデータパケットを送信しない。なぜならノードの移動により MA の算出経路が存在しなくなっている可能性があるためである。そこで、まずその経路を用いて経路確認パケットを送信する。同時に経路確認タイムアウトをセットする。経路確認パケットを受け取った送信元ノードは同じ経路を用いて経路確認応答パケットを返信する。データ送信要求ノードが経路確認応答パケットを受信した時点で、この経路を利用してデータ送信を行う。経路確認タイムアウトが経過した時点で経路確認応答パケットを受信できなかった場合には、MA から受け取った送信元ノードの位置情報のみを利用し、QUERY パケット送信と同じ手順で経路確認パケットを再送する。

経路確認応答パケットを受信できない理由としては、MA が保持しているノードの位置情報が不正確なため算出された経路が有効でなかったという場合と、単純にパケット衝突などの理由により経路確認あるいは経路確認応答パケットが欠落してしまったという場合が考えられる。失敗の原因が前者ならば位置情報を用いた経路確認パケット再送の効果は大きいと考えられる。

2.4 シミュレーションによる性能評価

提案方式の有効性を検証するためにシミュレーションによる性能評価を行う。比較対象には2.2.2で説明したOCTOPUSを用いる。以下ではシミュレーションの内容について詳しく述べる。

シミュレーションでは、エリア内にランダムに配置された全てのノードが Random Way Point に従い移動するモデルを用い、MAC プロトコルには IEEE802.11b を用いる。提案プロトコルの性能を評価するために経路構築成功率と経路構築に必要な制御パケットの送受信を測定する。経路構築成功率は、ランダムに選んだ2ノード間の経路構築を各プロトコルにより試みた回数に対して、経路構築が成功した割合で定義される。OCTOPUS ではフォワーディングパケットを送信して先ノードが受信できたことを経路構築の成功とし、提案方式ではデータ送信要求ノードが経路確認応答パケットを受信できたことを成功とする。制御パケット送受信量は各プロトコルで位置情報更新、位置/経路情報取得および経路構築に用いる制御パケットの総送受信バイト数と定義する。

また、ネットワーク負荷が性能に与える影響を調査するために、バックグラウンド UDP フローを定められた発生頻度に従いランダムなノード間に付加した状態で性能を評価する。そしてノード移動速度の影響も評価するため、Random Way Point アルゴリズムにおける速度更新の際の選択範囲を変化させた場合の測定も行う。さらに、ノードの数がプロトコルに与える影響評価のために4パターンのノード密度を用いた。具体的なシミュレーションパラメータを表2.1に示す。なお、表の移動速度欄にある1~3[m/s]の表記は、速度更新の際にこの範囲から一様分布に従うランダムな値を求めこれを用いるということの意味する。

次にOCTOPUS固有のパラメータを示す。仮想格子の幅は通信範囲と同じ100mとする。隣接ノードテーブル作成のためのHELLOパケットは各ノードが2秒間隔で送信する。仮想格子更新パケットについては、各ノードが10秒間隔で隣接ノードとの位置関係チェックを行い、もし自身が2.2.2で述べたようなネットワークエリアの端に存在するノードだった場合にパケット送信を開始する。位置情報取得の際に用いる探索タイムアウトは2秒とする。各制御パケットサイズについては、基本サイズを12バイトとし、ノードIDフィールドサイズを4バイト、位置情報フィールドサイズを8バイトと仮定した。そして基本サイズに必要なフィールド分を追加したものを制御パケットサイズとした。例えば仮想格子更新パ

ケットには仮想格子テーブルの情報（ノード ID, 位置情報）が書き込まれるため、そのサイズは可変であり $12+12n$ バイトとなる。ここで n はエントリノード数である。

続いて提案方式固有の設定値を示す。MA の存在エリアをシミュレーションエリアの中心に設定し半径 50m の円形とした。位置情報更新パケット送信タイミングを決める閾値 r は 20m とする。タイムアウト値については隣接応答タイムアウトを 0.03 秒、経路確認タイムアウトを 1 秒とした。各制御パケットサイズは OCTOPUS と同様の計算で定義する。なお MA のサイズについては 30 キロバイト+位置情報テーブルサイズとした。

最後にシミュレーションシナリオについて説明する。1 回のシミュレーション時間は 1000 秒であり、この間に各ノードが移動しながらそれぞれのプロトコルに基づき位置情報の共有・管理を行う。具体的に OCTOPUS では HELLO パケットと仮想格子更新パケットの交換を行い、提案方式では、MA に対して位置情報更新パケットを送る。シミュレーション開始 20 秒後からランダムな 2 ノード間でデータ送信要求を発生させ始め、以降 25 秒間隔で次々とデータ送信要求を発生させる。これは異なる経路構築のための制御パケットの衝突による影響を排除した状態で経路構築成功率の測定をするためである。1000 秒間では 40 回の経路構築が試みられるが、ノードの配置条件を変えて 200 回測定し、計 8,000 回の経路構築に対する性能測定を行った。

表 2.1 シミュレーションパラメータ

シミュレーションエリア	600[m]
通信範囲	100[m]
チャンネルレート	11[Mbps]
バッファサイズ	64[kbytes]
UDP フロー発生頻度	0~1[/s]
UDP フローサイズ	500[kbytes]
UDP レート	2[Mbps]
移動速度	1~3,1~5,1~7,1~9[m/s]
ノード密度	400~700[/km ²]
シミュレーション時間	1000[s]

2.5 シミュレーション結果・考察

初めに 1000 秒×200 回の中に全てのノードが送受信した制御パケットサイズの総計を求め、プロトコルで使用される制御パケットサイズの比較を行う。本シミュレーションでは、ブロードキャストパケット送受信における送信サイズと受信サイズの総計カウント方法が異なり、次の様に定義される。例えば 10 バイトのブロードキャストパケットを送信した場合、送信パケットサイズは 10 バイトとなるが、受信パケットサイズについては 3 ノードが受信した場合は 30 バイトとし、7 ノードが受信した場合は 70 バイトとする。これは送信サイズと受信サイズの差を調べることでブロードキャストパケットの多少が推測できるようにしたためである。

図 2.10 はノードの移動速度を 1~3m/s とした場合の総送信パケットサイズを示したグラフである。横軸には UDP フロー発生頻度を用い、ノード密度ごとの結果を表示している。図より、UDP フローが発生しない状況では、提案方式の総送信パケット量を OCTOPUS の 80%以下に削減できていることがわかる。これは、位置情報更新用のパケット送信頻度の違いによると考えられる。OCTOPUS では全てのノードに対して仮想格子更新パケットが東西南北の各方角から送信され多くのノードがそれを転送する必要がある。また、HELLO パケットの送信も定期的に行われる。これに対して提案方式では閾値以上の移動がない限り位置情報更新パケットを送信しない。従ってノードの移動速度が 1~3m/秒と比較的低速なノードが多い環境では提案方式の更新パケット送信頻度のほうが小さくなると考えられる。さらに、提案方式では隣接ノード情報の取得を必要な時点で行うことでも送信パケット量を削減できている。次に、OCTOPUS では UDP フローによる負荷が大きくなるにつれて送信サイズが減少するのに対して、提案方式ではほとんど変化しないことがわかる。これは、UDP フローとの衝突により制御パケットが欠落し、以降のノードが転送する機会を失うためである。特に仮想格子更新パケットはエリアの端から他端まで転送されるため経路ホップ数も多く欠落する可能性も高い。従って OCTOPUS で図のような傾向が現れる。

図 2.11 は図 2.10 と同じ条件における総受信パケットサイズを示したグラフである。送信パケットサイズとは異なり UDP フローが多い場合でも、提案方式の制御パケットのほうが少ないことが分かる。これは OCTOPUS における仮想格子更新パケットや経路構築を行う

際の QUERY パケットが転送端末を指定したブロードキャストパケットであり、受信サイズとして多くカウントされるためである。

図 2.12 は図 2.10 と同じ条件における経路構築成功率を示したグラフである。図よりノード密度や UDP フローの量に関係なく提案方式の経路構築成功率のほうが高いことが分かる。以上の結果より、ノードの平均速度が小さい場合には提案方式が有効であることが示された。

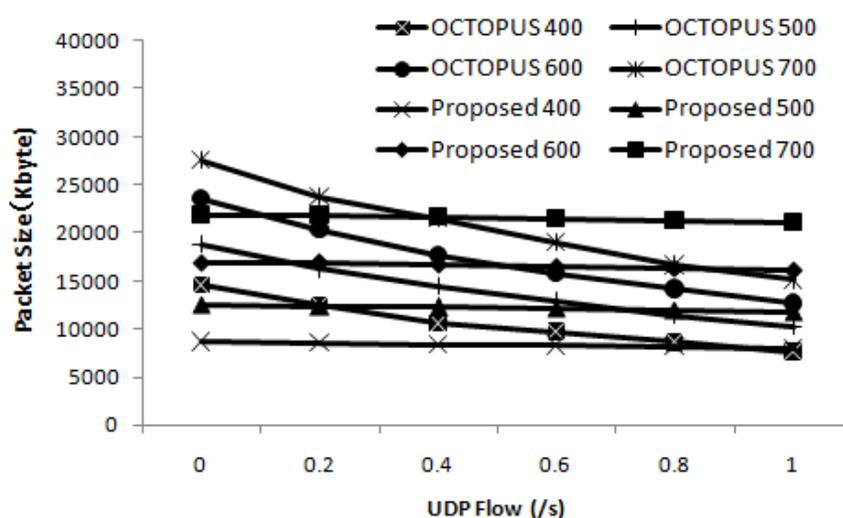


図 2.10 総送信パケットサイズ

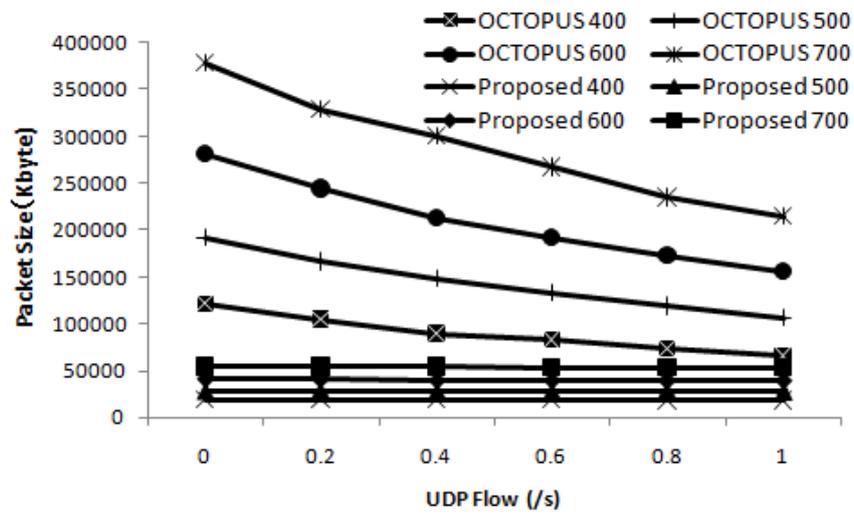


図 2.11 総受信パケットサイズ

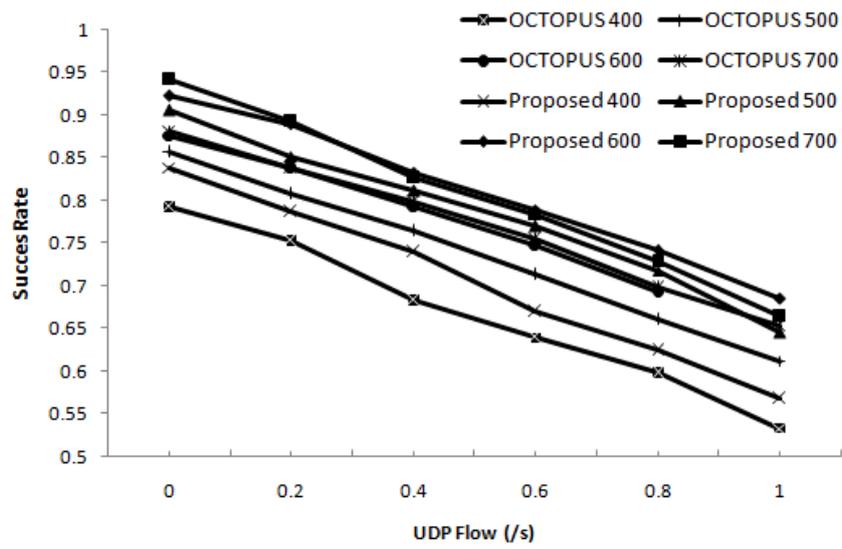


図 2.12 経路構築成功率

次に、ノードの移動速度による性能への影響を評価する。図 2.13, 図 2.14 は UDP フローがない状態での総送受信パケット量を示したグラフである。横軸にはノード密度を用いている。両方式ともに移動速度が大きくなるにつれてパケット量が増加していることが分かる。これは、提案方式では位置情報更新頻度が移動速度に依存するためだと考えられる。これに対し OCTOPUS ではノードの移動速度が大きくなると仮想格子更新パケットの転送効率が悪くなり、ホップ数が増加するためだと考えられる。また前述したように OCTOPUS ではブロードキャストパケットの量が多いため、ノード密度が高くなると送受信パケット量の増加が著しくなる。

図 2.15 は図 2.13 と同じ条件における速度別経路構築成功率を表したグラフである。図から全てのケースで提案方式の構築成功率が高いことが分かる。しかしながら、提案方式ではノード移動速度が大きい場合にノード密度が高くなると成功率が大きく低下することが分かる。これは移動速度が大きくなると位置情報更新パケットの送信頻度が増え、さらにノード密度が高くなることでパケットの衝突が起こり欠落したためと考えられる。以上の結果からノードが高速になると、経路構築成功率に関する提案プロトコルの優位性は少なくなる。しかし制御パケット量については十分な優位性を持つ。従って、提案プロトコルは有効であると考えられる。

提案方式では MA を利用することで制御パケットの削減に成功したが、MA ノードに負荷が集中することが考えられる。そこで最後に MA ノードへの負荷評価として、1 回のシミュレーション中で最も MA 起動による負荷の大きかったノードの送受信パケット量を調査した。このパケット量には MA として送受信した経路構築用制御パケットだけでなく、MA の移動に伴う制御パケットおよび MA の実体データも含まれる。ただし、経路計算やデータベース管理など CPU やメモリ関連の負荷についてはここでは考慮していない。

図 2.16 はノード数が少ないため各ノードへの負担が一番大きくなると考えられるノード密度 400 台/km² の状況で、最も負荷の大きかったノードの送受信パケットサイズを求め、これを 200 回のシミュレーションで平均化した値である。図 2.12, 図 2.13 の結果からノード一台当たりの送受信パケットに換算した値を図 2.16 の結果と比較すると、MA 起動により送信パケットサイズで最大 80%程度、受信パケットサイズでは最大 55%程度負荷が増えることが分かった。しかしながら、全体での制御メッセージが削減できたこともあり、1000

秒で 100~200 キロバイト程度のパケット送受信で済む。従って、特定のノードに極端に大きな負荷がかかることはないといえる。

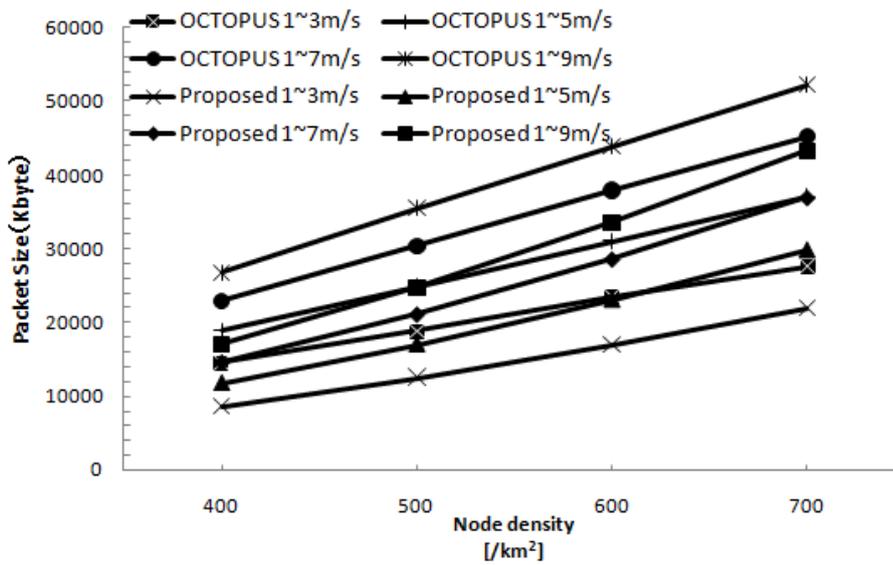


図 2.13 ノード速度と総送信パケットサイズ

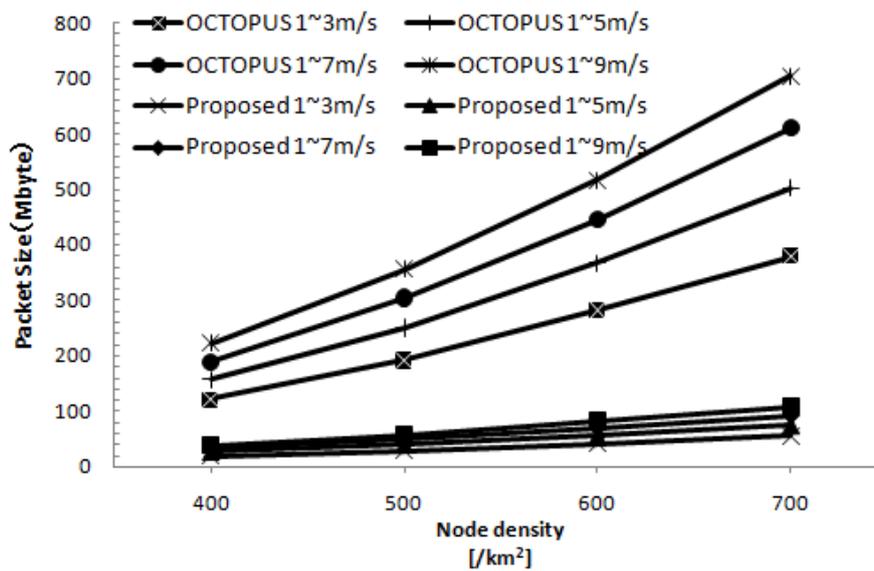


図 2.14 ノード速度と総受信パケットサイズ

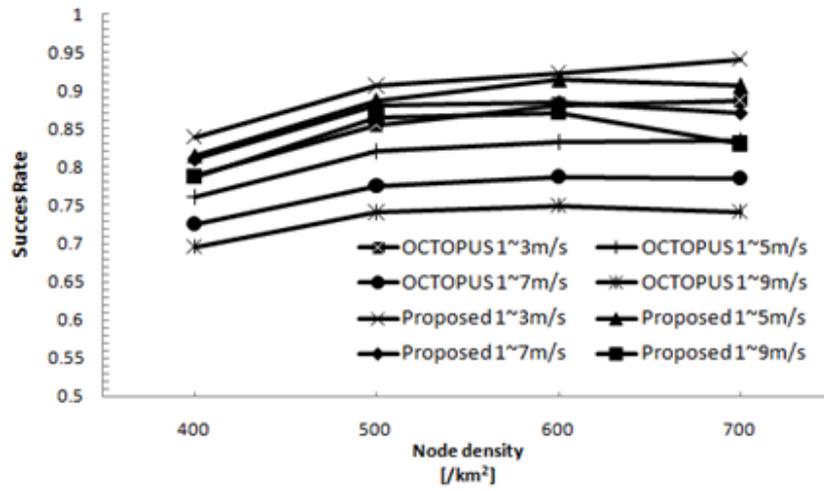


図 2.15 ノード速度と経路構築成功率

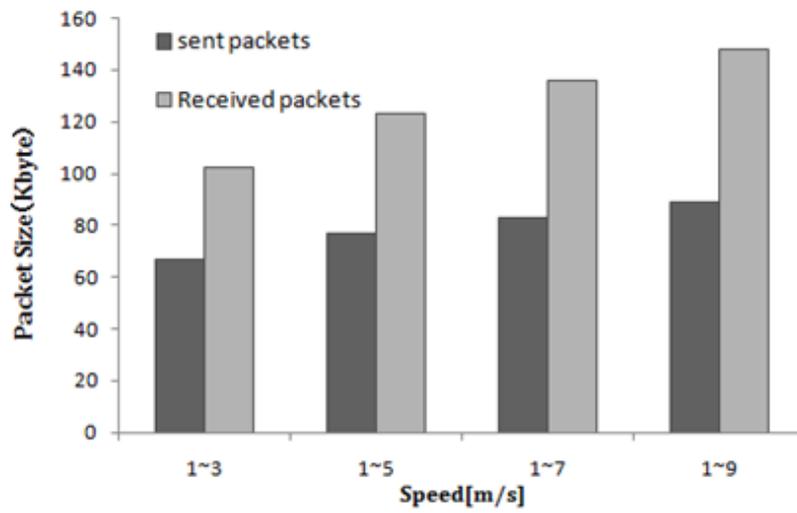


図 2.16 MA による最大負荷

2.6 2章のまとめ

本章では, アドホックネットワークにおける位置情報利用型ルーチングに対して, 制御パケット量の削減を目的に, MA を用いて位置情報を一元的に管理し経路構築を行うプロトコルを提案した. 性能評価の結果, 制御パケットを大幅に削減するとともに経路構築成功率も改善することができた. 今後の課題はノードの移動速度が大きい状況での経路構築成功率の改善と MA ノードに対するさらなる負荷分散があげられる.

参考文献

- [1] C. Perkins, E. Belding-Royer, S. Das " Ad hoc on- demand distance vector routing, " RFC3561, pp. 4089-4095, 2003.
- [2] IETF MANET Working Group, " The dynamic source routing protocol for mobile ad hoc networks (DSR), " draft-ietf-manet-der-10. txt, Internet-Draft, Jul. 2004.
- [3] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, "A distance Routing Effect Algorithm for Mobility (DREAM)," Proceedings of the International Conference on Mobile Computing and Networking, pp.76-84, 1998.
- [4] J. Urrutia, "Two Problems on Discrete and Computational Geometry," Proceedings of Japan Conference on Discrete and Computational Geometry, pp42-52, 1999.
- [5] R. Melamed, I. Keidar, and Y. Barel, "Octopus: A Fault-Tolerant and Efficient Ad-Hoc Routing Protocol," Proceedings of The 24th IEEE International Conference on Reliable Distributed Systems, pp.39-49, 2005.

第 3 章

3. モバイルエージェントを用いた経路干渉 考慮型マルチパスルーチング

3.1 はじめに

近年、無線通信の発展と無線端末の普及、利便性の向上、また災害時対策への期待に伴い、アドホックネットワークの研究が盛んに行われている。アドホックネットワークでは無線端末どうしが相互接続され、自身の電波が届く通信可能範囲から外れた端末に対しても、他端末を経由してパケットを転送することが可能である。

現在、アドホックネットワークに関する研究分野は広範囲にわたっている。アドホックネットワークには、有線ネットワークや有線インフラを併用する無線ネットワーク環境では存在しない特有の課題があり、それらを解決するために様々な研究がなされている。その中でもルーチングは大きな課題の1つである。アドホックネットワークには、その構成端末に移動端末が含まれることもあるため、頻りにネットワークトポロジが変化する。また、端末が蓄電池で動作する可能性も考慮し、通信に伴う消費電力を抑えることも必要となる。したがって有線ネットワークとは異なる独自の経路取得が必要であり、これまでに様々なルーチングプロトコルの提案がなされてきた。

代表的なアドホックネットワークにおけるルーチングプロトコルとして、AODV[1]、DSR[2]、OLSR[3]などが挙げられる。しかしながら、これらのプロトコルでは、常に各方式で適切であると判断された単一の経路によりパケットが転送される。そのため、特定の中継端末に負荷が集中することでパケット損失や通信遅延の増加を生じるなど、ネットワーク品質が劣化するという問題がある。なお、本論文における負荷とはデータの送受信より端末に生じる負担のこととする。そこでパケット転送効率の向上を目的としたマルチパスルーチングプロトコルの提案が盛んに行われている。代表的なプロトコルとしてEECA[4]、SMR[5]、AODVM[6]などがある。

本論文ではマルチパスの数を2つに限定する。また、同じパケットを複数の経路で送信することによる、到達信頼性向上を目的としたマルチパスルーティングではなく、個々のパケットをいずれかの経路に振り分けることによる、経路上の端末負荷軽減を目的としたマルチパスルーティングを扱う。したがって、到着順序の逆転が許容される TCP パケットを送信対象とする。このようなルーティングの中で、EECA は経路間の干渉を考慮した効率の良いプロトコルであることが示されている。

EECA では、2つの経路の距離を通信可能距離以上とすることで、各経路における通信が互いに干渉しないようにしている。しかしながら、ノードが移動しないことおよび全ノードが他ノードの位置情報を事前に完全に把握できることを前提条件としている。したがって、ノードが移動し、その位置が刻々と変化する状況では、EECA を適用できないという問題がある。これに対し、ノードが事前に全ノードの位置情報を得ることが困難な環境においても適用可能な経路干渉考慮型マルチパスルーティングが提案された[7]。この方式では、事前に他ノードの位置情報を把握しなくても EECA と同様に経路間干渉を考慮した効果的なマルチパスを得られる。しかしながら、経路取得のために多くの制御パケット送信が必要となり、ネットワーク全体のノードに大きな負荷を与える。また、得られるマルチパスが必ずしも適切でないという問題もある。

一方、ネットワーク負荷の軽減策として、単一経路のルーティング方式ではあるが、2章でも紹介したとおり Mobile Agent (MA) を用いた位置情報利用型ルーティングが提案されている。この方式では MA が各ノードの位置情報を一元的に管理して経路構築を代行することで経路構築に伴う制御パケット量の削減を図っている。

そこで本章では、MA を用いた位置情報利用型ルーティングをベースに MA が保持する位置情報を用いて効率的に経路間干渉を考慮したマルチパスを算出するルーティングプロトコルを提案し、スループットの向上を図る。

3.2 マルチパスルーチングプロトコル

現在主流のルーチングプロトコルでは、常に各方式で適切であると判断された単一の経路によりパケットが転送される。そのため、特定の中継ノードに負荷が集中することでパケット損失や通信遅延の増加を生じるなど、ネットワーク品質が劣化するという問題がある。そこでパケット転送効率の向上を目的としたマルチパスルーチングプロトコルの提案が盛んに行われている。代表的なプロトコルとして EECA, SMR, AODVM などがある。ここでは EECA と経路干渉考慮型マルチパスルーチングの具体的な動作について説明する。

3.2.1 EECA

EECA は経路間の干渉を考慮したマルチパスルーチングプロトコルである。この方式ではノードが移動しないネットワークを想定し、各ノードが事前に全ノードの位置情報を得られることを前提としている。

送信元ノード S があて先ノード D への経路構築を行いたい場合、まずルートリクエスト (RREQ) をブロードキャストする。RREQ には S と D の位置情報が含まれており、RREQ を受信したノードは、 S , D , および自身の位置情報を基に、 S と D を結ぶ線に対して自身が $R/2$ 以上離れた距離に位置しているかを判断する。ここで R は通信可能距離を表す。そして、もしも $R/2$ 以上離れた距離にいと判断した場合のみ RREQ を中継送信する。これを繰り返すことで、 S , D を結ぶ線を中心とした幅 R の帯状エリアを除外したノードのみで RREQ を D へ届けることができる。 D は帯状エリアのそれぞれにおいて最初に受信した RREQ に対してルートリプライ (RREP) を返信する。これにより 2 つの経路が構築されるが、これらの経路は R 以上の距離を保つことが保証されるため、互いの経路上での送受信が他の経路上の通信に影響を及ぼさない。これによりスループットの向上を図れる。

しかし、先述したように EECA では全てのノードが予め目的の位置情報を把握することが必要なことから、各ノードが移動できネットワークトポロジが絶えず変化する環境での適用が非常に難しい。

3.2.2 経路干渉考慮型マルチパスルーチング

ノードが移動する環境への適用が難しい EECA に対して、経路要求・応答パケットを工夫することで、予め送信先の位置情報を把握できなくても、経路間干渉を考慮したマルチパスを取得できるルーチングプロトコルが提案されている[7]。図 3.1 に経路要求 (RREQ) 送信例を示し、図 3.2 に経路応答 (RREP) 送信例を示す。ノード S がノード D への経路の取得を行う場合、S は自身の位置情報を含んだ RREQ を図 3.1 のようにネットワーク全体にフラッディングする。そして最初に D に届いた RREQ の経路が単一経路としては最適であるとみなし、D はその逆順経路を用いて S に RREP を送信する (図 3.2)。それと同時に、D は自身と S の位置情報を含んだ M-RREP をフラッディングする。M-RREP を受信した中継ノードは、EECA における RREQ 転送と同様に、D と S の位置情報から 2 つのノードを結ぶ仮想の直線を引き、直線と自身の距離が $R/2$ 以上であるかを計算する。そして、自身が $R/2$ 以上離れていない場合は M-RREP を破棄し、 $R/2$ 以上離れている場合は M-RREP の中継を行う。これを繰り返すことにより、分割されたエリアのそれぞれを用いて M-RREP を S へ送信することができる。上記の M-RREP 中継条件より、異なるエリアで転送される M-RREP の経路は互いに必ず R 以上離れることが保証される。これを利用して、S はそれぞれのエリアで最初に受信した M-RREP の転送経路の 2 つをマルチパスとする。パケット送信の際はマルチパスを優先して利用する。そしてマルチパスのうち 1 つでも使用できなくなった場合単一経路の利用に切り替え、同時にマルチパスの再要求を行う。

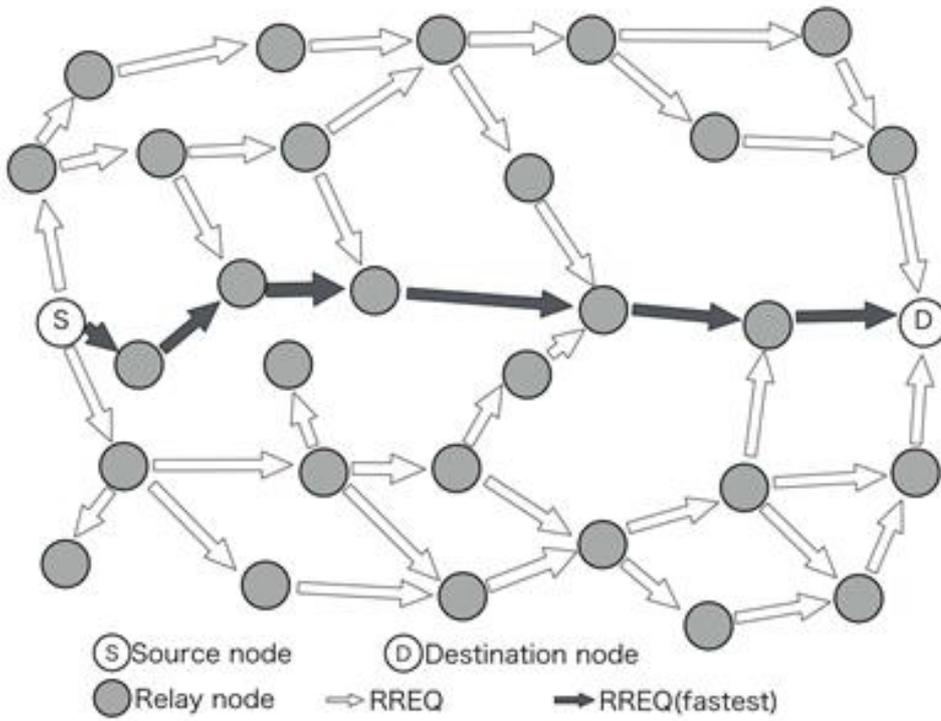


图 3.1 示例发送路由请求。

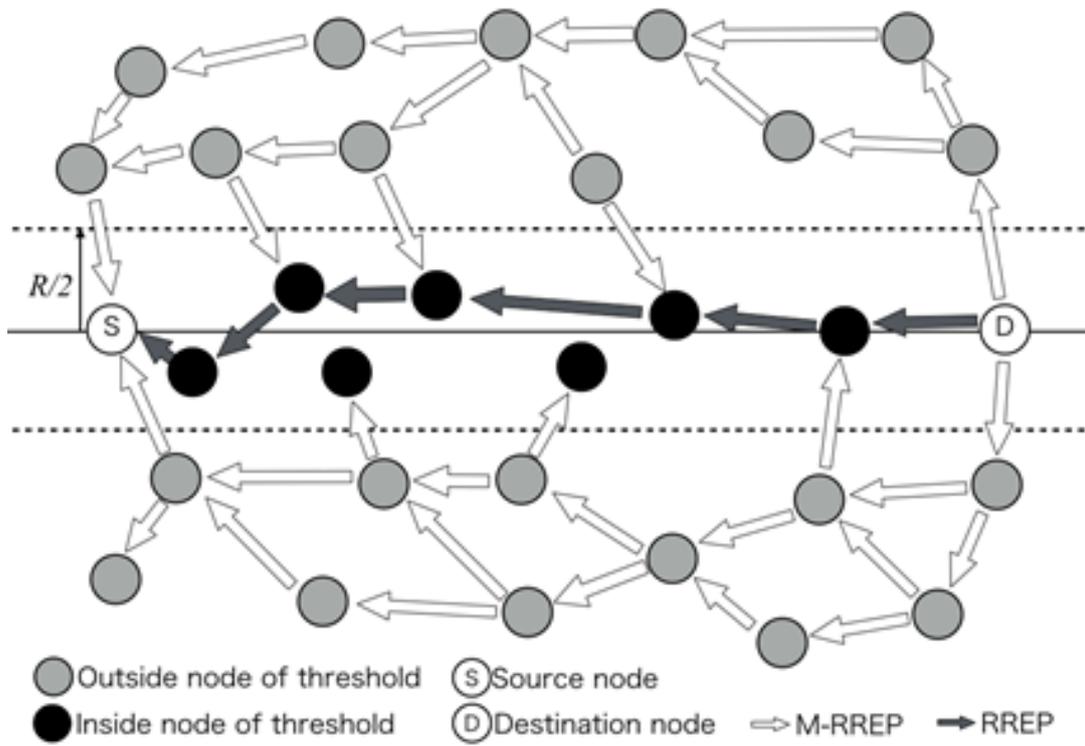


图 3.2 示例发送路由应答。

3.3 提案方式

2章で説明した通り，Mobile Agent (MA) を用いてネットワーク内に存在する全てのノードの位置情報を集中的に管理し，経路算出も MA に行わせることで，適切な経路を取得できる方式が提案されている．本章では，この方式をベースに MA が保持する位置情報を用いて効率的に経路間干渉を考慮したマルチパスを算出するルーティングプロトコルを提案し，スループットの向上を図る．MA とは，ネットワークに接続されたコンピュータ間を移動しながら様々な処理をするプログラムである．提案方式における MA の動作は2章の方式と同様で2.3.1で説明したとおりであり，MA に全ノードの位置情報管理と経路算出を行わせる．その機能は位置情報更新，経路情報取得，経路算出・ルーティングの3つから成り立つ．位置情報更新については2章の2.3.2，経路情報取得については2.3.3と同様のため説明は省略する．

3.3.1 経路算出

経路要求パケットを受信した MA は単一経路と1組の二重経路の算出を試みる．これら3つの経路を含めた経路応答パケットは，経路要求パケット転送経路の逆順により送信元ノードへと送信される．なお，二重経路が算出できなかった場合には経路応答パケットには単一経路のみが含まれる．

単一経路算出では，まず MA は保持している全ノードの位置と1ホップ通信可能平均距離の情報を利用してノード間の距離が通信可能平均距離の90%以下の場合無線リンクがあると仮定し，ネットワークトポロジを構築する．このネットワークトポロジから各無線リンクのコストを1としてダイクストラ法を用い，最もコストの小さい経路を求める．リンクコストを1としているため，この経路は最少ホップ数の経路であることが保証される．もし同じホップ数の経路が複数見つかった場合にはランダムでどれかを選択し採用する．この方法ではスループットを向上させるため，最小ホップとなるように経路を作成するが，リンクの有無を仮定するための基準値を通信可能範囲の90%とすることで各ホップのリンク品質の劣化を抑え，経路全体の品質を高く保つことが可能である．

二重経路の算出では、まず、ノード S から D へと仮想的な直線を引き、この直線と平行でかつ通信可能半径 $R/2$ 離れた直線を 2 本求める。この 2 本の直線を境界として、図 3.3 のように 2 つの二重経路エリアを設ける。そして、それぞれの二重経路エリア内のノードのみを用いて単一経路の場合と同様にダイクストラ法による算出を行う。こうして得られた 2 つの経路は互いに R 以上離れることが保証されるため経路間干渉が生じない。

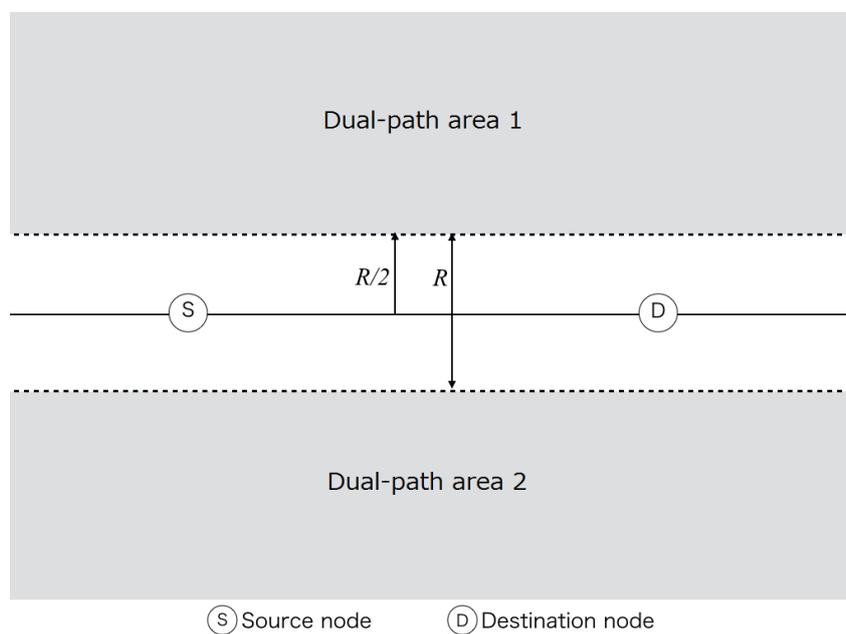


図 3.3 二重経路エリア

3.3.2 データフォワーディング

送信元ノードがあて先ノードへと通信をする際に MA から二重経路が取得できた場合は二重経路を用いて通信を行う。もし二重経路が取得できない場合には単一経路を用いる。位置情報利用型ルーチングの多くでは、あて先ノードの位置情報のみを利用し、Greedy フォワーディング等によりパケット送信を行うが、本方式では MA による算出経路を優先して利用する。具体的には DSR のようにパケットに経路情報を含め、中継ノードが次ホップノードを決定する際にはこの経路情報を最初に利用する。また、単一経路あるいは二重経路のどちらを用いて通信する場合にも、経路情報に記載のノードが次ホップノードとして存在しない場合には、あて先の位置情報を利用した GEDIR[8]によるフォワーディングに切り替える。そして経路情報に記載のノードが見つかり次第再び経路を利用したルーチングに戻す。これによりノードの移動により経路が局所的に使用不能になった場合にも通信を続行することができる。なお、1 つの TCP 通信において単一経路と二重経路の切り替えは行わない。

3.3.3 効率的なパスの割り当て

二重経路を用いてパケットを送信する際、経路ごとのパケット転送効率（ここでは経路の品質と呼ぶ）も考慮する必要がある。二重経路における 2 つの経路は、中継ノードが異なり、ホップ数も同じとは限らない。したがって、例えば片方の経路（経路 1 とする）内に負荷が集中している中継ノードが多く存在し、残りの経路（経路 2 とする）の全ての中継ノードの負荷が軽くホップ数も少ないような場合、経路 1 より経路 2 を多く使ってパケットを送信する方が、明らかに効率が良い。

そこで提案方式では、送信元ノードがパケットを送信する場合、二重経路における各経路の品質に応じてそのパケットの転送経路を決定する。ここでは TCP パケット送信を前提としていることを利用し、単純に TCP パケットの再送を経路品質の指標に用いる。

まず初期状態として経路 1 と経路 2 の選択確率をそれぞれ 50%とし、送信元ノードは各パケットの転送経路を選択確率に従いランダムに決定する。同時にそのパケットと転送経路の関係を送信成功が確認されるまで記憶しておく。もしもパケットの損失による ACK 受

信タイムアウトやパケット到達順序変更に伴う重複 ACK 受信等のため TCP パケットの再送が発生した場合、送信元ノードは再送対象パケットの転送経路を調べる。そして失敗した経路の選択確率を 10% 下げる。例えば送信失敗した転送経路が経路 1 であった場合には、送信元ノードは経路 1 の選択確率を 50% から 40% に下げる。これにより経路 2 の選択確率は 60% となる。なお、選択確率の変動幅の 10% は事前検証により最も良好な性能を示したものを採用した。これを繰り返すことでパケット再送が行われやすい経路の選択確率は下がっていき、パケット再送が行われにくい経路で多くのパケットを送信することができる。これにより再送制御の回数が減り TCP 通信の効率が上がるため、スループットを向上させることが可能である。

3.3.4 現在位置情報の推測

MA による経路算出の精度はノード位置情報の精度に依存する。そこで過去のノードの位置情報時系列から移動速度とベクトルを計算し、それらの情報を基に現在のノードの位置情報を推測する

あるノードの i 番目の位置情報として $\{x_i, y_i, t_i: i = 1, 2, \dots\}$ をサブセットとして定義する。ここでの x_i, y_i, t_i はそれぞれ X 座標, Y 座標, 時間を意味する。そして MA は連続した二つの差分 x_v, y_v, t_v を計算する。(3.1) ~ (3.3) では、最新の情報が x_n, y_n, t_n であると仮定した場合、これらの変動値を使用し、MA は式からノードの位置情報を推測する。(3.4) ~ (3.5) での x_e と y_e は推測された X 座標と Y 座標であり、そして t_{now} はそれぞれの現在時刻である。この推測位置情報を使用してダイクストラ法を用い経路を構築する。

$$x_v = x_n - x_{n-1} \quad (3.1)$$

$$y_v = y_n - y_{n-1} \quad (3.2)$$

$$t_v = t_n - t_{n-1} \quad (3.3)$$

$$x_e = x_n + x_v(t_{now} - t_n)/t_v \quad (3.4)$$

$$y_e = y_n + y_v(t_{now} - t_n)/t_v \quad (3.5)$$

3.4 シミュレーションによる性能評価

計算機シミュレーションにより提案方式と既存のルーチングプロトコルとの性能を比較し評価する。シミュレータは著者らの研究グループで開発したものをを用いる。電波伝搬モデルは two-ray ground としており、パケットの衝突や干渉によるパケットロスについても考慮する。またノード間における理想的な接続性が保たれにくい状況での評価も行うためシャドウイング環境を考慮した。シャドウイングのパラメータとして、距離減衰係数を 2 とし、電力の変動付加値を平均 0、標準偏差 4db の正規分布に従う変数とする。また平均電力は 100m 離れたノードが受信成功する値とする。移動モデルは Random way point とし、乱数によりノードの初期配置や移動方向、速度の変更を行った。また、シミュレーション時間は 2000 秒とし、複数回行った。

シミュレーションパラメータを表 3.1 に示す。MAC Layer には IEEE802.11g を用い、通信帯域を 54Mbps とする。シミュレーション範囲は一辺 1km の正方形とし各ノードの通信範囲を 100m とする。ノード数は 400~800 台とし密度が高い環境を想定した。比較対象は AODV, EECA, フラッドイングを用いた二重経路ルーチング : Conventional Dual-path [7], 提案方式の基となる MA を利用した位置情報利用型ルーチング : Single-path using MA, 提案方式 : Proposed Dual-path とする。また、評価の参考として MA からあて先の位置情報のみを取得して GEDIR により通信する方式: GEDIR using MA と提案方式において経路の選択確率を変動させない方式: Proposed Dual-path2 との比較も行った。

3.1 で述べたように、本研究で想定する二重経路通信ではパケット到着順序の逆転が頻繁に生じる。そこで順序制御を行える TCP をデータ通信の対象として用いる。具体的にはランダムなタイミングでランダムに 2 つのノードを選択し、そのノード間に TCP フローを発生させ、スループットを測定した。TCP フローでは、TCP reno に基づくパケット送信が行われ、ACK を利用した再送制御や輻輳制御も行う。TCP スループットは、1 フローあたりの合計データサイズを、送信元ノードによる先頭パケット送信から最終パケットに対する ACK 受信までの所要時間で割ったものと定義する。さらに、本シミュレーションの結果として示す TCP スループットは複数回のシミュレーション中に発生した 5000 フロー分のスループット結果を集計し平均値を取ったものとした。また、1 ホップの TCP フローについてはルーチングを行わないため、この 5000 フローには含まれないようにした。さらに、ス

ループットはノードの移動パターンと TCP フローの送受信ノードペアの決定がランダムに行われることによりばらつきが生じるため、各結果に対する標準偏差も示した。また、TCP フローを発生させる頻度を通信頻度と定義する。ノード密度の影響を検証するため通信頻度を 0.1flow/s に固定してノード数を変化させた場合と、ネットワークの負荷による影響を検証するためノード数を 500 台に固定して通信頻度を変化させた場合の測定を行った。どちらに関してもノードの平均速度が 1~3m/s の場合と 1~10m/s の場合で評価し、ノードの移動速度による性能への影響も検証した。スループット以外の結果についてはシミュレーション時間ごとに集計を行い、複数回行った結果の平均値をグラフに示した。MA を用いた単一経路ルーチングと提案方式における位置情報更新の閾値は 20m とした。また EECA については、ノードが移動する環境で評価するため、各ノードが 20m 移動するごとに自身の位置情報をフラッディングし、ネットワーク内の全ノードに広告することとした。GEDIR により通信する方式 (GEDIR using MA)における MA の動作は経路応答のかわりにあて先ノードの位置情報を返すという点以外は MA を用いた単一経路ルーチングと同じとした。

続いて各ルーチングにおける制御パケットサイズについて説明する。まず AODV では RREQ および RREP パケットサイズをそれぞれ 24 バイト、20 バイトとする。EECA では RREQ を 28 バイト、RREP を 12 バイトとする。また、TCP パケットには経路情報として経路の中継ノード数×4 バイトを付加する。そしてフラッディングを用いた二重経路ルーチングでは、RREQ を 24 バイト、RREP を 20 バイト、M-RREP を 28 バイトとする。また、TCP パケットには EECA と同様に経路情報として経路の中継ノード数×4 バイトを付加する。MA を利用したルーチングでは制御パケットサイズが固定ではない。そこで、基本サイズを 12 バイトとし、ノード 1 台当たりのノード ID フィールドサイズを 4 バイト、位置情報フィールドサイズを 12 バイト (座標情報 8 バイト、時刻情報 4 バイト) と仮定した。そして基本サイズに必要なフィールド分を追加したものを制御パケットサイズとする。なお MA のデータサイズについては起動に必要な MA 本体のサイズを 30 キロバイトとし、これに位置情報テーブルサイズ (ノード数×12 バイト) を加えた値とした。MA 利用ルーチングにおける具体的な制御パケットサイズを表 3.2 に示す。また、TCP パケットには経路情報として経路の中継ノード数×4 バイトとあて先ノードの位置情報の 8 バイトを付加する。

表 3.1 シミュレーションパラメータ

MAC Layer	IEEE802.11g
Channel rate	54Mbps
Simulation range	1000m ²
Transmission range	100m
Mobility model	Random way point
Speed	1~10m/s, 1~3m/s
Number of nodes	400, 500, 600, 700, 800
Flow	TCP
TCP flow size	1Mbyte

表 3.2 制御パケットサイズ

GEDIR Request	12[byte]
GEDIR Reply	20[byte]
Location Update	24[byte]
RREQ	$20+(h_1 \times 4)$ [byte]
RREP (Single-Path)	$24+(h_2 \times 4)+(h_a \times 4)$ [byte]
RREP (Dual-Path)	$24+4 \times (h_2+h_a+h_b+h_c)$ [byte]
MA Migration Request	8[byte]
MA Migration Reply	20[byte]
MA Migration	$30000+(n \times 12)$ [byte]

h_1 : 現在までのホップ数

h_2 : 要求受信時のホップ数

h_a : MA が算出した単一経路のホップ数

h_b : MA が算出した二重経路 1 のホップ数

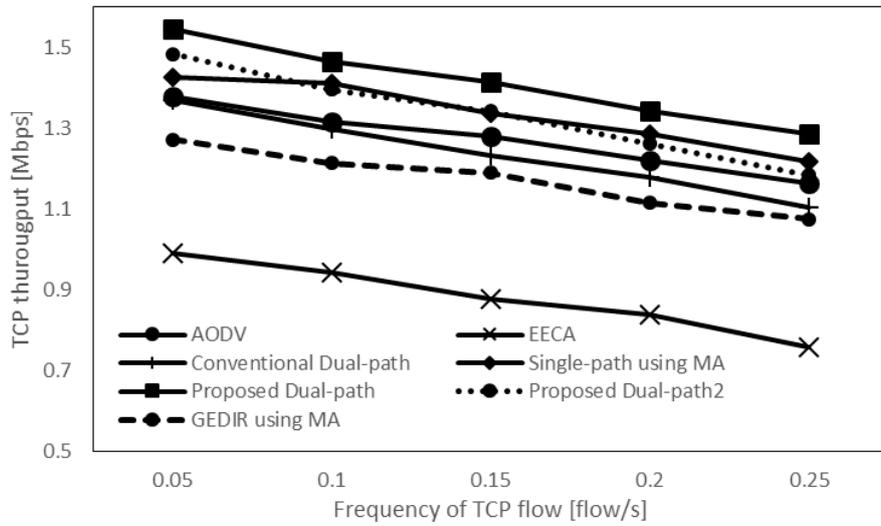
h_c : MA が算出した二重経路 2 のホップ数

n : MA が情報を保持するノード数

3.5 シミュレーション結果・考察

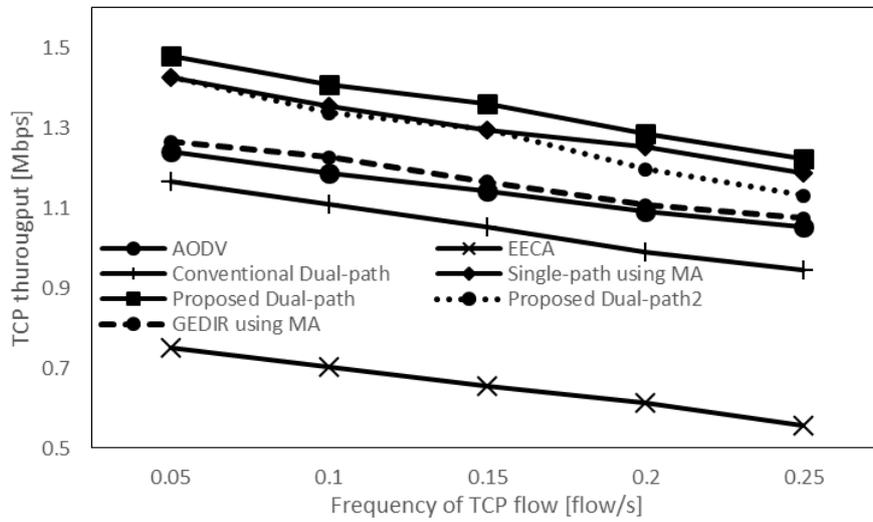
図 3.4 は 500 台のノードが平均速度 1~3m/s で移動する状況で通信頻度を 0.05~0.25flow/s と変化させながら比較を行った場合のスループットである。グラフに平均値，表に標準偏差を示す。図より従来方式と比較して全ての通信頻度において提案方式のスループットが高いことが確認できる。これは MA が効率的に位置情報を収集でき，その位置情報を基に効率的な二重経路を構築することができたためである。また，スループットは各プロトコルの制御により生じる負荷の影響も受けていると考えられる。例えば，EECA では各ノードが自身の位置情報を通知するためにネットワーク全体にフラッディングを行っているため，スループットが他の方式と比べて極端に低い。フラッディングを用いた二重経路ルーチングでは RREP, M-RREP の送信にフラッディングを用いるため，これらの負荷がスループットを低下させていると考えられる。AODV もフラッディングによる RREQ の送信が負荷となる。一方，提案方式を含む MA を使用する方式では制御パケットの送信は MA と各ノード間のユニキャスト通信となるため負荷が小さい。なお，MA を使用する方式の中で GEDIR using MA のみスループットが低い理由は制御パケットの負荷が原因ではなく，ノード台数が 500 台の場合には GEDIR のみでは効率的な経路が作られないためであると考えられる。また，各方式ともに通信頻度が高くなるにつれてスループットが低くなる傾向にある。これは通信頻度が高い状況下では同じタイミングで複数の TCP 通信が行われているためパケットの衝突による損失確率が大きくなったものと考えられる。TCP の同時存在フロー数については，ランダムな TCP 発生タイミングのため変化するが，提案方式におけるノード数 500 台，移動速度 1~3m/s，通信頻度 0.25flow/s の状況下で最大 9 フローが同時に存在していることを確認した。

図 3.5 は図 3.4 と同様の環境でノードの移動速度を 1~10m/s としてノードが高速で移動する環境で比較を行った結果である。この図からも提案方式のスループットが高いことが確認できる。さらに，図 3.4 と同様に各プロトコルの制御による負荷の影響が生じていることも確認できる。また，提案方式と MA を用いた単一経路ルーチング，GEDIR using MA では 1~3m/s の場合と比較してスループットに大きな差はないが，他の方式ではスループットが大きく低下している。



	0.05	0.1	0.15	0.2	0.25
EECA	0.5071	0.511131	0.482627	0.493497	0.457676
Conventional Dual-path	0.521843	0.508039	0.529095	0.526836	0.524925
AODV	0.586895	0.572647	0.58925	0.576919	0.569344
Single path using MA	0.595394	0.654989	0.639908	0.637578	0.648904
Proposed Dual-path	0.551713	0.563039	0.57578	0.553102	0.580009
Proposed Dual-path2	0.541047	0.524294	0.549542	0.550654	0.541472
GEDIR using MA	0.713816	0.680355	0.688915	0.688543	0.676409

図 3.4 通信頻度に対するスループット (移動速度 1~3m/s)



	0.05	0.1	0.15	0.2	0.25
EECA	0.456375	0.439244	0.426826	0.407645	0.388569
Conventional Dual-path	0.53927	0.542935	0.545658	0.527999	0.528232
AODV	0.61993	0.613252	0.611507	0.599929	0.592899
Single path using MA	0.617058	0.616664	0.595849	0.624961	0.628077
Proposed Dual-path	0.533826	0.533276	0.551042	0.546848	0.560841
Proposed Dual-path2	0.531835	0.499419	0.537112	0.555059	0.53401
GEDIR using MA	0.633438	0.647301	0.638537	0.635285	0.635173

図 3.5 通信頻度に対するスループット (移動速度 1~10m/s)

これは、提案方式と MA を用いた単一経路ルーチング, GEDIR using MA 以外の方式でノードが高速移動する環境では、経路が頻繁に切断され再構築が多く発生するためである。それに対して提案方式と MA を用いた単一経路ルーチングでは経路が切断された場合にも GEDIR に切り替わりあて先ノードの位置情報を元にフォワーディングすることが可能であるため、スループットが大きく低下しない結果となったと考えられる。

さらに提案方式が GEDIR に切り替わることの影響を検証するため、図 3.6 に提案方式において転送時に GEDIR に切り替わる割合を示した。これは転送回数の総数に対する切り替わり回数の比率とする。ノード台数は 500 台である。この結果より、実際に GEDIR への切り替えが行われていることが示された。また、移動速度が上がると GEDIR に切り替える頻度が上がることが示され、先に述べた考察が裏付けられた。またこの切り替わりが、提案方式が EECA やフラッディングを用いた二重経路ルーチングと経路作成の基本的な方針が同じであるのかかわらずスループットが高くなる一因であると考えられる。また、もう一つの要因としては、前述のとおり各方式の制御によるネットワークへの負荷が影響していると考えられる。

さらに提案方式の性能を詳細に評価するため位置情報更新パケットの到達率とパスの使用内訳を図に示す。ノード台数は 500 台である。図 3.7 の位置情報更新パケットの到達率は各ノードが送信した位置情報更新パケット数に対する MA が受信した位置情報更新パケット数の合計の比率である。この結果から、移動速度や通信頻度が位置情報更新パケットの到達率に与える影響は殆どないことが示され、通信頻度が高い場合にも MA が保持する各ノードの位置情報の精度が保たれていることが確認できた。

図 3.8 にはノード台数 500 台で移動速度 1~3m/s の時の経路の使用内訳を示す。これはシミュレーション期間中にパケットがどの経路を使用して送信されたかを比率で表したものである。経路選択時に選択確率の高い経路が使われた場合 Path1, 選択確率が低い経路が使われた場合 Path2, 単一経路が使われた場合 Single としてカウントし、それぞれの値を比率で表した。この図からどの通信頻度においても経路の使用内訳は大きく変わらないことが示された。また、二重経路を使用する場合には選択確率の高い Path1 が多く選択されていることが確認できる。

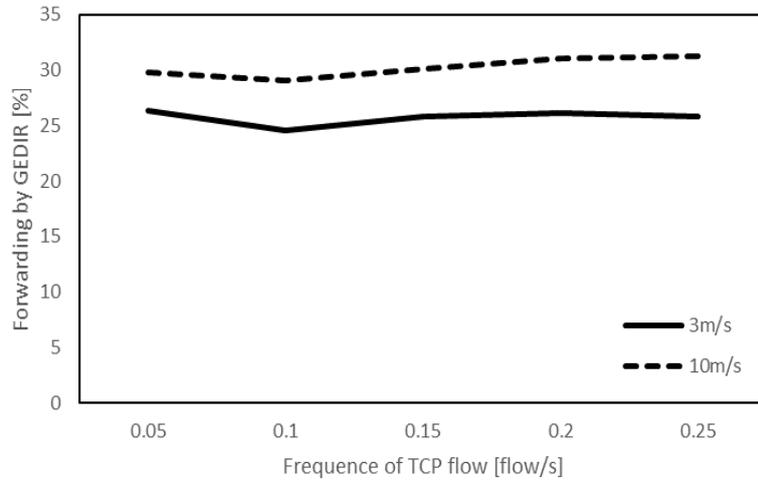


図 3.6 GEDIR に切り替わる確率

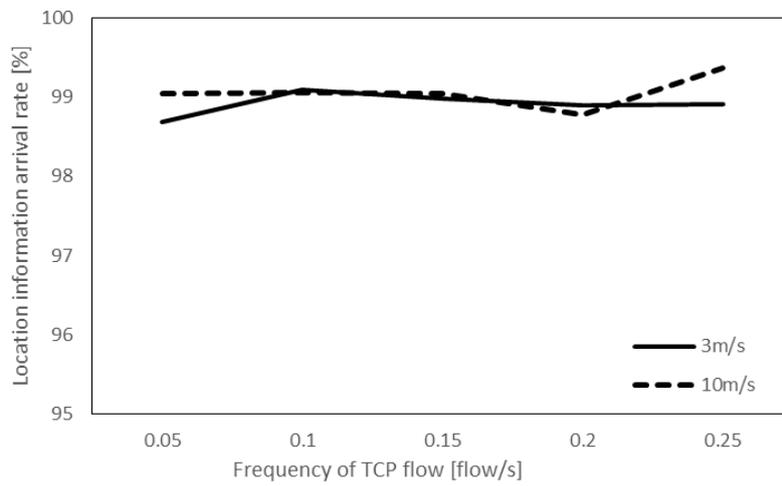


図 3.7 通信頻度に対する位置情報到達率

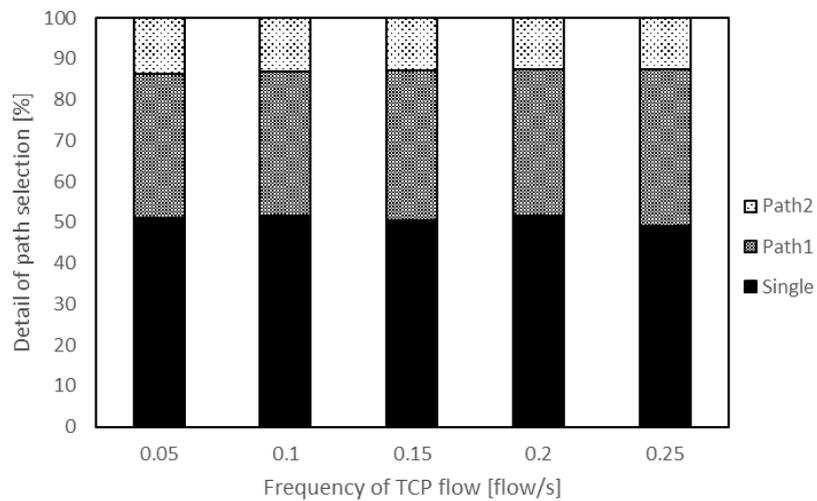


図 3.8 通信頻度に対するパス使用内訳

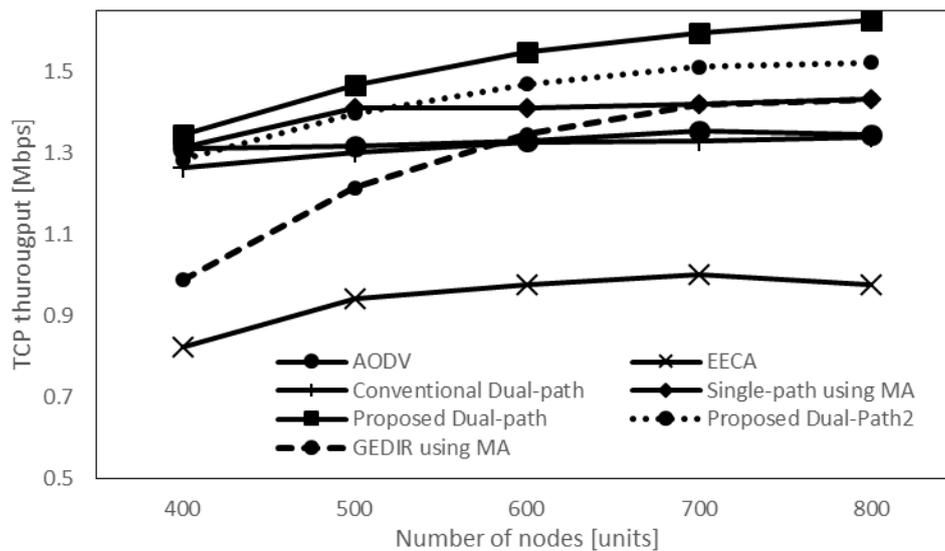
また、図 3.4, 図 3.5 において経路の選択確率を変動させる場合 (Proposed Dual-path) とさせない場合 (Proposed Dual-path2) のスループットを比較することで、経路の選択確率を変動させることがスループットを向上させていることを確認できる。

図 3.9 は通信頻度を 0.1flow/s に固定して、移動速度が 1~3m/s のノードを 400~800 台まで変化させた際のスループット平均と標準偏差である。ノード密度が低い場合は大きな差は無いが、ノード密度が高くなるにつれて提案方式の優位性が表れている。また、フラッディングを用いた二重経路ルーチングは提案方式と同等の経路が作成されるが、GEDIR への切り替えが行われなため、提案方式と比べてスループットが低い。

図 3.10 に提案方式における経路の使用内訳を示す。この図からノード密度が高くなるにつれて二重経路の使用頻度が上がることが確認でき、図 3.9 とあわせてみると二重経路の使用率がスループットに影響していることが確認できる。ノード密度が十分でない場合二重経路を構築できず単一経路の使用頻度が高くなるため MA を用いた単一経路とスループットが同じ程度になるが、二重経路の使用頻度が上がるにつれてスループットの差が広がっていくことが確認できた。また、図 3.9 においても経路の選択確率を変動させる場合 (Proposed Dual-path) とさせない場合 (Proposed Dual-path2) のスループットを比較することで、経路の選択確率を変動させることがスループットを向上させていることを確認できる。さらに、TCP の再送制御回数の平均についてノード数 500 台、移動速度 1~3m/s、通信頻度 0.1flow/s の状況下で調査したところ、経路の利用割合を変動させた場合 1 フローあたり 29.97 回、変動させない場合 39.74 回となることが確認でき、実際に再送制御回数が減少していることが示された。

図 3.11 は図 3.8 と同じ環境下でノードの移動速度を 1~10m/s としてシミュレーションを行った結果である。この図からもノードの密度に関係なく、提案方式のスループットが従来方式と比較して高くなっていることが確認できる。また、この結果においても提案方式と MA を用いた単一経路ルーチング、GEDIR using MA は他の方式と比べて移動速度が上がったことによる性能低下が少ないことがわかる。これは通信頻度別の考察で述べた理由と同様に経路切断時に GEDIR に切り替わることが影響していると考えられる。また、フラッディングを用いた二重経路ルーチングは先述のとおり経路構築に必要な負荷が大きいため、経路再構築が生じやすい高速移動環境ではよりスループットが低下している。

図 3.12 にノード台数別の MA における位置情報到達率を示す。通信頻度は 0.1flow/s とした。この図からノード台数が 400 台のときだけ位置情報到達率が低くなっていることがわかる。これもノード台数が 400 台のときに提案方式のスループットが低下する原因であると考えられる。



	400	500	600	700	800
EECA	0.522754	0.511131	0.473938	0.459042	0.43387
Conventional Dual-path	0.537645	0.508039	0.499749	0.479734	0.469558
AODV	0.595199	0.572647	0.570671	0.587014	0.574105
Single path using MA	0.649969	0.654989	0.63184	0.628937	0.610781
Proposed Dual-path	0.585549	0.563039	0.537992	0.498694	0.508606
Proposed Dual-path2	0.588609	0.524294	0.50308	0.510946	0.491646
GEDIR using MA	0.761776	0.680355	0.667393	0.644567	0.627205

図 3.9 ノード数に対するスループット (移動速度 1~3m/s)

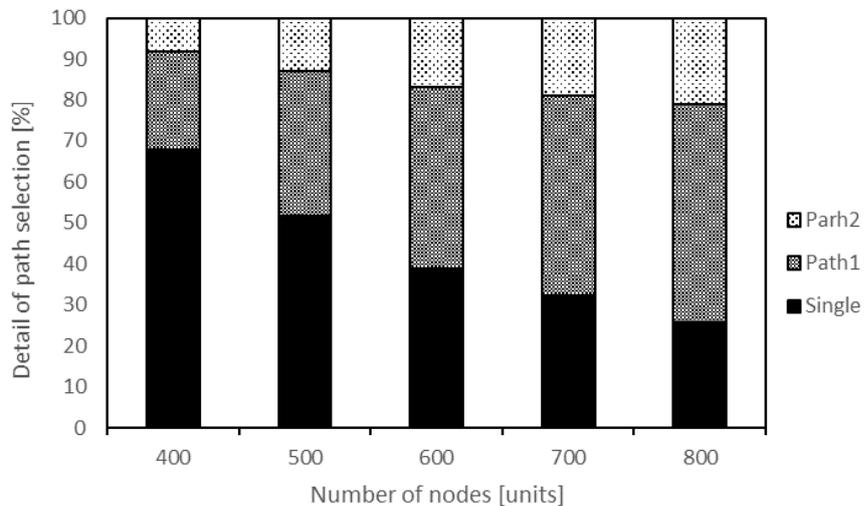
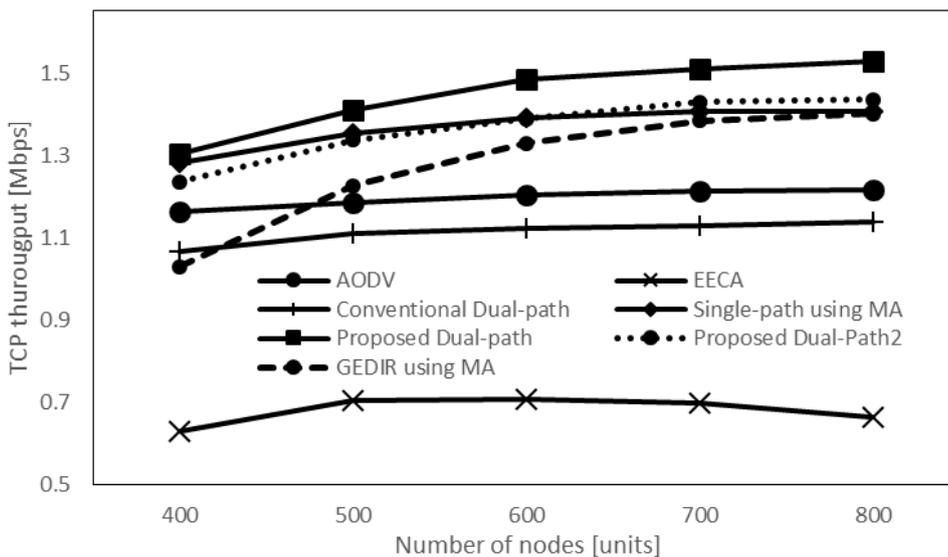


図 3.10 ノード数に対するパス使用内訳



	400	500	600	700	800
EECA	0.464181	0.439244	0.405753	0.385692	0.353313
Conventional Dual-path	0.560942	0.542935	0.539913	0.516303	0.515896
AODV	0.6109	0.613252	0.611922	0.624233	0.619421
Single path using MA	0.607607	0.616664	0.615313	0.632076	0.62544
Proposed Dual-path	0.573979	0.533276	0.51611	0.488607	0.506111
Proposed Dual-path2	0.559446	0.499419	0.4913	0.497078	0.473809
GEDIR using MA	0.710583	0.647301	0.646506	0.650603	0.640088

図 3.11 ノード数に対するスループット (移動速度 1~10m/s)

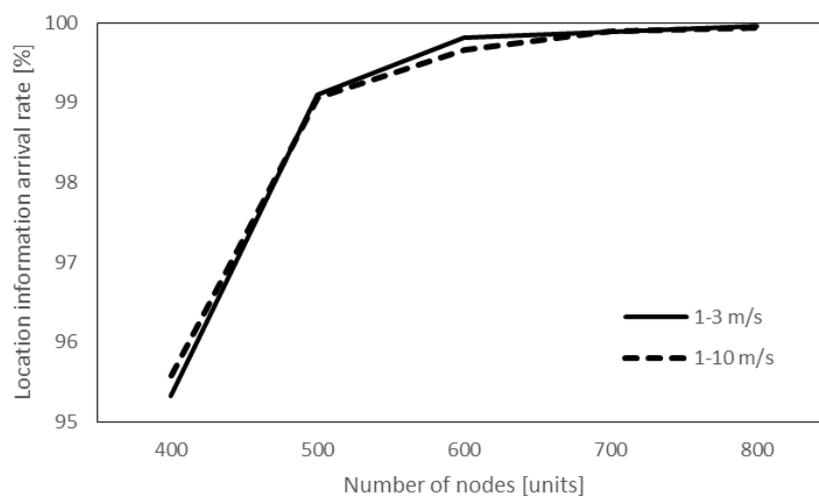


図 3.12 ノード数に対する位置情報到達率

図 3.13 でプロトコルの動作のために TCP パケットに付加されるオーバーヘッドを評価した。図で示す値はシミュレーション期間中に発生した TCP パケットに付加されるオーバーヘッドの合計とした。ノードの移動速度は 1~3m/s, 通信頻度を 0.1flow/s とした。このオーバーヘッドは AODV と GIDER using MA 以外の方式の場合は経路情報が付加されるため経路のホップ数の影響を受けると考えられる。提案方式と MA を用いた単一経路ルーチングを比べると二重経路を使うことで経路のホップ数が大きくなるため提案方式のオーバーヘッドが多くなっている。また, AODV ではオーバーヘッドがないため 0 となり, GIDER using MA は経路情報を持たないため小さい値を示している。図 3.9 のスループットの評価と合わせてみると, 提案方式や MA を用いた単一経路ルーチングはオーバーヘッドが大きいにもかかわらずオーバーヘッドのない AODV より高いスループットを示しており, オーバーヘッドによる性能低下の影響を受けても優位性が保てていることがわかる。

以上の結果より今回評価したすべてのシミュレーション環境において提案方式のスループットが優れていることが示された。また, ノードの移動速度が早い状況下では提案方式と MA を用いた単一経路ルーチングは他の方式と比べ, 性能低下が少ないことが確認された。この理由は先に述べたとおり, 経路が切断された場合に GEDIR に切り替わるためであると考えられる。一方 GEDIR を用いない方式では経路が切断されるたびに経路を再構築する必要があることから, ネットワークに大きな負荷を与え性能低下を招いている。

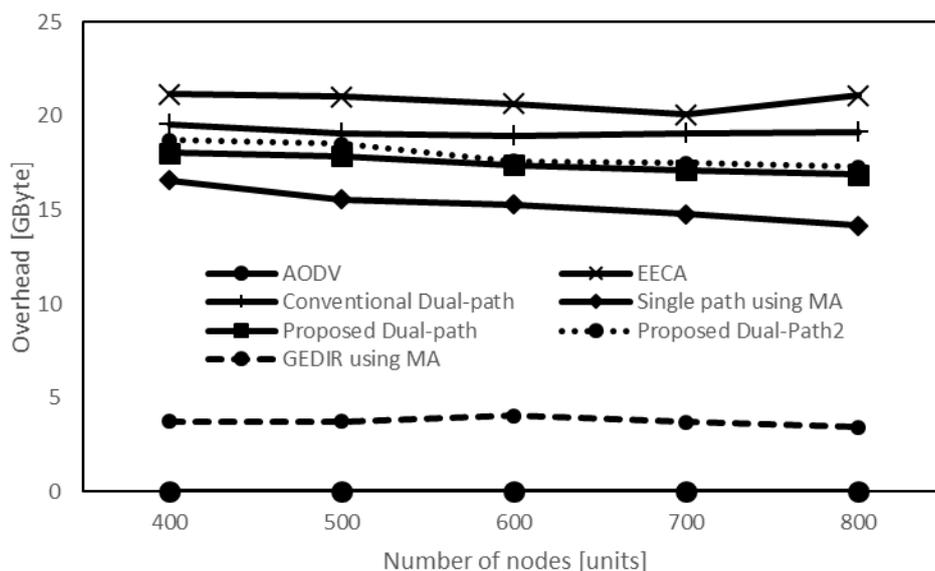


図 3.13 ノード数に対するオーバーヘッド

3.6 3章のまとめ

本章ではスループットの向上を目的とし、MAを用いることでフラッディングすることなく効率的に経路間干渉を考慮した二重経路を構築するルーチングプロトコルを提案した。計算機シミュレーションによる性能評価から提案方式の優位性を示すことができた。提案方式では二重経路を構築することで、ノードの台数が少ない場合にはスループットに大きな差はないが、それ以外の状況においては提案方式のスループットが大きく向上することが示された。さらにMAを利用することで、経路取得のための負荷が小さいという利点があり、これもスループット向上に貢献している。さらに、経路が切断された場合にも一時的にGEDIRに切り替わるため、他の方式に比べ移動速度が上がることによる性能低下が小さいことが確認できた。今後の課題としては、より効率の良い二重経路の作成方法の検討が挙げられる。また、二重経路の選択確率の変動についても効率化を図りたい。

参考文献

- [1] C. Perkins, E. Belding-Royer, S. Das " Ad hoc on- demand distance vector routing, " RFC3561, pp. 4089-4095, 2003.
- [2] IETF MANET Working Group, " The dynamic source routing protocol for mobile ad hoc networks (DSR), " draft-ietf-manet-der-10. txt, Internet-Draft, Jul. 2004.
- [3] T.Clausen, "Optimized Link State Routing Protocol" IETF, <https://www.ietf.org/rfc/rfc3626.txt>, Oct. 2003.
- [4] Z. Wang, E. Bulut and B. K. Szymanski "Energy Efficient Collision Aware Multipath Routing for Wireless Sensor Networks," ICC09 International Conference on Communication, pp. 1-5, Jan. 2009.
- [5] S. J. Lee, M. Gerla, "Split Multipath Routing with Maximally Disjoint Paths in Ad hoc Networks," IEEE International Conference on Communications, pp. 3201-3205, 2001.
- [6] Y. Ge, G Wang, Q. Zhang and M Guo, "Multipath Routing with Reliable Nodes in Large-Scale Mobile Ad-Hoc Networks," IEICE transactions on information and systems 92(9), pp. 1675-1682, 2009-9-1.
- [7] 小松 辰成, 塩川 茂樹, "経路間の干渉を考慮したマルチパスルーティングにおけるスループットの改善" 信学技報, AN2011-55, pp. 1-6, 2012 年 1 月.
- [8] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, "A distance Routing Effect Algorithm for Mobility (DREAM)," Proceedings of the International Conference on Mobile Computing and Networking, pp.76-84, 1998.

第 4 章

4. モバイルエージェントを用いた隣接ノード情報利用型ルーチング

4.1 はじめに

現在の無線通信技術は基地局やアクセスポイントなどの固定された通信基盤を必要とするものが多く、基盤が存在しない場合や、大規模な災害などにより物理的な通信網途絶が発生した場合、ネットワークの稼働率の大幅な低下が想定される。そのような際、固定の通信基盤を用いずネットワークを構成することができるアドホックネットワークの利用が有効とされている。アドホックネットワークとは、インフラストラクチャを用いず、マルチホップ通信を行う無線ノードによって構成されるネットワークである。アドホックネットワークでは、電波が届かず直接通信できないノード間の通信を、途中に存在するノードが中継ノードとして機能することで実現する。

アドホックネットワークの技術課題の一つとしてルーチングが挙げられ、多くのプロトコルが提案されている。ルーチングプロトコルは大まかにプロアクティブ型とリアクティブ型に分類され、代表的なプロトコルとしてプロアクティブ型では OLSR[1]等、リアクティブ型では DSR[2], AODV [3]等が提案されている。現在主流である、リアクティブ型ルーチングプロトコルの多くは、通信要求が発生するたびに経路要求パケットをネットワーク全体にフラッディングする必要があり、制御負荷が増加する問題がある。この問題を解決するため、DREAM[4], OCTOPUS[5]などの従来のリアクティブ、プロアクティブ型とは考え方の異なる位置情報利用型ルーチングが提案されている。位置情報利用型ルーチングの大きな特徴は、各ノードがパケットに含まれた送信元先への位置情報に基づき次の中継ノード（次ホップノード）を逐次選択していくことである。これにより経路情報がなくとも送信元ノードの位置情報さえわかれば通信が可能である。次ホップノード選択方式としては、GEDIR[4], COMPASS[6]などが提案されている。（詳細については 2.2.1 参照）位置情報利用型ルーチ

ングは、リアクティブルーチングと比較すると経路構築時の制御パケット量を削減できる一方、位置情報管理に伴う制御パケット量の増加という課題を有する。これに対し、2章で説明したとおりモバイルエージェント (MA: Mobile Agent) を用いて位置情報を集中的に管理し経路構築を行うルーチングプロトコルが提案された。このプロトコルでは、位置情報管理と経路算出の機能を持った MA を特定のエリア内に存在するノードが実行することで、ルーチングの効率化を図っている。MA の利用によるオーバーヘッドは増加するが、各ノードの位置情報をネットワークの全ノードへ広告する必要はなくなるため、プロトコル全体としての制御パケット量が削減される。また、MA が各ノードの位置情報を集中管理しているため、あて先ノードの位置情報を取得するだけでなく、適切な中継経路も算出できることからパケット到達率においても高い性能を示している。

これらの位置情報利用型ルーチングは制御メッセージによるオーバーヘッドの削減を達成しているが、位置情報の取得には一般的に GPS (Global Positioning System) を利用することが想定されているため、屋内での利用が困難である。また、GPS の測定誤差によるプロトコルの性能低下や GPS 利用のための消費電力の増加、端末自体のコスト増加等の問題も生じる。また、MA を用いた位置情報利用型ルーチングでは MA は各ノードの位置情報と理想的な通信可能範囲を用いて経路算出を行うが、実環境では位置情報より互いに通信範囲内に存在すると判断されたノード間でも建物等の遮蔽物により無線リンクが確立できない可能性がある。そのため、実際に存在しない無線リンクが経路に含まれてしまう可能性がある。

MA を用いた位置情報利用型ルーチングでは、MA の滞在エリアの定義や MA による経路算出を単純にするため位置情報を利用している。しかしながら、制御パケット量削減やパケット到達率向上には MA が各ノードの情報を一元管理している点が大きく貢献しているため、必ずしも MA が管理する情報を位置情報とする必要はないと考えられる。そこで、本章ではこの点に着目し、MA が位置情報の代わりに各ノードの隣接ノード情報を集中管理し、それを用いて経路構築を行うルーチングプロトコルを提案する。これにより、MA を用いた位置情報利用型ルーチングの利点を生かしつつ、上で述べた位置情報を利用することにより生じる問題を解決する。

4.2 提案方式

MA を用いた位置情報利用型ルーチング方式を含め、位置情報利用型ルーチングは一般的なリアクティブ型ルーチングに対して制御メッセージによるオーバーヘッドなどの点において優位性がある。しかしながら、4.1 で述べた通り、GPS を用いることによる欠点や、理想的な経路算出による実環境への適用上の問題がある。提案方式では、2 章で説明した MA を用いた位置情報利用型ルーチングを基本にし、位置情報の代わりに隣接ノード情報を利用することでこれらの問題を解決する。具体的には、MA に全ノードの隣接ノード情報を保持させ、通信要求時にはその情報を元に MA が経路の構築を代行する。

提案方式では MA を用いた位置情報利用型ルーチング同様に位置情報更新、経路情報取得、経路構築の 3 つの下位プロトコルから成り立つ。ただし、位置情報を扱わないことから MA の移動処理や経路情報取得処理など、MA を用いた位置情報利用型ルーチングと異なる部分がある。以降では提案方式の動作について説明する。

4.2.1 位置情報を利用しない Mobile Agent (MA) の運用

2 章で説明した MA を用いた位置情報利用型ルーチングでは位置情報を用いて MA の存在エリアを定義しているが、提案方式では位置情報を扱わないため同じ方法は使えない。しかしながら、提案方式では 4.2.2 で示すように MA が全ノードの隣接ノード情報を管理するため、ネットワークのトポロジを抽象的なグラフとして把握することができる。そこで、このグラフの中心にあるノードを推定して、そこへ MA を移動させる。本方式では中心を求める指標として近接中心性[7]を用いる。近接中心性はノードから他のノードへ平均的にどれくらい近いかが、によって定義される。これは情報がネットワーク全体に行き渡りやすいかを測る量ともいえる。近接中心性の定義は以下のとおりである。

$$\frac{N-1}{\sum_{\substack{j=1 \\ j \neq i}}^N d(v_i, v_j)} \quad (4.1)$$

式(4.1)において $d(v_i, v_j)$ はノード v_i からノード v_j までのホップ数である、すなわち式(4.1)はノード v_i から他の $N-1$ 点への平均ホップ数の逆数を意味する。全ノードに対して近接中心性を求め、その値が一番高いノードがネットワークの中心であると仮定し MA ノードとす

る。MA の移動のタイミングは従来方式では MA が存在エリアから外れた時であったが、提案方式ではあらかじめ定義した時間経過毎に行うこととする。MA はこのタイミングで最も近接中心性の高いノードに MA を移動させる。MA の移動は以下の通り行われる。まず MA ノードが隣接ノードに MA 移動要求パケットをブロードキャストし、受信したノードは自身の近接中心性の値を返信する。返信を受信したノードは、その中から値が最も大きいノードを MA の移動先として選択する。MA の移動に関する動作については 2.3.1 と同様のため説明は省略する。

4.2.2 隣接ノード情報更新

提案方式では隣接ノード情報を MA に通知する。MA を用いた位置情報利用型ルーチングでは MA の存在エリアに向かって GEDIR で送信することで位置情報の更新を実現していたが、提案方式では位置情報を使わないためこの方法は使えない。そこで、提案方式では MA が一定時間ごとに自身への経路を知らせるために MA 広告メッセージをフラッディングする。これを受信したノードはメッセージが辿ってきた経路を MA へのパスとして保存した後、自身の ID をメッセージパケットに加え転送する。ただし、受信メッセージの電波強度が予め定めた閾値以下だった場合は転送しない。これは、無線リンクが切れやすい MA への経路を作らないようにするためである。各ノードはこの経路を用いて一定時間ごとに隣接ノード情報を MA に通知し更新する。また、隣接ノード情報は保持しているすべてを送信するのではなく、受信電波強度が同じく事前に定めた閾値以上のノード情報のみ送信する。これも消滅しやすい無線リンクが経路として使われないようにするための工夫である。MA 広告メッセージの例を図 4.1 に示す。この図では、まず MA ノードであるノード 1 が MA 広告メッセージをフラッディングし、これを受信した各ノードはメッセージが辿ってきた経路を MA Path として保存し自身の ID を加えて転送している。

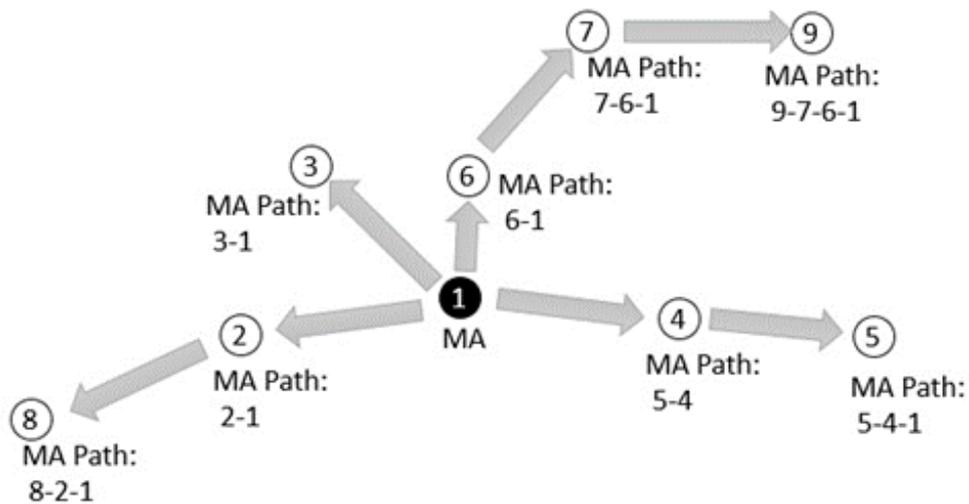


図 4.1 MA 広告メッセージ

4.2.3 経路情報取得

4.2.2 で述べたとおり各ノードは一定期間毎に MA が送信する広告メッセージから MA への経路を取得し保持する。通信要求が発生した時にはこの経路を用いて MA へ経路要求を送信する。経路要求を受信した MA は保持する各ノードの隣接ノード情報からリンクコスト 1 のダイクストラ法により経路を算出し要求元ノードへ作成した経路を通知する。

4.3 シミュレーションによる性能評価

計算機シミュレーションにより提案方式と既存のルーチングプロトコルとの性能を比較し評価する。シミュレータは著者らの研究グループで開発したものをを用いる。電波伝搬モデルは two-ray ground としており、パケットの衝突や干渉によるパケットロスについても考慮する。比較方式として提案方式の基となる MA を利用した位置情報利用型ルーチング（以下位置情報利用方式）の他にリアクティブ型ルーチングから AODV、プロアクティブ型ルーチングから OLSR を選び用いた。主なシミュレーションのパラメータを表 4.1 に示す。ノードの電力消費値は一般的な無線 LAN モジュールを参考にした。また、位置情報利用方式に有利な状況とするため、シャドウイングや遮蔽物などについては考慮しない。さらにノードが取得する位置情報は測定誤差を含まない理想的なものとする。

本シミュレーションでは、1 辺を 1000m とする正方形エリア内にランダムに配置された全ノードが Random Way Point に従い移動するモデルを想定する。そして、任意に選んだノードから同じく任意に選んだ他のノードへ UDP パケット送信を行わせる。この際の各ルーチングプロトコルにおける UDP パケット到達率（PDR: Packet Delivery Ratio）、平均経路構築遅延、消費電力を測定する。

PDR は構築された経路を用いて UDP 通信を行った場合の送信パケット量に対するあて先への到達パケット量の比とする。経路構築遅延は経路要求開始時から最初の UDP パケットがあて先へ到達するまでの時間とし、消費電力はノード 1 台あたりの通信モジュールが使用する電力の合計と定義する。なお、端末自体を動作させるための消費電力や位置情報を取得するために必要な GPS モジュール等の消費電力は含めないこととする。ネットワーク負荷が性能に与える影響を調査するために、UDP フローの発生頻度を変化させて評価する。また、ノード移動速度の影響も評価するため、速度を一様分布に従うランダムな値として選択する際の範囲として 1~3, 1~9m/s の 2 パターンを用いる。シミュレーション時間は 1 回あたり 2000 秒とし、ノードの配置や移動パターンを変化させた平均値を結果のグラフに示す。

また、提案プロトコルのパラメータは次のように設定した。まず、MA の移動間隔は 10 秒とした。そして、MA が自身の位置を知らせるための MA 広告メッセージの送信間隔を 1 秒とした。各ノードの隣接ノード情報の更新間隔は 1 秒と 3 秒の 2 パターンを用いた。ま

た, MA 広告メッセージの転送可否と隣接ノード情報の更新に載せる情報の選別に用いる電波強度の閾値は, two-ray ground のモデルにより通信可能範囲の 90% (今回は 90m) の距離だけ減衰させた電波強度を用いた.

また, 比較方式の 1 つである位置情報利用方式において MA の存在エリアはシミュレーションエリアの中心に設定し, 半径 50m の円とする.

表 4.1 シミュレーションパラメータ

パラメータ	値
MAC層プロトコル	IEEE802.11g
チャネルレート	54 [Mbps]
電波伝搬モデル	two-ray ground
通信範囲	100 [m]
シミュレーション時間	2000 [s]
移動モデル	Random way point
移動速度	1~3, 1~9 [m/s]
シミュレーションエリア	1000 × 1000 [m]
ノード台数	300,400,500,600 [台]
UDPフロー数	0.2, 0.4, 0.6, 0.8, 1.0 [フロー/s]
UDPフローデータサイズ	1024 [kbyte]
送信電力	530 [mW]
受信電力	326 [mW]
待機電力	0.4 [mW]

4.4 シミュレーション結果・考察

プロトコルの性能評価を行う前に、提案方式と位置情報利用方式において実際に MA がどのように位置しているかを確認した。位置情報利用方式では物理的にネットワークの中心に近いノードを選んで MA を配置しているが、提案方式では近接中心性により MA を配置するノードを決定している。図 4.2 は MA が移動したタイミングでの MA の位置座標をプロットしたものである。一辺 1000m の正方形のシミュレーションエリアにノードを 500 台配置し、各 300 回分の MA 移動後の位置を示した。本シミュレーションではノードは一樣に分布しているため MA の位置はネットワークエリアの中心に近いほど良いと考えられる。図より位置情報利用方式に比べると最大で半径にして 2 倍ほどの散らばりが見られるものの、多くが位置情報利用方式と同様にネットワークエリアの中心に存在していることが確認できる。この結果により、近接中心性による MA ノードの選択に問題がないことが確認できた。

次に 4 種類のルーチングプロトコルの比較により提案方式の評価を行った。提案方式については隣接ノード情報の更新間隔が異なる 2 パターンを評価し、グラフ上では更新間隔 1 秒のものを proposed 1, 3 秒のものを proposed 3 とした。図 4.3~4.5 は UDP フロー発生頻度別の結果をまとめたものである。ノード台数は全て 500 台とし、移動速度は 1~3m/s に設定した。

図 4.3 に UDP フロー発生頻度別の PDR を示す。UDP フローの発生頻度にかかわらず位置情報利用方式、提案方式、AODV、OLSR の順に到達率が高いことが示された。提案方式の PDR は位置情報利用方式に比べて少し悪いものの高い値を保っていることがわかる。これは、位置情報利用方式では MA から取得した経路が正しくなかった際にあて先ノードの位置情報を使い GEDIR による送信ができるが、提案方式では位置情報を使用しないためそのような仕組みがないからである。提案方式の隣接ノード情報更新間隔が 1 秒の場合と 3 秒の場合を比べると、UDP フロー発生頻度が低い場合には更新間隔が 1 秒のほうが高い PDR を示しているが、頻度が高い場合には PDR が低下している。これは、隣接ノード情報更新と UDP パケットの衝突が発生するためである。また、AODV は UDP フロー発生頻度が低い場合は高い値を示しているが、頻度が上がるにつれて値が低下している。AODV の場合、経路要求のたびに RREQ によるフラッディングなど負荷の大きい通信が行われる。

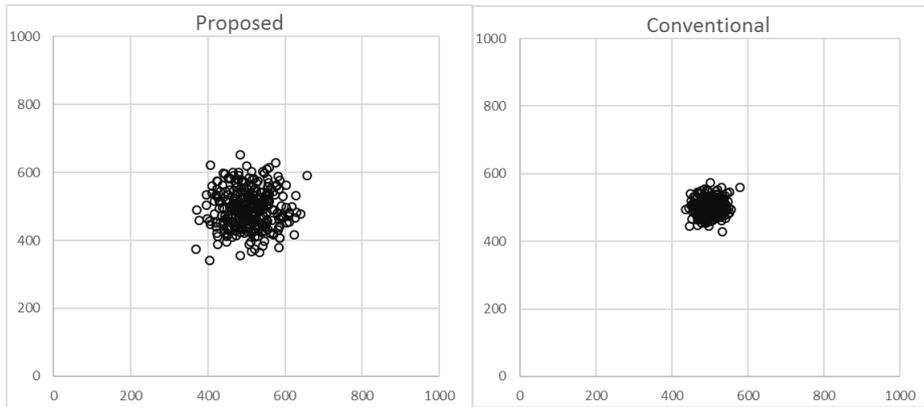


図 4.2 MA の位置

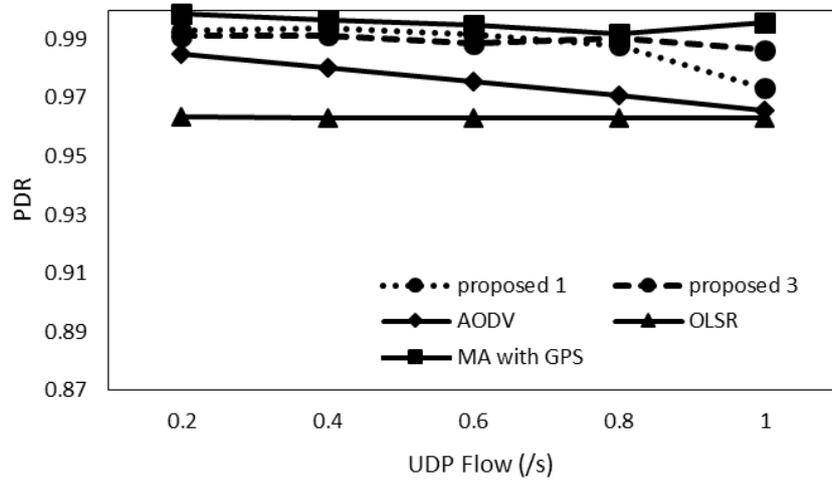


図 4.3 パケット到達率 (速度 1~3m/s)

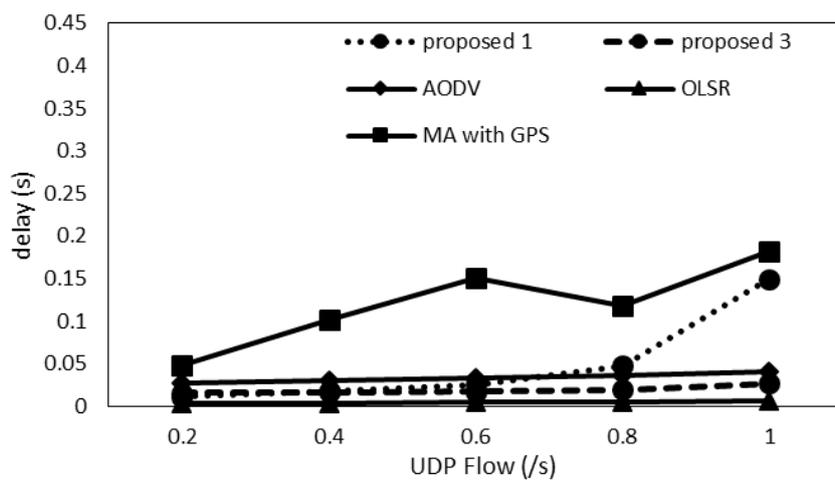


図 4.4 平均経路構築遅延 (速度 1~3m/s)

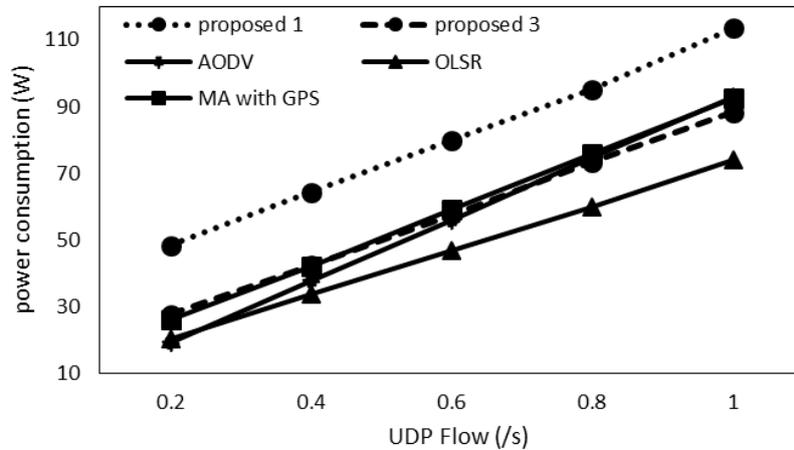


図 4.5 消費電力 (速度 1~3m/s)

これにより UDP パケットとの衝突が発生し PDR が低下していると考えられる。提案方式と位置情報利用方式では MA に対する経路要求・応答のユニキャスト通信のみであるため AODV に比べそのような影響は受けにくいと考えられる。OLSR については各ノードが常に同間隔で情報交換を行っているため、そもそも PDR は低いものの UDP フロー発生頻度の増加には影響されないことが示されている。

図 4.4 に UDP フロー発生頻度別の平均経路構築遅延を示す。位置情報利用方式以外の 3 方式は低い遅延時間を示している。位置情報利用方式の遅延が大きい理由は、算出した経路が使えず GEDIR を用いた通信に切り替わった時に遅延が大きくなる場合があるためである。他の方式ではこのようなことは生じない。また、提案方式の場合 RREQ の送信先が必ずシミュレーションエリアの中心付近にいる MA であるため、AODV に比べて平均的に RREQ/RREP の送信ホップ数が短くなり AODV よりも少し遅延が小さい結果となっている。ただし、隣接ノード情報更新間隔が 1 秒で UDP フロー発生頻度が 1 フロー毎秒の時には遅延が大きくなっている。これは更新間隔が高いことにより更新メッセージと UDP パケットの衝突が発生し再送による遅延が生じるためである。OLSR についてはプロアクティブ型ルーティングの特性上通信要求発生時に経路を作る必要がないため、当然遅延は最も小さくなっている。

図 4.5 に UDP フロー発生頻度別のノードあたりの消費電力を示す。提案方式の隣接ノード情報更新間隔が 1 秒の場合が最も消費電力が大きいことが示されている。これは高い頻

度の更新メッセージが消費電力を増加させているためである。UDP フロー発生頻度と消費電力の関係としては、AODV では経路構築にフラッディングを用いるため大きな負荷がかかることから UDP フロー発生頻度が上がると消費電力が大きく増えることが示されている。一方、他の方式は通信要求自体に対しては大きな負荷は生じないため消費電力は緩やかに上昇する。

次にノード台数別に評価したものを図 4.6~4.8 に示す。UDP フロー発生頻度は全て 0.6 フロー毎秒に設定した。

図 4.6 はノード台数別の PDR を示す。ノード台数が 400 台以上の場合は提案方式と位置情報利用方式の PDR が高い結果を示している。また、ノード数が 300 台のときだけ性能が大きく低下している。これはノード台数が少ないことで各ノードの隣接ノード情報や、経路要求・経路応答などの MA に対するメッセージが届きにくい状況になっていることが原因であると考えられる。そのため、隣接ノード情報更新間隔が 3 秒の時に特に PDR が低いことが示されている。また、提案方式の PDR が位置情報利用方式よりも低い理由は図 4.3 の考察で述べた理由と同様で GEDIR を用いた通信に切り替わる機能がないからである。この結果より、提案方式の性能を保つためにはある程度のノード台数が必要であるということが示された。

図 4.7 ではノード台数別の平均経路構築遅延を示す。図 4.4 の考察と同様の理由から位置情報利用方式の遅延が大きくなっていることがわかる。また、ノードが 300 台の時に提案方式の遅延が大きい理由としては、図 4.6 の考察と同様で MA に対する経路要求・経路応答が届きにくいことから再送による遅延が発生しているためと考えられる。

図 4.8 はノード台数別の消費電力を示す。図 4.5 の考察で述べたとおり消費電力は UDP 発生頻度の影響に左右されるため図 4.8 の 0.6 フロー毎秒のケースでは提案方式の消費電力は大きい。またこのグラフからノード台数が消費電力にあまり影響を与えないことがわかる。

次にノードの移動速度を 1~9m/s に変更して評価した。ノード台数は全て 500 台に設定した。図 4.9~4.11 に各方式の UDP フロー発生頻度別の評価を示す。

図 4.9 は UDP フロー発生頻度別の PDR を示したものである。位置情報利用方式、提案方式（更新間隔 1 秒）、AODV、OLSR、提案方式（更新間隔 3 秒）の順に到達率が高いことが示された。

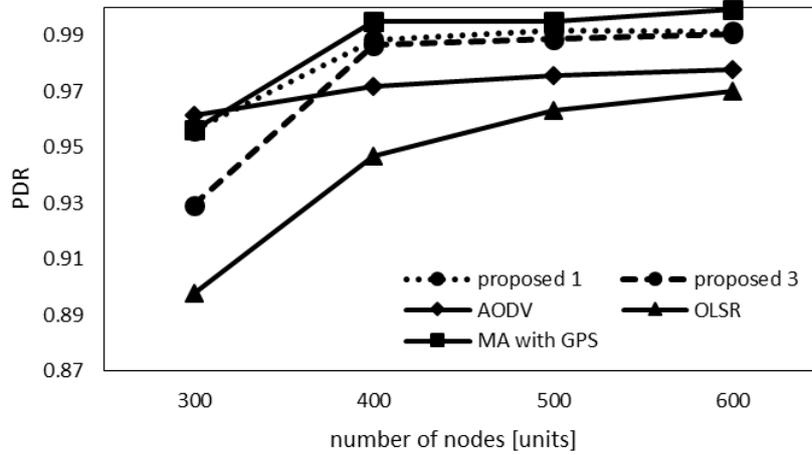


図 4.6 パケット到達率 (ノード数別)

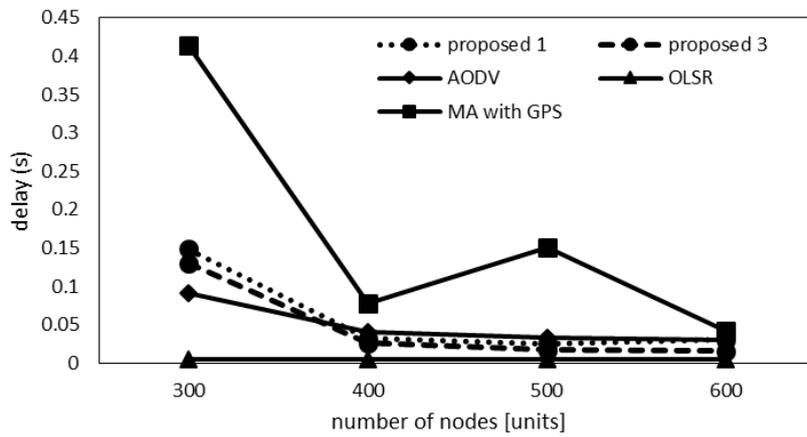


図 4.7 平均経路構築遅延 (ノード数別)

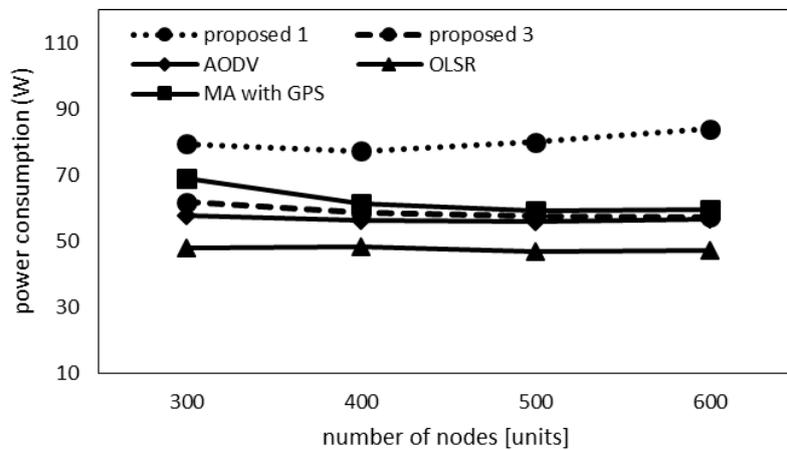


図 4.8 消費電力 (ノード数別)

また、UDP フローが 1 フロー毎秒のときは提案方式の性能が大きく低下している。これは隣接ノード情報の更新と UDP パケットの衝突が発生し、隣接ノード情報が届きにくくなることにより正しい経路が作られなくなることが原因である。また、提案方式の更新間隔が 3 秒の時に PDR が低いのはノードの移動速度に対して更新頻度が少ないためである。AODV と OLSR については UDP フロー発生頻度の変化による影響の傾向はそのまま、全体的に PDR が下がった結果になっている。これは単純に移動速度が上がったことにより取得した経路が実際に使用できない確率が上がるためである。

図 4.10 に UDP フロー発生頻度別の平均経路構築遅延を示す。OLSR 以外の方式では移動速度が 1~3m/s のときに比べると全体的に遅延が大きくなっていることがわかる。これも取得した経路が使用できなかった場合に経路の再取得が発生するためであると考えられる。また、提案方式の隣接ノード情報更新間隔が 3 秒の場合ではノード移動速度に対して更新頻度が少ないため、取得した経路が使用できない可能性が高くなることから特に遅延が大きくなっている。

図 4.11 に UDP フロー発生頻度別の消費電力を示した。この図では、提案方式（更新間隔 1 秒）と位置情報利用方式の消費電力が高いことがわかる。位置情報利用方式の消費電力が高いのは移動速度があがったことにより各ノードが情報を MA に通知する頻度が上がったことが直接の原因であると考えられる。

これらの結果から、PDR においては位置情報利用方式が優れていることが示された。ただし、これは GEDIR による送信手段への変更など位置情報を用いることにより実現できた機能が作用しているためである。本研究の目的が位置情報を利用することによるデメリットや制限を除くということを考えると、位置情報利用方式よりは少し低いものの AODV, OLSR よりも性能が高い提案方式の優位性は十分であると考えられる。また、前述したように、本シミュレーションでは各ノードが取得する位置情報は誤差を含まない理想的なものであり、さらに建物等を想定した障害物のない環境であることから、実際の環境では 4.1 で述べた理由により位置情報利用方式の性能低下が予想される。

平均経路構築遅延と消費電力においても提案方式の性能は、同等かそれ以上で十分な性能が示された。特に、消費電力についても今回の結果には GPS モジュール等の位置情報取得に必要な電力を含まないため、実際には位置情報利用方式の場合その分の消費電力が余分に必要となる。

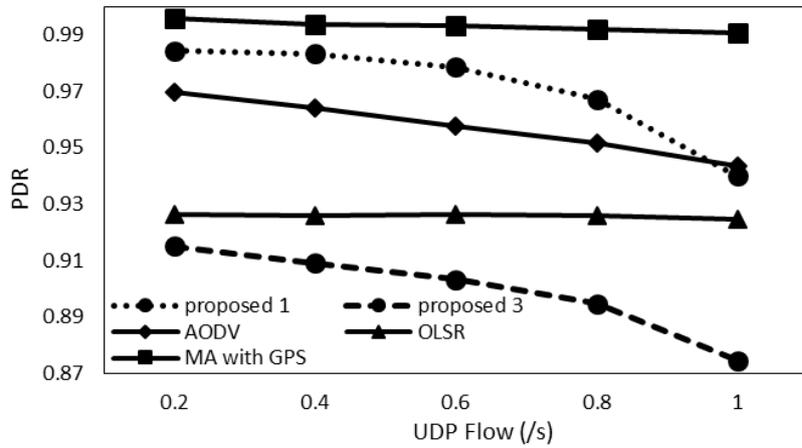


図 4.9 パケット到達率 (速度 1~9m/s)

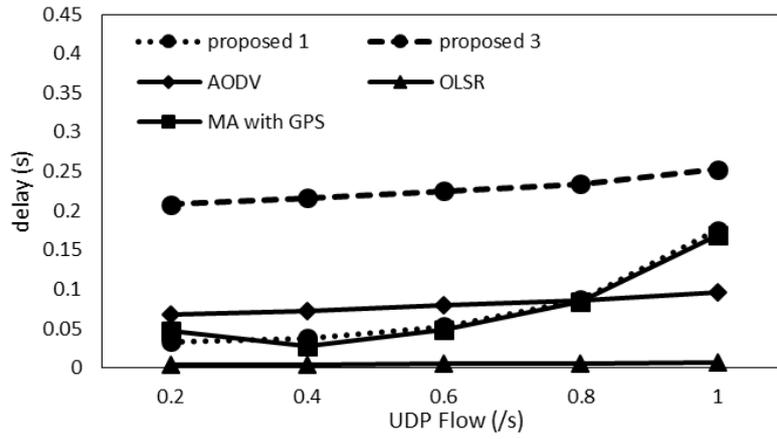


図 4.10 平均経路構築遅延 (速度 1~9m/s)

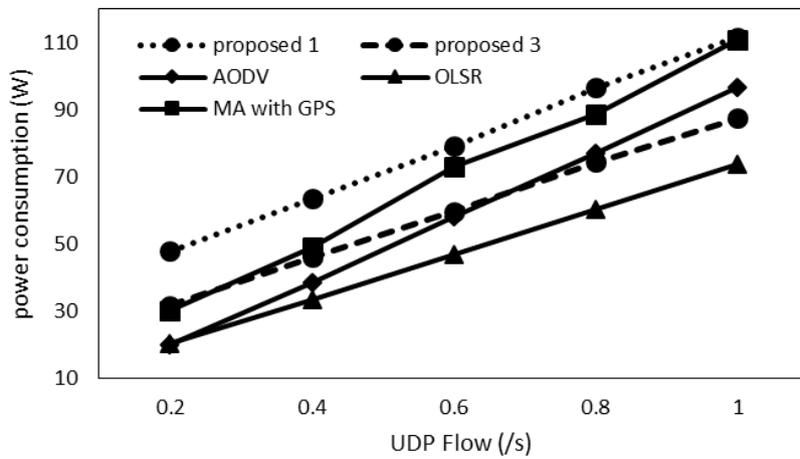


図 4.11 消費電力 (速度 1~9m/s)

また、今回提案方式について隣接ノード情報の更新間隔が 1 秒と 3 秒の 2 パターンについて検討したことにより、更新間隔がプロトコルの性能に大きく影響することがわかった。さらに、ノードの移動速度やノード密度の状況に応じて最適な更新間隔が異なるということが明らかになった。基本的には更新間隔を短くすることで PDR が向上するが、間隔を短くすると更新頻度が高い分消費電力が増加する。また、不必要に更新間隔を短くすると、UDP パケットとの衝突が生じて PDR が低下する場合があることもわかった。総合的に判断すると今回検討したケースではノードの移動速度が 1~3m/s のときは更新間隔 3 秒、移動速度が 1~9m/s のときは更新間隔が 1 秒の性能が良いと考えられる。ただし、今回採用した更新間隔は暫定的なものであり、実際には状況に応じた最適値があると考えられる。この点については今後詳細な検討が必要である。

4.5 4章のまとめ

本章では、位置情報利用型ルーチングの一つである MA を用いた位置情報利用型ルーチング方式において、GPS を利用することにより生じる問題を解決するため、各ノードの位置情報の代わりに隣接ノード情報を使用する方式を提案した。本方式では MA は隣接ノード情報を一元的に管理し、経路構築を行う。計算機シミュレーションによる性能評価の結果、PDR において位置情報利用方式よりも少し劣るものの AODV, OLSR よりは高い性能であることを示した。また、平均経路構築遅延と消費電力においても提案方式の性能は他の方式に比べ同等かそれ以上に優れていた。本研究の目的が位置情報を利用することによるデメリットや制限を除くことを考えると、総合的に判断して提案方式の有効性を示せたといえる。また、提案方式では隣接ノード情報の更新間隔が性能に大きな影響を与えることがわかった。そのため、ノードの移動速度やノード密度の状況に応じて最適な更新間隔を設定する必要がある。この点の検討は今後の課題である。

参考文献

- [1] T.Clausen, "Optimized Link State Routing Protocol" IETF, <https://www.ietf.org/rfc/rfc3626.txt>, Oct. 2003.
- [2] IETF MANET Working Group, "The dynamic source routing protocol for mobile ad hoc networks (DSR)," draft-ietf-manet-der-10. txt, Internet-Draft, Jul. 2004.
- [3] C. Perkins, E. Belding-Royer, S. Das "Ad hoc on- demand distance vector routing," RFC3561, pp. 4089-4095, 2003.
- [4] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, "A distance Routing Effect Algorithm for Mobility (DREAM)," Proceedings of the International Conference on Mobile Computing and Networking, pp.76-84, 1998.
- [5] R. Melamed, I. Keidar, and Y. Barel, "Octopus: A Fault-Tolerant and Efficient Ad-Hoc Routing Protocol," Proceedings of The 24th IEEE International Conference on Reliable Distributed Systems, pp.39-49, 2005.
- [6] J. Urrutia, "Two Problems on Discrete and Computational Geometry," Proceedings of Japan Conference on Discrete and Computational Geometry, pp42-52, 1999.
- [7] 増田直紀,今野紀雄,"複雑ネットワーク 基礎から応用まで", (株)近代科学社,東京,2010

第 5 章

5. コンテンツ指向型 MANET におけるモバイルエージェントを利用したコンテンツ取得手法

5.1 はじめに

近年, 次世代ネットワークのアーキテクチャとして, インターネットなどで広く用いられている IP アドレスに基づくホスト指向型ネットワークに代わり, 送受信データに着目して設計されたコンテンツ指向型ネットワーク (ICN; Information-Centric Network) [1]が注目を集めている. これは, インターネットの利用形態がホスト間の通信ではなく, 情報の配信・流通システムへ変化してきている流れを受けてのものである. コンテンツ指向型ネットワークの基本的なアイデアは特定のホストでなくコンテンツ名によってネットワークが機能する点にある. また, ネットワーク内キャッシングによる効率的なデータ取得が行えるという特徴を有する. この研究領域での有名な研究プロジェクトとしては NDN[2], CCN[3]等が挙げられる.

ICN の典型的な動作の具体例としては, まず, コンテンツを取得したいノードは **Interest** と呼ばれるリクエストパケットをネットワーク全体にブロードキャストする. このリクエストを受信したノードで, もし, 要求されているコンテンツを持っている場合は要求ノードへコンテンツを送信する. 持っていない場合は, **Interest** の転送を継続する.

これらの研究は有線環境でのものが主であったが, 近年では, モバイルアドホックネットワーク (MANET) 等の無線環境に適用した研究も行われている. MANET とは, 複数の移動体無線端末が自律分散的にネットワークを構成する無線通信技術である. 近年, 無線通信の発展と無線端末の普及, 利便性の向上, また災害時対策への期待に伴い研究が盛んに行わ

れている。MANET では無線端末どうしが相互接続され、自身の電波が届く通信可能範囲から外れた端末に対しても、他端末を経由してパケットを転送することが可能である。

MANET を構成するノードはスマートフォン等の小型の移動体無線端末を想定する場合が多いため、利用できる電力が限られている場合が多い。しかし、多くのコンテンツ指向型 MANET のコンテンツ取得手法では Interest パケットの送信にフラッディングを用いることから、制御パケットの負荷が大きい傾向にある。これらの問題を解決するため、いくつかの手法が提案されている。TOP-CCN[4] では MPR 集合を用いたパケットのフラッディングにより、制御パケットを抑制している。E-CHANET[5] ではカウンタベースのパケット転送抑制機構を用いている。REMIF[6] では Interest パケットの周囲の送信状況を用いて重複送信をチェックすることで制御パケットを削減している。このように、制御パケットの負荷の抑制についての改善は行われている。しかし、これらの方式では Interest パケットをフラッディングにより送信することから、要求したコンテンツを複数のノードがキャッシュしていた場合、Interest パケットを受信した複数のノードがコンテンツを送信するため重複したコンテンツ送信が発生するという問題がある。これはネットワークの負荷を増加させる。

そこで本章ではこの問題を解決するため、2 章で説明したホスト指向型の MANET のルーティングプロトコルであるモバイルエージェント (MA) を用いた位置情報利用型ルーティングをコンテンツ指向型 MANET に応用することで制御パケットの削減と重複したコンテンツ送信を抑制する手法を提案し、コンテンツパケット到達率、消費電力、送信遅延において優位性を示す。

5.2 REMIF

ここでは、本研究で比較方式として用いる REMIF (Robust and Efficient Multipath Interest Forwarding) の動作について説明する。REMIF は送信パケット量削減を目的とした Interest パケットのフラッディング方式である。REMIF では各ノードは Content Store (CS) と呼ばれるコンテンツ保持リストと Pending Interest Table (PIT) と呼ばれる Interest の送信履歴を持つ。具体的な動作について述べる。

まず、コンテンツを要求するノードは取得したいコンテンツ名を含む Interest パケットをブロードキャストする。Interest パケットを受信した中継ノードはその Interest が有効期限切れでないかを確認し、期限切れだった場合はパケットを破棄する。期限切れでない場合、自身の CS を確認し、CS に要求されたコンテンツの登録がある場合にはランダムに設定された時間 (*Defer_Data*) だけ待機する。その間に他のノードにより送信された同一コンテンツを受信した場合には自身からのコンテンツ送信を中止する。同一コンテンツを受信することなく待機時間が経過した場合には、要求ノードに向かってコンテンツを送信する。

要求されたコンテンツの登録が CS に登録されていない場合には以下の方法で Interest パケットの転送を行う。まず自身の PIT を確認し、もし同じ Interest の登録があった場合は、これを中継せず破棄する。PIT に Interest の登録がなかった場合は自身の残存電力を確認する。残存電力が閾値以下だった場合は PIT に Interest を登録し、Interest パケットを中継せずに破棄する。残存電力が閾値より大きい場合には PIT に Interest を登録しランダムに設定された時間 (*Defer_Interest*) だけ待機する。この間に他のノードによる同じ Interest パケットか要求コンテンツの送信を受信した場合には PIT から該当する Interest の登録を削除し Interest パケットも中継せず破棄する。そうでない場合 Interest パケットを中継する。REMIF ではこのような仕組みを用いて、重複した Interest やコンテンツの送信を削減する。また、残存電力の閾値を用いることで残存電力が少ないノードの負担を減らす仕組みも備えている。

5.3 提案方式

ここでは提案方式について説明する。本方式は、2章で説明したIPベースMANETのルーチングプロトコルであるMAを用いた位置情報利用型ルーチングをコンテンツ取得手法に応用したものである。この方式では、MAがネットワークに所属する全ノードの位置情報と各ノードが所有するコンテンツのリスト（コンテンツリスト）を集中的に管理する。このリストには各ノードが保持するコンテンツの名前が記録される。また、ノードがコンテンツを要求した際には、MAが保持する情報を用いて要求ノードから該当コンテンツを保持するノードまでの経路を作成する。このようにして制御パケットの削減を図っている。本方式は位置・コンテンツ情報のMAへの登録、コンテンツ要求・コンテンツ送信の下位プロトコルから成り立つ。なお、提案方式におけるMAの動作は2章の方式と同様で2.3.1で説明したとおりである。

5.3.1 位置・コンテンツ情報更新

各ノードは事前に定めた距離だけ移動すると、MAに向けて情報更新パケットを送信する。この距離を情報更新閾値と呼ぶ。このパケットには自身の位置情報と保持するコンテンツリストが含まれる。フォワーディングにはGreedyフォワーディングの代表的な手法の一つであるGIDER[7]を用いる。このように定期的にMAに位置情報を送信することで、MAは常に全ノードの位置情報とコンテンツリストを把握することができる。

5.3.2 コンテンツ要求

MAは全ノードの位置情報とコンテンツリストを保持できるため、コンテンツ要求ノード（リクエスタ）と要求するコンテンツを保持しているノード（プロバイダ）までの完全な経路を算出して伝えることが可能となる。

あるノードにおいてコンテンツ要求が発生した際、そのノードはMAに対してInterestパケットを送信する。Interestパケットを受け取ったMAは保持する各ノードの位置情報とコンテンツリストを元に、図5.1のように2つの経路をダイクストラ法により算出する。

1つ目の経路は MA とプロバイダノード間の経路であり、2つ目の経路はプロバイダノードとリクエスタノード間の経路である。プロバイダノードは、要求するコンテンツを保持するノードから決定されるが、該当するノードが複数いた場合はリクエスタノードに位置が最も近いノードとする。なお、ノードのネットワークからの離脱等を考慮し、一定時間更新のないノードの位置情報とコンテンツリスト情報は利用しないこととする。この時間の閾値をコンテンツリスト有効期限と呼ぶ。MA は以上の2つの経路、リクエスタノードおよびプロバイダノードの位置情報、要求コンテンツ名を付加したコンテンツ送信要求パケットをプロバイダノードへ送信する。

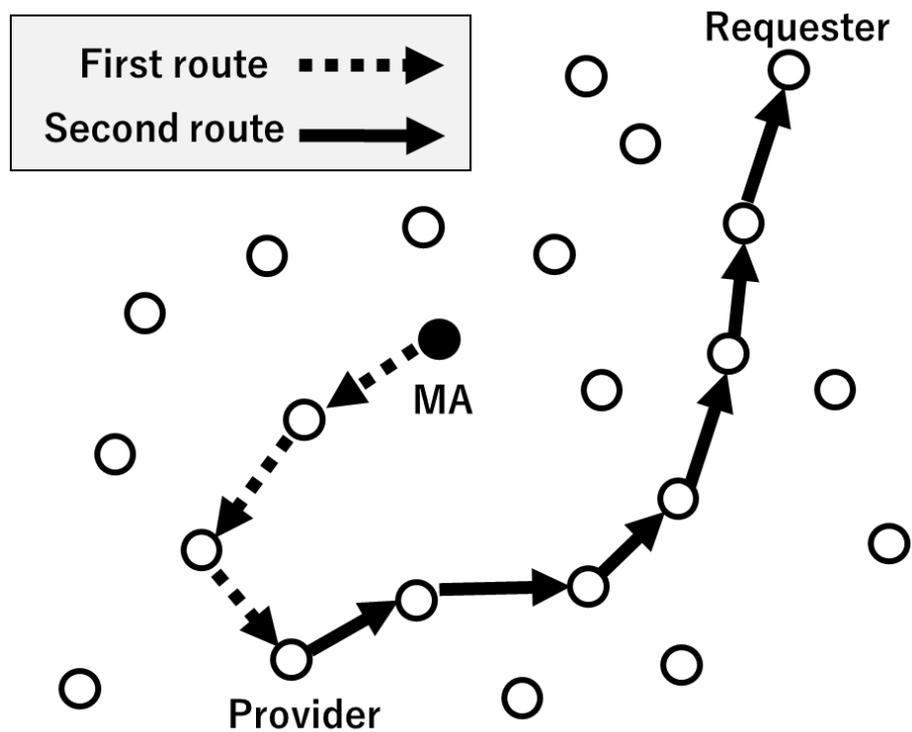


図 5.1 MA が作成する 2 つの経路

5.3.3 コンテンツ送信・キャッシング

コンテンツ送信要求パケットを受信したプロバイダノードはパケットに含まれる経路を用い、リクエスタノードに要求コンテンツを送信する。その際、コンテンツを中継するノードはそのコンテンツをキャッシュし、コンテンツ名を自身のコンテンツリストに追加する。また、経路に記載された中継ノードが見つからなかった場合には、リクエスタノードの位置情報を基にした GEDIR に切り替え送信を続行する。もし、経路に記載されたノードが見つかった場合には再び経路を用いた送信に切り替える。

5.4 シミュレーションによる性能評価

計算法シミュレーションにより提案方式と既存のプロトコルとの性能を比較し評価する。シミュレータには著者らの研究グループで開発したものをを用いる。電波伝搬モデルは two-ray ground としており、パケットの衝突や干渉によるパケットロスについても考慮する。移動モデルは Random way point とし、乱数によりノードの初期配置や移動方向、速度の変更を行った。また、シミュレーション時間は 1000 秒とし、複数回行って得られた数値の平均を結果として示す。

シミュレーションパラメータを表 5.1 に示す。MAC Layer には IEEE802.11g を用い、通信帯域を 54Mbps とする。シミュレーション範囲は一辺 1km の正方形とし各端末の通信範囲を 100m とする。端末数は 400, 500, 600, 700 台とし、密度が高い環境を想定した。移動速度は 1~3m/秒, 1~9m/秒の 2 パターンで測定した。比較方式には REMIF を用いた。

本シミュレーションでは、シミュレーション期間中ランダムにリクエストノードと要求コンテンツを指定し、コンテンツ要求を発生させる。この要求を発生させる頻度を要求頻度と定義し、0.05, 0.1, 0.15, 0.2 回/秒に設定した。また、コンテンツの送信には UDP を用いる。そして、全てのノードがコンテンツを 1 つ作成し、保持すると仮定する。よって、コンテンツの種類はノード台数と同じとなる。また、それに加えて自身のコンテンツ以外のコンテンツのコピーをシミュレーション開始の時点で 2 個ランダムにキャッシュしているものとする。キャッシュ用バッファの上限は 10 個分とした。また、各コンテンツのサイズは 1Mbyte とする。キャッシングポリシーには Leave copy everywhere (LCE) を用い、バッファに空きがない場合の置き換えのポリシーとして、Least recently used (LRU) を用いる。LCE ではコンテンツを中継するノードは無条件でキャッシュする。LRU ではキャッシュ時にバッファに空きがない場合は、参照した時間が一番古いものを削除して新しいコンテンツに置き換える。

評価指標としてはコンテンツ取得率、平均コンテンツ伝送遅延、消費電力を測定する。コンテンツ取得率は要求したコンテンツサイズに対する UDP 通信により取得できたコンテンツのサイズ量の比とする。平均コンテンツ伝送遅延は最初の UDP パケット送信時から最後の UDP パケットがあて先へ到達するまでの時間と定義する。消費電力はノード 1 台当たりの 1 秒に消費する電力と定義する。

提案方式における固有のパラメータはとして、情報更新閾値を 20m とした。コンテンツリスト有効期限は 10 秒とした。また、MA の存在エリアはシミュレーションエリアの中心に設定し、半径 50m の円とする。

続いてプロトコルにおける制御パケットサイズについて説明する。まず REMIF では Interest パケットのサイズを 8 バイトとする。提案方式では制御パケットサイズが固定ではない。そこで、基本サイズを 12 バイトとし、ノード 1 台当たりのノード ID フィールドサイズを 4 バイト、位置情報フィールドサイズを 12 バイト（座標情報 8 バイト、時刻情報 4 バイト）、コンテンツ名を 4 バイトと仮定した。そして基本サイズに必要なフィールド分を追加したものを制御パケットサイズとする。なお MA のデータサイズについては起動に必要な MA 本体のサイズを 30 キロバイトとし、これに情報テーブルサイズ（ノードの位置情報とコンテンツリストのサイズ）を加えた値とした。提案方式における具体的な制御パケットサイズを表 5.2 に示す。また、UDP パケットには経路情報として経路の中継ノード数×4 バイトとあて先ノードの位置情報の 8 バイトを付加する。

表 5.1 シミュレーションパラメータ

Parameter	Value
MAC layer protocol	IEEE802.11g
Channel rate	54 [Mbps]
Communication range	100 [m]
Simulation time	1000 [s]
Simulation area	1000×1000 [m]
Movement model	Random way point
Movement speed	1-3, 1-9 [m/s]
Number of nodes	400, 500, 600, 700 [units]
Content request rate	0.05, 0.1, 0.15, 0.2 [/s]
Number of kinds of content	400, 500, 600, 700
Content data size	1 [Mbyte]
Packet size	1024 [byte]
Caching policy	LCE
Replacement policy	LRU
Transmission power	530 [mW]
Received power	326 [mW]
Standby power	0.4 [mW]

表 5.2 制御パケットサイズ

Information Update	$28+(c \times 4)$ [byte]
Interest	$20+(h_1 \times 4)$ [byte]
Content Transmissions Request Packet	$24+(h_a \times 4)+(h_b \times 4)$ [byte]
MA Migration Request	8[byte]
MA Migration Reply	20[byte]
MA Migration	$30000+(n \times 12)+(c_m \times 4)$ [byte]

c : コンテンツリストに登録されているコンテンツ数

h_1 : 現在までのホップ数

h_a : MA が算出した経路 1 のホップ数

h_b : MA が算出した経路 2 のホップ数

n : MA が情報を保持するノード数

c_m : MA が保持するコンテンツリストの登録合計数

5.5 シミュレーション結果と考察

図 5.2~5.5 はコンテンツ要求頻度の影響を評価したものである。ノード台数は 500 台とし、ノードの移動速度は 1~3/s と 1~9m/s とした。

図 5.2 に PDR の結果を示した。結果より提案方式の PDR が REMIF に比べて高いことがわかる。また、コンテンツ要求頻度が上がっても提案方式の PDR は低下しないことが示された。これは、REMIF では経路が使用できなかった場合、コンテンツ送信に失敗してしまうが、提案方式では GEDIR に切り替わることで位置情報を基にコンテンツをあて先に届けることができることが影響している。また、提案方式は移動速度が速い 1~9m/s の時でも高い PDR が保たれていることが示された。一方 REMIF では経路が使用できなくなる可能性が高くなることから、PDR が低下している。

図 5.3 に消費電力の結果を示す。この図からも提案方式の優位性が確認できる。これは REMIF では Interest パケットの送信にフラッディングを用いるのに対し、提案方式では MA に対するユニキャストの送信で済むことが影響している。Interest パケットの送信にフラッディングを用いる場合、Interest パケット自体のデータ量が多くなるというデメリットもあるが、それ以上に複数のノードが同じ Interest を受信するため、複数のノードが要求するコンテンツを保持していた場合、重複したコンテンツ送信が発生し、これが消費電力を増加させる大きな原因となる。また、REMIF において、移動速度が 1~3m/s の場合に比べ、1~9m/s の場合の消費電力が低くなっている。これは、図 5.2 に示す通り多くのパケットロスが発生しているためである。

図 5.4 に重複して送信したコンテンツのデータ量を示す。この図から重複したコンテンツの送信は REMIF でのみ発生していることが示されており、図 5.3 の考察が裏付けられた。

図 5.5 は平均コンテンツ取得遅延を示す。この図から、提案方式の遅延が REMIF より短いことが示された。これは、提案方式では MA による最適な経路が使用されることが一因であると考えられる。また、図 5.4 で示されている通り、REMIF では重複したコンテンツ送信が発生するため UDP パケットの衝突が発生し、これが遅延を発生させている。また、REMIF ではノードの移動速度が速い場合のほうが遅延は短くなっているが、これは、経路の状態が良いパケットしかあて先に届かないためである。一方、提案方式ではノードの移動

速度による大きな影響は確認できない。これは提案方式では GEDIR を用いることから移動速度の影響を受けづらいためである。

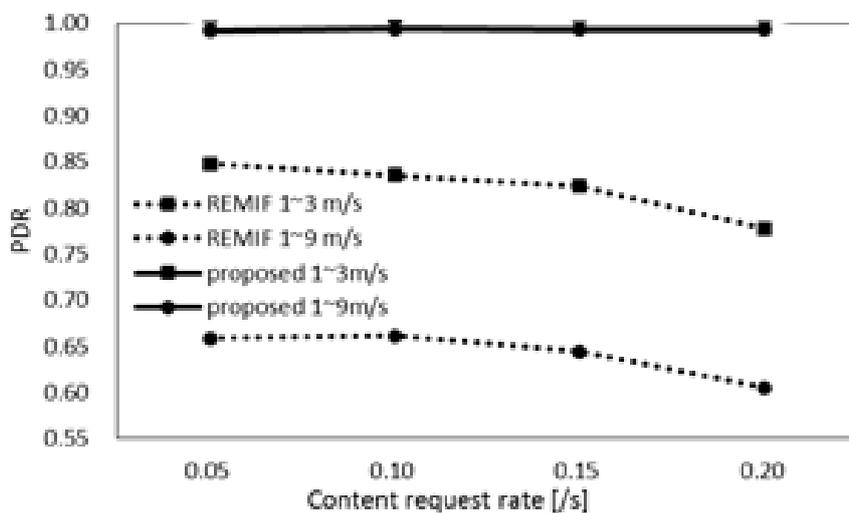


図 5.2 コンテンツ要求頻度に対する PDR

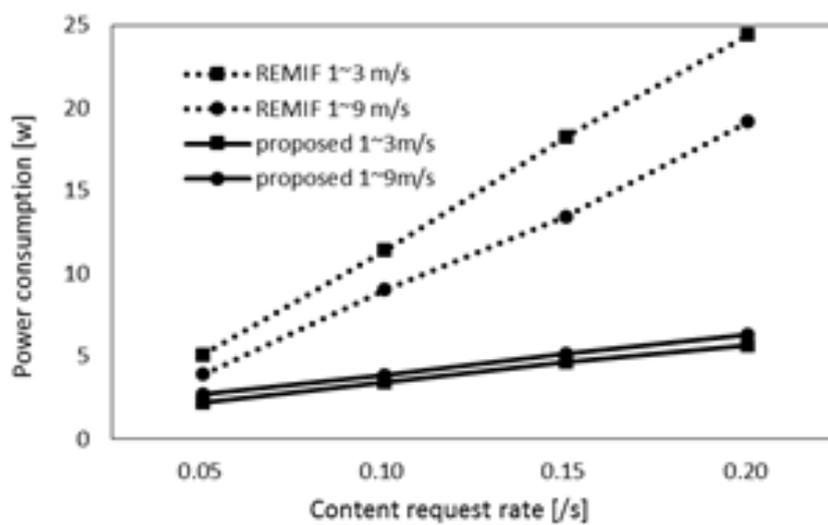


図 5.3 コンテンツ要求頻度に対する消費電力

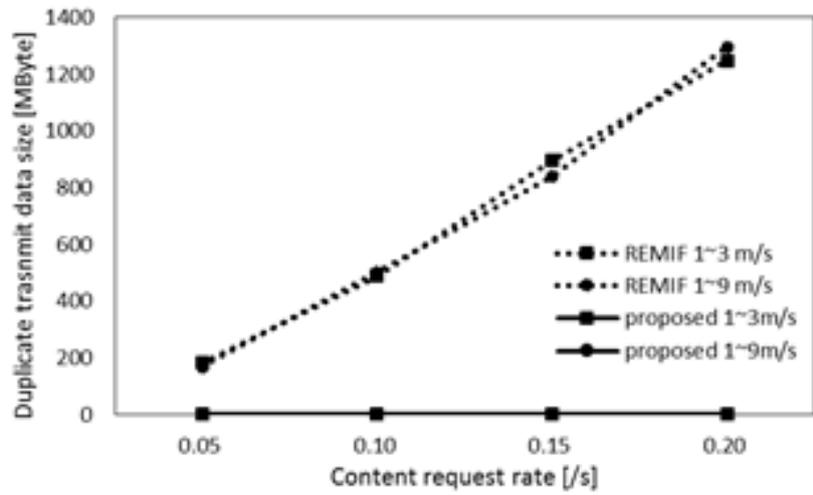


図 5.4 コンテンツ取得頻度に対するコンテンツの重複送信量

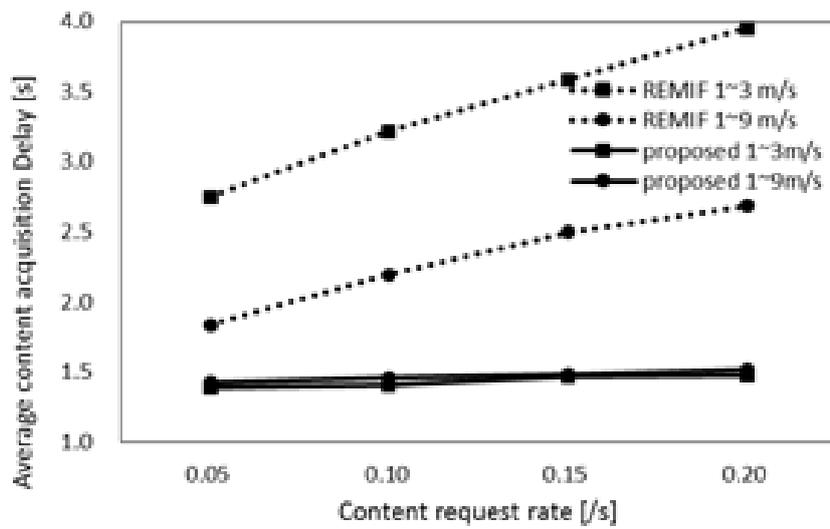


図 5.5 コンテンツ取得頻度に対するコンテンツ取得遅延

図 5.6~5.8 ではノード台数による影響を評価した結果を示す。コンテンツ要求頻度は 0.1 回/s に固定し、ノードの移動速度は 1~3m/s と 1~9m/s とした。これらの図から全ての条件において REMIF に比べ提案方式が優れていることが示された。また、ノード台数による影響は大きくないことが確認できる。さらに、図 5.2, 5.3, 5.5 と同様の理由により、ノードの移動速度の影響が生じていることも確認できる。

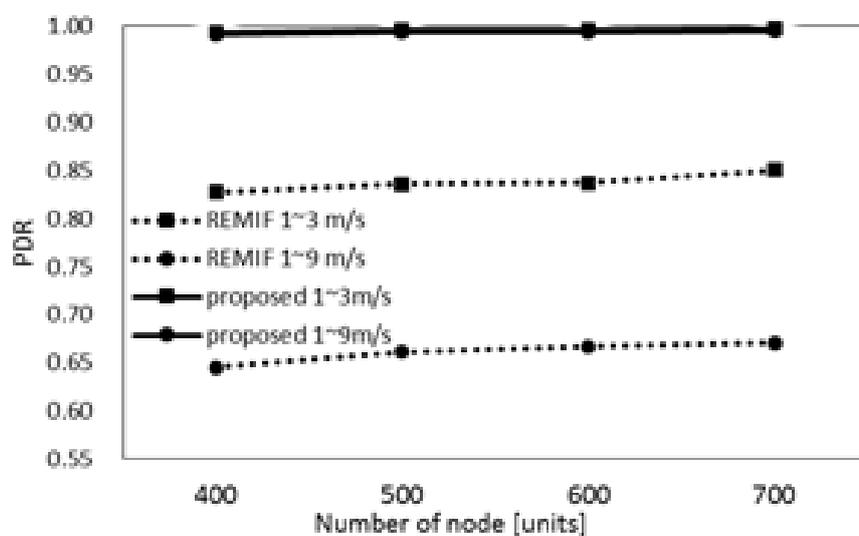


図 5.6 ノード台数に対する PDR

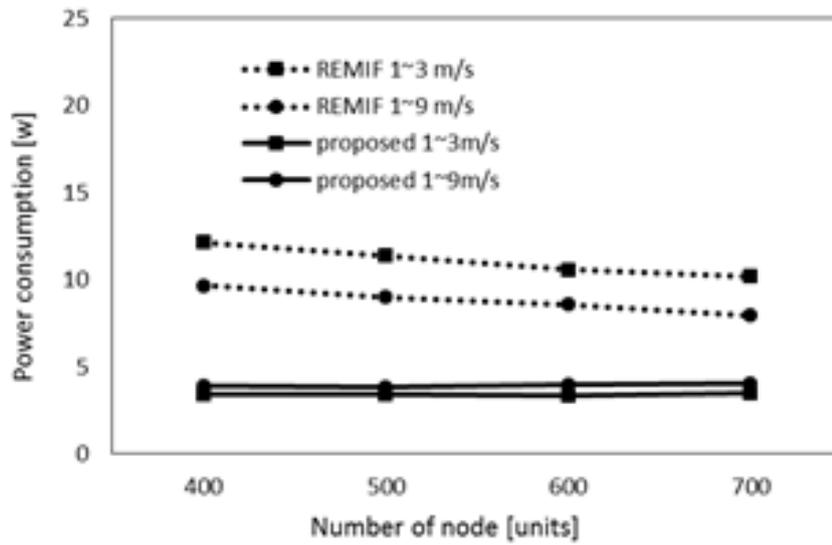


図 5.7 ノード台数に対する消費電力

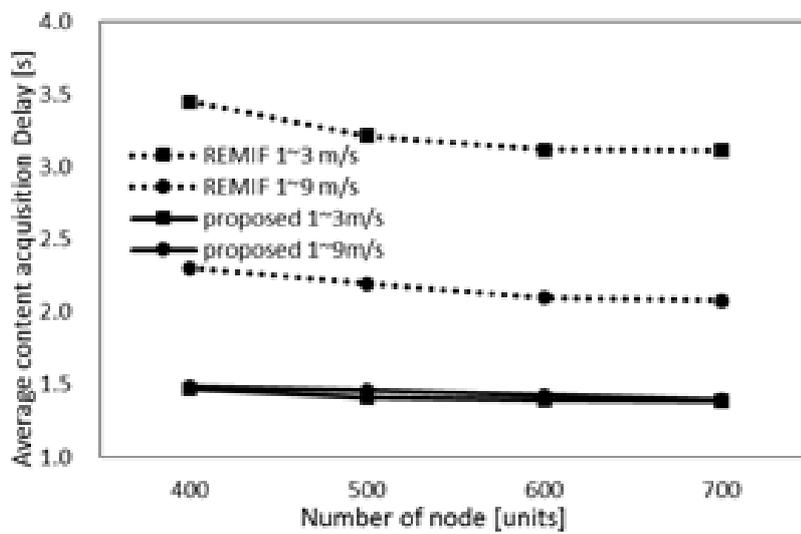


図 5.8 ノード台数に対するコンテンツ取得遅延

5.6 5章のまとめ

本章では、コンテンツ指向型ネットワークにおける MA を用いたコンテンツ取得手法を提案した。提案方式では MA が各ノードの位置情報とコンテンツリストを一括で管理し、コンテンツ取得のための経路を作成する。MA を用いるとことで、Interest パケットの送信にフラッディングを用いる必要がなくなり、コンテンツ要求に対して複数の要求コンテンツを保持するノードが存在した場合にも重複したコンテンツ送信が生じないため、負荷の削減につながる。計算機シミュレーションによる評価から、PDR、消費電力、コンテンツ取得遅延において優位性が示された。

参考文献

- [1] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking." *Communications Magazine*, vol. 50, no. 7, pp. 26-36, 2012.
- [2] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named data networking." *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 66-73, 2014
- [3] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content." In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, ACM, 2009, pp. 1-12.
- [4] J. Kim, D. Shin, and Y.B. Ko, "TOP-CCN: Topology aware Content Centric Networking for Mobile Ad Hoc Networks", 19th IEEE International Conference on Networks (ICON) 2013
- [5] M. Amadeo, A. Molinaro, and G. Ruggeri, "E-CHANET: Routing, forwarding and transport in Information-Centric multihop wireless networks." *Computer Communications*, vol. 36, no. 7, pp. 792-803, 2013.
- [6] R. A. Rehman, T. D. Hieu, H. Bae, S. Mah, and B. Kim "Robust and Efficient Multipath Interest Forwarding for NDN-based MANETs", *Wireless and mobile Networking Conference*, July.2016
- [7] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, "A distance Routing Effect Algorithm for Mobility (DREAM)," *Proceedings of the International Conference on Mobile Computing and Networking*, pp.76-84, 1998.

第 6 章

6. 結論

本論文ではモバイルアドホックネットワークの主要な課題であるルーチングの性能を向上させることを目的とし、分散処理技術の一つであるモバイルエージェント (MA) を利用したルーチング手法を提案した。さらに次世代のネットワークとして注目を集めているコンテンツ指向型ネットワークに MA を用いたルーチングを応用することで、効率的にコンテンツを取得する手法を提案した。

第 1 章では、研究の背景や目的について簡単に述べた。第 2 章では MA に全ノードの位置情報を管理させる方式について述べた。また、計算機シミュレーションによる性能評価の結果より、既存の方式に比べ制御パケットを大幅に削減するとともにパケット到達率が向上することを示した。第 3 章ではスループットの向上を目的とした MA 利用型マルチパスルーチングを提案した。計算機シミュレーションによる性能評価の結果より、既存の方式に比べスループットが大きく向上することを示した。第 4 章では GPS を使用せずに動作させることを目的とし、隣接ノード情報を利用した MA ルーチングを提案した。また、計算機シミュレーションにより、パケット到達率において位置情報を利用する方式よりも少し劣るものの AODV, OLSR よりは高い性能であることを示した。また、平均経路構築遅延と消費電力においても提案方式の性能は他の方式に比べ同等かそれ以上に優れていた。本方式の目的が位置情報を利用することによるデメリットや制限を除くことを考えると、総合的に判断して提案方式の有効性を示せたといえる。第 2~4 章の結果からアドホックネットワークのルーチングに対して MA を利用することでルーチングの性能を向上できることが示された。

第 5 章では MA を用いたルーチングをコンテンツ指向型ネットワークに応用し、効率的にコンテンツを取得する手法を提案した。計算機シミュレーションによる評価の結果、既存方式に比べ、PDR, 消費電力, コンテンツ取得遅延において高い性能を示すことが示された。

以上の研究成果から, **MA** をルーチングやコンテンツ取得に利用することでネットワーク性能の向上が見込めることが確認できた. モバイルアドホックネットワークは自律分散的に動作するという特性上ネットワーク負荷が大きいという問題があるが, **MA** を利用して情報を一元管理することがこの問題の解決策になることが示された.