

# MANET における階層化 QoS マルチキャストルーティング方式 とその性能評価

高島 栄一 村田 佳洋 柴田 直樹<sup>†</sup> 安本 慶一 伊藤 実

奈良先端科学技術大学院大学情報科学研究科 <sup>†</sup> 滋賀大学情報管理学科

あらまし モバイルアドホックネットワーク(以下, MANET)のアプリケーションの一つとして, 動画配信サービスがある. 質の良い動画配信サービスを実現するためには, 遅延やパケットロス率など様々なメトリックを考慮したマルチキャスト木を構築する必要がある. しかし, 2つ以上のメトリックを同時に考慮した QoS マルチキャスト木作成問題は NP 困難である. この問題に対する近似解法の一つとして遺伝的アルゴリズム(GA)を用いる方法が研究されてきた. しかし, 既存手法では, 集中制御方式が用いられているため, 各ノードの計算資源および通信資源に限られている MANET 上に適用するにはスケーラビリティの点で問題がある. 我々は, MANET 上でユーザの要求を満たすマルチキャスト木を複数ノードで分散して計算し構築する方法を提案している. 提案手法では, MANET を複数のクラスタに分割し, 各クラスタ内およびクラスタ間の 2 階層でそれぞれ配送経路を算出することにより, スケーラビリティを高める工夫を行っている. また, 配送経路算出の際, 並列 GA の一種である島モデル GA を用いることで並列計算による高速化も行っている. 本稿では, 提案手法の実装, および性能評価について述べる. 提案手法を評価するために, ネットワークシミュレータ GTNetS を用いて実験を行った. 実験の結果, クラスタ間の配送経路を DSR などの既存プロトコルで行った場合は切断が頻繁に起こるため改善を要すること, クラスタ内での配送経路は, 平均遅延時間, 平均データ到達率の面で十分実用的な性能が達成できることなどを確認した.

## Performance Evaluation of Hierarchical QoS Multicast Routing in MANET

Eiichi Takashima Yoshihiro Murata Naoki Shibata<sup>†</sup> Keiichi Yasumoto Minoru Ito

Grad. Sch. of Info. Sci., Nara Institute of Sci. and Tech. <sup>†</sup>Dept. of Info. Proc. and Man., Shiga Univ.

**Abstract** In this paper, we propose a new method to find a multicast tree which satisfies user's QoS requests on mobile ad-hoc network (MANET). The problem to find a multicast tree which satisfies restrictions over more than one QoS metrics such as delay and bandwidth is an NP-hard problem. Several methods utilizing Genetic Algorithm (GA, hereafter) which calculate an approximate solution of the problem have been proposed. However, existing GA-based methods assume that a multicast tree is calculated by one of nodes in a centralized manner. This lacks scalability due to costs of gathering topology information and calculating the tree. In this paper, a GA-based method to find a multicast tree which satisfies user's request with more than one QoS metrics in MANET. In our method, the entire MANET is divided into clusters, and a multicast tree is calculated hierarchically for inter-cluster and for intra-cluster. Our algorithm calculates an approximate solution which satisfies all of QoS restrictions. We used island GA, which is a kind of parallel GA, to increase scalability. In order to evaluate effectiveness of our method, we conducted experiments using network simulator called GTNetS. As a result, we confirmed that we need to improve overlay link robustness in an inter-cluster multicast tree since links established with DSR are likely to break, and that intra-cluster multicast trees can achieve expected performance with respect to packet delivery ratio and delay.

### 1 はじめに

モバイルアドホックネットワーク(以下, MANET)のアプリケーションの一つとして, 動画配信サービスがある. 動画配信の経路を決定するために, リンクのホップ数の総和を最小化するようなマルチキャスト木を算出する手法が提案されている [1]. しかし, 動画配信サービスには, 遅延, 帯域幅やパケットロス率など複数の異なるパラメタに対する品質保証の要求があり, これら複数の品質保証要求を同時に満たしかつコスト最小となるマルチキャスト木を算出する問題は NP 困難であることが知られている. また, MANET 上では, ノードの移動にともないリンクが生成・消滅するため, トポロジが動的に変化する. 従って, マルチキャスト木をトポロジの変化にあわせ再計算する必要がある.

コスト最小のマルチキャスト木を算出する問題に対する近似解法の一つとして遺伝的アルゴリズム(GA)を用いた方法が研究されている [2, 3]. しかし, これらの既存手法は集中制御方式を用いており, 全てのトポロジ情報を一箇所に集めるための通信コストと, 全経路を算出する計算コストが大きい. 従って, 大規模なアドホックネットワーク上では, 計算資源に乏しい移動ノード上でこれらの算出を行わせることは困難である.

我々は, MANET 上で, 要求された帯域, 遅延などの制約を満たし, かつ, リンクのホップ数の総和などの基準で準最適なマルチキャスト木を効率よく構築する手法 (Hierarchical QoS Multicast Routing Using GA in MANET, 以下 HQMGA) を提案している [4]. この手法ではネットワークを複数のクラスタに分割し, 各クラスタ内およびクラスタ間の 2 階層

でそれぞれ経路を算出することにより、負荷分散を行っている。これにより、クラスタ毎にトポロジ情報を集めるため、通信コストを小さくできる。また、マルチキャスト木をクラスタ間、クラスタ内で独立に算出するため、木の算出を行うノードの計算負荷を小さくできる。

本稿では、提案手法 HQMGA の実装、およびネットワークシミュレータ GTNetS [5] により性能評価を行った結果を報告する。実験の結果、クラスタ間の配送経路を DSR [6] などの既存プロトコルで行った場合は切断および再構築が頻繁に起こり、パケット到達率、遅延は期待した性能を達成できなかった。今後、切断しにくいクラスタ間経路の確立方法を考案する必要がある。一方、クラスタ内での配送経路は、平均遅延時間、平均データ到達率の面で十分実用的な性能が達成できることが確認できた。

## 2 関連研究

MANET における QoS ルーティングについて、幾つかの手法が提案されている。

単一の QoS のメトリックの制約条件を扱う手法として、文献 [7, 8, 9, 10] が提案されている。文献 [7] は、帯域幅（遅延）に対する QoS のユニキャストルーティングの手法である。これは、ある一定量の論理 ticket を使って、限定的フラッディングを行い、経路の QoS を調べる。各 probing メッセージには、一つ以上の ticket が含まれる。probing メッセージがあるノードに到達したとき、次ホップ毎に子の probing メッセージを作り、ticket を分配する。そして、それぞれ異なる次ホップに転送する。残り ticket 数が一つになると、それ以上、分けることはできないので、フラッディングを限定させることができる。一つ以上の probing メッセージが目的ノードに到着したとき、経路とその帯域幅（遅延）が分かる。これにより、QoS を満たす経路を予約することができる。

文献 [8, 9] は、帯域幅に対する QoS のユニキャストルーティングの手法である。前者は、プロアクティブ型のユニキャストルーティングである。後者は、AODV を用いた TDMA に基づくオンデマンド型のユニキャストルーティングである。これらの手法では、TDMA を用いることで、時間をタイムスロットに分け、利用可能な帯域幅を分ける。その上で、文献 [8] では、利用可能な帯域幅から、QoS を満たすタイムスロットの割り当てを行う。この割り当て問題は、NP 困難であるため、ヒューリスティックを用いたアルゴリズムが提案されている。一方、文献 [9] では、AODV のルート要求の過程を使って、ホップ毎の利用可能な帯域幅を算出する。目的地のノードでは、QoS の要求を満たす要求メッセージだけ応答を返す。これにより、ソースノードから目的地までの QoS の要求を満たす経路を予約できる。

文献 [10] は、複数の経路を用いた QoS ルーティングの方法である。文献 [7] などの他の多くの手法では、ソースから目的地までの QoS を満たす 1 つの経路を探索する。この手法では、まず、文献 [7] と同様に ticket を使って、限定的フラッディングを行い、さまざまな経路の QoS を調べる。そして、QoS 情報が分かっている複数の経路を組み合わせることで、要求された帯域幅を確保する。1 つの経路では要求された QoS を満たすことができないネットワーク資源が限られた環境でも複数の経路を組み合わせることで確保できる。

複数の QoS のメトリックの制約条件を同時に扱う手法には、以下のものがある。文献 [2] は、MANET 上の GA を用いたユニキャストルーティングの手法である。この手法では、トポロジ情報を抽象化することで、マルチキャスト木を高速に算出することができる。文献 [3] は、無線環境下は不

安定なネットワークであることを考慮し、QoS のメトリックを確率的に満たすマルチキャスト木を算出する手法を提案している。

MANET 環境下では計算資源が限られていることを考慮し、ノードの集合を分けそれぞれ分散制御で配送経路を算出する手法がいくつか提案されている。文献 [11] は、帯域幅に対する QoS のユニキャストルーティングの手法 CEDAR を提案している。CEDAR では、ネットワークのコアとなるグラフを動的に確立する。そして、コアとなるグラフ内で安定した経路を用意する。ソースノードと目的ノードは、このコアとなるグラフを経由することで、経路を効率よく算出している。

文献 [12] は、文献 [11] の手法を拡張し、マルチキャストルーティングに対応させた手法である。文献 [11] では、コアとなる経路を用意したが、文献 [12] では、コアとなるメッシュ型のマルチキャストグループを用意し、このマルチキャストグループを経由してデータを伝播させることで、効率よく配送経路を算出している。

以上述べたように、MANET 上での QoS を考慮した経路構築手法として様々なものが提案されているが、複数の QoS メトリックに対する制約を同時に満たすコスト最小なマルチキャスト木を分散制御により算出する方法は提案されていない。

## 3 ネットワークモデル

本稿では、MANET を重みつき無向グラフ  $G = (V, E)$  で表す。ここで  $V, E$  は、それぞれノードの集合とノード間に存在するリンクの集合を表す。各ノードには固有の ID (整数値) が与えられるとする。ここで、2 つのノード  $v_i, v_j$  が 1 ホップの無線範囲内にある時に限り、これらのノード間にリンク  $\langle v_i, v_j \rangle$  が存在するとする。また、各リンクの遅延、帯域幅、コスト (リンクパラメタと呼ぶ) は、そのリンクの両端ノードが知ることができると仮定する。マルチキャスト木  $T' = (V', E')$  は、 $G$  の部分グラフであり、1 つのソースノード  $s \in V'$  を必ず含む。したがって、 $v \in V'$  から  $s$  までの  $T'$  上の経路は一意に決まる。経路は、 $\text{path}(v, s, T') = \{ \langle v, v_{i1} \rangle, \langle v_{i1}, v_{i2} \rangle, \dots, \langle v_{ik}, s \rangle \}$  とする。経路がない場合、 $\text{path}(v, s, T') = \emptyset$  とする。

各リンク  $e \in E$  に対する遅延、帯域幅、コストを、それぞれ  $\text{delay}(e)$ ,  $\text{band}(e)$ ,  $\text{cost}(e)$  と表記する。 $\text{path}(v)$  に対する遅延と帯域幅を以下のように定義する。

$$\text{Delay}(\text{path}(v, s, T')) = \sum_{e \in \text{path}(v)} \text{delay}(e),$$

$$\text{Band}(\text{path}(v, s, T')) = \min_{e \in \text{path}(v, s, T')} \text{band}(e)$$

ただし、 $\text{path}(v, s, T') = \emptyset$  のとき、遅延  $\text{Delay}(\text{path}(v, s, T'))$ 、帯域幅  $\text{Band}(\text{path}(v, s, T'))$  は、それぞれ無限大、0 とする。また、木  $T'$  に対するコストを以下のように定義する。

$$\text{Cost}(T') = \sum_{e \in T'} \text{cost}(e)$$

## 4 問題設定

グラフ  $G$  において、いくつかのノードが動画の配信要求を送信するものとする。これらのノードをユーザとし、その集合を  $U \subset V$  として表す。また、配信要求には、2 つの制約条件 (必要帯域幅  $B_r$  以上と許容遅延時間  $D_r$  以内) が与えられるとする。このとき、これらの制約条件が両方満たされたユーザの数を最大にし、リンクのコストの和を最

小にするようなマルチキャスト木を算出することが目的である。

制約条件が満たされたユーザ数とコストの和のトレードオフは重み付け係数  $\alpha$  によって調整することとする。従って、本問題の目的関数  $F$  は最大化関数であり、求めた木  $T'$  に対して次の式で評価する。

$$\max. \sum_{u \in U} \text{deliver}(u, s, T', Dr, Br) - \alpha * \text{Cost}(T')$$

ただし、 $\text{deliver}(u, s, T', Dr, Br) =$

$$\begin{cases} 1 & \text{if } \text{path}(u, s, T') \neq \emptyset \wedge \text{Delay}(\text{path}(u, s, T')) \leq Dr \\ & \wedge \text{Band}(\text{path}(u, s, T')) \leq Br \\ 0 & \text{otherwise.} \end{cases}$$

である。

## 5 提案手法

### 5.1 提案手法の概要

提案手法では、スケーラビリティの確保、トポロジの変化への対応の2点の達成を目指す。

スケーラビリティの確保: MANET 上で GA を用いて QoS を考慮したマルチキャスト木を算出する際、従来の集中制御方式 [2, 3] を用いると、すべてのリンクパラメタの情報を一箇所に集める必要があり、そのための通信量が多くなる。また GA では、解候補となるマルチキャスト木を何度も評価して解を進化させる必要があり、この評価は木の大きさに依存した計算量が必要であるため、ノード数が増えるとともに計算量が線形以上の割合で増える。MANET ではノードの計算資源が限られていることが多く、単一ノードでの処理では比較的小規模のネットワークに限られる。

提案手法では、ネットワーク上のノード集合をクラスタに分割する。次に、クラスタを結ぶバックボーンとなる木を算出する。また、各クラスタ内で配送するための木(これをクラスタ内配送木と呼ぶ)を算出する。そして、これらの木を組み合わせることで、マルチキャスト木を算出する。さらに、各クラスタ内配送木の算出は、それぞれのクラスタ内で行うことにより、負荷分散を行う。さらにこれらの木の算出には、各クラスタ内で島モデル GA [13] を用いて並列計算を行う(図 5)。以上の方法により、大規模な MANET にも提案手法が適用できるようにする。

トポロジの変化への対応: MANET 環境では、ノードは時間とともに移動する。そのために生じる経路の切断等は、経路上の QoS が悪化する原因となる。そのため、ノードの移動に応じたマルチキャスト木の再構築が必要となる。

小規模の局所的なトポロジの変化は、高い頻度で発生するため、クラスタ内配送木だけを短期的に、再構築することにより対応する。一方、大規模の大域的なトポロジの変化は、短期的にはあまり起こらないと考えられるので、比較的長い周期でクラスタ間配送木を再構築する。

また提案手法では GA を用いているため、前回の算出に用いた解候補を流用して、新しい木を高速に算出することができる。

### 5.2 クラスタ

提案手法で用いるクラスタとそのクラスタを管理するクラスタヘッドについて定義する。クラスタとはネットワーク上の全てのノード集合  $V$  のある分割  $S$  の各要素と考える。これを  $V_i$  とする。 $V_i$  内のノードを結ぶリンクの集合を考える。これを  $E_i$  とする。このとき、クラスタ  $C_i$  は  $(V_i, E_i)$  で示される部分グラフである。各クラスタ  $C_i$  は、クラスタヘッド  $h_i$  を1つもち、また、 $h_i$  から  $C_i$  の全ての要素へ  $q$

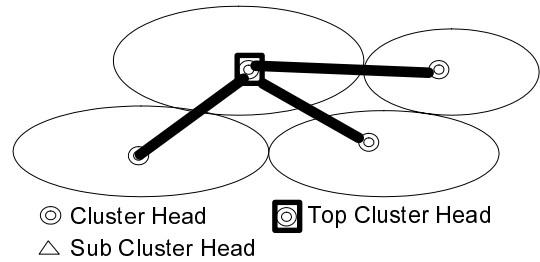


図 1: クラスタ内情報収集

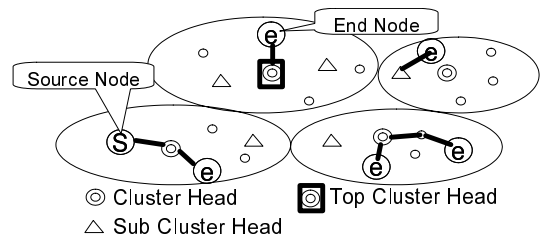


図 2: クラスタ内木の算出

ホップ以内の最短経路をもつようにクラスタ  $C_i$  を構築する ( $q$  は既定の定数)。クラスタの ID はクラスタヘッドのノード ID を用いる。ネットワーク上の全クラスタのクラスタヘッドの集合を  $H \subseteq V$  で表す。 $H$  の要素のうち最も ID の小さいノードをトップクラスタヘッドと呼ぶ。サブクラスタヘッドは、各クラスタにおいて、クラスタヘッドからの最短距離が  $q/2$  ホップ以内にあるノードの中から  $k$  個選ぶものとする ( $k$  は既定の定数)。

なお、各ノードの動画配信要求は自身が所属するクラスタヘッドに送信するものとする。

### 5.3 各フェイズの処理

#### 5.3.1 フェイズの全体的な流れ

提案手法では、マルチキャスト木の構築を (A) クラスタ構築, (B-intra) クラスタ内情報収集, (B-inter) クラスタ間情報収集, (C-inter) クラスタ間配送木の算出, (C-intra) クラスタ内配送木の算出, (D) クラスタ再構築の 6 フェイズで行う。処理手順は図 6 のとおりである。

この過程を繰り返すことにより、ノードの移動にともなうトポロジの変化に対応して、マルチキャスト木を構築することができる。クラスタ内のトポロジの変化に比べ、クラスタ間のトポロジの変化の方が小さいと考えられるため、(B-intra) フェイズと (C-intra) フェイズを高い頻度で繰り返す。

#### 5.3.2 クラスタ構築

フェイズ (A) ではネットワークをクラスタに分割し、クラスタヘッドとサブクラスタヘッドを決定する。サブクラスタヘッドはクラスタヘッドとともに、島モデル GA における計算資源となる。

ネットワーク上のクラスタの生成とクラスタヘッドの決定をするアルゴリズムには、ノードの ID を基準にクラスタヘッドを決定し、クラスタを構成する Lowest-ID algorithm [14]、隣接ノードの数、ノードの移動速度などからクラスタヘッドの適性度を求め、隣り合うノード間で比べることによりクラスタヘッドを決定し、クラスタを構成する WCA (Weighted Clustering Algorithm) [15] などがある。

クラスタヘッドからの TTL =  $q/2$  のフラッディングにより、各ノードで生成した乱数を収集し、上位  $k$  個のノード

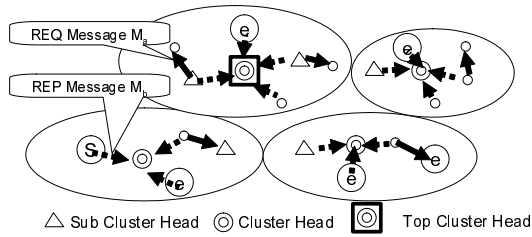


図 3: クラスタ間情報収集

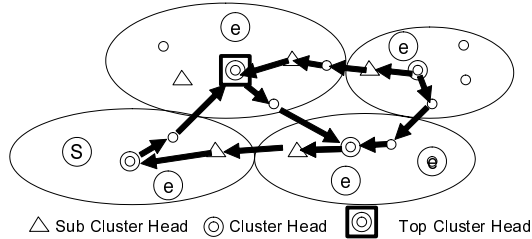


図 4: クラスタ間情報収集

をサブクラスタヘッドとして決定する。

### 5.3.3 クラスタ内情報収集

フェイズ (B-intra) ではクラスタ毎にリンクパラメタをクラスタヘッドに集める。

クラスタ  $C_1$  のクラスタヘッド  $h_1$  から全隣接ノードへ情報収集要求メッセージ  $M_{REQ}$  (シーケンス番号, クラスタ ID, TTL,  $R_a$ ) を送信する。但し,  $R_a$  は, クラスタヘッドからの経路である。 $M_{REQ}$  を受け取ったノード  $v_j$  は,  $M_{REQ}$  をまだ受信していない場合,  $R_a$  に  $v_j$  の ID を加える。また,  $v_j$  の経路制御表に  $R_a$  を加える。そして, 送信元以外の隣接ノードにメッセージ  $M_{REQ}$  を転送する。メッセージ  $M_{REQ}$  は TTL が 0 になるまで次々と再転送される。メッセージ  $M_{REQ}$  を転送したノードは, 他に転送するノードがない場合, またはすでに TTL が 0 である場合, クラスタヘッド  $h_i$  に向けプライメットメッセージ  $M_{REP}$  (シーケンス番号, クラスタ ID,  $R_b, p_b, r$ ) を送信する。シーケンス番号,  $R_b, p_b$  と  $r$  は, それぞれ受信したメッセージ  $M_{REQ}$  のシーケンス番号,  $h_i$  への経路, リンクパラメタと接続要求を要求するノードの ID である。この送信の際, 経路制御表の最小ホップの経路を用いる (図 5)。

ノード  $v_i$  が属するクラスタ  $C_i$  以外のクラスタヘッド  $h_j$  からメッセージ  $M_{REQ}$  を受け取った場合, 経路制御表のなかに,  $C_i$  のクラスタヘッド  $h_i$  への経路  $r_i$  があれば, その  $v_i$  は, クラスタヘッド  $h_j$  に向けて境界メッセージ  $M_{BDR}$  (シーケンス番号,  $v_i$  の ID,  $C_i$  の ID,  $r_{BDR}$ ) を送信する。但し, シーケンス番号,  $r_{BDR}$  は, それぞれ受信したメッセージ  $M_{REQ}$  のシーケンス番号と  $h_i$  から  $h_j$  への経路である。この  $h_i$  から  $h_j$  への経路は,  $h_i$  への経路  $r_i$  と受信したメッセージ  $M_{REQ}$  にある  $r_a$  を組み合わせたものである。 $M_{BDR}$  を受信したクラスタヘッド  $h_j$  は, 隣接経路表に  $r_{BDR}$  を加える。

以上によりクラスタヘッドにクラスタ内のトポロジ情報が集められる。

また, クラスタ毎に接続要求を出しているノードからのクラスタヘッドへの最大遅延時間  $Dr'_i$  を算出する。また, ソースノードを含むクラスタでは, ソースノードからソースの属するクラスタのクラスタヘッドまでの遅延時間  $Dr''_i$  を算出する。

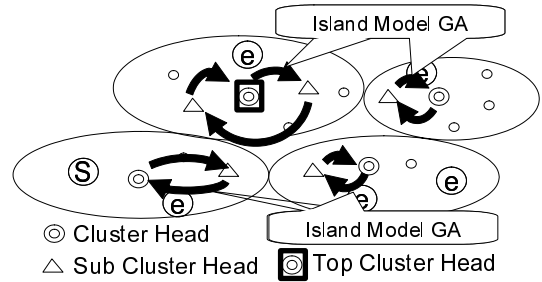


図 5: 島モデル GA

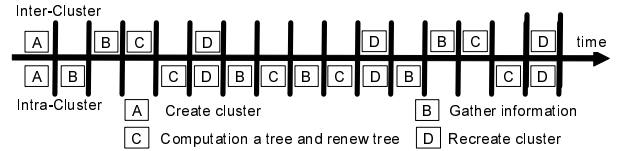


図 6: マルチキャスト木の算出手順

### 5.3.4 クラスタ間情報収集

フェイズ (B-inter) では隣接するクラスタ間の経路の帯域幅, 遅延の情報を調べ, クラスタヘッドに集める。そして, これらの情報とユーザの接続要求を各クラスタヘッドからトップクラスタヘッドに集める。トップクラスタヘッドは, クラスタヘッドの中から選ばれ, すべてのクラスタに関する情報を統括する。

クラスタ  $C_1$  のクラスタヘッドから, 隣接するクラスタ  $C_2$  のクラスタヘッドへの最短経路をそれぞれ作る。これを隣接経路と呼ぶ。それらの隣接経路とそれらの隣接経路の遅延, 帯域幅の情報の組 (隣接パス情報) をクラスタヘッドに送る (図 6)。隣接経路情報とクラスタ外のノードへの接続要求を各クラスタヘッドからトップクラスタヘッドに集める。

### 5.3.5 クラスタ間配送木の算出

フェイズ (C-inter) では, クラスタ間配送木の算出を行う。これは, トップクラスタヘッドが属するクラスタにおいて, 島モデル GA [13] を用いて算出される。クラスタ間配送木とは, ソースノードを含むクラスタのクラスタヘッドから各クラスタのクラスタヘッドまでの経路から形成されたマルチキャスト木である。この結果は全てのクラスタヘッドに送られる (図 1)。この問題の問題設定は以下の通りである。

クラスタ間配送木の算出問題: 各クラスタヘッドをノード, 隣接経路をエッジとするグラフ  $(V'', E'')$  を考える。

隣接の経路の遅延と帯域幅は隣接経路情報で与えられる。以上の条件下で, 式 (1) を最大化するようなマルチキャスト木  $T''$  を算出することが目的である。

$$\max. \sum_{C_i \in \mathcal{C}} \text{deliver}_c(C_i, T'') * |U'_i| - \alpha' \text{Cost}(T'') \quad (1)$$

ただし,  $\text{deliver}_c(C_i, T'') =$

$$\begin{cases} 1 & \text{if } \text{path}_c(C_i) \neq \emptyset \wedge \text{Band}(\text{path}_c(C_i)) \leq Br \\ & \wedge \text{Delay}(\text{path}_c(C_i)) \leq Dr - Dr'_i - Dr''_i, \\ 0 & \text{otherwise.} \end{cases}$$

ここで,  $|U'_i|$  はクラスタ  $C_i$  上の接続要求を出しているノードの数,  $\mathcal{C}$  はネットワーク上にあるクラスタの集合,  $\text{path}_c(C_i)$  は,  $T''$  上のソースノードを含むクラスタのクラスタヘッドから  $T''$  上のノードを通り, クラスタ  $C_i$  のクラスタヘッドに至る経路のことである (もし, そのような経路が存在しない場合,  $\text{path}_c(C_i) = \emptyset$  とする)。 $\alpha'$  は重み付け係数である。

遺伝的アルゴリズムの適用法：GA に適用するためには、染色体表現、交叉方法、突然変異方法と評価方法を定める必要がある。

染色体は、 $(V'', E'')$  上のすべてのノード番号を要素とする数列である。逆符号化の方法は以下の通りである。まず、ソースノード  $s$  が属するクラスタのクラスタヘッドのみから構成される木  $T''$  を考える。そして、つぎのように逆符号化する。

- (1)  $i := 1, \text{flag} := 0$  とする。
- (2) 染色体において  $i$  番目の遺伝子が示すノードが、 $T''$  に含まれておらず、かつ、接続可能であるかどうか調べる。接続可能ならば、そのノードを  $T''$  に接続し、 $\text{flag} := 1$  とする。
- (3)  $i$  に 1 を加える。もし、 $i$  が  $|V''| + 1$  に達していなければ、2. に戻る。
- (4)  $\text{flag} = 0$  ならば終了。そうでなければ、1. に戻る。

解候補の交叉方法として、EXX [16] を用いる。突然変異法として、一つの染色体から 2 つの遺伝子をランダムに入れ換える方法を用いる。

### 5.3.6 クラスタ内配送木の算出

フェイズ (C-intra) ではクラスタ内で配信要求を出したノードを、クラスタ間配送木の算出で計算されたマルチキャスト木に、つぎ木する経路 (帯域幅、遅延の制約を満たす) を各クラスタヘッドが算出する。これは、各クラスタ毎に、島モデル GA を用いて算出する (図 4)。そして、その算出結果を各ノードへ送る。

この問題設定は以下の通りである。

問題設定：クラスタ  $C_i$  において、クラスタ内配送木の算出の際の制約条件は、帯域幅  $Br$  と遅延  $Dr'_i$  である。

そのクラスタがソースノードを含む場合、そのクラスタをネットワーク全体とみなし、式 (2) を最大化するようなマルチキャスト木  $T'''$  を算出する。

ソースノードを含まない場合、クラスタヘッドをソースノードとみなして、式 (2) を最大化するようなマルチキャスト木  $T'''$  を算出する。

$$\max. \sum_{u \in U'_i} \text{deliver}(u, s, T''', Dr'_i, Br) - \alpha'' * \text{Cost}(T''') \quad (2)$$

ここで  $U'_i$  は、クラスタ  $C_i$  のユーザの集合、 $\alpha''$  は重み付け係数である。 $T'''$  の算出には、遺伝的アルゴリズムの適用法：フェイズ (C-inter) と同じ方法を用いる。

### 5.3.7 クラスタ再構築

フェイズ (D) では、ノードの移動によるトポロジの変化に対応するため、フェイズ (A) と同様の方法を用いて、クラスタの再構成を行う。新しく現れた、あるいは、所属していたクラスタに接続できなくなったノードは、接続要求信号を周囲のノードに流す。それを受け取ったノードは、自分の所属するクラスタのクラスタヘッドに信号を伝達する。

この信号を受け取っていたクラスタのみが、このフェイズにおいて、クラスタの再構成を行う。

## 6 評価実験

本章では、提案手法により算出されたマルチキャスト木に沿ってマルチメディアデータを配信した際の性能 (パケット到達率と遅延) およびトポロジの変化に従った性能の変化について評価を行う。なお、トポロジ情報収集のための通信コストおよび GA によるマルチキャスト木の算出時間は、文献 [4] にて報告している。

### 6.1 実験における提案手法の実装

実験には、提案手法を一部単純化したプロトタイプを用いた。プロトタイプは以下のように実装した。

クラスタ間配送木の構築：クラスタ間のデータ配送は、ネットワーク層で行った。クラスタ間配送木を構成するリンクは DSR [6] を用いて確立させた。従って、パケットは DSR が確立した経路に基づいて通信を行うことになる。なお、クラスタ間配送木はトポロジ情報から GA により算出したものを使用し、その情報は各クラスタヘッドにあらかじめ与えた。

クラスタ内配送木の構築：クラスタ内のデータ配送は、ネットワーク層における選択的フラッディングにより行った。

なお、今回の実験では、クラスタ内配送木をクラスタ間配送木とは独立に、クラスタヘッドを根とする木として構築している。クラスタ間配送木につぎ木をする機能は実装していない。クラスタ間の場合と同様に、クラスタ内配送木はトポロジ情報から GA により算出したものをクラスタ内の各ノードにあらかじめ持たせた。

クラスタ内配送は、以下の手順で行った。まず、クラスタヘッドが送信パケットをブロードキャストする。そして、パケットを受信したノードは、そのパケットの送信元が、クラスタ内配送木の上流のノードになっている場合にだけ、そのパケットをさらにブロードキャストする。この過程を繰り返すことでパケットを木の下流のノードへ伝達する。

### 6.2 実験の設定

今回の実験における設定は、以下のとおりである。実験には、ネットワークシミュレータ GTNetS [5] を用いた。実験空間は  $3000 \times 3000 \text{ m}^2$ 、ノード数は 1000 である。電波の到達距離を  $200 \text{ m}$ 、データの送信レートを  $8 \text{ Kbps}$  とした。ノードの移動モデルは、ランダムウェイポイントとした。実験空間を、 $4 \times 4$  の格子状に固定分割し、それぞれの領域に属するノードの集合をクラスタとして扱った。また、それぞれの領域の中心付近にあるノードをクラスタヘッドとした。

提案手法により構築されたマルチキャスト木に対して、平均遅延時間、パケット到達率についての評価を行った。各試行回数は、10 回である。ノードの移動速度を 2, 4, 10  $\text{km/h}$  と変化させ実験を行った。

### 6.3 平均遅延時間、パケット到達率

提案手法により構築されたマルチキャスト木の評価をするために、シミュレーション時間の経過に対する平均遅延時間とパケット到達率をそれぞれ計測した。平均遅延時間とは、ソースノードがパケットを送信してからユーザが受信するまでの時間の平均である。パケット到達率とは、送信されたパケットを受信できたノードの割合である。計測結果をそれぞれ図 7、図 10 に示す。なお、計測結果の図では、60 秒間に計測された値の平均値を折れ線でプロットしている。

パケット到達率に関する実験の結果について：図 7 を見ると、パケット到達率は、シミュレーションの初期のころで、0.4 以下の低い値となり、シミュレーション時間の経過とともに 0.1 付近の値までさらに減少している。パケット到達率が期待した通りにならなかった原因として、ノードの移動による配送経路の切断が考えられる。特に、上流の方 (クラスタ間の経路など) で経路が切断すると、下流にパケットが到達しなくなり全体としてパケット到達率が大幅に落ちる。

このことを検証するために、シミュレーション時間の経過に対する、ソースノードからパケットが到達しているクラスタヘッドの数を計測した。その結果を図 8 に示す。また、各クラスタにおいてクラスタヘッドをソースノードと

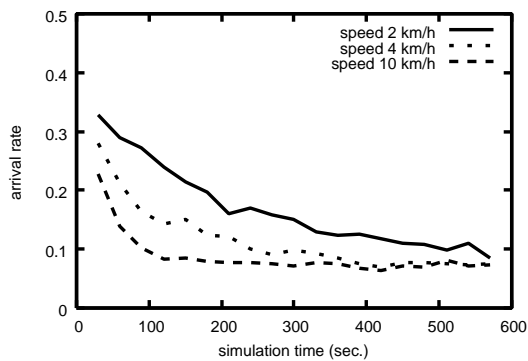


図 7: パケット到達率

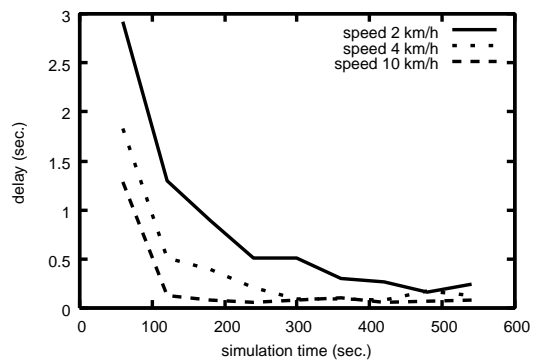


図 10: 平均遅延時間

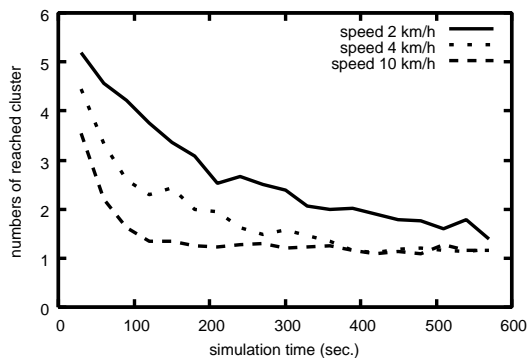


図 8: パケットの到達したクラスタの数

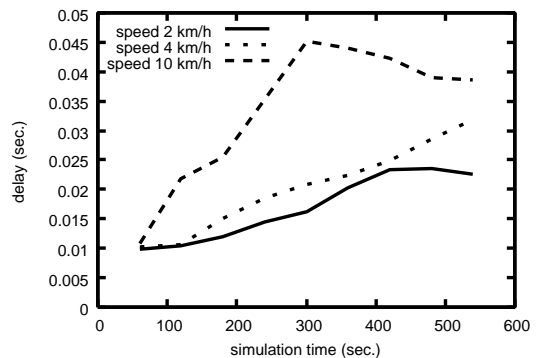


図 11: クラスタ内の平均遅延時間

想定した場合の(すなわちクラスタ間の経路を經由しない場合の), シミュレーション時間の経過に対するパケット到達率を計測した. 計測結果を図 9 に示す.

図 8 を見ると, 予想したとおり, 総クラスタ数 16 に対し, シミュレーション初期で 5 つのクラスタにしかパケットが配送されておらず, 時間が経過するとどのクラスタにもパケットが配送されていないことが分かる. 以上の結果から, ノード密度にも依存するが DSR でクラスタ間の経路を確保するのは問題があることが分かった.

また, 図 9 では, クラスタ内のパケット到達率は, 0.9 付近の高い値を推移している. シミュレーション時間が経過するとともに, クラスタ間のパケット到達率は減少しているが, 0.8 以上の高い値を示している.

平均遅延時間に関する実験の結果について: 図 10 を見ると, 平均遅延時間は, シミュレーションの初期のころで, 1 秒以上の高い値となっていたが, シミュレーション時間の

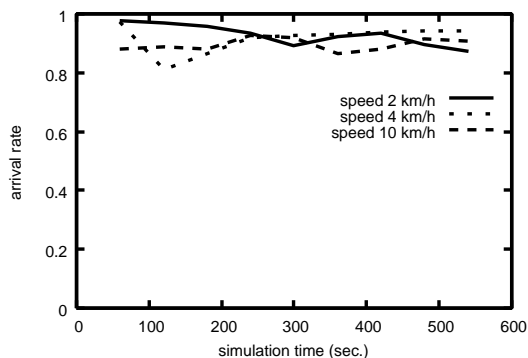


図 9: クラスタ内のパケット到達率

経過とともに 0.5 秒以下の低い値に減少している. 平均遅延時間が減少した原因は, クラスタ間の経路の切断により, ソースノードからのホップ数が比較的小さいノードのみがパケットを受信しているためだと思われる.

さらに, クラスタ内のクラスタヘッドをソースノードとした場合のシミュレーション時間の経過に対する平均遅延時間を計測した. 計測結果を図 11 に示す. これを見ると, クラスタ内の遅延は, シミュレーション初期のころで, 0.01 程度の値であったが, ノードが移動するにつれて 0.03 付近まで大きくなっているが, 実用上問題ない性能が達成できていることが分かった.

## 7 おわりに

本稿では, MANET 上でユーザの QoS 要求を満たすマルチキャスト木を構築する方法を提案しその性能をシミュレーションにより評価した. 提案手法ではネットワークをクラスタに分割し, 各クラスタ内およびクラスタ間の 2 階層で算出を行うことにより, スケーラビリティを高めている. また, 並列 GA の一種である島モデル GA を用いることで計算の高速化を行っている.

今回の実験により, クラスタ間の経路構築に DSR などの既存プロトコルを用いた場合には経路の切断が頻繁に起こり, パケット到達率, 遅延ともに期待した性能を達成できないことが分かった. 今後, ABR (Associativity-Based Routing)[17] などの切断しにくい経路を確立可能なプロトコルを使用し, クラスタ間データ伝送を含む場合のパケット到達率および遅延を改善させて行く予定である.

## 参考文献

- [1] Mohapatra, P., Li, J. and Gui, C.: QoS in mobile ad hoc networks (2003).
- [2] L. Barolli, A. Koyama, T. S. and Shiratori, N.: GAMAN: A GA Based QoS Routing Method for Mobile Ad-hoc Networks, *Journal of Inetrconnection Networks (JOIN)*, Vol. 4, No. 3, pp. 251–270 (2003).
- [3] Layuan, L. and Chunlin, L.: QoS Multicast Routing in Networks with Uncertain Parameters, *Web Technologies and Applications, 5th Asian-Pacific Web Conference (APWeb 2003)*, pp. 430–441 (2003).
- [4] 高島栄一, 柴田直樹, 村田佳洋, 安本慶一, 伊藤実: MANET における島モデル GA を用いた QoS マルチキャストルーティングの一手法, マルチメディア, 分散, 協調とモバイル ( DICOMO2005 ) シンポジウム 論文集情報処理学会シンポジウムシリーズ, No. 6, pp. 413–416 (2005).
- [5] Riley, G. F.: The Georgia Tech Network Simulator, *MoMeTools '03: Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research*, New York, NY, USA, ACM Press, pp. 5–12 (2003).
- [6] Johnson, D. B. and Maltz, D. A.: Dynamic Source Routing in Ad Hoc Wireless Networks, *Mobile Computing* (Imielinski and Korth(eds.)), Vol. 353, Kluwer Academic Publishers, chapter 5, pp. 153–181 (1996).
- [7] Chen, S. and Nahrstedt, K.: Distributed Quality-of-Service Routing in Ad-Hoc Networks, *IEEE Journal on Special Areas in Communications*, Vol. 17, No. 8, pp. 1–18 (1999).
- [8] Lin, C. R. and Liu, J.-S.: QoS Routing in Ad Hoc Wireless Networks, *IEEE JSAC*, Vol. 17, No. 8 (1999).
- [9] Zhu, C. and Corson, M. S.: QoS Routing for Mobile Ad Hoc Networks (2002).
- [10] Liao, W.-H., Tseng, Y.-C., Wang, S.-L. and Sheu, J.-P.: A Multi-path QoS Routing Protocol in a Wireless Mobile ad Hoc Network, *ICN '01: Proceedings of the First International Conference on Networking-Part 2*, London, UK, Springer-Verlag, pp. 158–167 (2001).
- [11] Sinha, P., Sivakumar, R. and Bharghavan, V.: CEDAR: A Core-Extraction Distributed Ad Hoc Routing Algorithm, *Proceedings of IEEE Conference on Computer Communications INFOCOM'99*, pp. 202–209 (1999).
- [12] Sinha, P., Sivakumar, R. and Bharghavan, V.: MCEDAR: Multicast core extraction distributed ad-hoc routing, *Proceedings of the Wireless Communications and Networking Conference* (1999).
- [13] Cantú-Paz, E.: A Survey of Parallel Genetic Algorithms, Technical Report 97003, Illinois Genetic Algorithms Laboratory (1997).
- [14] A. Ephremides, J. E. Wieselthier, and D. J. Baker: A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling, *Proc. IEEE* 75, pp. 56–73 (1987).
- [15] Chatterjee, M., Das, S. K. and Turgut, D.: WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks, *Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks)*, Vol. 5, No. 2, pp. 193–204 (2002).
- [16] K. Maekawa, N. Mori, H. Tamaki, H. Kita, and H. Nishikawa: Genetic Solution for the Traveling Salesman Problem by Means of a Thermodynamical Selection Rule, *Proc. 1996 IEEE ICEC*, pp. 529–534 (1996).
- [17] Toh, C.-K.: Associativity-Based Routing for Ad Hoc Mobile Networks, *Wireless Personal Communications*, Vol. 4, No. 2, pp. 103–139 (1997).