

アドホックネットワークにおける階層ルーティングの 適応性と安定性について

藤本 宗彦 大田 知行 角田 良明

広島市立大学

あらまし アドホックネットワークでは、ノードの移動によりトポロジ変化が頻繁に起こるので、トポロジ変化に対する適応性が必要であるが、同時にトポロジ変化に依存しない安定性も必要である。以前、我々は、自律的に構成されたクラスタリングに、クラスタ内はテーブル駆動方式、クラスタ間はオンデマンド方式を適用した階層ルーティング Hi-TORA(Hierarchical TORA) を提案している。オンデマンド方式のルーティングでは、トポロジ変化に適応して経路を維持する必要がある、テーブル駆動方式のルーティングでは、トポロジ変化に適応して経路表を更新する必要がある。ところが、トポロジ変化に過度に適応すると、ノード間の接続性が悪くなり不安定な経路を提供する恐れがある。本稿では、Hi-TORA を適応性が高いと言われる TORA と比較して、ノード間の接続性の向上により、Hi-TORA は適応性だけでなく安定性も高いことを示している。

Adaptability and Stability of Hierarchical Routing in Ad Hoc Networks

Munehiko FUJIMOTO, Tomoyuki OHTA, and Yoshiaki KAKUDA

Hiroshima City University

Abstract Adaptability and stability to topology change are necessary for routing in ad hoc networks because nodes are frequently moving in the networks. Previously we proposed a hierarchical routing called Hi-TORA, in which, based on an autonomously configurable clustering, on demand routing is applied to routing within clusters and table driven routing is applied to routing among clusters. According to topological change adaptive route maintenance and routing table update are respectively required for on demand routing and table driven routing. However, due to excessive adaptability connectivity between nodes becomes worse and as a result these routings may provide unstable routes. In this paper, comparing Hi-TORA with TORA which is said to have high adaptability, we show that Hi-TORA has not only high adaptability but also high stability because of good connectivity between nodes.

1. はじめに

アドホックネットワーク [2] とは、基地局を必要としないモバイルホストの集まりによって形成されるネットワークである。アドホックネットワークは、災害救助活動、軍事活動、会議などのイベントなど様々な目的で利用可能である。近年のアドホックネットワークの進化に伴い、ネットワークサイズの増大が予想される。結

果として、大規模なネットワークを効率良く管理する必要がある。

現在、アドホックネットワークのためのルーティングが数多く提案されている。それらは、大別してプレーンルーティングプロトコルと階層ルーティングプロトコルがある。さらに、プレーンルーティングプロトコルには、テーブル駆動方式と [15] オンデマンド方式 [8], [13], [14] が存在する。テーブル駆動方式では、各モバイルホスト

が持つ経路表に基づいてデータが転送される。しかしながら、モバイルホストの頻繁な移動やネットワークの増大に伴い経路表を維持するための負荷が非常に高くなる。また、オンデマンド方式では、送信元がデータを転送したい時のみ、送信元と受信先間の経路を設定する。ネットワークサイズの増大に伴い、送信元と受信先間の経路の設定にかかる時間が増大する可能性がある。

一方、階層ルーティングプロトコル [7], [9], [11], [12] は、ネットワークを複数のクラスタに分割し、送信元から受信先への経路は、クラスタ内の部分経路とクラスタ間の部分経路で構成される。階層ルーティングプロトコルは、効率性と頑健性の目的で設計されている。

本稿では、文献 [4] で提案した階層ルーティングプロトコル Hi-TORA (Hierarchical TORA) を適応性が高いと言われる TORA と比較して、ノード間の接続性の向上により、Hi-TORA は適応性だけでなく安定性も高いことを示す。階層構造を実現するために、文献 [1] で提案したクラスタリング手法を使用している。Hi-TORA では、クラスタ内ルーティングにリンクステートプロトコル、クラスタ間ルーティングに TORA [13] を適用した。TORA は、トポロジ変化に対する適応性が高く、複数経路を提供する。しかしながら、TORA は、適応性が高すぎることに伴い、ノードの移動速度が高く、かつ、送信元と受信先間の通信が増加すると、パケットロスの発生により安定した通信を提供することができないことが、文献 [10] で報告されている。以後、モバイルホストは、ノードと記述する。

本稿の構成は、以下の通りである。2. は、文献 [6] で提案したクラスタリング手法について説明する。3. では、TORA について説明する。4. では、Hi-TORA について説明する。5. では、シミュレーション実験により、Hi-TORA の有効性を示す。最後に、6. では、将来課題とともにまとめを述べる。

2. クラスタリング手法

2.1 諸定義

アドホックネットワーク (以下、ネットワーク) は、有線リンクやそれを經由する基地局を持たず、移動ノードだけで構成される。各ノードは無線が有効な通信範囲内のノードとだけ通信可能である。ネットワークを無向グラフ $G = (V, E)$ で表す。各接点 $v_i \in V$ はノード ID i のノード (以下、ノード i と呼ぶ) に、各無向枝 $(v_i, v_j) \in E$ はノード i, j 間の無線リンクに対応する。このときノード v_i と v_j は隣接しているという。

2.2 クラスタの構成

クラスタは、ノードの部分集合で、一つのクラスタヘッ

ドと複数のノードからなる。クラスタヘッドとなるノードの識別子がこのクラスタのクラスタ ID となる。このとき、クラスタヘッドとクラスタヘッド以外のノードは、同じクラスタ ID を持つノードで連結していなければならない。また、クラスタの大きさを制限するために、あらかじめ最小構成数と最大構成数を定める。

[定義 1] クラスタヘッドが v_i であるクラスタ C をクラスタ C_i と呼ぶ。

[定義 2] クラスタ C_i のクラスタメンバとなるノードの個数 $|C_i|$ は、 $L \leq |C_i| \leq U$ でなければならない。

2.3 ノード

ノードは、以下の二つの基本的な動作を行なう。

- (1) 隣接ノードとの情報交換による情報収集。
- (2) クラスタ内ブロードキャストによるメンバ情報の収集。

ノードの持つ情報。ノード ID, クラスタ ID, 状態, クラスタメンバリスト。ブロードキャストによりクラスタメンバリストが更新される。

2.4 クラスタヘッド

クラスタを管理するノードをクラスタヘッドと呼ぶ。クラスタメンバリストにより、クラスタの大きさを管理する。大きさが L よりも少ない場合、隣接クラスタのいずれかと合体を行なう。隣接クラスタの大きさについては、ゲートウェイから情報を収集する。合体は、他のクラスタとの関係が必要となるため、合体するためにはネットワークに多少の影響を与える。

また、大きさが U よりも大きい場合、クラスタを二分割する。しかしながら、分割はそのクラスタ単独で行なうため分割中に他のクラスタに影響を与えることはない。どちらの場合においても、合体の場合はクラスタの消滅、分割の場合は新しいクラスタの生成により、その周辺のクラスタでは、隣接クラスタ情報の書き換えが必要となるためネットワークに多少の影響を与える。しかしながら、どちらの場合においても、隣接クラスタのみにしか影響を与えないため局所的にクラスタ構成の更新を行なうことができる。分割、結合がクラスタ構成の更新に与える影響については、文献 [5] にて考察を行なっている。

2.5 ゲートウェイ

ノード v_i の隣接ノードの中に、 v_i と異なるクラスタ ID を持つノード $v_j (j \neq i)$ が存在する時、ノード v_i をゲートウェイと呼ぶ。ゲートウェイ v_i は、 v_j とノード動作 1 により、 v_j の持つクラスタメンバリストにより、隣接クラスタのサイズを知ることができる。その情報をノード動作 2 により、 v_i のクラスタヘッドに伝える。

2.6 状態とノードの動作

この階層構造構成法では五つの状態を定義し、その状

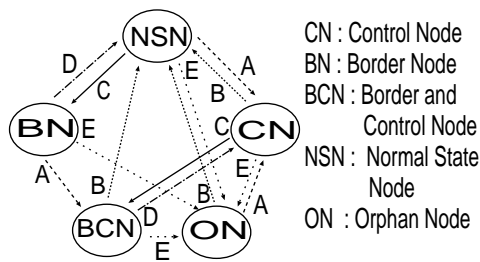


図1 ノード v_i の状態遷移図

態により、果たす役割を変化させる。各状態にあるノードの役割について説明する。

Normal State Node(NSN): ノードはクラスタヘッドや、ゲートウェイの機能を持たないノード。

Control Node(CN): ノードはクラスタヘッドの役割を果たす。

Border Node(BN): ノードはゲートウェイとなる。

Border Control Node(BCN): ノードはクラスタヘッドとゲートウェイの役割を担うことになる。

Orphan Node(ON): ノードの状態が ON のとき、ノードはクラスタ ID を持たない。新しくネットワークに加えられたノードは、最初、この状態である。

アドホックネットワークでは、常にノードが移動するので、ノードの移動に対して適応力があることは重要な要素の一つである。本階層構造構成法は、ノードの移動に対して高い適応性を示している [5]。

2.7 ノードの状態遷移

アドホックネットワークでは、クラスタを構成するための中心的存在となるクラスタヘッドや、クラスタとクラスタを結びゲートウェイの移動に伴い、別のノードがそのクラスタでの新しいクラスタヘッドやゲートウェイとして機能する必要がある。このような理由からノードは自律的に状態遷移をすることにより、クラスタの維持を行う必要がある。各ノードの状態の変化は、それらのノードの隣接ノードの状態によって決まる。図1にその状態遷移を示す。A から E までの5つの状態遷移条件を元に、クラスタ維持を各ノードの状態遷移により行う。

A: ノードは現在の役割に、クラスタヘッドの役割を追加する。B: ノードの状態は NSN に遷移し、通常のノードになる。C: ノードは現在の役割に、ゲートウェイの役割を追加する。D: ノードは現在の役割から、ゲートウェイの役割を削除する。E: ノードが保持していたクラスタ ID を削除し、ノードは状態 ON に遷移し孤立したノードになる。

3. TORA [13]

Temporally-Ordered Routing Algorithm (TORA)

[13] について説明する。TORA は、アドホックネットワークルーティングのための link reversal アルゴリズムの一種である。すべての到達可能な目的地のために、TORA は destination-oriented directed acyclic graph (DAG) を維持する。このグラフ構造では、あらゆるノードからリンクの論理的な方向に従うことにより、目的のノードへ経路が導かれる。TORA は、各リンクの方向を決定するために、“height” という表記法を用いている。通信要求がおこると、送信先から受信先に向けて、各ノードに順に小さい height が付けられる。このとき受信先の height には、ネットワーク中で最も小さい0がつく。また、動的に故障が起きても TORA は DAG を保ち続けるため、送信元からのデータは、受信先に到達可能である。

TORA は三つの基本的な機能を実行する。それは、経路生成、経路維持、経路削除である。それぞれの機能を実行するために、三種類の制御パケット (query (QRY), update (UPD), clear (CLR)) が使われる。

height の表記法について説明する。height は、 τ, oid, r, δ, i の五つの要素で構成される。各要素の意味は以下の通りである。

τ : new reference level

oid: new reference level を定義したノードの識別子

r: 反映指標ビット

δ : 伝搬順番変数

i: ノード識別子

TORA のルート生成について図2を用いて説明する。QRY と UPD パケットがルート生成のために使用される。図2において、二重丸のノードは、“route-required” フラグをセットされたノードを表す。ノード間の矢印は、論理リンクを表す。ここで、route-required フラグはノードがすでに隣接ノードから QRY パケットを受信したかどうかを管理するために使用される。

最初、図2(a)で示すように、各ノードの height は受信先ノード F を除いて NULL となっている。NULL は、 $H_i = (-, -, -, -, i)$ と表す。受信先ノード F は、height を ZERO = (0, 0, 0, 0, F) と設定する。ノード A が受信先ノード F への経路を必要とする時、QRY パケットをすべての隣接ノードへブロードキャストし、それを受信したノードは、route-required フラグをセットする。図2(b)で示すように、QRY パケットを受信したノードで route-required フラグが設定されていないノードは、route-required フラグをセットし、隣接ノードへ QRY を転送する (ノード B, C)。論理リンクを持つノード E と D が QRY パケットを受信した場合、height の δ を隣接ノードの height に基づいて修正する (図2(c))。このように、現在 height が NULL であるノードは、隣接ノード

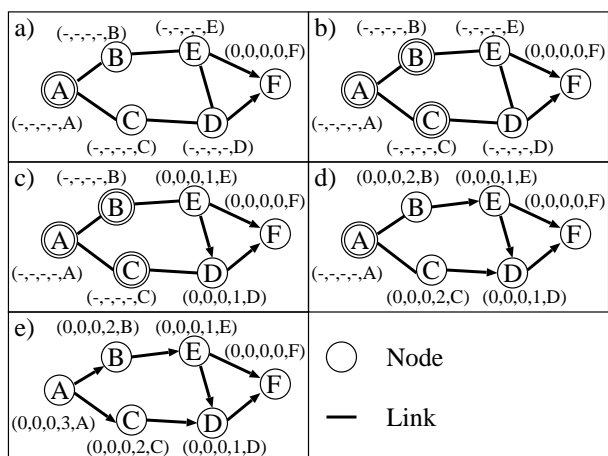


図 2 TORA の経路生成の例

ドの中で $NULL$ でない最小の height を持つノードに基づいて $(\tau, oid, r, \delta + 1, i)$ を実行する。そして、その新しい height を UPD パケットを使用して隣接ノードへブロードキャストする (図 2(d))。結果として、ノード A の height は、隣接のノード B の $H_B = (0, 0, 0, 2, B)$ を基に $(0, 0, 0, 3, A)$ と設定される。図 2(e) に示すような A から F までの DAG が生成されることにより、すべてのノードはこの論理リンクに沿ってデータパケットを転送することにより、受信先へ伝達することができる。

4. 階層ルーティングプロトコル Hi-TORA

本節では 2. で述べた階層構造構成法を基にした階層ルーティング Hi-TORA について述べる。Hi-TORA は、クラスタ内は最短経路で、クラスタ間はクラスタを仮想ノードとみなし、TORA でルーティングを行う。TORA は、送信先と受信先間に複数の経路を維持する。その処理はネットワーク全体が対象範囲になる。よって、ネットワーク中のノード数が増加すると、経路制御パケットが必然的に増加する。しかし、クラスタを 1 つの仮想ノードとしてルーティングを行うことで、ルート制御パケットを削減できる。Hi-TORA では文献 [1] の階層構造構成法により、TORA を大規模なネットワークに適用することを容易にした。

4.1 クラスタ内ルーティング

クラスタ内では、リンクステート型のルーティングを適用し、ホップ数による最短経路で通信を行う。送信先、または異なるクラスタ ID を持つゲートウェイからデータパケットを受け取ったゲートウェイは、そのクラスタ内通信の送信元になる。クラスタ内通信の受信先には、本来の受信先とゲートウェイがある。送信先と同じクラスタ内に受信先が存在すれば、送信先は最短経路で受信先にデータを送る。同じクラスタ内に受信先がない場

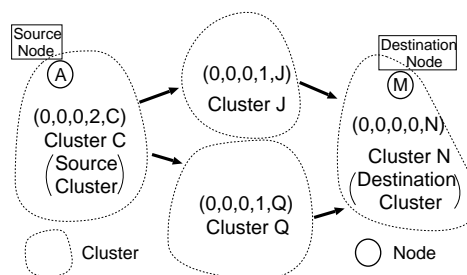


図 3 Hi-TORA でのルーティングのコンセプト

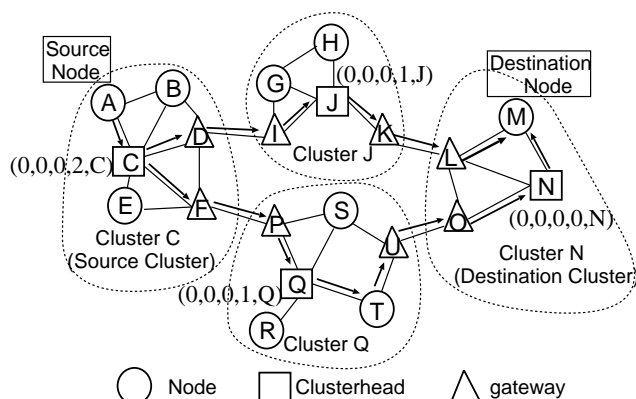


図 4 Hi-TORA でのルーティング例

合、送信先は次に送信すべきクラスタに隣接するゲートウェイにデータを送る。このとき複数のゲートウェイが存在するとき、近い位置にいるノードを選択する。

4.2 クラスタ間ルーティング

クラスタ間では、オンデマンド方式でネットワークの変化に高い適応性がある TORA を適用する。送信先が受信先に通信要求を出すと、受信先のいるクラスタ (以下、受信クラスタ) から順に height が設定される (図 3)。受信クラスタの height は、ネットワーク中で最も小さい 0 に設定される。このとき、図 4 で示すように、height は受信先ノードに設定されるのではなく、そのノードが所属するクラスタのクラスタヘッドに設定される。送信先のいるクラスタ (以下、送信クラスタ) から TORA の仕様に従い、より低い height をもつ隣接クラスタにデータを送信することで受信クラスタまでデータを届ける。

4.3 階層化による問題点とその解決法

TORA における通信では、送信先と受信先の ID は唯一の値に固定されているため、データの宛先は通信中固定である。しかし Hi-TORA では、送信、受信クラスタの ID が動的に変化するために、データが届かないという問題が生じる。文献 [1] の階層構造構成法に基づきネットワークを階層化すると、クラスタの ID にそのクラスタのクラスタヘッドの固有の ID がつけられる。クラス

タのクラスタヘッドの役割が、ノードの移動により別のノードに移ると、通信中に送信クラスタや受信クラスタの ID が変化する。このことにより、受信先になるクラスタを送信クラスタが見失ってしまい、データ通信が不可能になる。受信先のクラスタ ID の変化に伴い、受信、送信クラスタを動的に認識することが必要になる。

送信先、または受信先の所属するクラスタが変化した場合の処理を以下に示す。

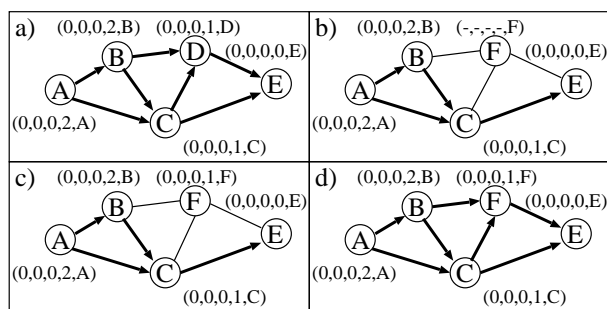
a) 送信先の場合

クラスタ ID が変化した送信先はクラスタヘッドの定期通知により、そのクラスタの新しい経路情報を得る。送信先の移動先のクラスタが受信クラスタのための経路情報を保持していれば、その情報に従うことで通信は可能である。受信クラスタのための経路情報がなければ QRY パケットを隣接クラスタにブロードキャストする。その後、height が更新されると、再び通信を開始する。

b) 受信先の場合

Hi-TORA では、受信クラスタはネットワーク中で最も低い height を持つ。この定義に基づき、受信先は新しく所属するクラスタのクラスタヘッドに height を最も低くするように促すパケットを送信する。height を更新したクラスタヘッドはこの更新した情報を、隣接するクラスタにブロードキャストする。一方、ノードの移動によって受信先を失ったクラスタは、受信先がいなくなったことを認識したら、height を NULL にして UPD を待つ。このようにして受信先が存在するクラスタが、ネットワーク中で最も低い height を持つことで問題を解決している。

最後のケースとして、送信元クラスタと受信先クラスタ間に存在する中間クラスタのクラスタ ID が変わった場合についての処理方法を図 5 を用いて述べる。図中の丸はクラスタ、線はリンク、矢印は論理リンクを表す。Hi-TORA では、ネットワーク中に DAG を設定する。あらゆるノードは受信先ノード E へ論理リンクの方向に沿ってデータパケットを送る。しかし、ノードは移動するため、クラスタヘッドの移動によりクラスタ ID の変更が起こる可能性がある。図 5a) のクラスタ D のクラスタヘッドがクラスタ外へ移動したと想定する。そのとき、クラスタ D はクラスタリング手法に基づいて、クラスタ ID を F へ変更する (図 5b))。新しくクラスタ ID が設定されたクラスタの height は NULL に設定される。TORA の height の計算に従って、クラスタは、NULL から $(\tau, oid, r, \delta + 1, i)$ へ変更する必要がある。この例では、すべての隣接クラスタの height はそれぞれ、 $(0, 0, 0, 2, B)$ 、 $(0, 0, 0, 1, C)$ 、 $(0, 0, 0, 0, E)$ である。この中で、最小の height は、 $(0, 0, 0, 0, E)$ である。TORA



○ Cluster — Link

図 5 Hi-TORA での中間クラスタのクラスタ ID が変更した場合の例

の height の計算に従って、クラスタ F はその height を NULL から $(0, 0, 0, 1, F)$ へ変更する (図 5c))。結果として、クラスタ F は図 5d) で示すように、すべての隣接ノードとの間に論理リンクを持つ。クラスタからクラスタヘッドが出ていってしまったとしても、DAG を再構成することにより、データパケットを伝達することが可能となる。

5. シミュレーション実験

ルーティングの安定性と適応性に関する評価を行なうために、制御パケット数、データパケットの到達数、送信元から受信先までの遅延時間 (ホップ数) の点においてシミュレーション実験を行なった。それぞれの点において、Hi-TORA と TORA の比較を行なった。

5.1 実験環境とパラメータ

今回のシミュレーション実験での実験環境について説明する。ネットワークの大きさを $750m \times 750m$ とし、その中に存在するノードを 200 ノードとする。ノードの初期位置はランダムに配置する。ノードの移動方法は、randomway point model [10] を採用している。この方法は、目的座標まで与えられた平均速度で移動し、目的地に到着すると、ある一定時間その目的地で停止する。これを停止時間と呼ぶ。今回の実験では、この停止時間 (pause time) を 0 とした。そして、新しい目的座標を決め、再び新しい目的座標まで移動を開始する。以上の動作を繰り返す。ノードの平均移動速度を $0m/s$ 、 $1m/s$ 、 $5m/s$ 、 $10m/s$ 、 $20m/s$ とする。

また、各ノードの通信可能範囲は半径 $100m$ とし、任意のノード間の遅延時間は、パケットを送受信の処理時間も含めて $1msec$ とする。今回の実験では、ノード間の帯域は考慮しない。

次に、各ノードが隣接ノードの状態やクラスタ ID を定期的にチェックするための Hello の間隔を $200ms$ とし、

表 1 TORA の平均制御パケット数

V(m/s)	A
0	3419.60
1	20564.20
5	108421.60
10	194460.20
20	373946.20

V:ノードの移動速度
A:平均制御パケット数

すべてのノードでの定期通知の間隔もそれぞれ $200ms$ とする．シミュレーション全体の時間を $65sec$ に設定し，最初の $5sec$ のデータはすべて破棄し， $5sec$ 以降から， $60sec$ 間ノードを動かす．

シミュレーションは，以下のように進める．送信元と受信先ノードを一つずつ決定し，一つの SD ペアを選択する． $5sec$ 経過後，送信元が受信先までの経路を生成し，生成後から $200ms$ ごとに $65sec$ までデータパケットを送信する．TORA では，経路を生成し，維持するための制御パケットが必要である．また，Hi-TORA では，経路を生成し，維持するための制御パケットに加え，階層構造を維持し，クラスタ内の経路表を作成するための制御パケットが必要である．制御パケット数の比較として，TORA で必要となる制御パケット数と Hi-TORA で必要となる制御パケット数の数を比較する．

$5sec$ から $65sec$ までの間に，SD ペア間に最大で 300 個のデータパケットを送信することができる．受信先のノードが 300 個のデータパケットの中で何個のデータパケットを受信することができたかを測定することで，パケットの到達度に関する比較を行なう．

最後に，この到達したパケットが通過したノードの個数を計測することによって，送信元と受信先間の経路のホップ数を計測する．以上の三つの要素を計測することにより，Hi-TORA の有効性を示す．

また，Hi-TORA の場合の各クラスタのクラスタメンバ数の範囲 (L, U) を，それぞれ， $(10, 25)$ ， $(20, 50)$ ， $(30, 75)$ とした．

5.2 制御パケットに関する結果

表 1 に TORA の各速度での平均経路制御パケット数の計測結果を示す．各値は，1 つの SD ペアの通信を 60 秒間管理するために必要な平均制御パケット数である．表 2 に Hi-TORA の各速度での平均経路制御パケット数と，ネットワークの階層化に使用された平均制御パケット数の計測結果を示す．TORA は，1 つの SD ペアの通信のためにすべてのノードが経路を持つため，ノードの移動によるトポロジ変化に適応するために設定した経路を維持するために制御パケット (UPD パケット) を送信する．

一方，Hi-TORA はトポロジ変化に対してクラスタ単位で適応するため経路維持のために必要とする制御パケットの数は TORA と比べて少ない．

次に，クラスタリングのために必要な制御パケットも含めた全制御パケットに関しての Hi-TORA の有効性を述べる．Hi-TORA は，クラスタ単位でトポロジ変化に適応するため，経路制御パケットの数は少ないが，クラスタリングのために，非常に多くのパケットを使用している．Hi-TORA の制御パケットの総数が TORA より少なくなる (以下，Hi-TORA が有効になると言う) ためにはネットワークに複数の SD ペアが存在する必要がある．TORA は SD ペア毎に違った経路を維持する．よって経路制御パケット数は SD ペア数と平均経路制御パケット数の積で表すことができる．ここで，Hi-TORA が有効になる最小の SD ペア数を以下の手順で求めた．

TORA と Hi-TORA における平均経路制御パケット数に注目する．SD ペア数を N とし，TORA における経路制御パケット数を T_p ，Hi-TORA における経路制御パケット数を HiT_p ，とするとそのときに必要なそれぞれの経路制御パケット数は

$$N \times T_p, N \times HiT_p$$

と表すことができる．以上により，Hi-TORA が有効になる最小の SD ペア数 N_{min} を求める．ここで階層構造を維持するために必要な制御パケット数を Hi_i_p とおくと，

$$N \times T_p > N \times HiT_p + Hi_i_p \quad (1)$$

を満足する最小の自然数 N が N_{min} になる．(1) を変形すると，

$$N > \frac{Hi_i_p}{T_p - HiT_p} \quad (2)$$

となり，この不等式 (2) は

$$N_{min} = \left\lceil \frac{Hi_i_p}{T_p - HiT_p} \right\rceil + 1 \quad (3)$$

となる．

ここで表 1，2 の各速度における値を用い，(3) によって N_{min} を求めた．表 3 はその結果を示したものである．

表 3 で示されるように，ノードの移動速度が上がるに伴い，少しの SD ペア数で Hi-TORA の方が TORA と比べて効率的になっている．ネットワークの規模の増加に対し，SD ペア数は自乗的に増加すると考えられる．SD ペア数の増加に伴い，経路を生成するための制御パケットの数が Hi-TORA を使用した場合が，TORA を使用した場合と比較して制御パケット数の増加を抑えることができる．つまり，ネットワークの規模が大きいほど Hi-TORA

表 2 Hi-TORA の平均制御パケット数

V (m/s)	Cluster Range					
	10-25		20-50		30-75	
	B	C	B	C	B	C
0	72.20	234223.40	34.20	204655.80	18.00	215148.40
1	224.00	243414.60	56.40	211626.60	24.60	211571.60
5	1932.40	323456.00	363.40	252834.20	84.60	230652.60
10	6518.00	380812.00	1251.80	268497.80	907.00	235619.00
20	9614.80	434067.00	2559.00	281446.00	1465.20	257470.80

V:ノードの移動速度 B:ルーティングのための平均制御パケット数 C:クラスタリングのための平均制御パケット数

表 3 Hi-TORA が有効になる最小の SD ペア数

V(m/s)	Cluster Range		
	10-25	20-50	30-75
0	70	61	64
1	12	11	11
5	4	3	3
10	3	2	2
20	2	1	1

V:ノードの移動速度

表 4 受信先が送信元から受信したデータパケットの数

V(m/s)	TORA	Hi-TORA		
		Cluster Range		
		10-25	20-50	30-75
0	299.80	299.00	299.00	299.20
1	172.80	286.80	298.00	298.00
5	141.80	229.40	283.00	263.80
10	180.40	181.00	193.00	274.80
20	77.20	132.20	171.20	261.60

V:Moving Speed of Node

が有効になる。以上より、ノードの移動が高速の場合や、ネットワークの規模が大きい場合に Hi-TORA が有効になることを示した。

5.3 パケットの到達数に関する結果

表 4 にデータパケットの到達数を示す。ノードの移動速度が高くなるに伴い、到達数が下がっていることがわかる。また、Hi-TORA ではクラスタサイズが大きいほど到達数の減少が抑えられている。

クラスタのサイズが大きい場合、送信元と受信先が同一のクラスタ内に所属する可能性が高くなる。クラスタ内ルーティングは、クラスタ間ルーティングと比較して高い接続性を持つ。クラスタのサイズが (30, 75) の場合、受信先が受け取るデータパケットの数はノードの移動速度に関係なく高い水準となっている。

TORA は、高い適応性を持っていることを制御パケットに関する実験で示したが、表 4 で示されるように、ノードの移動速度が高くなるに伴いデータパケットの到達数が下がっていることがわかる。TORA により設定される経路の安定性が低いことを意味する。結果として、Hi-TORA は、TORA と比較して高い安定性を持つと言える。

5.4 ホップ数に関する結果

表 5 は TORA と Hi-TORA における送信元から送られたデータパケットが受信先に到達するまでに必要としたホップ数を示している。Hi-TORA で必要としたホップ数は、すべての結果において TORA よりもホップ数を

余分に必要している。この理由は、各クラスタが自律的に動作していることが原因であると考えられる。クラスタ間では、各クラスタの持つクラスタメンバ数のみを行なう。クラスタメンバ数以外の情報は、クラスタ間で交換しなくても、各クラスタは自律的に送信元から受信先への経路を生成する。

図 6 を例として説明する。ノード A - F はゲートウェイである。同じ隣接クラスタと隣接するゲートウェイは各クラスタ内に一つ以上存在する状況を考える。クラスタ I 内のゲートウェイ A が隣接クラスタからデータパケットを受信したとする。このとき、そのデータパケットの宛先となるノードがクラスタ I 内に存在しないため、そのデータパケットをクラスタ間ルーティングにより得られた経路情報に基づいて隣接クラスタ J に転送する。今、クラスタ I 内には、隣接クラスタ J と隣接するゲートウェイ B, C が存在する。このとき、ゲートウェイ A は、ホップ数の近いゲートウェイ C に向けてデータパケットを転送する。同様に、ゲートウェイ C からデータパケットを受信したクラスタ J 内のゲートウェイ E は、さらに隣接クラスタへ転送するためにゲートウェイ F へ向けてデータパケットを転送する。結果として、A, C, E, F という経路が選択される。しかしながら、クラスタ I と J の両方のクラスタを合わせた場合、先ほど選択した経路よりも短い、A, B, D, F という経路が存在する可能性がある。

表 5 データパケットのホップ数

V(m/s)	TORA	Hi-TORA		
		Cluster Range		
		10-25	20-50	30-75
0	8.00	9.40	10.40	10.60
1	7.66	10.92	9.42	9.97
5	7.87	9.55	9.35	9.14
10	5.74	7.08	6.66	6.37
20	3.63	4.72	4.97	5.40

V:ノードの移動速度

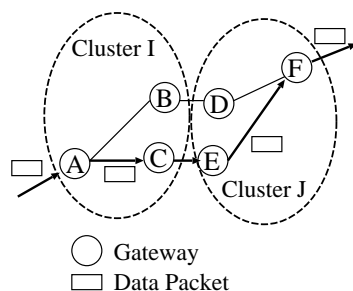


図 6 階層ルーティングのホップ数に関する問題例

また、ノードの移動速度が高くなるに連れて、ホップ数が短くなっているのは、ノードが活発的に動いたことにより、送信元と受信先の間が近くなった時が比較の長い間があったと考えられる。今回の実験環境では、ネットワークの両端に存在するノード間は、最も理想的に通信範囲を利用した場合で 8 ホップで到達可能である。0m/s や 5m/s の場合は、送信元と受信先の距離は十分に離れていて、さらに、ほとんどノードが移動したいため、ホップ数が多いと考えられる。

アドホックネットワークでは、データパケットの到達率の方が遅延時間より重要であると考えられる。Hi-TORA のデータパケットの到達数は、ノードの移動速度が高い場合においても、TORA の到達数と比較して十分高い。TORA の方が良い結果を示しているが、到達数に関する結果において、Hi-TORA な安定性を持つことを示したため、ホップ数に関しては十分に許容可能な差であると考えられる。

6. おわりに

本稿では、文献 [4] で提案した階層ルーティングプロトコル Hi-TORA を適応性が高いと言われる TORA と比較して、ノード間の接続性の向上により、Hi-TORA は適応性だけでなく安定性も高いことをシミュレーション実験により示した。今後の課題として、実際のネットワーク環境に基づいた実装により Hi-TORA を評価していく予定である。

文 献

- [1] T.Ohta, S.Inoue, Y.Kakuda, and K.Ishida: *An adaptive multihop clustering scheme for ad hoc networks with high mobility*, IEICE Trans. Fundamentals, Vol.E86-A, No.7, pp. 1689–1697 (2003).
- [2] 角田, 大田: アドホックネットワークルーティング, オペレーションズ・リサーチ, Vol.48 No.3, pp. 189–195 (2003).
- [3] T.Ohta, S.Inoue, and Y.Kakuda: *An adaptive multihop clustering scheme for highly mobile ad hoc networks*, Proc. 7th IEEE Int'l Symp. on Autonomous Decentralized Systems (ISADS2003), pp. 293–300 (2003).
- [4] T.Ohta, M.Fujimoto, S.Inoue, and Y.Kakuda: *Hi-TORA: hierarchical routing protocol in ad hoc networks*, Proc. 7th IEEE Int'l Symp. on High Assurance Systems Engineering (HASE2002), pp. 143–148 (2002).
- [5] T.Ohta, S.Inoue, Y.Kakuda, K.Ishida, and K.Maeda: *An adaptive maintenance of hierarchical structure in ad hoc networks and its evaluation*, Proc. 1st Int'l. Workshop on Assurance in Distributed Systems and Networks (ADSN2002), pp. 7–13 (2002).
- [6] T.Ohta, K.Ishida, Y.Kakuda, S.Inoue, and K.Maeda: *Maintenance algorithm for hierarchical structure in large ad hoc networks*, Proc. Int'l Conf. on Fundamentals of Electronics, Communications and Computer Sciences (ICFCS2002), pp. S3-12–S3-17 (2002).
- [7] C.E.Perkins: *Ad hoc networking*, Addison-Wesley, pp. 221–253 (2001).
- [8] C.E.Perkins and E.M.Royer: *Ad hoc on-demand distance vector routing*, Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications, pp. 90–100 (1999).
- [9] S.Basagni: *Distributed clustering for ad hoc networks* Proc. Int'l Symp. on Parallel Architectures, Algorithms, and Networks (I-SPAN'99), pp. 310–315 (1999).
- [10] J.Broch, D.A.Malts, D.B.Johnson, Y.-C. Hu, and J.Jetcheva: *A performance comparison of multi-hop wireless ad hoc network routing protocols*, Proc. ACM/IEEE MOBICOM, pp. 85–97 (1998).
- [11] C.Chiang, H.Wu, W.Liu and M.Gerla: *Routing in clustered multihop, mobile wireless networks with fading channel*, Proc. IEEE Singapore Int'l Conf. on Networks (SICON), pp. 197–211 (1997).
- [12] C.R.Lin and M.Gerla: *Adaptive clustering for wireless networks*, IEEE Journal on Selected Areas in Communications, Vol. 15, No. 7, pp. 1265–1275 (1997).
- [13] V.D.Park and M.S.Corson: *A highly adaptive distributed routing algorithm for mobile wireless networks*, Proc. IEEE INFOCOM'97, pp. 1405–1413 (1997).
- [14] D.Johnson and D.A.Maltz: *Dynamic source routing in ad hoc wireless network*, Mobile Computing, Kluwer Academic Publishers (1996).
- [15] C.E.Perkins and P.Bhagwat: *Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers*, Proc. SIGCOMM'94, pp. 234–244 (1994).