

# ユビキタス情報環境におけるサービス品質に基づく 適応型経路制御法

|        |   |
|--------|---|
| 著者     | 長田 俊明   |
| 学位授与機関 | Tohoku University   |
| URL    | <a href="http://hdl.handle.net/10097/39886">http://hdl.handle.net/10097/39886</a> |

平成 20 年度 博士学位論文

# ユビキタス情報環境における サービス品質に基づく適応型経路制御法

東北大学大学院 情報科学研究科  
情報基礎科学専攻  
コミュニケーション論講座  
白鳥研究室

A3ID1003

長田 俊明

2009 年 1 月

# 目次

|                                     |           |
|-------------------------------------|-----------|
| <b>第1章 序論</b>                       | <b>1</b>  |
| 1.1 本研究の背景                          | 1         |
| 1.1.1 ユビキタス情報社会                     | 1         |
| 1.1.2 ユビキタス情報環境におけるサービス品質向上のための課題   | 2         |
| 1.1.3 サービス品質向上のための経路制御と技術課題         | 7         |
| 1.2 本研究の目的                          | 8         |
| 1.2.1 本研究の目的                        | 8         |
| 1.2.2 本研究の課題                        | 9         |
| 1.3 本研究の位置付け                        | 10        |
| 1.4 本論文の構成                          | 11        |
| <b>第2章 サービス品質向上のためのネットワークの経路制御</b>  | <b>14</b> |
| 2.1 はじめに                            | 14        |
| 2.2 ネットワーク資源を積極的に割り当てるネットワーク経路制御    | 15        |
| 2.3 経路制御の現状と課題                      | 19        |
| 2.3.1 有線の経路制御の関連研究                  | 19        |
| 2.3.2 無線の経路制御の関連研究                  | 21        |
| 2.3.3 サービス品質向上のための適応型経路制御を実現するための課題 | 27        |
| 2.4 要約                              | 32        |

|              |   |           |
|--------------|---|-----------|
| <b>第 3 章</b> | <b>無線アドホックネットワークのための高精度な帯域予測に基づく経路制御法</b>                             | <b>34</b> |
| 3.1          | はじめに . . . . .  | 34        |
| 3.2          | 既存の無線アドホックネットワークの経路制御法の問題点 . . . . .                                  | 35        |
| 3.3          | ALRS: An effective Application-Level QoS Routing Scheme の提案 . . . . . | 37        |
| 3.3.1        | ALRS の概要 . . . . .  | 37        |
| 3.3.2        | キャリアセンスの範囲を基準とした帯域予測 . . . . .  | 39        |
| 3.3.3        | アドミSSION制御に基づく経路の構成 . . . . .   | 45        |
| 3.3.4        | 経路の動的な再構成 . . . . .   | 51        |
| 3.4          | 性能評価 . . . . .  | 54        |
| 3.4.1        | シミュレーションのパラメータ . . . . .  | 55        |
| 3.4.2        | 性能評価に関する測定項目 . . . . .  | 56        |
| 3.4.3        | 端末の移動がない場合の性能 . . . . .   | 58        |
| 3.4.4        | 端末の移動がある場合の性能 . . . . .   | 61        |
| 3.4.5        | 考察 . . . . .  | 62        |
| 3.5          | 要約 . . . . .  | 63        |
| <b>第 4 章</b> | <b>部分木の帯域予測に基づくエンドシステムマルチキャストの配信経路の制御</b>                             | <b>71</b> |
|              | <b>法</b>  | <b>71</b> |
| 4.1          | はじめに . . . . .  | 71        |
| 4.2          | 既存のエンドシステムマルチキャストのデータ配信木 ( ツリー ) の構成法<br>の問題点 . . . . .               | 72        |
| 4.3          | 反復型ツリー構成法の提案 . . . . .  | 73        |
| 4.3.1        | 反復型ツリー構成法の提案 . . . . .  | 73        |
| 4.3.2        | ネットワークモデルおよび問題設定 . . . . .  | 74        |
| 4.3.3        | ツリー構成アルゴリズムの詳細 . . . . .  | 77        |

|            |                       |            |
|------------|-----------------------|------------|
| 4.3.4      | 各ノードの持つ情報 . . . . .   | 78         |
| 4.3.5      | 各種状態情報の更新 . . . . .   | 79         |
| 4.3.6      | 帯域分配アルゴリズム . . . . .  | 81         |
| 4.3.7      | 再構成判断アルゴリズム . . . . . | 81         |
| 4.4        | 実験と評価 . . . . .       | 84         |
| 4.4.1      | 実験概要 . . . . .        | 84         |
| 4.4.2      | 機能検証実験 . . . . .      | 85         |
| 4.4.3      | 比較実験 . . . . .        | 86         |
| 4.4.4      | 考察 . . . . .          | 90         |
| 4.5        | 要約 . . . . .          | 91         |
| <b>第5章</b> | <b>結論</b>             | <b>96</b>  |
| 5.1        | 本研究の概要 . . . . .      | 96         |
| 5.2        | 各章の要約 . . . . .       | 97         |
| 5.3        | 本研究の貢献 . . . . .      | 99         |
| 5.4        | 今後の課題 . . . . .       | 99         |
|            | 謝辞                    | <b>101</b> |
|            | 発表論文                  | <b>107</b> |

# 目次

|      |   |    |
|------|---|----|
| 1.1  | サービス品質の分類 . . . . .                               | 3  |
| 1.2  | ネットワーク QoS 技術の対象分野 . . . . .                      | 9  |
| 1.3  | 本研究の位置付け . . . . .                                | 11 |
| 1.4  | 本論文の構成 . . . . .                                  | 13 |
| 2.1  | インターネットの構造 . . . . .                              | 16 |
| 2.2  | 無線アドホックネットワークの物理的な構造 . . . . .                    | 17 |
| 2.3  | 無線アドホックネットワークをリンクの数に注目して抽象化 . . . . .             | 18 |
| 2.4  | エンドシステムマルチキャストを行うネットワークの物理的な構造 . . . . .          | 18 |
| 2.5  | エンドシステムマルチキャストを行うネットワークをリンクの数に注目して抽象化 . . . . .   | 19 |
| 2.6  | エンドシステムマルチキャストと IP マルチキャスト . . . . .              | 20 |
| 2.7  | エンドシステムマルチキャストの配信木の構成法の分類 . . . . .               | 21 |
| 2.8  | 無線アドホックネットワークの概要 . . . . .                        | 26 |
| 2.9  | アドミッション制御に基づく無線アドホックネットワークの経路制御法の分類 . . . . .     | 27 |
| 2.10 | 多対多通信のためのエンドシステムマルチキャストの配信経路の制御に関する技術課題 . . . . . | 28 |
| 2.11 | 端末 A における消費帯域 . . . . .                           | 29 |

|      |   |    |
|------|---|----|
| 2.12 | アドミッション制御に基づく無線アドホックネットワークの経路制御に関する技術課題 . . . . .   | 32 |
| 3.1  | (a) HELLO メッセージのフォーマットおよび (b)NIT の構造 . . . . .  | 39 |
| 3.2  | NIT の作成・更新アルゴリズムの擬似コード . . . . .  | 42 |
| 3.3  | 帯域情報の交換メカニズムの動作例 . . . . .  | 65 |
| 3.4  | 経路探索のメッセージのシーケンス . . . . .  | 66 |
| 3.5  | Final Check アルゴリズムの擬似コード . . . . .  | 66 |
| 3.6  | $CBR_{40}$ の場合の実験結果 (30 シナリオの平均) . . . . .  | 68 |
| 3.7  | $CBR_{200}$ の場合の実験結果 (30 シナリオの平均) . . . . .   | 69 |
| 3.8  | 各セッションにおけるスループットと遅延の散布図( 端末が移動無し, $CBR_{40}$ , Load = 1200Kbps の場合の, ボトルネックが各所で起こりやすいシナリオ (excessive) と比較的ボトルネックの少ないシナリオ (moderate) ) . . . . . | 69 |
| 3.9  | パケットの配信率に対するセッション数の累積度数分布 . . . . .   | 70 |
| 4.1  | 本研究におけるネットワークのモデル . . . . .   | 74 |
| 4.2  | ツリー再構成の手順 . . . . .   | 78 |
| 4.3  | (Case A) の試算手順 . . . . .  | 82 |
| 4.4  | (Case B) の試算手順 . . . . .  | 83 |
| 4.5  | 機能検証実験における受信ビットレートの変化 . . . . .   | 87 |
| 4.6  | パターン 1 の全てのアクセスリンク帯域を一律 1/10 にした場合の受信ビットレートの変化 . . . . .  | 88 |
| 4.7  | 受信ビットレートの変化 (パターン 1) . . . . .  | 93 |
| 4.8  | 受信ビットレートの変化 (パターン 2) . . . . .  | 94 |
| 4.9  | 受信ビットレートの変化 (パターン 3) . . . . .  | 95 |

|                                   |     |
|-----------------------------------|-----|
| 5.1 ユビキタス情報社会に対する本研究の貢献 . . . . . | 100 |
|-----------------------------------|-----|

# 第1章 序論

## 1.1 本研究の背景

### 1.1.1 ユビキタス情報社会

近年，ユビキタス情報社会と呼ばれる「いつでも・どこでも・誰でも，利用者がコンピュータやネットワークなどを利用しサービスを楽しむ情報化社会」を目指した取り組みが，行政・企業・研究機関で盛んにおこなわれている。『ユビキタス(ubiquitous)』は，ラテン語で「あらゆるところで」を意味する「ubique」が英語化したもので「遍在する」と訳されている。米国では『Pervasive』，欧州では『Ambient』といった言葉が同様の概念を表す言葉として使われることが多い。コンピュータやネットワークが遍在し，利用者が意識せずにそれらを利用する技術として，ユビキタスコンピューティング，ユビキタスネットワークがある。これらが実現され，日常生活の中で社会基盤として利用されている社会がユビキタス情報社会である。ユビキタスコンピューティングは，1988年にゼロックス・パロアルト研究所(Palo Alto Research Center: PARC)のMark Weiser氏により提唱された概念であるが，携帯電話に代表される情報端末の小型化・高性能化，インターネットの爆発的な普及などによる情報・通信技術の発展に伴い注目を集めるようになった。また日本国内においては，坂村健氏により1980年代初頭に同様の概念が提唱されていたと言われている。現在，総務省が主導する国家プロジェクトとしてu-japan政策が打ち出され，国家レベルでもさまざまな研究・開発が進められている。

ユビキタス情報社会のための情報通信基盤であるユビキタス情報環境は，無線アドホックネットワークなどの無線ネットワーク，携帯電話・PDAなどの携帯型端末，ICタグ・RFID

システム・センサネットワークなどのセンサ技術，情報家電やウェアラブルコンピュータなどに代表される埋め込みコンピュータなどから構成される．

ユビキタス情報社会における情報サービスには，チャットやビデオ会議などのような「利用者同士のコミュニケーション」を目的としたサービスや，検索サイトやホームページ，掲示板，動画配信サイトなどが提供する「情報の取得・閲覧・視聴」を目的とした行うサービス，さらにはセンサやタグなどを用いた「環境情報の収集」を目的とするサービスなどが挙げられる．これらの情報サービスは近年，商工農業・行政・教育・医療等の様々な分野で活用されつつあることから分かるように，今後，ユビキタス情報環境の普及・技術の発展が進み，様々な情報通信基盤上で利用者に多種多様な情報サービスが提供されることで，人間の社会生活をより便利で快適に営むことができるような情報化社会の実現が期待される．

### 1.1.2 ユビキタス情報環境におけるサービス品質向上のための課題

情報サービスの中でも，動画メディアや音声メディアを，コンピュータネットワークを介して提供するネットワークサービスであるマルチメディア通信サービスは，近年の高速・広帯域ネットワークの普及や計算機端末の高性能化により普及が進んだ．マルチメディア通信サービスの具体的な実現例としては，ビデオ会議や多人数参加型のゲーム，IP電話，インターネットラジオ，ワンセグ放送，音楽・映像のストリーミング配信などがある．また，これらのマルチメディア通信サービスが提供されるネットワーク環境（通信網）には，携帯電話と基地局によるセルラネットワークを含む電話網，衛星ネットワーク，専用回線網，NGN，インターネット上のIPネットワーク，無線ホットスポットサービスをはじめとする無線LAN，さらには無線アドホックネットワークなどが挙げられる．

このように，マルチメディア通信は，我々の生活環境の多くの場面で活用されているが，中でもビデオ会議などのように，自身の状況または通信相手側の状況を直感的に把握

したり把握させたりすることにも適用できるため，利用者同士のコミュニケーションのために特に有用な手段である．そのため，提供されるマルチメディア通信サービスの品質が向上することによって，充実した利用者同士のコミュニケーションを実現することが期待できる．

ここで，本研究では「サービス品質」を説明する上で，以下のような階層化した概念を導入する．

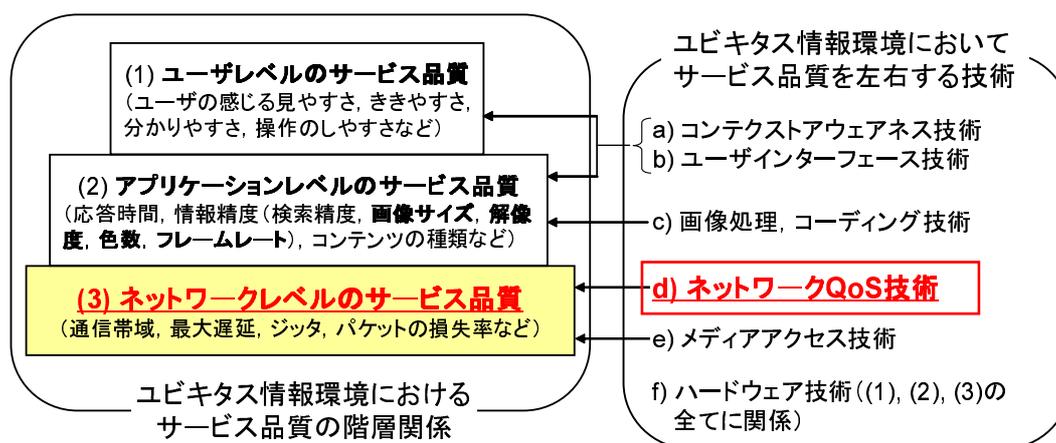


図 1.1: サービス品質の分類

### (1) ユーザレベルのサービス品質

情報サービスを提供するアプリケーションやシステムが出力する内容について，利用者の主観的な評価に基いた品質を表す．例えば，画面の見やすさ，音声のききとりやすさ，提示される情報の意味についての分かりやすさ，システムの操作のしやすさなどがユーザレベルのサービス品質に当たる．

### (2) アプリケーションレベルのサービス品質

アプリケーションやシステム自身が出力するデータ自体の定量的に解析可能な品質や，システムの要求に対する相対的または絶対的な評価値を指す．応答時間，情報精度（検索精度，画像サイズ，解像度，色や色数，フレームレート），音量，コンテンツやメディアの種類などがアプリケーションレベルのサービス品質に当たる．

### (3) ネットワークレベルのサービス品質

アプリケーションから見たネットワークの性能のことである。通信帯域，遅延，ジッタ，パケット損失率などがネットワークレベルのサービス品質と定義できる。

以上の3つのサービス品質の間には，図 1.1 に示すように，下位層のサービス品質によって上位層のサービス品質が制約を受けるという，階層関係が成り立つ。どの層から影響を受け，その制約の大きさはどのくらいであるかは，対象となるサービスの種類や特徴によって異なるが，マルチメディア通信においては，ユーザレベルのサービス品質およびアプリケーションレベルのサービス品質がネットワークレベルのサービス品質に特に左右される。

マルチメディア通信により利用者同士のコミュニケーションを行うビデオ会議と，情報検索サービスを例にとり比較してみる。アプリケーションが確保できる通信帯域が充分でなければ，単位時間あたりにネットワーク上で扱えるデータの絶対量が少ない。ビデオ会議システムの場合，通信相手から受信する映像データの受信量によっても大きく影響を受ける要素が（アプリケーションレベルの）サービス品質に含まれるため，通信帯域は，システムが使用する計算機の処理能力や処理の負荷，画像の圧縮技術や出力インターフェースの能力と同様に，画面に出力される映像の画質や解像度，フレームレートを改善する際の制約である。これらのアプリケーションレベルのサービス品質が向上すれば，利用者が視覚的に相手側の状況が分かりやすくなるなど，利用者同士のコミュニケーションのサービス品質が向上することから，ユーザレベルのサービス品質の向上が直接上位の階層のサービス品質の改善に繋がるのが非常に多い。一方，情報検索に関するサービスでは，通信帯域の向上や遅延の低減は，検索の際の応答速度の向上が期待できる。しかしながら，利用者が自分が知りたいジャンルに関してなるべく有用で正確な情報を獲得したい，というユーザレベルのサービス品質の向上が求められる場合には，アプリケーションレベルのサービス品質である情報検索の精度の向上，信頼性の向上を行う技術が最重要であ

り，ユーザレベルのサービス品質の向上が上位層の改善に貢献する割合は大きくない．

本研究では，ビデオ会議のような，利用者同士のコミュニケーションに有効なマルチメディア通信サービスに焦点を当て，マルチメディア通信サービスの品質を向上させるためのアプローチとして，ネットワークレベルのサービス品質を向上させることを目的とする．ここで，ネットワークレベルのサービス品質の向上する技術として重要となるものは，ネットワーク QoS 技術である [1]．図 1.2 は，ネットワーク QoS 技術の対象分野を整理したものであり，研究課題として以下の 1) から 4) の 4 つの制御技術が存在する．

1) アドミッション制御（または受付制御とも呼ばれる）は，各アプリケーションからのサービス要求を受け付けるかどうかを決定する仕組みである．そのサービス要求を受け付けるかどうかは，要求元となるアプリケーションがサービス品質要求を行う資格の有無や，要求されたサービスに使用するネットワーク資源と現在のネットワーク資源の使用状況などから総合的に判断する．帯域予約はサービスのフロー（セッションのデータ）に対してサービス品質を保証するために経路上で必要なネットワーク資源を予約する動作を行う仕組みである [2]．動的な資源予約の際には RSVP[3] やなどのシグナリングプロトコルが用いられる．また RSVP をベースとし，無線ネットワーク環境向けに改良したシグナリングプロトコルも研究されている [4, 5]．

2) 経路制御は，データの送信元から宛先へデータを到着させるための経路の探索や経路の更新・維持を行う仕組みである．経路制御の方式は，(A) 有線ネットワークと (B) 無線ネットワークで大きく異なる．まず，有線ネットワークの経路制御の方式は，2 種類の方式に大きく分類され，その一つは基幹ネットワークのための経路制御である．これは，管理者のポリシーに基づき IP ルータ（L3 ルータ，L4 の場合もある）間でのデータの中継経路の制御を行うインターネットの経路制御の方式 [6, 7, 8] を指す．また，IP アドレスに基づきルータ上で実行される経路制御だけではなく，MPLS のようにラベルを用いて L2 スイッチや L3 スイッチ上で高速なスイッチングを行う仕組みも有線ネットワークにおける

経路制御の方式の一つといえる [9]。もう一つは、利用者端末間で仮想的に構成されるネットワークにおいて、利用者端末すなわちエンドシステムによりデータの複製・転送を行いマルチキャストを行うエンドシステムマルチキャストの、データの配信経路の制御するための経路制御方式 [10, 11, 12, 13, 14, 15] である。無線ネットワークの経路制御では、ネットワークの利用目的や特性に応じて様々な経路制御法が提案されており、中でも AODV や DSR 等が有名である [16, 17]。

3) 優先度制御（または優先制御）とは、データ（パケット）に優先度をつけて転送する仕組みであり、基本的に優先度の高いパケットを先に処理させる制御を行う。回線の混雑やルータなどの通信機器の処理能力の限界などでパケットが待ち行列を作っている場合でも、優先度の高いパケットの遅延時間は大きくなり、パケットの廃棄も避けられる。リアルタイムアプリケーションやバックグラウンドトラフィックなどで各サービス（のためのデータ）を分類し（サービスクラスと呼ぶ）、それぞれのサービスクラスごとの相対的な優先度を基準とした優先度制御を行う方式である DiffServ [18] などが有名である。

4) 輻輳制御は、ネットワーク内で生じる輻輳を、ルータやエンドシステムで抑制や解消を行うことであり、到着したパケットをキューに收容するか廃棄するかや、既にキューの中に存在するパケットを廃棄するかなどを処理するキューマネジメントなどの技術がある。また、輻輳制御の技術を活用した研究領域として、送信レートを決めるための輻輳ウィンドウサイズの適切な調整によって輻輳を回避する TCP 輻輳制御のが様々な手法が提案されている。

以上の4つのネットワーク QoS 技術の中で、本研究では、配信経路の構成によって確保できるネットワークの資源の量が左右されるなど、上述の他の3つの制御技術における処理動作や性能に大きな影響を与える経路制御技術を適用し、ネットワークレベルのサービス品質の向上に取り組む。

### 1.1.3 サービス品質向上のための経路制御と技術課題

ユビキタス情報環境においては無線ネットワークおよび有線ネットワークに対して様々なネットワークの通信形態が存在し、従来の経路制御の対象すなわちインターネットのための経路制御 [6, 7, 8] とは異なる経路制御の技術が必要となる情報通信基盤が存在する。特に無線ネットワークのひとつである無線アドホックネットワークにおけるマルチホップ通信と、有線ネットワーク上で利用者端末（エンドシステム）同士でデータの複製・転送を行うエンドシステムマルチキャストは、データの配信のための経路制御を行う上で、基本的にポリシーベースで行われるインターネットの経路制御とは異なる複雑さがある。

まず、無線アドホックネットワークにおいては、通信セッションのデータに対する送信元から宛先までデータを運ぶ際、通信を行う空間を多くの無線端末で共有しながらマルチホップ通信を行うため、ネットワークの規模に対して各無線端末における論理的なリンクの接続数（データが直接届く相手の数）すなわち次数が多く、無線端末の移動により次数の変動が大きい。加えて、無線アドホックネットワークにおいては、通信を行う空間を多くの無線端末で共有しながらマルチホップ通信を行う。そのため、データの配信経路上に通信帯域を割り当てる際には、有線で接続している場合のように単純に配信経路上の各リンクに通信帯域を割り当てを行うことができない。無線の場合、ある端末の送信によって、少なくともその端末と通信空間を共有する他の全ての端末においても帯域を消費することになるからである。このように、次数の多さによる宛先に到達するための配信経路の候補の数の問題に加え、経路の候補ごとに各端末で消費する帯域の予測が難しい。以上のことから、セッションのデータの配信経路上で所望の通信帯域を割り当てる（すなわち、通信帯域を充足する）ことは難しい。

また、エンドシステムマルチキャストは、インターネット上の利用者端末などのエンドシステム同士で物理的なネットワーク上に仮想的なネットワークであるオーバーレイネットワークを構成し、マルチキャスト通信の際にエンドシステムによってデータを複製・転送

するアプリケーションレベルのマルチキャストである。ここで、オーバーレイネットワーク上には、マルチキャストに参加する端末の数だけアプリケーションレベルの仮想的なリンクが存在する。さらに、この仮想ネットワーク上では端末の頻繁な参加・退出があることから次数の変動が大きい。以上の特徴をもつオーバーレイネットワーク上のそれぞれの利用者端末とアクセスルータとの間のアクセスリンクの帯域が、全ての利用者間で均一ではないことから、利用者端末から送信されるデータが使用する通信帯域を考慮して各利用者端末の通信品質を向上するようなマルチキャストのためのデータ配信経路を効果的に構成することが難しい。

これらのように、無線ネットワークのひとつである無線アドホックネットワークにおけるマルチホップ通信と、有線ネットワーク上で利用者端末（エンドシステム）同士でデータの複製・転送を行うエンドシステムマルチキャストは、データの配信のための経路制御を行う上で、次数が多く、また動的に次数が変動するため、通信帯域を各経路に分配できる帯域を予測することが困難である。以上のことから、利用可能な帯域が変動する情報通信基盤においてマルチメディア通信サービスを提供する場合、すなわち、無線アドホックネットワークのマルチホップ通信および有線ネットワーク上のエンドシステムマルチキャストにおいて、マルチメディア通信サービスの品質を向上するためには、正確な帯域予測に基づく適応型の経路制御技術が重要となることが分かる。

## 1.2 本研究の目的

### 1.2.1 本研究の目的

本研究では、ユビキタス情報環境における利用者同士のコミュニケーション手段として有用であるマルチメディア通信に対して、ユーザレベルのサービス品質を向上させるため、その土台となるネットワークレベルのサービス品質の向上させることを目的とする。

そこで本研究では、ネットワークレベルのサービス品質の向上を実現するためのアプ

ローチとして，ネットワーク QoS 技術の一つである経路制御技術を活用する．ここで，マルチメディア通信のネットワークレベルのサービス品質として，特に重要である通信帯域を本研究では主に取り扱い，ネットワークレベルのサービス品質に基づき，利用可能な帯域を適切に割り当てられる経路を構成する，ネットワークの経路制御法の確立を目指す．

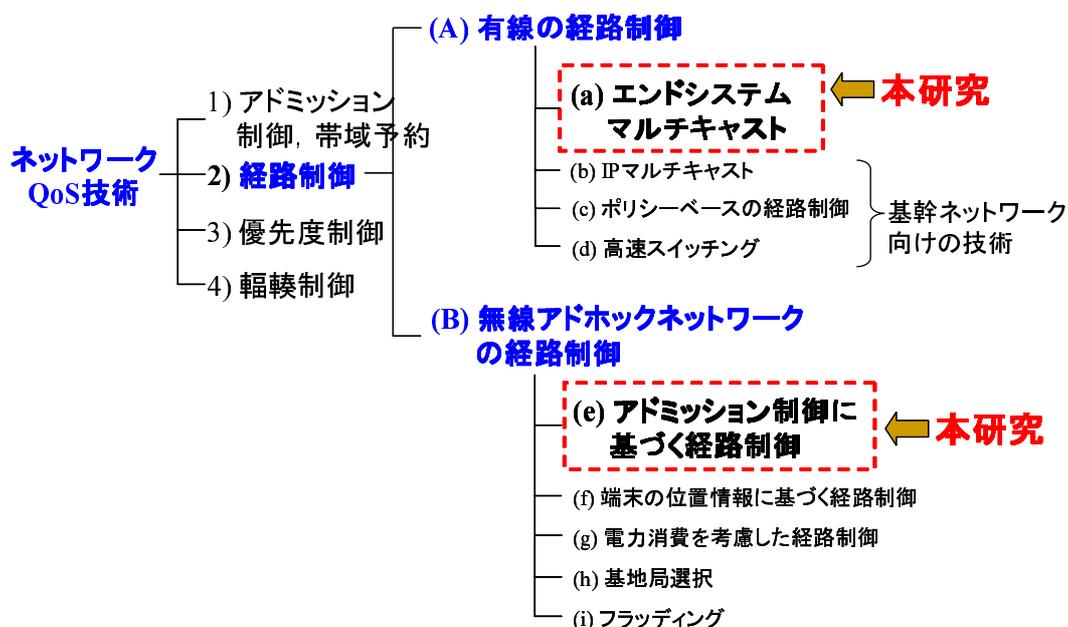


図 1.2: ネットワーク QoS 技術の対象分野

### 1.2.2 本研究の課題

本研究の目的であるネットワークレベルのサービス品質の向上のために，本研究では，無線ネットワークおよび有線ネットワークにおいてセッションに対して通信帯域を積極的に割り当てるための適応型の経路制御を行う．無線アドホックネットワークのマルチホップ通信および有線ネットワーク上のエンドシステムマルチキャストにおいてネットワークレベルのサービス品質を向上する経路制御法の実現のためには，以下のような解決すべき技術課題が存在する．

(T1) 無線アドホックネットワークの経路制御において，帯域予測の精度が低いためセッ

ションが所望の通信帯域の充足が困難である

(T2) エンドシステムマルチキャストのデータ配信木において、多対多通信においては各端末が複数の配信木の間で消費する帯域を予測することが困難であるため、利用者端末間で通信品質に偏りが生じる

(T1) に関しては、第3章において、各端末自身が把握している帯域情報を端末間で交換しCSR内の端末(2-hopの隣接端末)が各々把握している帯域情報を収集することでCSR内の消費帯域を高精度に予測し、収集した情報を用いたアドミッション制御を行う経路制御手法を適用することにより解決を図る。

(T2) に関しては、第4章において、1回の配信木の再構成の対象を局所的な部分木にしぼることで複数の配信木の間で消費する帯域の予測を可能にし(容易にし)、これに基づき配信木の局所的な再構成を自律分散的に反復して行うことで通信品質の偏りを段階的に軽減する配信木の構成法を適用することにより解決を図る。

### 1.3 本研究の位置付け

本研究の位置付けを図1.3に示す。この図は、ユビキタス情報環境におけるネットワーク技術の対象領域をネットワークアーキテクチャのように階層化して表したものであり、その最上位層に位置するのは人間の生活に直接的につながる情報処理のためのネットワーク技術であり、最下位層がユビキタス情報環境における情報通信の基盤となるネットワーク(技術)である。

本研究では、ネットワークQoS技術の重要な一要素である経路制御によって、有線ネットワーク及び無線ネットワークにおいてネットワークレベルのサービス品質を向上することに従事する。ここで、1.1.2節で述べたように、ネットワークレベルのサービス品質が向上することでアプリケーションレベルのサービス品質およびユーザレベルのサービス品質の向上が促進するため、本研究では結果として全てのネットワーク技術の領域に関わる

サービス品質の向上をさせ、ユビキタス情報社会のための情報通信サービスの高度化に貢献すると考えられる。

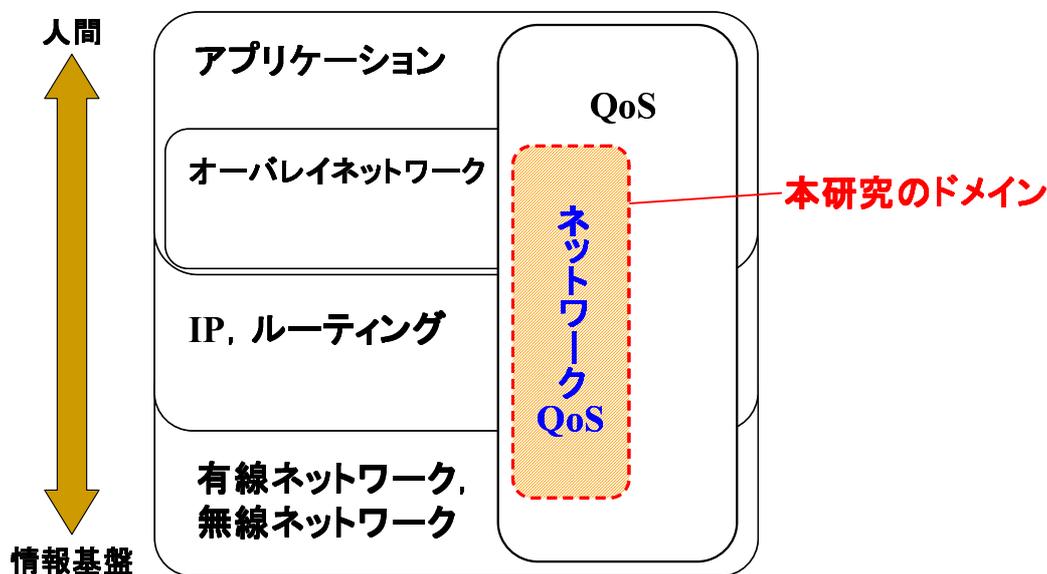


図 1.3: 本研究の位置付け

## 1.4 本論文の構成

図 1.4 に本論文の構成を示す。本論文は全 5 章からなる。第 1 章「序論」では、研究の背景を示し、本研究の目的と課題、および本研究の位置付けを示した。

第 2 章「サービス品質向上のためのネットワーク経路制御に関する課題」では、ネットワークレベルのサービス品質の向上のための経路制御における課題を解析し、その結果、(A) 無線ネットワークの経路制御および (B) 有線ネットワークの経路制御に関して、それぞれの技術課題を明確にする。

第 3 章「無線アドホックネットワークのための高精度な帯域予測に基づく経路制御法」では、ネットワークレベルのサービス品質を向上するための無線ネットワークの経路制御における技術課題である、(T1)「無線アドホックネットワークの経路制御において、帯域予測の精度が低いためセッションが所望の通信帯域の充足が困難である」(高精度な帯域

の予測を行う必要性，すなわち問題の所在は「帯域予測の精度が低い」ということ)ことを解決するため，各端末自身が把握している帯域情報を端末間で逐次交換しCSR内の端末(2-hopの隣接端末)が各々把握している帯域情報を収集することでCSR内の消費帯域を高精度に予測し，収集した情報を用いたアドミッション制御を行う経路制御手法を提案する．そして，計算機によるネットワークシミュレーションによる性能評価を通し本提案手法の有効性を示す．

第4章「部分木の帯域予測に基づくエンドシステムマルチキャストの配信経路の制御法」では，ネットワークレベルのサービス品質を向上するための有線ネットワークの経路制御における技術課題である，(T2)「エンドシステムマルチキャストのデータ配信木において各端末が複数の配信木の間で消費する帯域を予測することが困難であるため，利用者端末間で通信品質に偏りが生じる(すなわち全ての配信木の全ノードにおけるそれぞれの受信ビットレートの間隔のばらつきが大きい)」(高精度な帯域予測の必要性)ことを解決するため，1回の配信木の再構成の対象を局所的な部分木にしぼることで複数の配信木の間で消費する帯域の予測を可能にし(容易にし)，これに基づき配信木の局所的な再構成を自律分散的に反復して行うことで通信品質の偏りを段階的に軽減するツリー構成法を提案する．そして，シミュレーションによる評価により，本提案手法の有効性を示す．

第5章「結論」では，まとめとして研究成果を要約し，本研究の結論を述べ，今後の課題を挙げる．

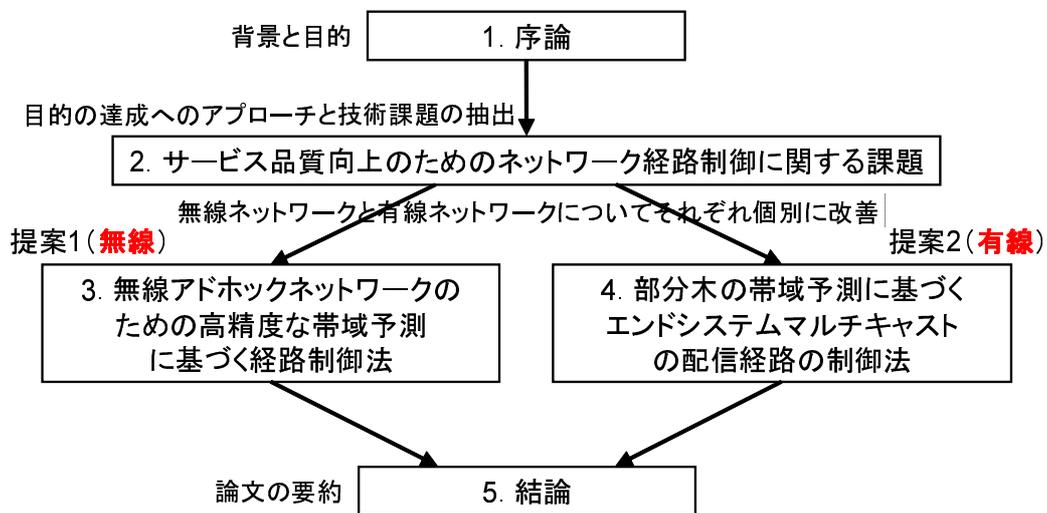


図 1.4: 本論文の構成

## 第2章 サービス品質向上のためのネットワークの経路制御

### 2.1 はじめに

1章で述べたように，ユビキタス情報社会における利用者同士のコミュニケーションを実現するサービスの品質を向上するためには，ネットワークレベルのサービス品質を向上させることが非常に重要となる．本研究では，そのアプローチとして，ネットワーク QoS 技術の一つである経路制御の技術を適用することを述べた．特に，基幹ネットワークの性能向上という視点からではなく，いつでも・どこでも，利用者同士のコミュニケーションを行う上で有効となる (A) 有線ネットワークにおいて利用者端末間でデータの複製・転送を行うエンドシステムマルチキャスト，および (B) 無線ネットワークの中でも，無線端末のみで構成され，無線端末間でマルチホップ通信を行う無線アドホックネットワーク，の2つのネットワーク形態に着目した．これらは，有線ネットワークおよび無線ネットワークのそれぞれにおいて，従来のインターネットの経路制御の方式とは異なり，利用者端末による経路制御を土台として通信を行う環境である．

本研究では，各セッションに対する経路に通信帯域を積極的に割り当てる経路制御法によって，上述の2つの環境におけるネットワークレベルのサービス品質の向上を実現する．

本章では，通信帯域を積極的に割り当てる経路制御法に関して，(A) および (B) におけるデータの配信に関する特徴および関連研究について述べた後，それぞれの従来手法の問題点すなわち解決すべき技術課題を明確にする．

## 2.2 ネットワーク資源を積極的に割り当てるネットワーク経路制御

エンドシステムマルチキャストを適用するネットワーク（オーバーレイネットワーク）や無線アドホックネットワークにおいては，インターネットにおける経路制御の方式とは異なる経路制御が必要な大きな要因として，構造的な違いが挙げられる．

それぞれのネットワークの構造を図 2.1～ 図 2.3 に示す．

まず，インターネットは，図 2.1 に示すように，経路制御の適用範囲ごとに階層化した構造かつ静的な構成であり，その規模に対して考慮すべきルータ間のリンク（辺）の次数が少なく，かつ半固定的である．このようにインターネットにおいては，経路制御に必要な要素について運用・管理の容易性のためのネットワーク間の相互の接続性の向上・ネットワークアドレスの集約化，障害に対する安定性，ネットワークの高速化などの視点に立ったものである．このように，インターネットの経路制御は基幹ネットワークのルータによる経路制御であるといえる．

一方，エンドシステムマルチキャストを行うネットワークは，図 2.5 に示すように，各アクセスネットワーク上の，サービスに参加する利用者端末の数の 2 乗だけ仮想的なリンクが存在し，端末の頻繁な参加・退出が多いことから，ネットワークの規模に対して次数が多く，次数の変動が大きいという傾向がある．P2P 通信であることから柔軟なネットワーク環境である反面，これらの特徴は分散的に経路制御を行う上では大きな障害となる．また，基幹ネットワークのネットワークリソース（帯域）は潤沢であることから，アクセスネットワークのリンク（アクセスリンク）がボトルネックとなる．すなわちエンドシステムマルチキャストによって配信されるデータの通信帯域は，それぞれのアクセスリンクの帯域の大きさと配信経路の構成の仕方の両方に左右される．すなわち，エンドシステムマルチキャストの配信経路の制御はインターネットの経路制御と異なる仕組みが必要であり，また，それぞれのアクセスリンクの帯域の不均一性を考慮しつつ，各マルチキャ

ストセッションに通信帯域を適切に割り当てることができるデータ配信木の構成法の実現は、ネットワークレベルのサービス品質を向上する上で重要な課題である。

また、無線アドホックネットワークは、通信空間の共有してマルチホップ通信を行うことに加え、参加・退出以外にも、端末の移動性があるネットワークであり、次数が多く、次数の変動が大きい。無線アドホックネットワークでは、エンドシステムマルチキャストで考慮するネットワークレベルのサービス品質「通信帯域の公平化」などを実現する場合にも、それ以前に通信帯域の充足する必要があるという部分問題を抱えている。

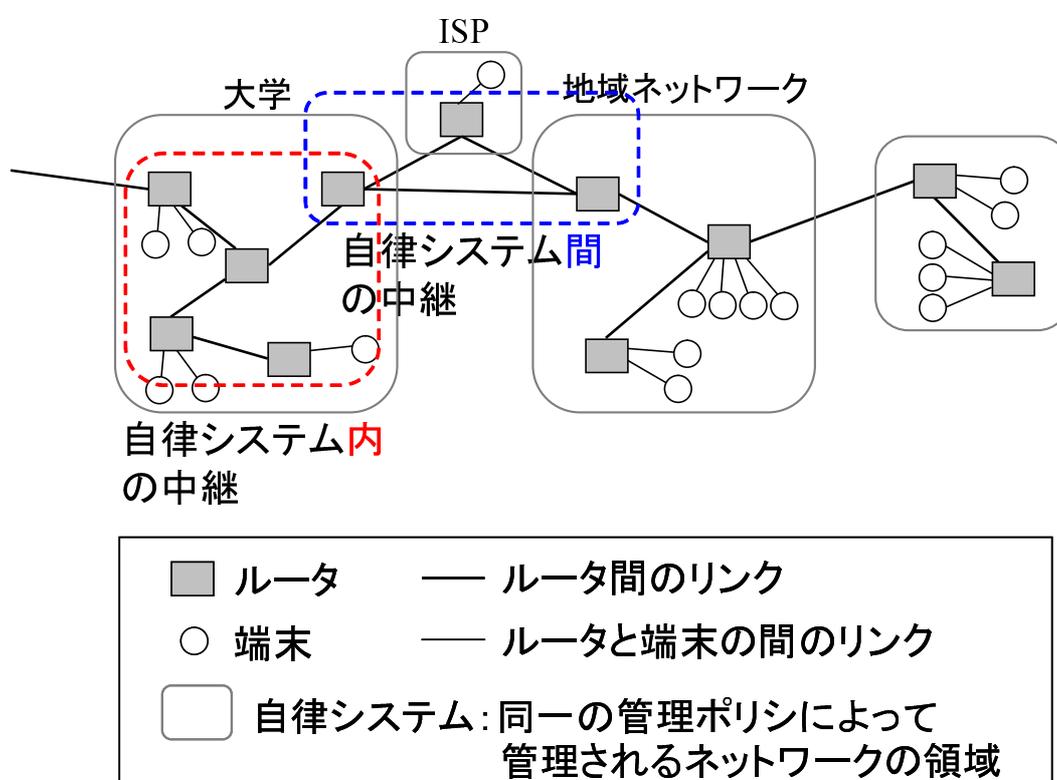


図 2.1: インターネットの構造

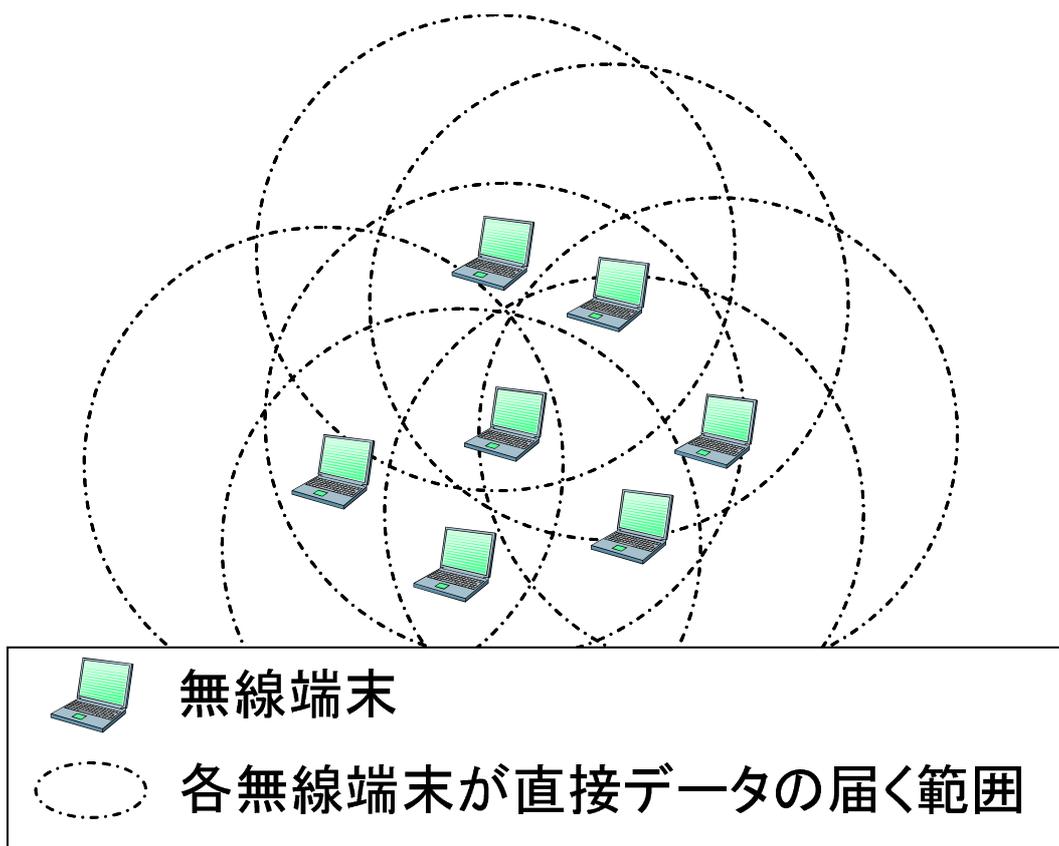


図 2.2: 無線アドホックネットワークの物理的な構造

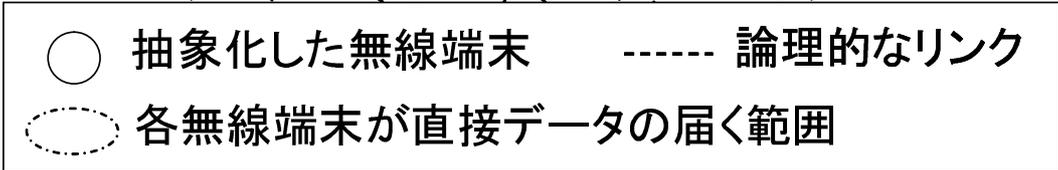
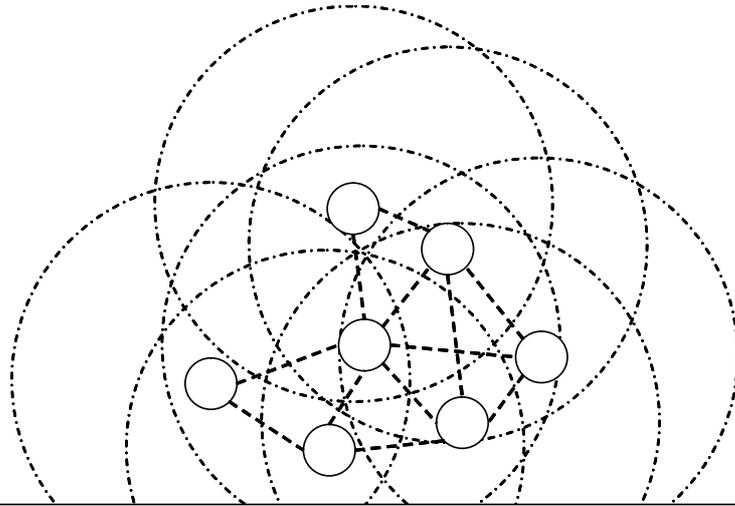


図 2.3: 無線アドホックネットワークをリンクの数に注目して抽象化

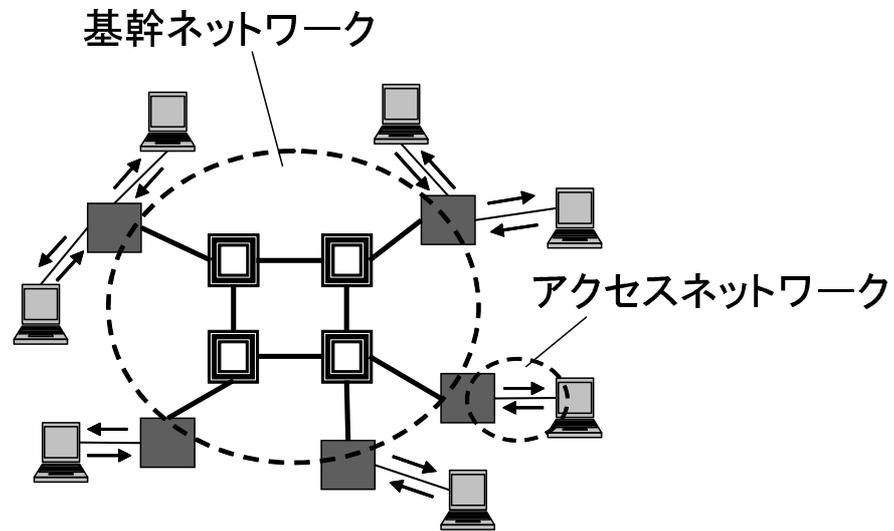


図 2.4: エンドシステムマルチキャストを行うネットワークの物理的な構造

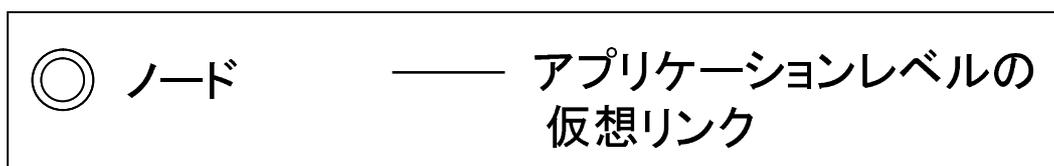
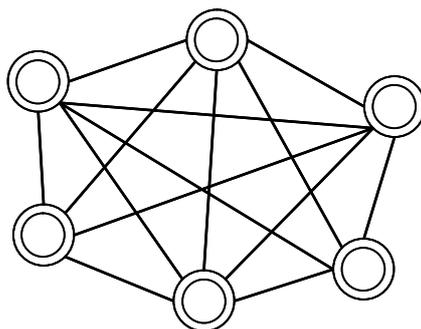


図 2.5: エンドシステムマルチキャストを行うネットワークをリンクの数に注目して抽象化

## 2.3 経路制御の現状と課題

### 2.3.1 有線の経路制御の関連研究

マルチキャストの方式には、IP マルチキャストとエンドシステムマルチキャストがある。IP マルチキャストは、図 2.6 のように、ルータ間でマルチキャストデータを複製・転送しそれぞれの受信先に配信を行う。IP マルチキャストは帯域の利用効率が高いというメリットがあるが、マルチキャストを行う専用のルータが必要となり、またその管理システムの必要性があり、利用できる環境が限定的である。インターネットにおいても試験的に運用されるに留まっている。

これに対し、図 2.6 のように、エンドシステムマルチキャストはエンドシステムすなわち利用者端末においてマルチキャストデータを複製・転送しそれぞれの受信先に配信を行う。エンドシステムマルチキャストの仕組みを適用する理由は、適用の容易さと全てユニキャストでデータを配信することに比べての通信品質の向上の可能性があることの 2 点である。利用者端末間で構築される P2P 通信のための仮想ネットワークにおいては、基幹

ネットワークのネットワークリソース（帯域）は潤沢であり，エンドシステムとインターネットに繋がるルータ（アクセスルータ）との間のネットワークであるアクセスネットワークのリンク（アクセスリンク）がボトルネックとなる．特に，各アクセスネットワークの間に（リンク帯域の）不均一性があることから，広帯域のアクセスリンクを持つ端末が多くデータの複製・転送を受け持つことで，ユニキャストに比べ通信品質の向上が可能であるため，エンドシステムマルチキャストは有用といえる．また，IP マルチキャストに比べて帯域の利用効率は低いが，マルチキャストに際して専用のルータを必要としないため，インターネット上のどこでも使えるという大きなメリットがある．

図 2.7 は，エンドシステムマルチキャストのデータ配信木の構成法に関する関連研究を，その達成目標で分類したものである．

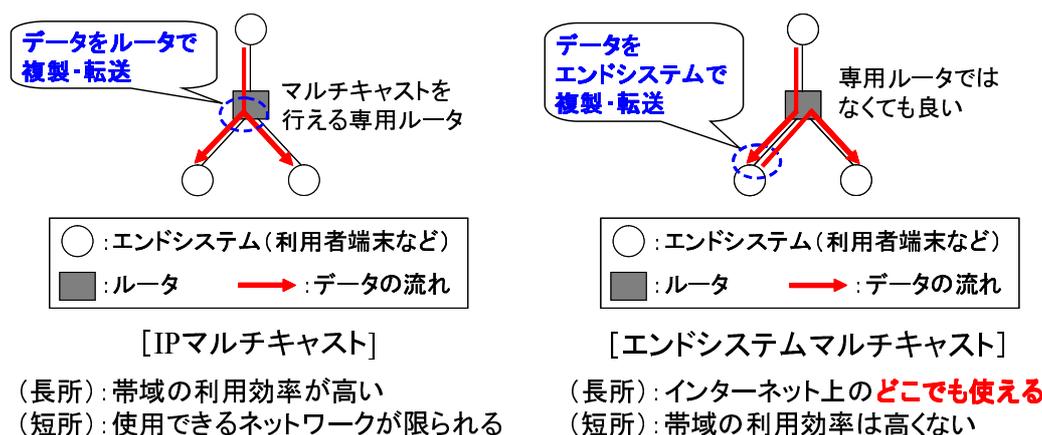
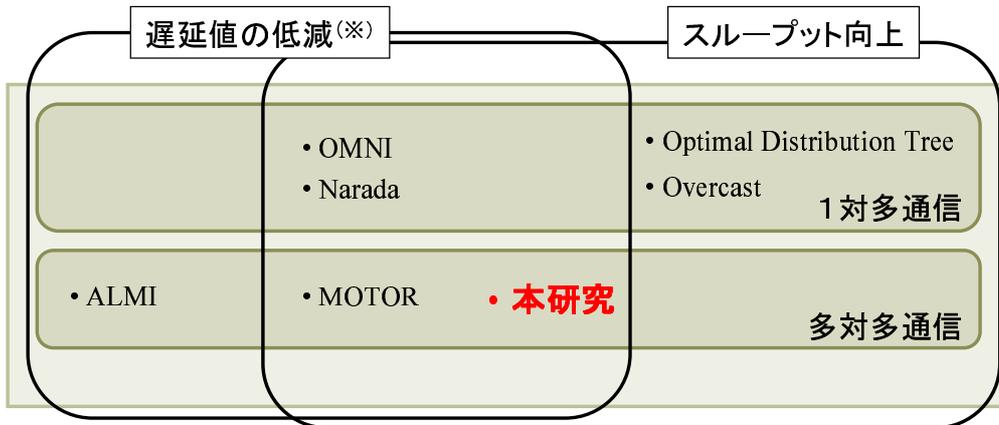


図 2.6: エンドシステムマルチキャストと IP マルチキャスト

一対多，および多対多マルチメディア通信サービスにおける ALM ツリー構築に関する既存研究は，ALM ツリー構築アルゴリズムが，通信遅延と通信帯域のどちらを優先的に考慮するかによって大きく 2 つに分類できる [19]．

[15, 14, 10, 20, 21, 22] 等は，通信遅延の低減を目的とした研究例であり，遅延最小全域木や最短経路木を目標とする ALM ツリーの構築手法をそれぞれ提案している．このような，遅延をメトリックとした研究は，リアルタイムのラジオストリーミング放送やオンラ

(※)本手法の場合は, 最大許容遅延値の保証



[ エンドシステムマルチキャストの配信木の構成法の分類 ]

図 2.7: エンドシステムマルチキャストの配信木の構成法の分類

インゲームのような, 参加数が多く比較的大規模な P2P 環境において, リアルタイム性を重視するマルチキャスト通信が行われるようなサービスに対して効果を発揮するものと位置付けられる. 一方 [13, 11] 等は, 通信帯域の効果的な利用を目的とした研究である. 本研究で対象とするビデオ会議等のマルチメディア通信サービスにおいては, 各利用者になるべく公平に, かつなるべく高品質なサービスを提供することが望まれる. 従って, 異なったアクセスリンクの通信帯域の制限の中で, 各受信端末のマルチメディアデータの受信ビットレート向上を実現するために, マルチメディアデータの送受信に必要な帯域を効率的に分配できるツリーの構成が問題となる. すなわち, 上記の分類では後者にあたる.

後者を目的とした研究 [13, 11] では, 各利用者端末間で使用する帯域を指標に ALM ツリーの構築を行っている.

### 2.3.2 無線の経路制御の関連研究

無線ネットワークの種類には, 衛星ネットワーク, 無線 LAN, 携帯電話と基地局によるセルラネットワーク, 無線端末のみで構成される無線アドホックネットワーク等がある. 無線 LAN およびセルラネットワークは, 有線のインフラ側への接続口となるアクセスポ

イント（無線 LAN の場合）や基地局（セルラネットワークの場合、無線 LAN でもこの語句を用いる場合もある）とエンド端末との間を無線によって通信を行う。基本的に無線の区間であるアクセスポイントとエンド端末の間では他の端末を一度も経由せずに直接データを送受信する「シングルホップ」の通信であり、インターネットのラストワンマイルといった形で実現されている。無線アドホックネットワークは、図 2.8 に示すように、PDA やノート PC、センサなどの無線端末のみによって自律分散的に構成されるインフラレスな無線ネットワークであり、データの送信元から宛先までのデータの配信には、無線端末がルータとして動作しパケット（すなわちデータ）を中継する「マルチホップ」の通信により行われる。無線アドホックネットワークは、事前に準備された固定的な通信インフラを必要とせず、一時的・即興的に構成が可能であることから、災害などの緊急時の連絡網としてや、通信インフラの整備が不十分な場所での代替の通信手段、イベント会場や会議のような恒久的でない情報通信が行われる場合など、様々な場面でのサービスの提供手段として注目されている。また、インフラから完全に独立した形態のネットワークとして利用されることに加え、近年では、無線アドホックネットワークとインターネットとを相互に接続し、無線アドホックネットワークの技術を応用して無線によるマルチホップ通信が行われるアクセスネットワークを構成することで、拡張性の高い柔軟な情報通信基盤を構築する研究も行われている [23, 24, 25]。このように、無線アドホックネットワークはユビキタス情報環境のコンセプトのひとつである、いつでも・どこでも情報通信を行うためのネットワークとして特に有用な無線ネットワークである。

無線アドホックネットワークを構成する際の無線通信デバイスの技術（または規格）には、無線 LAN にも採用されている IEEE802.11x、bluetooth (IEEE802.15) や主にセンサネットワークで採用される ZigBee (IEEE802.15.4) 等がある。

無線アドホックネットワークは、柔軟かつ利便性の高いネットワーク構成である反面、通信空間を共有するという性質および端末の移動があるという性質により、比較的静的に

かつ構造化されたインターネットで利用される「基幹ネットワーク向けの経路制御」法を適用することが難しい。すなわち、ネットワークの規模に比べて次数が多く次数が変動するような、動的なネットワークの変動に追従した独自の経路制御の手法が必要である。また、経路制御の位置付け自体もインターネットと大きく異なり、汎用性が高い経路制御法を確立することは非常に困難である。まず、インターネットにおいては、そのネットワークのドメインや階層毎に利用される経路制御の方式は異なるが、基本的には様々なネットワークが相互に接続された環境において全体として適切な経路制御を行うことが求められる。一方、無線アドホックネットワークにおいては、無線アドホックネットワーク毎の、提供する情報サービスの目的やネットワークの環境それぞれに適した経路制御を行うことが重要となる。以上のことから、無線アドホックネットワークにおける経路制御の研究は非常に盛んであり、IETFのMANET Working Groupなどで、無線アドホックネットワークにおける様々な経路制御方式についての検討がなされている [26]。

まず、無線アドホックネットワークの各経路制御法には、経路の探索や作成・維持に関して技術的にベースとなる経路制御手法があり [16, 17, 27, 28, 29]、ベースとなるこれらの手法は、送信元から宛先までの近似的な最短経路の構築を自律分散的に行う事を主目的にしつつ、データが届くかどうかというリンクの接続性を考慮するだけでなく、無線アドホックネットワークを構成する各端末がバッテリー駆動であることや計算処理能力などのネットワーク資源が少ないことを考慮し、それぞれ効率的に経路制御を行う工夫が施されたものである。ここで、無線アドホックネットワークのための経路制御の基本となる技術的な特徴を説明するため、経路探索・維持の方式、経路情報の保持形式、データの配信先（宛先）の数の、それぞれの観点から分類・整理する。その後、本研究で対象とするアドミッション制御に基づく経路制御を含め、対象とするサービスやネットワーク環境に応じたそれぞれの経路制御についての関連研究を紹介する。

## 経路探索・維持の方式

無線アドホックネットワークにおける経路制御法は、オンデマンド（またはリアクティブ型と呼ばれる）の方式 [17, 16] と、テーブル駆動（またはプロアクティブ型と呼ばれる）の方式 [28, 27] の2つの方式に大きく分類できる。オンデマンド方式は、データの送信の必要が生じた際に経路の構築を行う方式であり、データの送信が開始されるまでの待ち時間が比較的大きくなる一方、その場で必要となる経路に関する制御のみを行うため、経路情報に関するメッセージおよび計算に関するオーバーヘッドが小さくできるというメリットがある。一方、テーブル駆動方式は、ネットワーク内の全ての端末を宛先とした経路表を常時作成・更新することで、データの送信までの待ち時間が小さく、また通信のタイムアウトが少ないというメリットがあるが、経路表の整合性を保つために、経路情報に関するメッセージをネットワーク全体で逐次交換するため、メッセージおよび計算に関するオーバーヘッドが大きいというデメリットがある。また、オンデマンド方式とテーブル駆動方式の複合方式として、ZRP[29] などがある。ZRPでは、ネットワークを階層化して、自分の近くに存在する端末に関してはテーブル駆動方式で経路表を作成し、遠方に存在する端末との通信を行なう場合にはオンデマンド方式の制御を用いて経路を構築する方式である。

## 経路情報の保持形式

各経路制御手法における経路情報の保持形式については、次の転送先（次ホップ）の端末の指定と自身から宛先までの距離（ホップ数）を保持するような、距離ベクトルでの表現を採用しているものが多い。この場合、中継端末はデータを受信した際に、自身の経路表を参照して次ホップの端末を把握し、データをその端末へ転送する。一方、DSR[17] などでは、送信元始動型であるソースルーティング方式を採用している。この方式は、距離ベクトル方式と異なり、データの送信元が宛先への経路を全て把握し、データに経路に関

する情報を（パケットの拡張ヘッダ内に）付加して送信する方式である．経路情報とは，宛先に到着するまでに経由する全ての端末のリストであり，これによって中継端末は経路表を参照せず，データに付加されている経路情報を参照して次ホップの端末に向けてデータを転送する．

#### データの配信先（宛先）の数による方式の違い

無線通信は，ケーブルで繋がっている相手のみとしか物理的な通信ができない有線の通信とは異なり，電波が届く範囲にいる端末と通信チャネルを共有する，すなわち通信する空間を共有しているため，通信は物理的にはブロードキャストである．正確には「電波が届く範囲」が論理的なデータの送受信が可能な範囲とは異なるが，これについては次節で詳述する．しかしながら，論理的にデータを送受信する上では通信相手を指定することが必要なことは，有線の場合と同様であり，上述の AODV や DSR などの単一の宛先へのデータの配信（ユニキャスト）のための経路制御の他，複数の宛先へのデータの配信（マルチキャスト）の経路制御や，ネットワーク内の全ての端末へのデータの伝播（フラッディング）を効率に行うための手法なども提案されている．

マルチメディア通信向けのセッションに所望な通信帯域に適応した（通信帯域を積極的に割り当てる）手法，主に車車間通信などに適した手法である GPS などを利用した端末の位置情報に基づく手法，無線アドホックネットワークのひとつでもあるセンサーネットワーク向けの電力消費量を抑制する手法などの，それぞれの通信の主目的に応じた経路制御法が提案されている．

本研究では，無線アドホックネットワークにおける経路制御に対して様々な方向性がある中で，利用者同士のコミュニケーションのためのサービスである音声チャットやビデオ会議などのマルチメディア通信に重要となるネットワークレベルのサービス品質の中の「通信帯域」の観点から，セッションの経路に通信帯域を積極的に割り当てる経路制御方式，すなわちアドミッション制御に基づく無線アドホックネットワークの経路制御を行う．

無線アドホックネットワークを構成する際の無線通信デバイスの技術（または規格）には、無線LANにも採用されているIEEE802.11x, bluetooth (IEEE802.15) や主にセンサネットワークで採用されるZigBee (IEEE802.15.4) 等がある。また、図2.9は、アドミッション制御に基づく無線アドホックネットワークの経路制御法に関する関連研究を、そのメディアアクセス方式で分類したものである。CDMA/TDMA方式は、セルラネットワークで利用されているアクセス方式である。携帯電話網からの片方向な通信サービス向けである。一方、IEEE802.11方式は、ネットワークの設置およびサービスの利用に対するコストが小さいこと、および利用できる帯域が比較的大きいといった、どこでも自由にアドホックネットワークを構築し無線端末（利用者端末）間でマルチメディア通信を行う上での長所がある。そこで、本研究では、IEEE802.11を使用する無線デバイスとした無線アドホックネットワークを対象として、従来の手法の問題点およびそれを解決した経路制御手法の検討を行う。

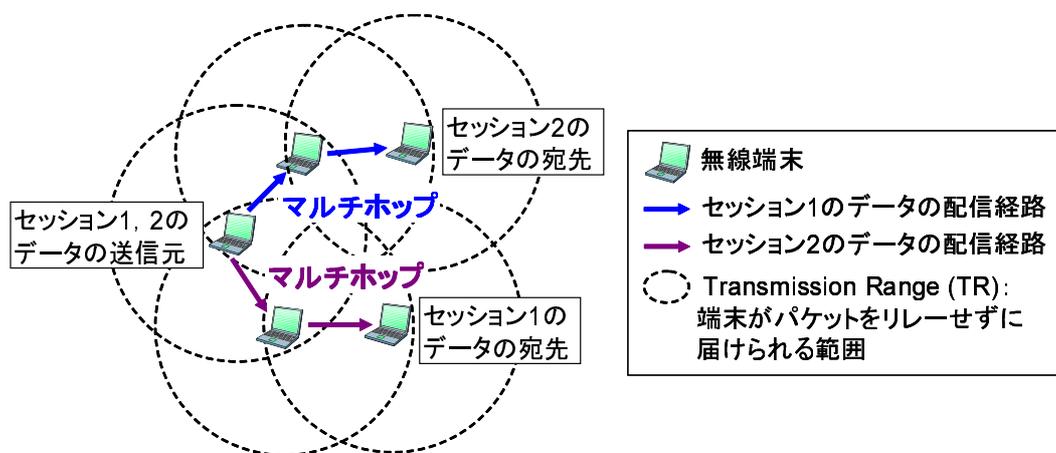
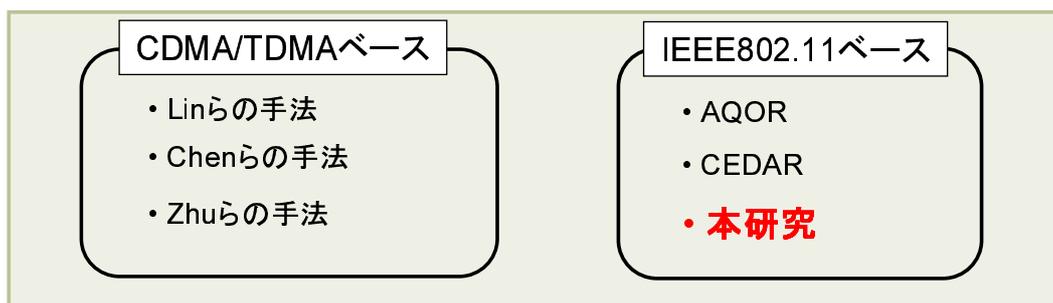


図 2.8: 無線アドホックネットワークの概要



[ アドミッション制御に基づく経路制御の関連研究の分類 ]

図 2.9: アドミッション制御に基づく無線アドホックネットワークの経路制御法の分類

### 2.3.3 サービス品質向上のための適応型経路制御を実現するための課題

#### (A) 有線ネットワーク

多対多通信の特徴として、複数のセッションの間で各アクセスリンクの帯域を共有することが挙げられる。

従来手法は、単一の送信元によるマルチキャスト（1対多通信）のためのデータ配信木の構成法であり、1つのセッションの配信木でノードを管理し、各ノードに分配する通信帯域を最大化する配信木を構成する [11, 13]。このような配信木の構成法は、単一の配信木に対しては利用者端末間の通信帯域の公平性を実現することが可能である。しかしながら、図 2.10 に示すように、多対多通信に適用した場合、それぞれの配信木は、他の配信木に対してどのように通信帯域が適切に分配できるかを考慮せずに独立して構成していくことになる。そのため、他より大きな上り帯域を持つアクセスリンクに各配信木の子ノードが過剰に集中し上り帯域が圧迫され、その結果、自身が受信するデータの品質が高いとしても、各子ノードの通信品質が悪化するのである。

以上のように、有線ネットワークにおいては、(T2) エンドシステムマルチキャストの配信経路の制御における帯域予測が困難なことによる、通信品質の偏りの発生（公平性のある多対多通信の実現が困難）、という技術課題があることを示した。

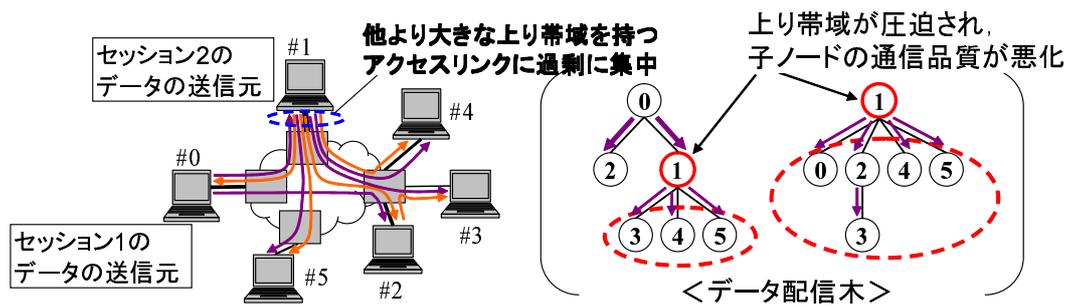


図 2.10: 多対多通信のためのエンドシステムマルチキャストの配信経路の制御に関する技術課題

## (B) 無線ネットワーク

図 2.12 のように，無線アドホックネットワークにおいては，各端末はデータを送信する際，Transmission Range (TR) 内の各端末の通信だけではなく TR の外側の IR 内の各端末の通信に関しても通信を妨害する / される対象となる．

本論文において，任意の端末の「自己発生トラヒック」とは，その端末自身が送信すべきセッションのデータ（フロー）を表す．この場合の送信とは，セッションの送信元としての送信の場合と中継端末として転送する場合の両方の意味である．また，自己発生トラヒックの量とは，その端末自身が単位時間あたりに送信すべきデータの量と定義する．これはすなわち，自身が送信元または中継端末である各セッションの通信帯域の総計と等価である．

ここで，任意の端末 ‘X’ が単位時間に発生する自己発生トラヒックの量を  $B_{self}(X)$  と定義する．端末 ‘X’ の Carrier Sensing Range (CSR) 内に存在する端末の集合を  $N_X$  と定義すると，端末 ‘X’ の消費帯域  $B_{cons}(X)$  は少なくとも，以下のようになる．

$$B_{cons}(X) = \sum_{i \in N_X} B_{self}(i). \quad (2.1)$$

今，端末 ‘X’ が送信元または中継端末となる任意のセッションにおいて，端末 ‘X’ が次にデータをリレーする相手が端末 ‘Y’ である場合，端末 ‘Y’ はこのセッションのための経

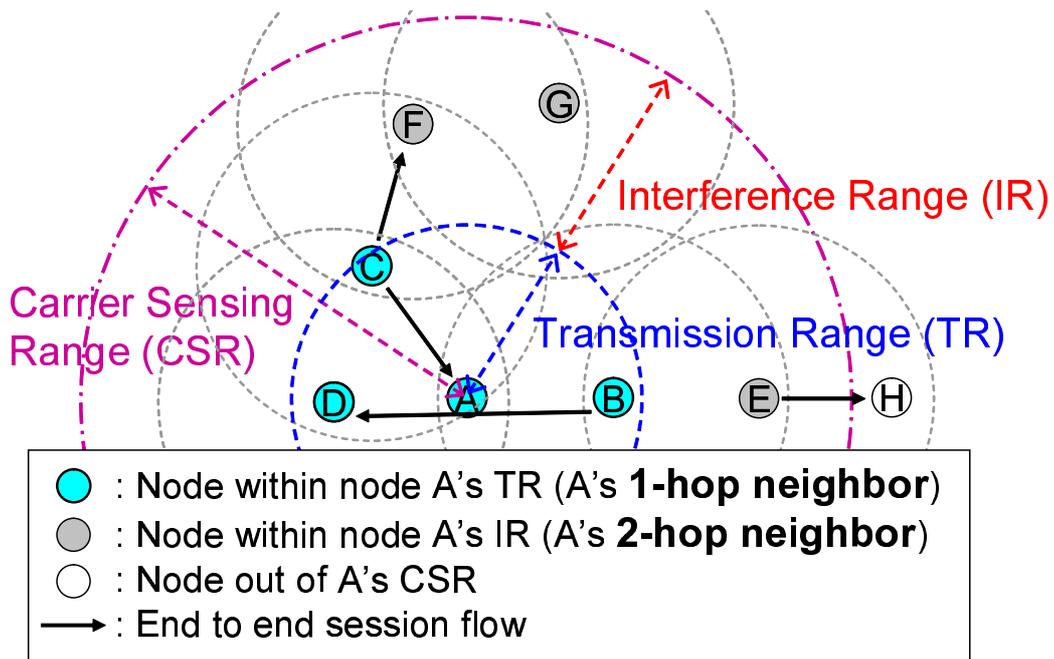


図 2.11: 端末 A における消費帯域

路における「端末 'X' の次ホップ」と表現する．ここで，端末 'Y' が端末 'X' の次ホップである各セッションの経路において必要とする通信帯域の合計，すなわち端末 'X' から端末 'Y' へのトラヒックの総量を  $T_{XY}$  とおく．図 2.11 の場合を例にとると，端末 'A' の自己発生トラヒック  $B_{self}(A)$  については  $B_{self}(A) = T_{AD}$  であり，また，端末 'A' の消費帯域の総計  $B_{cons}(A)$  は，

$$B_{cons}(A) = T_{BA} + T_{AD} + T_{CA} + T_{CF} + T_{EH}. \quad (2.2)$$

ここで，IEEE 802.11 DCF 方式は，CSMA/CA によるメディアアクセス（キャリアセンス）方式に加え，RTS/CTS/ACK ハンドシェイク機構を組み込んでおり，隠れ端末によるデータ（フレーム）の衝突を回避することが可能である．しかしながら，基本的には複数の端末からの同時のデータ送信を回避することによる衝突の回避が行われるとしても， $B_{cons}(X)$  が無線デバイスがサポートする最大の利用可能帯域  $B_{max}$  を超える通信の処理は

スムーズに行われない。これは、単位時間内に処理が可能なチャネルの容量  $B_{max}$  を超える過剰なトラヒックに対しては、どれほど時間的に重複させずに各トラヒック（データ）の送信を完了させたとしても、単位時間内には全て処理しきれないからである。また、それぞれ他のデータとの衝突の回避のために送信が長く待たされ大きな遅延が生じたり、データフレームの衝突回避を行う制御フレームである RTS/CTS フレーム同士またはデータフレームと制御フレームの衝突の結果、データフレーム同士の衝突の回避がうまく行われな可能性が増大する。

このように、 $B_{cons}(X)$  が  $B_{max}$  を超える場合、この端末 'X' の CSR 内の端末の自己発生トラヒックが過剰であるためボトルネックとなり、送信の失敗が頻繁に起こるため、エンドツーエンドにおいては伝送遅延の増加およびパケットの到達率の低下が起こり、セッションが必要とする通信帯域を充足することができなくなる。

以上のように、各セッションが発生させるトラヒックはネットワークの通信資源の容量を超えることができないため、確立要求があるセッションを全て確立しようとしても、セッション数や必要とする通信帯域が大きいセッションが多数ある場合には共倒れとなる可能性がある。

よって、ネットワークの通信資源の容量を超えないようにトラヒックを抑制できる範囲で、確立したセッションの通信帯域を充足することができると試算できるセッションの経路のみを様々な指標（基本的な指標については以下を参照）に基づき選ぶ（受け付ける、許可する）必要があり、これをアドミッション制御と呼ぶ。経路を構築する際に、セッションに必要となる通信帯域を確保できないセッションは確立しない。なぜならば、このようなセッションを確立することでボトルネックを発生させ、自身および確立済みのセッションに対してそれぞれ必要な通信帯域を充足できなくなるからである。

以上のように、アドミッション制御に基づく経路制御とは、発生させるトラヒックを抑制しつつ、利用可能な通信資源の容量内で可能な限り多くの「必要とする通信帯域を充足

できる」セッションのみを確立するような経路制御法であり，そのためにも各端末における消費帯域を高精度に予測することが重要である．さらに，ノードの移動による動的なネットワークトポロジの変動に追従して，リンクの消滅だけではなく，CSR 内に存在する端末の変動による自身の消費帯域の変化がある場合にも，セッションの必要とする通信帯域の充足を維持する経路の再構築を適切に行う仕組みも必要となる．

このような特徴を持つ無線アドホックネットワーク環境において，従来手法は，各端末の消費する帯域の予測の際，自身の TR 内に存在する各端末が行う通信の（単位時間当たりの）通信データ量に関する情報のみを利用して消費する帯域の予測を行うものであった [30] ．

そのため帯域予測の精度が低くなり，結果として，各セッションが単位時間あたりに送受信するデータ量，すなわちセッションの所望する通信帯域を充足することが可能である経路を構築することが困難である．

以上のように，従来手法には，(T1) 無線アドホックネットワークにおいて帯域予測の精度が低いためセッションが所望の通信帯域の充足が困難であるという問題があることを示した．

補足であるが，気温や天候などの因子を無視すれば，建物などの障害物が内見通しのよい空間では，指向性がない一般的な無線 LAN デバイスの場合は通常 TR，IR，および CSR は同心円（同心球）である．本研究では，同一のベンダおよびスペックの無線 LAN デバイスをもつ各端末で構成されるアドホックネットワークを前提とする．よって，ネットワーク内の各端末の TR の半径は全て等しいと仮定できる．IR および CSR についても同様である．また，他の信号との競合が発生しない状態を考えた場合に，ある端末 'X' が送信した信号を感知した端末 'Y' が，その信号を誤りなく正しく復号し，意味のある情報を持つ「フレーム」として認識できる条件とは，その端末 'Y' が信号を送信した端末 'X' の CSR の境界までのほぼ半分の距離の中に存在することである．すなわち，TR の半径は

CSR の半径の半分である .

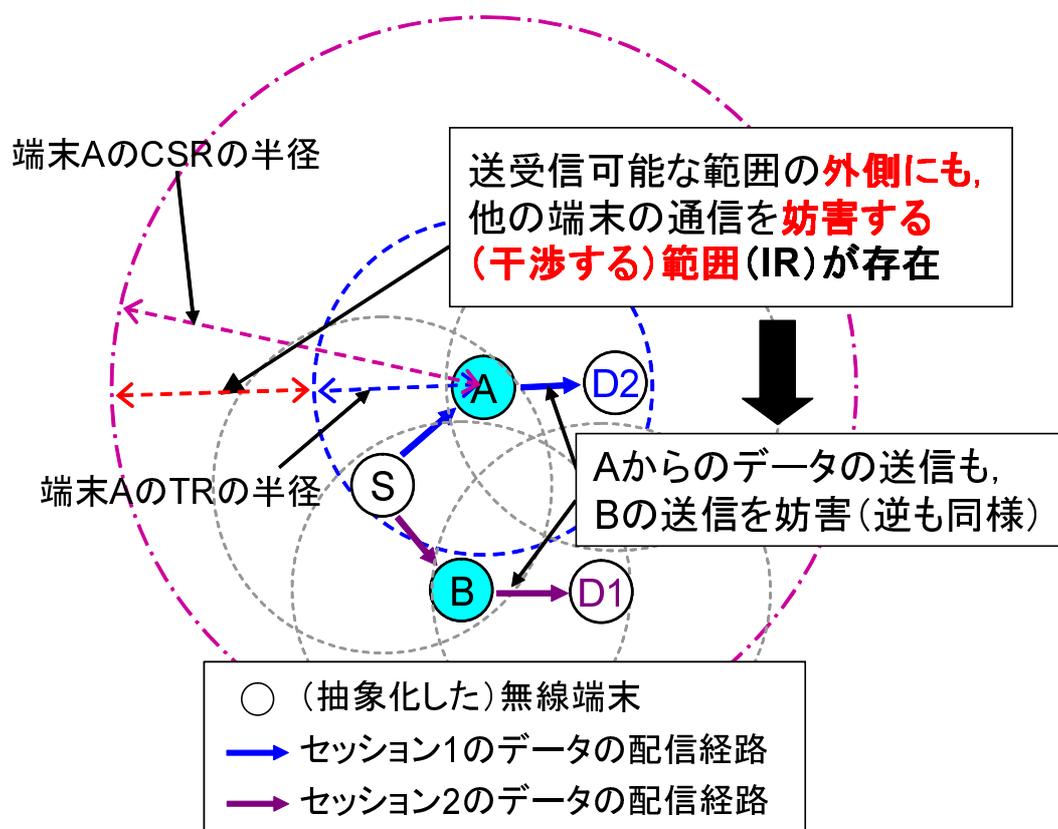


図 2.12: アドミッション制御に基づく無線アドホックネットワークの経路制御に関する技術課題

## 2.4 要約

本章では、ユビキタス情報環境におけるネットワークレベルのサービス品質を向上させるための経路制御における課題を解析し、技術課題を明確にした。その結果、有線ネットワークにおけるオーバーレイネットワーク上でのエンドシステムマルチキャスト、および無線アドホックネットワークにおけるマルチホップ通信を行う環境とは、利用可能な帯域が変動する情報通信基盤であり、ネットワークレベルのサービス品質を向上させるためには、正確な帯域予測に基づく適応型の経路制御技術が重要となることを確認した。

有線ネットワークおよび無線ネットワークの従来の経路制御法における技術的課題はそれぞれ以下のように整理できる。

(T1) (無線の技術課題) 無線アドホックネットワークにおける帯域予測の精度が低いことによる、通信品質の低下(データの配信に対する配信経路上の通信帯域の充足が困難)。

(T2) (有線の技術課題) エンドシステムマルチキャストの配信経路の制御における帯域予測が困難なことによる、通信品質の偏りの発生(公平性のある多対多通信の実現が困難)。

まず、有線ネットワークの経路制御については、多対多の端末間で行うエンドシステムマルチキャストにおいて、従来手法では、複数の配信木を構成する際に各端末に適切に分配できる通信帯域を予測することが困難であったため、公平性のある多対多通信の実現が困難となる問題を指摘した。また、無線ネットワークの経路制御については、従来手法においてはデータを送受信可能な領域のみを考慮して消費帯域を予測するため精度の高いアドミッション制御が行うことができなくなり、サービスの単位であるセッションに必要な通信帯域の充足が困難である問題を指摘した。

# 第3章 無線アドホックネットワークのための高精度な帯域予測に基づく経路制御法

## 3.1 はじめに

2章で述べたように、無線アドホックネットワークはインフラレスで構築できるという利便性があり、イベント会場や即興的なネットワークによる会議運営、災害時の救助支援や連絡網など様々な場面での活用が期待されている。

しかしながら、無線アドホックネットワークにおいては、動的にトポロジが変動することによる通信帯域の時間的な変化があるため、マルチメディアアプリケーションのような高トラヒックな通信を効果的に管理し、ネットワークレベルのサービス品質の向上するための研究は、アドホックネットワークにおける重要な研究課題と位置付けられている。

本研究では、ネットワークレベルのサービス品質としてマルチメディア通信に重要である通信帯域を扱い、無線アドホックネットワークにおける動的なネットワーク資源の変動に適応し、セッションが必要とするエンドツーエンドの通信帯域を充足する経路制御法、Application-level QoS Routing Scheme (ALRS) を提案する。

ALRS は以下の3つの特徴を持つ。(1) Carrier Sensing Range 内の帯域情報を収集するために、拡張した HELLO メッセージによって2-hop 先の隣接端末の帯域情報を獲得する。(2) 各セッションのデータ(フロー)の確立要求があった際、収集した帯域情報に基づきアドミッション制御を行い、フローの配信経路をオンデマンドに構築する。(3) QoS 違反すなわち通信帯域の充足が維持できない場合には、古い配信経路に割り当てた帯域を解

放しつつ経路を削除し，経路の再構築を行う．ここで，「帯域情報」とは，「自己発生トラフィック」および「消費帯域」の2つの要素を含む情報を示している．詳細な定義は次節で述べる．

ここで，各セッションが必要とする通信帯域を充足することとは，各セッションのデータの送信元から宛先まで，すなわちエンドツーエンドのデータパケットの到達率を可能な限り100%にすることを意味する．これに加え，ビデオ会議や音声チャットにおいてスムーズな双方向のコミュニケーションを実現する必要があることを考えると，エンドツーエンド間のパケットの到達遅延は小さいことが望まれる．本手法は上述のメカニズムによる高精度な帯域予測に基づき，高精度なアドミッション制御を行うことを可能とする経路制御法である．これにより従来手法では達成が困難であった，各セッションに必要な通信帯域を充足する経路の構築および再構築を行う経路制御を実現する．

本章では以下，3.2節で関連研究および既存の無線アドホックネットワークの経路制御法の問題点について述べ，3.3節では提案である高精度な帯域予測に基づく経路制御手法，ALRSの詳細について述べる．3.4節では，シミュレーションによる提案手法の性能評価，および従来手法との比較を中心とした考察を行う．最後に，3.5節で本章のまとめを述べる．

## 3.2 既存の無線アドホックネットワークの経路制御法の問題点

利用者同士のコミュニケーションにおいては，経路を構築する際の待ち時間，すなわちセッションの確立自体にかかる待ち時間を短縮すること以上に，マルチメディア通信のひとつのセッションとして比較的長い時間行われるデータ通信に対する通信帯域を充足することが重要である．また，インターネットのような（有線）ネットワークとは異なり，基本的にはその場で一時的に構成される無線アドホックネットワークにおいては，帯域情報の更新に加えて全ての端末に対する経路の維持を行ってしまうとさらにオーバーヘッドが大きくなる．以上の理由から，本手法では全ての端末への経路表を維持しないで，セッ

セッションの確立要求毎に経路探索を行うようなオンデマンド方式の経路制御をベースとし、経路探索とアドミッション制御を各端末で自律分散的に行う。

ここで、各セッションに必要な通信帯域を充足する経路制御を行う際には、高精度な帯域予測に基づいた高精度なアドミッション制御が重要となる。

アドミッション制御に基づく経路制御法には、メディアアクセス方式としてCDMA/TDMAをベースとしたものが多数提案されている [31, 32, 33]。このアクセス方式は、時間同期のもとで集中管理によってチャンネル（またはタイムスロット）を各端末に割り当てる方式であり、チャンネルの競合が生じないアクセス制御を実現しており、QoSの保証が容易である。しかしながら、CDMA/TDMAは携帯電話網などで利用されるメディアアクセス方式であり、従来のIPネットワークにおける利用が難しい。本研究で対象とする、ノートPCやPDAなどの携帯端末（さらには携帯電話においても基地局を利用しない利用者同士の直接の通信に用いられる）において利用されるIEEE 802.11では、干渉範囲を事前に把握し集中制御を行うアクセス方式ではないため、これらの経路制御手法を適用できない。

IEEE802.11方式に対応した、アドミッション制御に基づく経路制御法もいくつか提案されている。Xueらは、周辺の端末のトラフィック量の情報から自身の消費帯域を予測し、アドミッション制御に基づき経路確立および帯域の分配を行う経路制御法であるAQORを提案した [30]。しかしながらAQORは、直接データ通信が可能である範囲 (TR) 内の端末すなわち 1-hop の隣接端末までのトラフィック情報を収集するが、TRの外側にも存在する干渉範囲、Interference Range (IR) の端末のトラフィックの情報が不足しているため、信頼性の高い、すなわち高精度なアドミッション制御が行われない。このため、構築した経路を通してデータが配信される際、アドミッション制御時に予測した以上に過剰なトラフィックが流通することによりボトルネックが発生し、セッションが必要とする通信帯域を充足しないという問題がある。Sivakumarらは、コアノードを用いて効果的にアドミッション制御を行う経路制御手法である、core-extraction distributed routing algorithm (CEDAR) を

提案している [34] . CEDAR は , OSPF のように分散して通信帯域を充足する経路を算出する手法であり , ネットワークのコアとなるグラフを動的に構築・維持する . 送信端末と宛先の端末は , このコアとなるグラフを経由することで , 必要な帯域を持つ経路を効率よく算出することができる . CEDAR はノードの移動がないネットワーク環境をシミュレートした性能評価において高い性能を示している . しかしながら , コアノード移動して選択された経路から離れてしまうと , 経路の再構築を行う際のコストが非常に高くなるのが問題点として挙げられている .

### 3.3 ALRS: An effective Application-Level QoS Routing Scheme の提案

#### 3.3.1 ALRS の概要

2.3.3 で述べたように , 経路上の各端末に対して , セッションの通信帯域を充足できる分の帯域を割り当てることができるかどうかを判断するアドミSSION制御を正確に行うためには , 各端末における消費帯域について高精度に予測することが不可欠となる .

そこで本研究では , 高精度な帯域予測に基づきアドミSSION制御を行い各セッションの必要とする通信帯域を充足する経路制御手法 , ALRS を提案する . ALRS は以下の 3 つの特徴を持つ . (1) 高精度な帯域予測を実現するために , Carrier Sensing Range 内の帯域情報を収集する上で , 拡張した HELLO メッセージにより 1-hop 先の隣接端末の帯域情報に加えて 2-hop 先の隣接端末の帯域情報を獲得する . (2) 各セッションのデータ (フロー) の確立要求があった際 , 収集した帯域情報に基づきアドミSSION制御を行い , フローの配信経路をオンデマンドに構築する . アドミSSION制御は , 候補となる経路上の各端末において自律分散的に行う . 各端末は , 確立済みの経路に加え , 新たに自身が中継端末となる経路が構築された場合に  $B_{cons}(X)$  が  $B_{max}$  を超えるかどうか (ボトルネックを発生させるかどうか) を試算し , 中継端末となることを了承するか拒否するかを決定する . (3)

端末の移動により端末の分布が変化する場合にも，定期的な HELLO メッセージの交換により，その変動に追従する．QoS 違反すなわち通信帯域の充足が維持できないことを検知した場合，即時的に HELLO メッセージを送信することで直ちに古い配信経路に割り当てた帯域を解放しつつ経路を削除し，経路の再構成を行う．

本手法の核となる，HELLO メッセージによる CSR 内の端末の帯域情報の交換メカニズムについては 3.3.2 で述べる．また，通信帯域を充足ができない原因となる過剰なトラヒックの発生を抑制し，通信帯域を充足するセッション（に対するデータの配信経路）を適切に確立するための，上述の収集した帯域情報を利用したアドミッション制御に基づく経路の構築アルゴリズムについては 3.3.3 で述べる．また，QoS 違反の際の経路の動的な再構成について 3.3.4 で述べる．

本手法と従来のアドミッション制御に基づく経路制御法の効果の違いを簡単に述べる．最短経路（最短ホップ数）に基づいて経路を構築した場合にはボトルネックが生じるようなネットワーク環境を考えると，アドミッション制御に基づく経路制御では，通信資源の容量の上限を考慮して確立するセッションの数を制限し，通信帯域が充足するような経路を構築できるセッションのみを確立する．ここで，従来のアドミッション制御に基づく経路制御法においては，通信に影響する範囲を正確に見積もっていないことから帯域予測の精度が低く，アドミッション制御を通じて確立した各セッションにおいても，どこかでボトルネックが生じる可能性があるため常には通信帯域が充足することが困難であるという問題があった．一方，本手法は，従来のアドミッション制御に基づく経路制御法に比べて帯域の予測の精度が高いため，確立したセッションに対してはエンドツーエンドにおいて高いパケット到達率および低遅延でのデータの配信が可能となる．

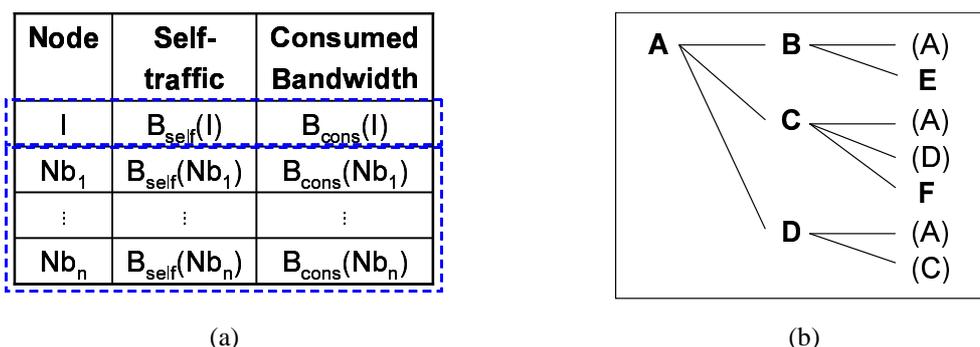


図 3.1: (a) HELLO メッセージのフォーマットおよび (b) NIT の構造.

### 3.3.2 キャリアセンスの範囲を基準とした帯域予測

トラフィックが過剰になることは、既に確立している各セッションのデータ送信に対しての性能の低下も引き起こす。

前述のように、経路を構築する際のアドミッション制御を高精度に行い、さらに、端末の分布の変動により消費帯域が利用可能な帯域を超えるような過剰なトラフィックが生じたことを高精度に検知するためには、消費帯域の高精度な予測が必要となる。そこで本手法では、HELLO メッセージを用いて隣接端末の存在およびそれらの帯域情報を動的に獲得（および通知）し、CSR 内の端末の帯域情報の収集を行うことで、精度の高い帯域予測を実現するメカニズムを提案する。同時に、本メカニズムの機能の一部となる、Neighbors' Information Tree (NIT) を用いた帯域情報の管理と、自身が送信する HELLO メッセージを生成する仕組みを導入する。

本手法では、TR の半径が CSR の半径の半分であることを利用し、消費帯域の試算を行う。以降、本手法が提供する、各端末が自身から 2-hop までの各隣接端末の自己発生トラフィックの情報の収集を行うことにより自身の消費帯域を試算を行う仕組みについて説明する（さらに、各 CSR 内の端末の消費帯域を把握することにより、次節で述べるアドミッション制御をさらに高精度に行うことが可能となるため、本メカニズムでは各 CSR 内の消費帯域の情報の収集も同時に行う。）

## HELLO メッセージ

本手法で情報交換のために利用する HELLO メッセージとは、従来の経路制御法においても、各端末が自身の存在を周囲に通知するために用いられているブロードキャストのメッセージである。ここで、無線においては有線の場合と違い、通信する空間を隣接端末と共有しているため、フレームの送信は物理的には常にブロードキャストで行われることになる。すなわち、送信されたひとつのメッセージを全ての隣接端末で受信することが可能である。また、手法によっては、端末の存在を通知だけでなく、他の情報を付加する場合もある。

そこで、本研究では、CSR 内に存在する端末の帯域情報を収集する上で、図 3.1(a) のような論理的なフォーマットからなる HELLO メッセージを準備する。本研究で用いる HELLO メッセージは論理的に各行 3 つの要素を持つ構造である。各行は、ひとつの端末の情報を保持しており、左から端末 ID、自己発生トラヒック、消費帯域を表す。また、HELLO メッセージは 2 つの領域から構成されている。最初の領域は HELLO メッセージの送信端末についての情報を示しており、これはメッセージの第 1 行目に対応している。2 番目の領域は送信端末からみた 1-hop の各隣接端末の情報を示しており、これはメッセージの第 2 行目から最終行までに対応している。従って、端末 'X' が隣接端末 'Y' から HELLO メッセージを受信した場合、端末 'X' は「端末 'X' からみた 2-hop の隣接端末（端末 'Y' の各隣接端末の分）」の帯域情報を把握することが可能であり、さらに端末 'X' は各隣接端末から HELLO メッセージを受信することで CSR 内の端末の帯域情報を収集することが可能となる。

## NIT: Neighbors' Information Tree

また、各端末は、自身の CSR 内の全ての端末のトポロジおよび帯域情報を図 3.1(b) に示すような論理的な木により管理している。本研究では、これを Neighbors' Information

Tree (NIT) と呼ぶ。ここで、図 3.1(b) は、図の場合の端末の分布に対する端末 'A' がもつ NIT を示している。NIT の根はその端末自身を示しており、深さ  $d$  の各ノードは、根であるこの端末 A からみて  $d$ -hop の隣接端末の ID を示している。この図において、括弧でくくられているノードは、NIT 上の自身の位置より浅いところに同じ ID の端末を示すノードがあることを意味する。これらは、状況に応じて括弧でくくられていないノードと区別できるようになっており、括弧でくくられていないノードによる木の構成が、根が示す端末 A の TR 内に存在する端末および IR 内に存在している端末、すなわち CSR 内のネットワークのトポロジを表すことが分かる。また、任意のノードが示す端末の ID を 'X' とすると、そのノードは端末 'X' の帯域情報である  $B_{self}(X)$  および  $B_{cons}(X)$  を端末の ID とともに保持している。ここで、同じ端末 ID を持つ複数のノードについては重複しないようにしながら（すなわち図 3.1(b) においては括弧でくくられたノードを除いて）、木の根を含め全てのノードについての自己発生トラヒックを累積することで、根であるこの端末についての消費帯域  $B_{cons}(root)$  を算出することが可能である。

図 3.2 は、各端末が HELLO メッセージを受信した際の NIT の更新手順についての擬似コードである。

#### セッション単位のルートエントリ

各端末は、セッション単位のエントリである、ルートエントリの集合からなる経路表を保持する。ルートエントリは以下の書式である。

$$ENTRY = \{sid, seq\_num, src, dest, pre\_node, next\_node, alloc\_bw, state\}$$

ここで、 $sid$  は、どのセッションに対するエントリであるかを表す識別子である。また、 $src$ 、 $dest$ 、 $pre\_node$ 、 $next\_node$  は、データを中継する際に直接関係する他の端末を示す要素であり、 $alloc\_bw$  は帯域情報に関わる要素、経路探索処理に主に使われる要素 ( $seq\_num$ 、 $state$ ) である。

```

Procedure UpdateNIT(he_nodeID[], he_self[], he_cons[], length) {
/* edge(x,y): combination of node x and node y ( $edge(x,y) \neq edge(y,x)$ ) */
/* C: set of combination  $edge(x,y)$  */
/* selftraff(x,y): value of self-traffic on x's children y */
/* consbw(x,y): value of consumed bandwidth on x's children y */

/* The following three parameters show 1st element, 2nd one, and 3rd one
shown in ith line (in HELLO) */
/* he_nodeID[i]: (neighbor) node (ID) shown in ith line (in HELLO),
i.e. one of neighbor node (ID) of the node which sent the HELLO */
/* he_self[i]: value of self-traffic on he_nodeID[i] */
/* he_cons[i]: value of consumed bandwidth on he_nodeID[i] */
/* length: the number of lines in the HELLO */

root:= root on NIT, i.e. which received the HELLO;
x:= the node (ID) which sent the HELLO;
if ( $edge(root,x) \notin C$ )  $C \leftarrow C + \{edge(root,x)\}$ ;
extend  $edge(root,x)$ 's lifetime on C;
i := 1; TEMP =  $\phi$ ;
while(i < length){
  y:= he_nodeID[i];
  if ( $edge(x,y) \notin C$ ){
     $C \leftarrow C + \{edge(x,y)\}$ ;
    TEMP  $\leftarrow$  TEMP + {y};
  }
  extend  $edge(x,y)$ 's lifetime on C;
  selftraff(x,y):= he_self[i]; consbw(x,y):= he_cons[i];
  i ++;
}
while(TEMP! =  $\phi$ ){
  y:= extracted one from TEMP;
  if ( $edge(x,y)$  has expired) remove  $edge(x,y)$  from C;
  remove y from TEMP;
}
}

```

図 3.2: NIT の作成・更新アルゴリズムの擬似コード.

この中で、本節では、帯域情報の取得・更新に関わる要素である、 $alloc\_bw$  および、その一部に関わる  $state$  について簡単に説明する。 $alloc\_bw$  は、このエン트리への帯域の仮想的な割り当て分を表す。割り当てる帯域は、このセッションに関してのデータの送信または中継を行う際の、単位時間当たりの自己発生トラフィック量となる。ここで、その自己発生トラフィック量は対象としているセッションの必要とする通信帯域  $Req\_bw$  と同値である。 $state$  は、対象となるセッションについて経路探索処理のどの段階であるかを表す状態情報である。各端末は、任意のセッションに対する経路探索処理の途中で中継端末の候補となった場合に、そのセッションについてのルートエン트리を作成する。ただし、経路探索処理の途中の段階では  $state$  には非アクティブな状態の値が設定される。一方、端末がそのセッションに対して実際にデータの中継端末である場合、 $state$  は“ACTIVE”という値を持つ。以降、このようなエントリを「“ACTIVE”なエン트리」と表現する。すなわち、各端末‘X’の自己発生トラフィック  $B_{self}(X)$  は、端末 X がもつ経路表内の全ての“ACTIVE”なエントリの  $alloc\_bw$  の値の総計と等しい。

#### 帯域情報の更新メカニズムの動作例

図 3.3 に、CSR 内の端末の帯域情報の収集・更新を行う本メカニズムの動作例を示す（各図にトポロジの図も添えるようにする。特に (a) では端末 E が A の CSR の外にいる図にしておく）。尚、この動作例は、図 2.11 のネットワークのシチュエーションに従う。ここで、 $t_n$  は時刻を離散的に表しており、 $t_n < t_{n+1}$  である。また、 $HELLO(X, t_n)$  は、時刻  $t_n$  に端末‘X’が送信した HELLO メッセージを意味する。まず最初の段階として、図 3.3(a) で示すように、時刻  $t_0$  において端末 B は、端末 E がまだ自身の TR 内の外に存在するため、端末 E の存在を知らないものとする。この状態においては、端末 A が端末 B から HELLO メッセージを受信したとしても、端末 B の生成する HELLO には端末 E の存在およびその帯域情報が付加されることはないため、端末 A は自身の NIT に端末 E を示すノードを登録させることはない。その後、端末 E が端末 B の TR 内に移動した場合を考える。図 3.3(b)

は，端末 E がある時刻  $t_1$  に送信した HELLO メッセージを端末 B が受信した状態である．端末 B は，端末 E および端末 H をそれぞれ自身の 1-hop の隣接端末および 2-hop の隣接端末として NIT に追加し，同時にそれらの帯域情報も登録する．端末 B は NIT を更新することで，以降，CSR 内の端末のトポロジおよびそれら端末の帯域情報に関して時間的に鮮度の高い内容に更新された HELLO を準備することが可能となる．よって，図 3.3(c) に示すように，端末 A は，時刻  $t_2$  に端末 B から送信されたは HELLO を受信することで，自身の 2-hop の隣接端末である端末 E の存在を確認し，NIT に追加する．同時に端末 A は以下のように，自身が把握する各端末の帯域情報 ( $B_{cons}(A)$  および  $B_{cons}(B)$ ) を正しく更新することができる．

$$B_{cons}(A) = B_{cons}(B) = T_{BA} + T_{AD} + T_{CA} + T_{CF} + T_{EH}$$

このように，各端末が自身の周辺の端末と HELLO メッセージを用いた情報交換を逐次行うことで，CSR 内の全ての端末の存在およびそれらの帯域情報を把握することが可能であり，これはすなわち消費帯域の高精度な予測を行えることを意味する．さらに，HELLO による帯域情報の収集メカニズムによって，セッションの確立・終了や端末の移動があったとしても，追従して NIT の更新を容易に行うことが可能であり，消費帯域の高精度な予測が実現できる．

ここで補足であるが，端末 E が端末 B の TR 内に移動した際に，端末 E が HELLO を送信するよりも先に端末 B が HELLO を送信した場合は，端末 A (およびその他の端末 B の隣接端末) はまだ端末 E の存在および帯域情報を把握できないため，端末 A 上では， $B_{cons}(A)$  および  $B_{cons}(B)$  は実際より小さい値が算出され，正確ではない帯域予測を行う時間が存在することになる．ただし，端末の分布が変わらない内に端末 E から端末 B が HELLO を受信し，その後の端末 B が送信する HELLO を端末 A が受信すれば，NIT の構成および各ノードの帯域情報は正しく修正されるため，各端末の HELLO の送信間隔が端末の移動速度に比べて充分短い場合は問題なく，本提案手法の適用する環境としては，そ

のような状況を想定する。

### 3.3.3 アドミッション制御に基づく経路の構成

本節では、本提案手法における、アドミッション制御に基づく経路の構築法について説明する。前提として、各セッションの送信元は対象となるアプリケーションから、必要とする通信帯域を含めセッションの確立要求を受け取ることが可能であるとする。

本提案手法の経路探索においては、オンデマンド型の経路探索で広く用いられる RREQ/RREP メッセージングの仕組みを改良して利用する。以下に、本提案手法における、RREQ および RREP メッセージの書式（ただし、特に重要な項目のみ表記）および使用用途、特徴について簡単に説明する。

- **Route REQuest(RREQ)**

経路の候補を探索するためのメッセージである。対象とするセッションの送信元端末から送信され、宛先に届くまでフラッディングされる。そのため、各端末には、様々な端末を経由して、複数の RREQ が届く可能性がある。RREQ は、以下のような書式のメッセージである。

$$RREQ = \{sid, seq\_num, src, dest, req\_bw, \mathbf{RL}, TTL, timestamp\}$$

ここで、**RL** は、このメッセージを受信した端末の ID を経由順にリスト形式で示したものである。RREQ を受け取った端末は、**RL** の最後尾に自身の端末 ID を登録した後、この RREQ を再度ブロードキャストする。*seq\_num* は、経路探索に関するメッセージについて（メッセージの種類とセッションの識別子 *sid* が同じ）複数のメッセージに対して到着順序を識別するために用いられる。詳細は、後述の経路探索手順の中で説明する。

- **Route REPLY(RREP)**

RREQ に対して宛先から返信されるメッセージであり，論理的にユニキャストで送信元まで転送することで，候補となる経路（すなわち中継端末群）の絞込みを行う．また，本手法においては，RREP を受信した端末は，“Final Check” プロシジャを用いたアドミッション制御を行う．RREP は，以下のような書式のメッセージである．

$$RREP = \{sid, seq\_num, src, dest, target\_id, timestamp\}$$

経路探索の処理は，送信元，宛先，およびその他の中間の端末で分散的に行われる．図 3.4 は，本手法の経路探索処理手順を表すシーケンスチャートである．以下に，本手法の経路探索手順を説明する．ここで今，確立要求のあるセッションの ID を 's1'，アプリケーションが必要とする通信帯域を *bw* とする．

**(Step 1)：送信元端末による RREQ の送信**

送信元端末は， $sid = 's1'$ ， $req\_bw = bw$  とした RREQ の作成し，RREQ をブロードキャストする．RREQ には， $seq\_num$  や，送信元および宛先の端末の ID 情報に加え，RREQ メッセージの最大ホップ可能数を表す *TTL*（初期値は 5 としている）や送信時刻である *timestamp* も設定する．また，空のリストである *RL* に自身の端末 ID を登録する．ここで送信元端末は，RREQ の送信に加え，新たなルートエントリを作成する．このルートエントリにおいては，*state* を，RREP メッセージ待ちの状態を表す 'WAIT\_RREP' に設定する．*expired\_time* には，RREP メッセージを待つ期限の時刻を設定する．この値は，現在時刻および *TTL* の値に基づき算出される．

**(Step 2)：中間の端末による RREQ のフラッディング**

RREQ を受信した中間の端末（これを端末 'm' とする）は，受信した RREQ の内容を一部変更してその RREQ を再度ブロードキャストする．送信する RREQ の変更部

分としては、*TTL* の値の減算 (-1)、*timestamp* について自身が RREQ を送信する時刻への更新、*RL* の最後尾に自身の端末 ID を追加、などである。

動作の詳細としては、まず、端末 *m* は RREQ を受信した際、自身の保持する経路表の中から、RREQ に記入されている *sid* と同じ *sid* のルートエントリを探し、該当するエントリが見つからない場合には新たなルートエントリを作成する。ルートエントリの作成は、送信元端末におけるエントリの作成と同様に行う。加えて、*pre\_hop* として RREQ を送信した端末の ID を、また *src* および *dest* として RREQ の *src* に示されている送信元の ID および宛先の ID をそれぞれ登録する。該当するエントリが見つかった場合に、*state* が 'ACTIVE' ではなく、かつ *seq\_num* の値が RREQ に記入されている *seq\_num* の値より小さい場合、このエントリ内の各種要素を上述の新規エントリの作成の場合と同様に更新する。

補足事項として、本手法では、経路探索メッセージのループの回避や必要以上の RREQ の伝播によるオーバーヘッド（帯域消費）の抑制、経路探索処理自身のループやのデッドロックの回避などの目的で、中間の端末において RREQ を廃棄する必要がある。端末 'm' が RREQ の廃棄を行う（すなわち、これ以上 RREQ を伝播させない）条件を以下に示す。

1. *TTL* の値を 1 減らした場合に 0 になる場合（後述の拡大リング探索を参照）
2. 該当するエントリが見つかった場合に、*state* が "ACTIVE" である、または RREQ に記入されている *seq\_num* の値がルートエントリに記入されている *seq\_num* の値と同じまたはそれより小さい場合

### (Step 3)：宛先による RREP の返信

宛先端末は、RREQ を受信した際、アドミッション制御である "Final Check" プロシジャを実行する。"Final Check" については、後で詳細に述べる。このプロシジャの戻り値が "acceptance" であった場合、RREQ を送信した端末宛に RREP を返信する。この際、経路表に対象のセッションに対するルートエントリがまだ存在していない

場合には、新規にエントリを追加し、*state* に“WAIT\_RACK”を設定する。一方、戻り値が“refusal”であった場合、この RREQ を廃棄する。

ここで、RREP に含む **RL** は、RREQ に含まれた **RL** の最後尾に自身の端末 ID を追加したものである。また、対象のセッションに対するルートエントリが既に存在（かつ、*state* が“WAIT\_RACK”である）している場合、宛先端末は中間の端末と異なり、RREQ の *seq\_num* の値に関わらず RREQ を受信する毎に (Step 3) を実行する。

**(Step 4)： 中間の端末によるアドミッション制御および RREP の返信**

自分宛の RREP を受信した任意の中間の端末 ‘m’ は、自身の経路表の中に *sid = ‘s1’* かつ *state = “WAIT\_RREP”* であるエントリ（‘en’ とする）が存在する場合、“Final Check” を実行する。戻り値が“acceptance”であった場合、エントリ ‘en’ の内容を更新し、このエントリの *pre\_node* に示される端末に RREP を転送する。エントリの更新の際は、*next\_node* をこの端末 ‘m’ に RREP を送信した端末の ID に、*state* を“WAIT\_RACK”にそれぞれ設定する。

**(Step 5)： 送信元によるアドミッション制御とデータの送信開始**

送信元は RREP を受信すると（無論、自身の経路表の中に *sid = ‘s1’* かつ *state = “WAIT\_RREP”* であるエントリが存在する場合である）、“Final Check” を実行する。戻り値が“acceptance”であった場合、このエントリの *next\_node* を RREP を送信した端末の ID に、*state* を ‘ACTIVE’ にそれぞれ更新し、このエントリの *next\_node* に示される端末に向けて、セッション ‘s1’ のデータパケットの送信を開始する。また、データの送信の開始と共に、“Immediate-Hello”（以降 “Imm-HELLO”）メッセージを送信する。

ここで、Imm-HELLO とは、新たな経路の構築により自身のトラフィック量が更新されたことを周辺の端末に直ちに通知するため（すなわち周辺の端末にも帯域情報の更新を促

すため)のメッセージである。Imm-HELLOには、定期的送信するHELLOが含む各種情報に加え *target* という項目を付加しており、ここには送信元端末にRREPを送信した端末のIDが登録される。Imm-HELLOを受信した端末は、通常のHELLOと同様に自身のNITの更新を行うが、*target* に示される端末の場合のみNITの更新に加え、*target* をエントリに示される *next\_node* に示される端末のIDに変更してブロードキャストする。このように、Imm-HELLOはセッションの経路の構築時に一度だけ、データの各中継端末上でデータと同様に転送される。

(Step 6): データの中継端末の正式決定および中継開始

*sid* = 's1' かつ *state* = "WAIT\_RREP" であるルートエントリをもつ端末が、セッション 's1' のデータパケットを初めて受信するか *target* が自身のIDを示している "Imm-HELLO" を受信すると、この端末は自身が実際の中継端末に決定したと判断し、このエントリの *state* を 'ACTIVE' に更新する。以降の処理は (Step 5) の後半部分と同様である。

補足であるが、既に *state* = 'ACTIVE' であるのならば、データパケットの中継端末に既になっているため、当然パケットの中継が行われるのみである。

"Final Check" processes following three checks in rotation:

図 3.5 に "Final Check" プロシジャの擬似コードを示す。

"Final Check" では、まず、RREPに付加された **RL** が示す経路を採用してセッションを確立した場合の自身の消費帯域の増加分を利用可能な残りの帯域で確保できるかどうかを判断する (5, 6 行目)。ここで、消費帯域は自身のCSR内に存在する各端末上の自己発生トラヒックの量の総計であるため、消費帯域の増加分は、**RL** と自身のNITに登録されている端末の積集合をとり (5 行目)、積集合に含まれる端末の個数とセッションが必要とする帯域 *bw* との積により算出できる (6 行目)。

また、ネットワーク内のトラヒックが過剰にならないようにするためには、消費帯域が

それぞれ異なる他の端末に対しても、アドミッション制御が適用される必要がある。本手法では、**RL**には含まれない端末が自らアドミッション制御を行わない代わりに、RREPを受信する各端末、すなわち **RL**に含まれる端末上の“Final Check”内で実現する（7行目以降）。ここで、1-hop、2-hopの隣接端末それぞれに対して、この経路を採用したセッションを確立した場合の消費帯域の増加分を算出し、増加分を利用可能な残りで確保できるかの判断を繰り返す（7- 13行目および14- 22行目）。

以上のように、自身および根、1-hop、2-hopのそれぞれの端末全てにおいて、トラヒック過剰とはならない場合にのみ、“Final Check”は“acceptance”の戻り値を返すことになる。

このように、**RL**に含まれる各端末において以上のような“Final Check”を行うことで、この経路を使用した場合に消費帯域が増加しうる全ての端末に対するアドミッション制御を、正確かつ効率的に代行することが可能である。

本手法では、セッションが必要とする通信帯域を充足すると試算できる経路候補が複数ある場合は、一番先に経路探索処理が完了した経路を採用することが基本方針である。これは、可能な限り短い待ち時間で経路を構築するためであり、さらに、経路構築処理が短い待ち時間で完了する経路（無論アドミッション制御が完了した経路である）が、セッションのデータを実際に流す際にもエンドツーエンドにおけるデータの伝送遅延が短く抑えられることが期待できるからである。

以上が本提案手法における経路探索処理の手順およびアドミッション制御である。

ここで、経路探索に用いる二次的な処理について、以下で簡単に説明する。

- 拡大リング探索

本手法では、経路探索を行う際のフラッディングのためにネットワークにかかる負荷を抑制する目的で、DSR や AODV でも用いられている拡大リング探索を適用している。これは、経路探索について初めはネットワーク全体を対象とせず、RREQの転送可能なホップ数を制限し、適切な経路が見つからなかった場合は転送範囲を

拡大して再度経路探索を行う手法である。この際、*TTL*の値を到達ホップ数の制限値として利用する（シミュレーションにおいては、*TTL*の初期値の増加回数は2回まで、すなわち再探索は最大3回までに設定している）。

- メッセージ待ち状態に対する制限時間の設定

各端末は、RREQ や RREP の送信・転送すると、次の段階で受け取る経路探索メッセージに対する待ち状態（*state* が “WAIT\_RREP” または “WAIT\_RACK”）に入り、返信メッセージを待つ時間を *expired\_time* に登録する。これは、遅延の値を一定値以内に抑える必要があるリアルタイムアプリケーションのセッションに対するアドミッション制御の一部としても機能する。その場合、エンドツーエンドにおいて許容されるデータの伝送遅延に閾値を *delay* に登録し、その値を *delay* の値に基づき *expired\_time* の値を設定する。具体的な閾値の設定がない場合には、*delay* には ‘0’ が設定され、*expired\_time* は、現在時刻および *TTL* の値に基づき算出される。

### 3.3.4 経路の動的な再構成

本手法は、3.3.2 で述べた帯域予測に基づいて、3.3.3 で述べたアドミッション制御に基づきボトルネックを発生させない経路の構築を行うことで、確立したセッションに対して必要とする通信帯域を充足することを実現することを可能とする。しかしながら、無線アドホックネットワークにおいては各端末の移動がある場合が多く、このような環境では、端末同士の分布が変わることにより各端末の消費帯域が変動するため、ボトルネックが発生する（すなわち QoS 違反）場合がある。ボトルネックが発生すると、経路上にボトルネックが存在する各セッションに対して通信帯域の充足が維持できなくなるため、本手法では、確立したセッションのために構築していた経路を再構成することでボトルネックを解消することで、それぞれのセッションが必要とする通信帯域を維持（再び充足）することが可能となる機構を導入する。ここで、経路の再構成とは、通信帯域を充足できない状

態である「古い」経路の解放，および，帯域情報の更新を前もって行った上で，新しい経路を構築することである．本手法においては，RERR および Imm-HELLO を用いて，迅速かつ信頼性のある古い経路の解放および帯域情報の更新を行う．

本節では，QoS 違反が発生したことを検知した際の，確立したセッションに対する経路の再構成方法について説明する．

各端末は，定期的な HELLO メッセージの送受信により帯域情報の更新（NIT の更新）を逐次行うため，端末の分布が変わることでトラヒックが過剰となる（ボトルネックが発生する），すなわち QoS 違反を検知することが可能である．

QoS 違反があった場合には，ボトルネックを解消しなければ，経路上にボトルネックが存在する各セッションに対して通信帯域の充足が維持できなくなる．

よって，以下の処理を各端末で分散的に行うことで，各セッションに対して必要とする通信帯域を再び充足させることを目指す．(1) ボトルネックとなる端末が中継端末である任意のセッションに対しての経路を解放する．経路の解放により，ボトルネックであった端末が発生するトラヒックを減らすことに加え，このセッションに対する他の中継端末のトラヒックも減る．このようにして，ボトルネックの解消をねらう．これによって，確立した他のセッション（すなわち今まで通信帯域を充足していたセッション）の通信帯域を再び充足させる．(2) 経路を解放するセッションに対しても可能な限り通信帯域を充足する経路を再度用意しようと試みる．これは，ボトルネックを発生させない新たな経路の探索が行われるように，送信元に経路の再探索の開始を促す RERR メッセージを伝達させることにより実現させる．

## 消費帯域の補正

本章ではこれまで，本提案手法における帯域予測に関する各アルゴリズムおよびメカニズムの説明を行った．ここで，アプリケーションのセッションが必要とする通信帯域としては，基本的には IP パケットの送信に関する通信帯域を示している．しかしながら，実

際のデータパケットの送信の際には IP パケット自身の送信に加え定常的に付加コストがあるため、以降この節ではパケットの送信に対する付加コスト分を補正する値について説明する。

まず、基本的にデータパケットの送信の際には、IP ペイロードにあたるデータパケットに加え、下位のレイヤにおいてヘッダが付加されて送信されるため、送信の際に消費される帯域は、これらを含めて考慮する必要がある。また、IEEE802.11 では、通信の際にはフレームの衝突の可能性があるチャネル競合方式（メディアアクセス方式）であるが、制御フレームや待ち時間を設定することにより、衝突を可能な限り回避するように試みる。そのため、データパケットの送信の際には、衝突の回避の仕組みにより、データフレーム自体の送信時間に加え、制御フレームの送信にかかる時間および制御のための適度な待ち時間を要する。そこで、本手法の各アルゴリズムを適用するにあたり、上述のような制御に要する時間を、データパケットの送信の際の定常的なオーバーヘッドとし、以下のように消費する帯域の補正值  $ov\_Wt$  を定義する（この補正は、ここまでに説明した本手法自身を変更するものではない）。

データパケットのサイズ（すなわち IP ペイロード）を  $DATA$ 、IP ペイロードに物理ヘッダを除く MAC ヘッダ等下位レイヤのヘッダを加えた“データフレーム”を送信する際のビットレート（チャネルのビットレート）を  $DataRate$  とする。ここで、制御にかかる時間を含め、ひとつのデータパケットの送信を完了させるために必要な最小の時間  $T_{total}$  と  $ov\_Wt$  の関係を次式のように定義する。

$$ov\_Wt = T_{total} \times \frac{DataRate}{DATA} \quad (3.1)$$

右辺の分数を左辺に移項すれば分かるように、 $ov\_Wt$  は、データパケットを送信する際に必要となるトータルの時間的なコストを算出する倍数を意味するものである。ここで、RTS/CTS ハンドシェイク使用時の IEEE 802.11 DCF の送信動作に基づけば、 $T_{total}$  は次式のように表せる（ただし、コンテンションウィンドウ（contention window: CW）に基づ

くバックオフタイムのような，静的ではない因子は無視するもととする）。

$$T_{total} = T_{data} + T_{control} + 4 \times T_{PHY} + 3 \times SIFS + DIFS \quad (3.2)$$

式中の， $T_{data}$ ， $T_{control}$ ，および $T_{PHY}$ は，それぞれデータフレームの送信時間，制御フレームの送信時間および物理ヘッダの送信時間である。

今，式( )および[35]を参考にしてまとめたIEEE802.11 DCFの諸元[35]（表3.1を参照）から $ov\_Wt$ を算出すると， $ov\_Wt \approx 1.71$ である。本章の次節におけるシミュレーションにおいて，本手法ではこの値を用いてシミュレーションを試行する。具体的には，セッションが必要とする通信帯域を $ReqBw$ とした場合，この値はデータパケットの通信のみに対して必要な通信帯域であるため，本提案手法では，経路発見処理の最初にこのセッションの送信元で $ReqBw' = ReqBw \times ov\_Wt$ を行い，以降のこのセッションに関する経路構築・再構築処理および帯域情報の更新時には $ReqBw'$ を用いる。

補足であるが，表3.1においてはData Rateの項目が，式(3.1)における $DataRate$ に対応する。

### 3.4 性能評価

本提案手法であるALRSの性能評価を行うため，計算機シミュレーションを行った。

まず，確立されたセッションに対してセッションが必要とする通信帯域を充足できるような精度の高いアドミッション制御が行われるかを確認するために，端末の移動がないシナリオを用いたシミュレーションを行う。特に詳細な解析として，最短経路に基づく無線アドホックネットワークの標準的な経路制御手法（AODV[16]）の場合には通常ボトルネックが発生するシナリオを用い，その場合でも本提案手法が精度の高い帯域予測によりセッションに必要な通信帯域を充足するかどうかの解析を行う。

また，通信範囲の分布の変動に起因してボトルネックが発生した場合にも，分布の変化に追従して精度の高い帯域予測を行いボトルネックを検知し，セッションが必要とする通

信帯域を充足する経路の再構築を適切に行えるかを検証するため，端末が移動するシナリオを用いたシミュレーションを行う．

ここで，各計算機シミュレーションには，ネットワークシミュレーションを行う際に広く用いらてれる，NS-2シミュレータ [36] を用い，全てのシミュレーションを通じて本提案手法である ALRS と，標準的な経路制御手法である AODV についてそれぞれ比較を行った．

ここで，AODV は，本手法と同様のオンデマンドな経路探索を行う．ただし，AODV は構築する経路の目標は送信元から宛先までの最短経路を構築することであり，帯域の充足可否については経路の構築の基準として含まれていない．しかしながら，最短路のみを経路の構築目標として扱う経路制御を行う手法の中では，AODV は経路制御のオーバーヘッドが少なく，効率的で性能が高い手法である．

### 3.4.1 シミュレーションのパラメータ

本シミュレーションにおける設定を述べる．端末としては IEEE 802.11 DCF 準拠の無線デバイスを持つ移動端末を想定している．表 3.2 に，全ての端末に共通の設定項目を示す．ここで，random way-point とは，端末や人間の移動モデルのひとつのとして広く適用されており，各端末は以下の手順に基づいて移動する [37]．(1) pause time で設定した時間だけ現在の位置で静止する．(2) 設定時間の経過後，新しくランダムに次にの目標の移動先の位置を決定し，その位置に向けて移動を開始する．(3) 移動完了後，再び (1) から繰り返す．移動速度のパターン (type of speed) にはいくつかの種類があり，目標の位置の設定毎に一様分布の確率に基づき min と Max の間で速度を選択し，目標位置まで速度を一定に保って移動するパターン (uniform) や，シミュレーション中は pause time 以外は常に一定の速度 (constant) である (すなわち min と Max が同じ) パターンなどがある．本シミュレーションでは type of speed は “constant” としている．

また，各シナリオに対して，ネットワークのサイズは  $1500\text{m} \times 1500\text{m}$ ，ノード数は 50 ノードに設定する．各セッションに対する送信元と宛先は，50 ノードからランダムに選ばれる．各セッションのデータとしては，一定の通信帯域を必要とするデータをストリームとして送出するアプリケーションを想定し，Constant Bit-Rate (CBR) trafficを使用する．CBR trafficは，一定の時間間隔でパケットを定常的に送出する方式である．ここで，CBR trafficのパケットサイズは 512 bytes に設定する．CBR trafficの送出ビットレート (CBR-rate と表記する) は，全ての送信元で同一なものとする．また， $x$  Kbps の CBR-rate を “ $CBR_x$ ” とし，ひとつのシナリオにおいて，確立要求のあるセッション数に CBR-rate の値を掛けたものを “Load” と表す．

表 3.3 に，シミュレーションで使用する，nodes' speed，CBR-rate，および Load のそれぞれのパターンを示す．ここで，nodes' speed，CBR-rate，および Load の 3 つの組み合わせひとつに対して，端末の初期位置および各セッションの送信元と宛先の組み合わせがそれぞれ異なる 30 個のシナリオを用意し，それぞれ試行する．

以上のような条件の下，各シナリオについて，シミュレーションを 500 秒 (シミュレーション上のクロックで換算) 間試行した．

### 3.4.2 性能評価に関する測定項目

本提案手法の性能評価に際し，シミュレーション時には以下の 4 項目について測定を行う．それぞれの項目については，各セッションの送信元と宛先の間，すなわちエンドツーエンドで測定する．

#### 1. トラフィックアドミッション率 (Traffic admission ratio ( $Ad$ ))

全ての送信元で生成使用と試みるデータパケットの総数に対する，送信元から実際にネットワーク上に送信されたデータパケットの総数の割合である．各端末が移動しない場合は，確立要求があるセッションの数に対する，実際にネットワーク内で

確立したセッションの数，すなわち構築した経路の割合を表すことにもなる．ただし，送信した各データパケットは宛先に到達するとは限らないため， $Ad$  から確認できるのは，経路を構築し，送信元からパケットが送出されたかどうかのみである．

## 2. パケットの配信率 ( End-to-end packet delivery ratio (*Deli*) )

送信元から送信されたデータパケットの総数に対する，宛先で受信されたデータパケットの総数の割合である．高い配信率であるということが，セッションに必要な通信帯域が充足していることを表しており，経路を構築する際に信頼性の高いアドミッション制御が行われたこと，すなわちアドミッション制御の精度が高いことを表す．さらに端末が移動するシナリオの場合には，最初の経路構築の際に高精度なアドミッション制御が行われたことを表すだけでなく，端末の移動に起因する QoS 違反を精度良く検知し適切に経路の再構築が行われたことを意味する．

## 3. スループット ( Throughput (*Th*) )

各宛先で受信された単位時間当たりのデータ量，すなわち単位時間当たりに受信されたデータパケットの数にデータパケットのサイズ ( 本シミュレーションにおいては 512 bytes に固定 ) を掛けたものであり，帯域の利用効率を表す．スループットには，経路制御のための各種制御メッセージ ( HELLO , RREQ , RREP ) は含めず，純粋にデータパケットのみの帯域の利用効率を表している．

## 4. 平均遅延 ( Average end to end delay (*Delay*) )

宛先で受信された各データパケットに関して，パケットひとつ当たりの伝送遅延は，宛先にそのパケットが到着した時間から送信元でそのパケットが生成された時間を差し引いた値である．セッション単位の平均遅延とは，対象のセッションに関する全てのデータパケットに関しての伝送遅延の平均値を表す．また，シナリオ単位の平均遅延とは，ネットワーク内の全ての宛先で受信されたデータパケット全てに関

しての伝送遅延の平均値を表す。

### 3.4.3 端末の移動がない場合の性能

まず、各端末が移動しない場合についての本提案手法の性能を検証する。図 3.6 に、CBR トラヒックとして  $CBR_{40}$  を用いた場合の実験結果を、また図 3.7 に、 $CBR_{200}$  を用いた場合の実験結果をそれぞれ示す。ここで、グラフ上の各点は、30 個のシナリオに対する平均値である。

Load が 200Kbps の場合、図 3.6(a)~ (d) および図 3.7(a)~ (d) から分かるように、本提案手法である ALRS を用いた場合および既存手法の AODV を用いた場合の両方とも、 $Ad$  は 0.95 付近に達し、 $Deli$  の値はほぼ 1 になる。よって、 $Th$  が約 200Kbps、すなわち Load とほぼ同値になることは必然である。また、 $Delay$  も極めて小さい値である。以上の結果から、Load が小さい場合は、AODV を用いてもネットワーク内にボトルネックが発生するほど過剰にトラヒックが発生することはないことが分かる。そのため、仮にアドミッション制御によって必要以上に  $Ad$  が抑えられてしまうようでは非効率であるといえるが、本提案手法は、 $Ad$  が AODV に比べても低くはないことから、トラヒックが少ない場合にも消費帯域を必要以上に大きく見積もらず、精度の高いアドミッション制御が行われることが確認できる。

次に、Load が 400, 600, 800 Kbps の場合の結果を確認すると、AODV を用いた場合、Load が増加するにつれて  $Th$  は増加していく。ただし、Load の増加に伴い、 $Ad$  は 0.9 以上を維持している一方、 $Deli$  は大幅に減少し、 $Delay$  は大幅に増加する。これはすなわち、AODV のように、送信元から宛先までがマルチホップで接続できるのであれば経路を構築する方針の場合は、Load が増加するにつれて、ネットワーク内にボトルネックが発生することが分かる。Load がさらに増加した場合には、 $Deli$  の減少量および  $Delay$  の増加量が小さいことが確認できる。

一方，ALRS を用いた場合は，全ての Load において  $Del_i$  は 0.7 以上を達成している．さらに，Load が増加しても，AODV と比較して  $Delay$  の増加は小さく，Load が 800Kbps 以上の場合には，AODV を用いた場合の 25% 以下に抑えられている．また， $Ad$  は減少しながらも，AODV を用いた場合とほぼ同様の  $Th$  の値を実現している．

このように，本提案手法を用いた場合，確立したセッションに対し，高いパケット配信率を実現し，また，パケットを小さな遅延で宛先へ到達させることができる．さらに，送出するパケットの量を制限しているにも関わらず，スループットについては AODV と同様の性能を維持することが可能である．従って，HELLO，RREQ，RREP のような，経路制御に関わる制御メッセージに対するオーバーヘッドが，本提案手法の性能に悪影響を及ぼさないことが確認できる．ここで，図 3.6(c)，(d)，および図 3.7(c)，(d) をみると分かるように，本手法を用いた場合，Load が増加するに従い， $Delay$  は減少し  $Del_i$  は増加する．この要因とは，以下のように分析できる．

送信元から宛先までの距離が遠いことで経路長が必然的に長くなるようなセッションに対しては，Load が増加すると，必要とする通信帯域が充足できない場合が多いため，本手法を用いた場合は，Load が増加すると，短い経路長の経路を構築可能なセッションを確立する状況が増える．ネットワーク内に発生するトラヒックの量にも左右されるが，基本的には経路長が短い場合はエンドツーエンドの伝送遅延が小さいため，上述のような傾向を示したといえる．

実験結果に対する全体的な解析結果は以上である．ここからは実験結果に対して，詳細な解析を行う．

#### ボトルネックの発生しやすさの違いに対する性能

図 3.8 は，CBR-rate を 40Kbps，Load を 1200Kbps とした場合の 2 つのシナリオの実験について，各セッションにおける， $Th$  と  $Delay$  の関係を表す散布図を示す．また，図 3.9(a) および (b) は，達成したパケット到達率毎のセッションの数を表す累積度数分布である．

ここで，図 3.9(a) は，図 3.8 における，比較的ボトルネックが少ないシナリオの実験結果に対応しており，図 3.9(b) は図 3.8 における，ボトルネックが各所で起こりやすいシナリオに対応している．尚，確立要求のあるセッションの数はどちらのシナリオも当然 30 セッションであり，確立しなかったセッションについては，パケットの到達率は 0 として取り扱っている．

まず，比較的ボトルネックが少ないシナリオの場合について検証する．まず，図 3.8 に示すように，各セッションについての  $Th$  と  $Delay$  については，本手法を用いた場合と AODV を用いた場合には差は無い．ここで， $Th$  が比較的小さいセッションは  $Delay$  も比例して大きいことが分かる．このことから，これらのセッションについては，多少のボトルネックが生じるような経路を構築しており，ボトルネックがある部分をデータが経由することになり，データが宛先に届かなくなった，または遅延が大きくなったものと考えられる．しかしながら，全てのセッションの総数に比べるとその割合は 1/10 程度である．また，図 3.9 (a) に示すように，本提案手法および AODV とともに確立要求のある全てのセッションに対して多くのセッションのパケットの配信率が高いことが確認できる．元々 AODV のようにリンクの有無のみにより経路を構成する場合に比較的ボトルネックが少ないシナリオであるため，AODV を用いた場合に多くのセッションのパケットの配信率が高いことは当然といえる．このような状況においても，本提案手法は AODV と同等の性能を発揮することから，トラヒックを必要以上に抑制せず，また，経路制御におけるオーバーヘッドも AODV と差がない．

次に，ボトルネックが各所で起こりやすいシナリオの場合について検証する．まず，AODV を用いた場合，図 3.8 に示すように，比較的ボトルネックが少ないシナリオに比べて， $Th$  が低く  $Delay$  が大きいセッションの割合が高くなる．一方，本提案手法を用いた場合，確立したセッションについては，2 つのセッションを除いて残りは全て  $Th$  が上限近くまで達し， $Delay$  も AODV と比べて非常に小さく抑えられている．また，図 3.9(b) に

示すように、確立したセッションに対して、*Deli* が低いセッションはひとつしかなく、残りの全てのセッションが 0.9 以上の *Deli* を達成している。さらに、AODV に比べて *Deli* が高いセッションの数も AODV に比べて約 2 倍である。

以上の実験結果より、本提案手法は、ボトルネックが起これないように、且つ、必要以上にトラヒックの発生を抑制しないように、確立可能なセッションの判断と適切な経路の選択を同時に行うアドミッション制御を、高い精度で行えることが分かる。これは、本手法が消費帯域を高精度に予測することができるからであり、以上のことから、本提案手法は、ネットワーク上に発生しうるトラヒック量に柔軟に対応し、セッションが必要とする通信帯域を充足可能である経路制御方であることが確認できる。

#### 3.4.4 端末の移動がある場合の性能

次に、端末の移動がある場合には、本提案手法の性能にどのように影響を与えるかについて検証する。3.4.3 節でも参照した図 3.6 および図 3.7 には、端末の移動がある場合の実験結果も示している。

移動速度が速くなるに従い、ネットワークのトポロジの変化が激しくなるため、必然的に、本提案手法を用いた場合でも AODV を用いた場合でも  $Th$  が低下している。ここで、本提案手法を用いた場合でも、AODV を用いた場合の  $Th$  とほとんど変わらない。

しかしながら、移動速度が速い場合には、AODV を用いた場合は大きく *Deli* が低下し、また Load の増加につれて、端末が移動しない場合と同様、*Deli* は減少する。一方、本提案手法を用いた場合は、移動速度が速くなるに従い、*Deli* を多少低下させるものの、AODV と比較して常に高く、端末が移動しない場合に比べて本提案手法と AODV との *Deli* の差は広がる。また、本提案手法を用いた場合は、移動速度が速くなったとしても、*Delay* の増加はみられない。

このように、本提案手法は、端末の移動があることで極端に不正確なアドミッション制

御になることはなく，スループットについても AODV を用いた場合と同程度を維持することが分かる．

特に，端末が移動する場合でも歩行速度程度の移動性であれば，セッションを確立すれば，そのセッションが必要とする通信帯域の 7 割のデータは配信可能であることが確認できる．

### 3.4.5 考察

既存研究である AQOR および CEDAR は，TR と CSR が等しいとした無線アドホックネットワークのモデルに基づき，アドミッション制御の信頼性を向上させることを目的とした経路制御法である．しかしながら，Muqattash らをはじめとした様々な文献において，シングルホップの無線ネットワークにおいて，TR の外側に存在する CSR についての影響を論じている [38]．Muqattash らは，シングルホップの環境のモデルを用いた解析を行っていたが，CSR の概念はマルチホップ通信も行われる無線アドホックネットワークにおいても同様に考慮すべきであることが分かる．したがって AQOR および CEDAR を適用した場合，帯域の予測の精度が低くなり（消費帯域を低く見積もる），アドミッション制御上ではセッションの必要とする通信帯域を確保しても，実際に構成した経路上でデータを配信した際，その通信帯域（すなわち通信量）を充足できる保証がない．一方，本提案手法では，CSR 内のトラフィック量に基づき消費帯域を予測することで，高精度なアドミッション制御を実現する．すなわち，アドミッション制御において通信帯域が確保できることが確認されれば，実際に構成した経路上でデータを配信した際，その通信帯域（すなわち通信量）を充足できる．

さらに本手法の性能を向上させるためには，以下のことが考えられる．

本提案手法の場合は，消費帯域の予測を行いアドミッション制御を行うため，端末の移動がある場合に性能が低下することについては，以下の要因が大きい．端末の移動する速

度が速くなるほど、各端末間で交換している隣接端末の存在の情報や帯域情報を各端末で交換している間にも、動的にネットワークのトポロジが逐次変化し、自身が把握する端末情報が実際の各端末の消費帯域とにずれが生じる。その結果、本提案手法では、アドミッション制御により、確立可能であると判断したセッションについても、必要とする通信帯域を充足しない頻度が高くなる。従って、帯域情報の交換メッセージのオーバーヘッドが大きくなる範囲で帯域情報交換メッセージの送信間隔を短くすることで、さらに速い移動速度の場合であっても通信帯域の充足が可能となると考えられる。

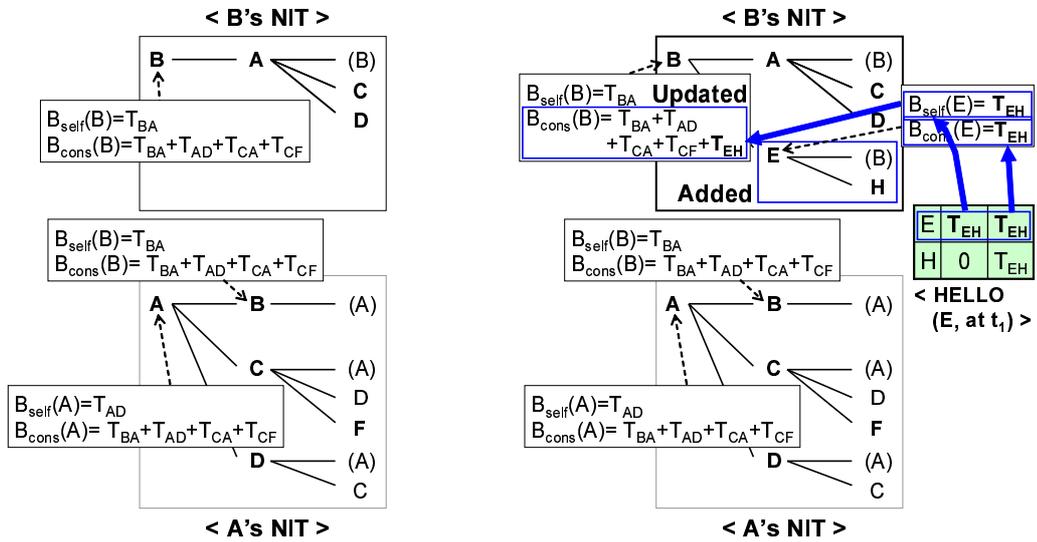
### 3.5 要約

本章では、無線ネットワークネットワークレベルのサービス品質を向上する上で、従来の経路制御法の問題である、(T1) 無線アドホックネットワークの経路制御における帯域予測の精度が低いことにより、通信帯域の充足が困難であったことを解決するために、無線端末間で互いの干渉範囲にあるトラヒック量を把握することで消費帯域を高精度に予測しアドミッション制御を行う経路制御法を提案した。また、計算機シミュレーションによる評価実験の結果、従来の最短経路探索を基にした経路制御手法 [16] に比べてスループットの低下させることなくパケットの到達率を最大 50% 向上し、更にエンドツーエンドの到達遅延を 1/4 以下に抑制することを確認し、セッションに必要な通信帯域の充足が達成可能な手法であることを確認した。

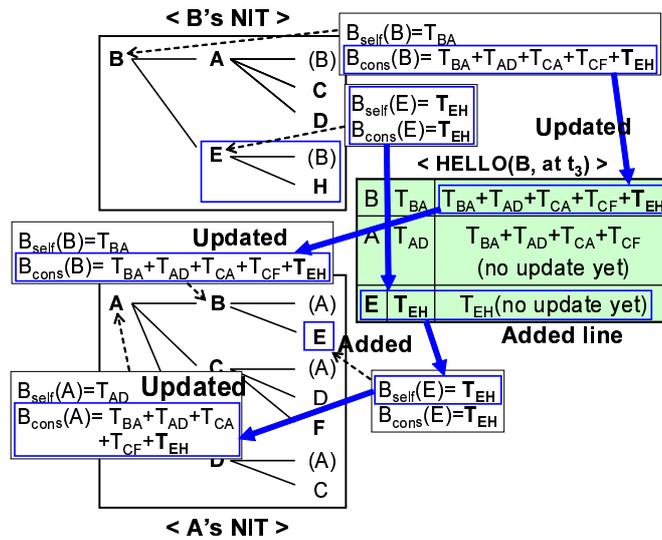
ネットワーク上に発生しうるトラヒック量に柔軟に対応し、セッションに必要な通信帯域の充足することが可能であり、また、端末が移動する場合でも本手法が十分機能することを確認した。

また、議論を通じて、従来のアドミッション制御に基づく経路制御法と比較して本手法が高精度な帯域予測が可能であり、これにより精度の高いアドミッションが可能となるため、通信帯域の充足が困難であるという従来手法の問題の解決が可能であることを確認し

た．以上のことから，本手法を適用することで，独自の経路制御が必要となる無線ネットワークである無線アドホックネットワークにおいてネットワークレベルのサービス品質を向上することが可能であるという結論を得た．



(a) 端末 B が端末 E からの HELLO メッセージを受け取っていない段階 (b) 端末 B が時刻  $t_2$  に受信した  $HELLO(E, t_1)$  に基づき自身の NIT を更新した状態



(c) 端末 A が時刻  $t_4$  に受信した  $HELLO(B, t_3)$  に基づき自身の NIT を更新した状態

図 3.3: 帯域情報の交換メカニズムの動作例

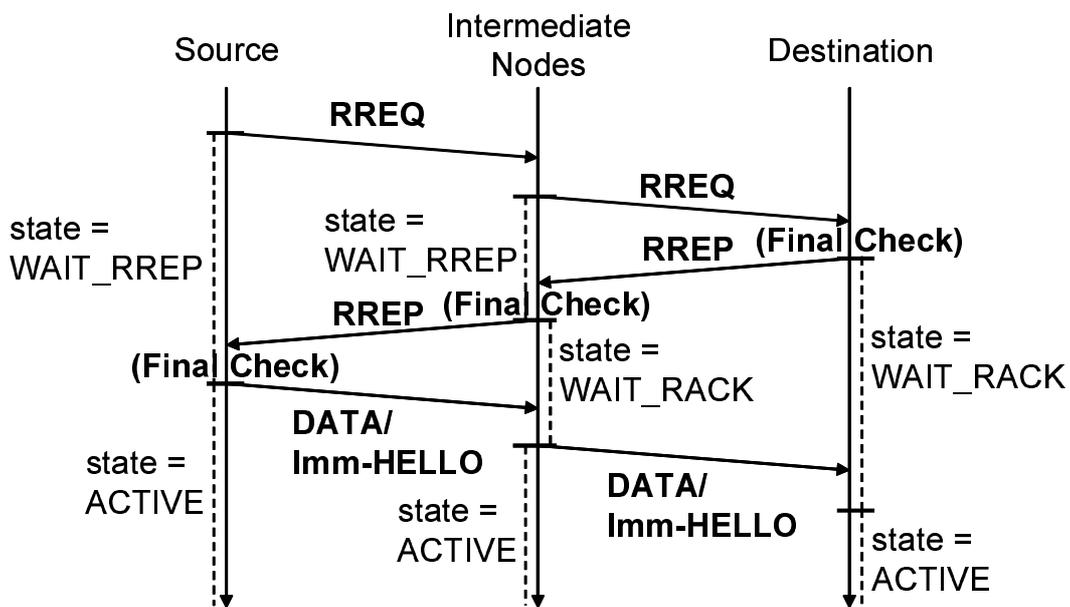


図 3.4: 経路探索のメッセージのシーケンス

```

Procedure FinalCheck(RL, NIT, SNIT(d), req) {
  /* RL := {x | x is in route list except Destination} */
  /* SNIT(d) := {x | x ∈ NIT, x is node at a depth of d} */
  /* req := requested bandwidth for the session */
  n := number of nodes in RL ∩ NIT;
  if ( $B_{max} - B_{cons}(root) < n \times req$ ) return ("refusal"); /* check for self-node */
  copy SNIT(1) to CI;
  while (CI ≠ ∅) {
    en := an extracted node from CI;
    n := number of nodes in RL ∩ NIT;
    if ( $B_{max} - B_{cons}(en) < n \times req$ ) return ("refusal"); /* check for 1-hops */
    else remove en from CI;
  }
  copy SNIT(2) to C2;
  while (C2 ≠ ∅) {
    en := an extracted node from C2;
    F := {x | x ∈ SNIT(1), x has en as its child node};
    E := {x | x ∈ SNIT(2), x is child of y (for y ∈ F)};
    n := number of nodes in RL ∩ E ∪ SNIT(0);
    if ( $B_{max} - B_{cons}(en) < n \times req$ ) return ("refusal"); /* check for 2-hops */
    else remove en from C2;
  }
  return ("acceptance");
}

```

図 3.5: Final Check アルゴリズムの擬似コード

表 3.1: IEEE802.11 DCF (DSSS) の諸元表.

|                 |                      |
|-----------------|----------------------|
| MAC header(*1)  | 28bytes              |
| PHY header(*2)  | 24bytes@1Mbps        |
| RTS/CTS/ACK(*1) | 20/14/14 bytes@1Mbps |
| Basic Rate(*3)  | 1Mbps                |
| Data Rate       | 2Mbps                |
| Slot Time       | 20microsec           |
| SIFS/DIFS       | 10/50microsec        |
| CWmin/CWmax     | 31Slot/1023Slot      |

(\*1) FCS (4 bytes) is also included

(\*2) Total of PLCP preamble and PLCP header.

(\*3) Basic Rate is used to send control frame such as RTS/CTS/ACK and PHY header.

表 3.2: シミュレーションにおける各端末共通の設定

|                           |  |
|---------------------------|--|
| radius of TR              | 250 m                                    |
| radius of CSR             | 550 m                                    |
| RTS Threshold             | 0  |
| channel Data Rate         | 2Mbps                                    |
| initial position          | randomly (based on uniform distribution) |
| mobility pattern          | random way-point                         |
| pause time, type of speed | 0, "constant"                            |

表 3.3: speed , CBR-rate , および Load のパターン

|              |   |
|--------------|---|
| nodes' speed | 0, 2, 5, 10 m/sec                         |
| CBR-rate     | 40, 200 Kbps                              |
| Load         | 200, 400, 600, 800, 1200, 1600, 2000 Kbps |

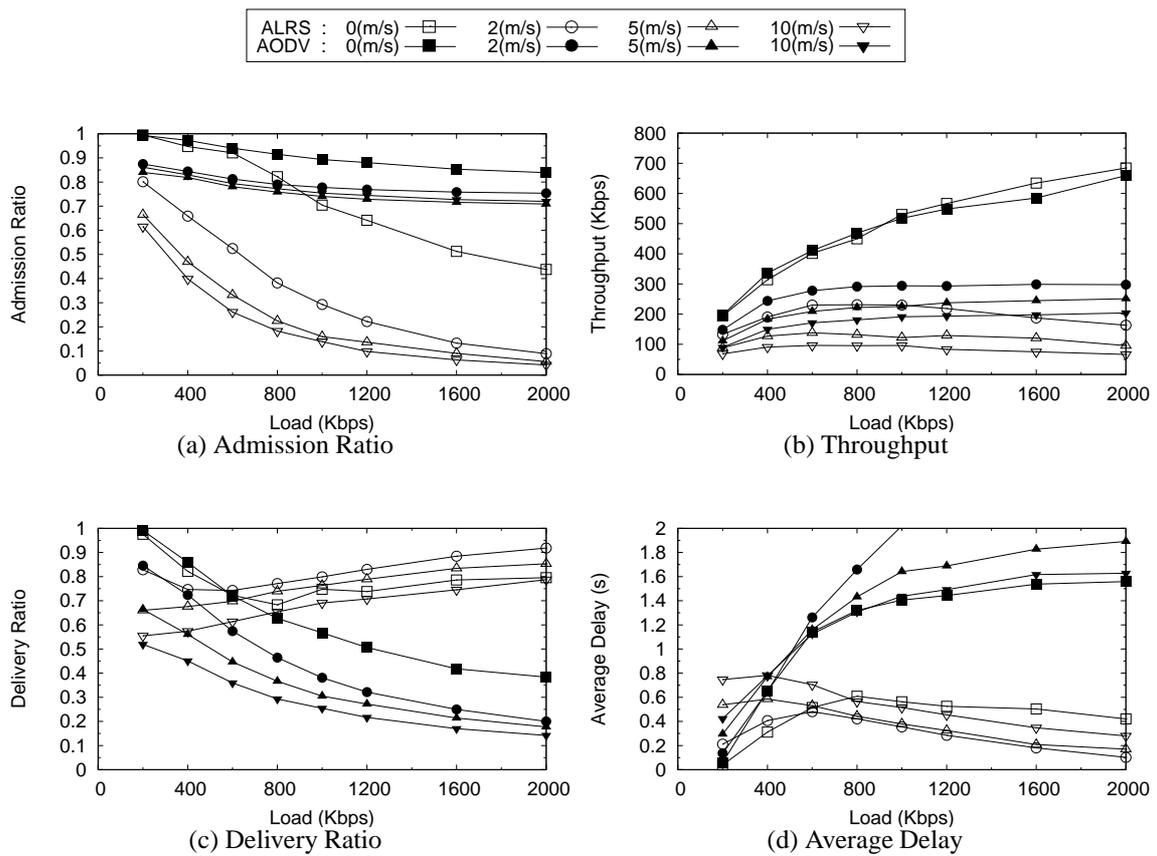


図 3.6:  $CBR_{40}$  の場合の実験結果 (30 シナリオの平均)

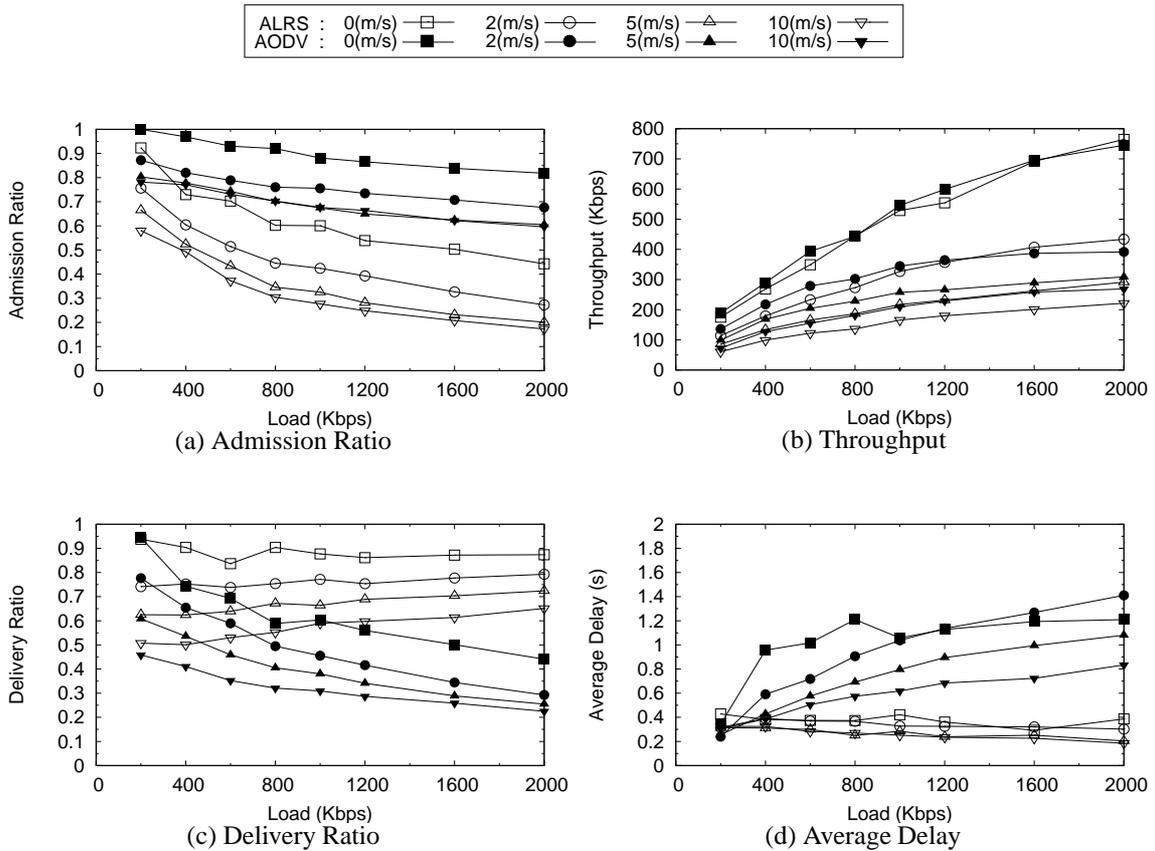


図 3.7:  $CBR_{200}$  の場合の実験結果 (30 シナリオの平均)

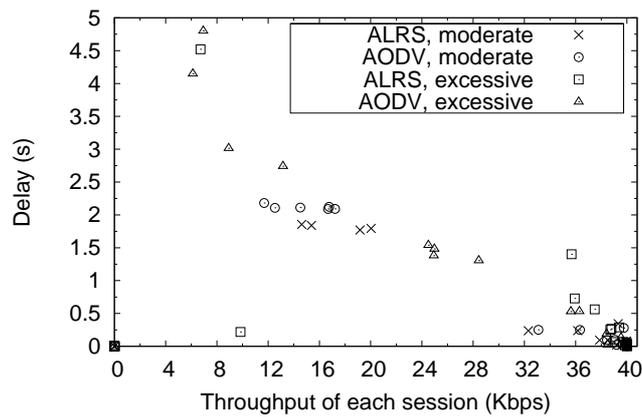
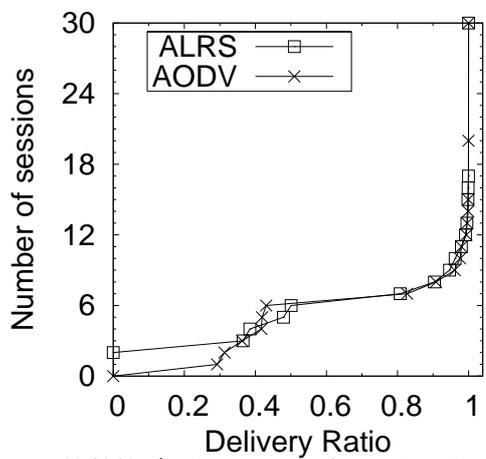
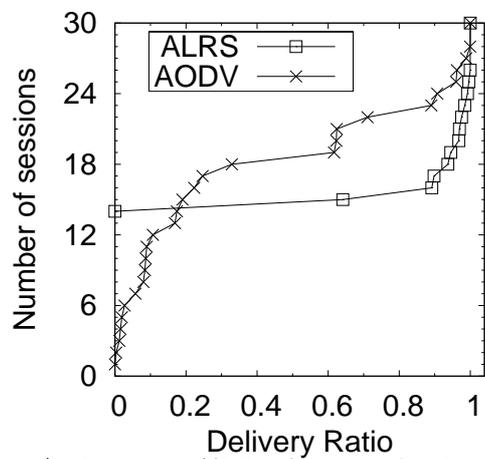


図 3.8: 各セッションにおけるスループットと遅延の散布図 (端末が移動無し,  $CBR_{40}$ , Load = 1200Kbps の場合の, ボトルネックが各所で起こりやすいシナリオ (excessive) と比較的ボトルネックの少ないシナリオ (moderate) )



(a) 比較的ボトルネックの少ないシナリオ



(b) ボトルネックが各所で起こりやすいシナリオ

図 3.9: パケットの配信率に対するセッション数の累積度数分布

# 第4章 部分木の帯域予測に基づくエンドシステムマルチキャストの配信経路の制御法

## 4.1 はじめに

ユビキタス情報社会においては、マルチメディア通信サービスが利用者同士のコミュニケーションに非常に役立つことが期待されている。そのためには、様々な情報通信基盤において、マルチメディア通信サービスの品質に大きく関わるネットワークレベルのサービス品質を向上することが重要となる。特にエンドシステムマルチキャストは、有線ネットワークのP2P通信技術を有効に利用したマルチキャスト技術であり、有線ネットワークにおける利用者同士のコミュニケーションに有効な手段であることから、エンドシステムマルチキャストにおけるネットワークレベルのサービス品質を向上させることは重要な取り組み課題である。

しかしながら、従来のエンドシステムマルチキャストの配信経路の制御法では、(T2)エンドシステムマルチキャストの配信経路の制御における帯域予測が困難なことによる、通信品質の偏りの発生（公平性のある多対多通信の実現が困難）という問題があった。

この問題に対し、本章では、エンドシステムマルチキャストにおいて発生する利用者端末ごとの通信品質の偏りを段階的に改善する、反復型のエンドシステムマルチキャストのデータ配信木の構成手法を提案する。具体的には本手法は、各参加者の利用者端末を根とする複数の配信木に対するネットワーク帯域の割り当てを考慮し、各利用者端末が自律分散的かつ局所的に配信木の再構成を反復して行うことで、全端末中で下限となる受信ピッ

トレートの値を積極的に上昇する．これにより利用者端末間の受信ビットレートの偏りを軽減することが可能となる．

本手法を実装し計算機シミュレーションによる評価実験を行った．その結果本手法は，既存手法と比べ受信ビットレートの最小値を 1.31 倍に向上することを確認した．また，各利用者端末の受信ビットレートの標準偏差を既存手法の 1/4 以下に抑えられることを確認した．以上から本手法は，利用者端末間の受信ビットレートの偏りを軽減し，参加者全体の通信品質を向上していることを示した．

ここで，エンドシステムマルチキャストは，アプリケーション層マルチキャスト (Application Layer Multicast : ALM) とも呼ばれており，本章では，以後エンドシステムマルチキャストのデータ配信木のことを ALM ツリーと略記する．

本章の構成は以下の通りである．4.2 では関連する既存研究について述べる．4.3 では再構成判断アルゴリズムを中心に本提案手法の詳細について述べる．4.4 では既存手法との比較に基づく本提案手法の評価を行う．最後に 4.5 で本章のまとめと今後の課題について述べる．

## 4.2 既存のエンドシステムマルチキャストのデータ配信木(ツリー)の構成法の問題点

本研究で対象とするビデオ会議等のマルチメディア通信サービスにおいては，各利用者になるべく公平に，かつなるべく高品質なサービスを提供することが望まれる．従って，異なったアクセスリンクの通信帯域の制限の中で，各受信端末のマルチメディアデータの受信ビットレート向上を実現するために，マルチメディアデータの送受信に必要な帯域を効率的に分配できるツリーの構成が問題となる．

Overcast[13] では，各端末間の仮想リンクのうち，最大の利用可能帯域を持つものから順に，閉路を作らないように選び，ツリーを作成する．Minら [11] は，各利用者端末が，

次にデータを転送する各端末へ自身の最大利用可能帯域を均等に分配し，端末の受信ビットレートの平均値が最大となるようなツリーを構築する分散アルゴリズムを提案している．

これらの手法ではいずれも，一つのデータ送信元に対し，一つの最適なツリーを算出するアルゴリズムを提供しているが，これを用いて多対多通信のために複数のツリーを作成する場合に問題が生ずる可能性がある．例えば，Minらの手法[11]では，自身のアクセスリンク帯域が大きく，他の利用者端末間のデータ転送に帯域を提供可能な端末がツリーの上位に配置され，一つのデータを多くの端末に転送する役割を担うこととなる．この手法に従って複数のツリー上でこの端末を上位に配置した結果，その端末のアクセスリンク帯域が圧迫され，その端末からデータを受信する端末の受信レートが他に比べ極端に低下してしまう．

以上のように，既存手法を用いて多対多マルチメディア通信のための複数のツリーを作成した場合，利用者端末が受信するマルチメディアデータの通信品質に偏りが生ずるといふ問題が発生する．

## 4.3 反復型ツリー構成法の提案

### 4.3.1 反復型ツリー構成法の提案

前述の問題を解決するため，本研究では，利用者端末が受信するマルチメディアデータの通信品質の偏りを段階的に軽減する，反復型 ALM ツリー構成手法を提案する．本手法は，多対多通信のために形成される全ての ALM ツリーを対象に，利用者端末が自律分散的に自身の受信レートを改善するための局所的なツリーの再構成を，反復して行う．これにより，複数の送信者端末を根とするツリーが共存する場合においても，全ての ALM ツリーにおいて下限となる受信ビットレートの値を上昇し，利用者端末間の偏りを軽減し，公平性を向上することができる．

以下，本研究で対象とするネットワークモデル，および本手法の達成目標について述

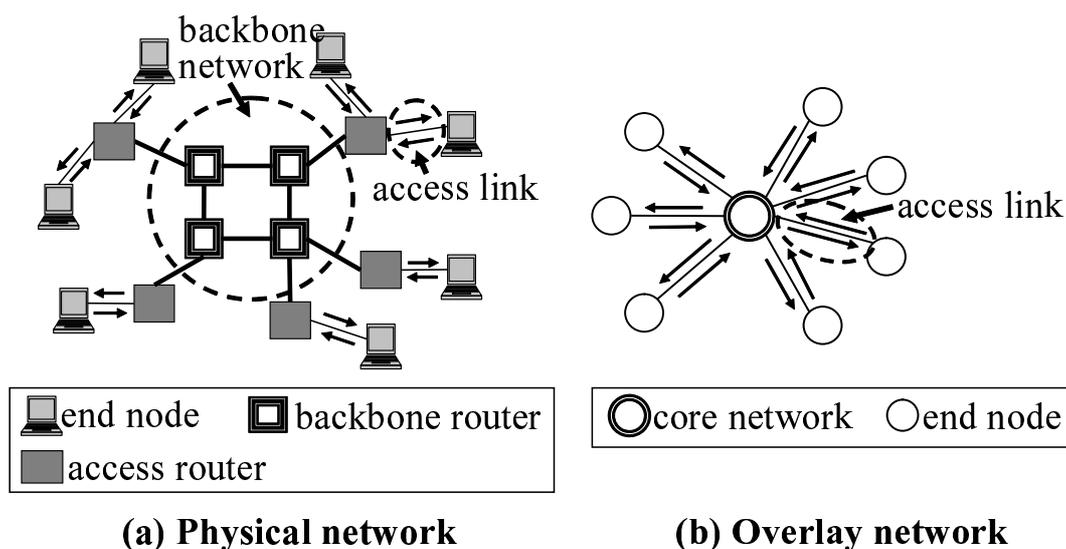


図 4.1: 本研究におけるネットワークのモデル

べ，その後，提案手法の中核となる ALM ツリー構成アルゴリズムについて詳述する．

### 4.3.2 ネットワークモデルおよび問題設定

#### ネットワークモデル

本研究では，図 4.1(a) に示すように，各利用者端末がそれぞれ上り帯域（送信帯域）と下り帯域（受信帯域）の異なる非対称なアクセスリンクを通じてバックボーンネットワークに接続する，一般的な物理ネットワークを対象とする．ここで，一般に，バックボーンネットワークの帯域はアクセスリンクに比べて十分に大いため，ALM におけるボトルネックは，アクセスネットワークにおいて生じると仮定できる [39]．よって本研究で対象とするネットワークは，図 4.1(b) に示すように，バックボーンネットワークを広帯域・低遅延のコアネットワークとして抽象化し，各利用者端末すなわち，エンドノード（ノード）が，コアネットワークとそれぞれのアクセスリンクを通じて他の全てのノードと接続されているオーバーレイネットワークとして，モデル化する．

また本研究では，ノード数が数十ノード程度までの比較的小規模なオーバーレイネット

ワークにおいて、複数のノードが送信元ノードとなる双方向ビデオ会議のような、多対多マルチメディア通信を対象とする。本研究では、これ以降、マルチメディアデータを単にデータと表す。また、あるデータをノードが受信するビットレートを受信レートと表し、同様に、送信するビットレートを送信レートと表す。

さらに各ノードは、データを中継する際、トランスコーディングにより自身が受信したビットレートよりも低いビットレートで次のノードへ中継できることを前提とする。

ここで、オーバーレイネットワーク上のノードの集合およびデータの送信元ノード、すなわちツリーのルート（根）の集合を、それぞれ  $N, S$  とする。また、送信元ノード  $s$  をルートとしたツリーを  $T_s$  と表す。以降、特に明記しない限り、 $i, j \in N$  かつ  $s \in S$  とする。表 4.1 に、本研究で用いる変数の定義を示す。なお、各ノードが中継を行う際に、受信レートより高いビットレートにトランスコーディングすることは無く、また、アクセスリンクの帯域には上限があるため、構築・再構成したツリーは、式 (4.1) および式 (4.2) の条件を満たす必要がある。

$$L_i^{\text{in}} \geq \sum_{s,j} {}_s B_{j,i}, \quad L_i^{\text{out}} \geq \sum_{s,j} {}_s B_{i,j} \quad \forall i \quad (4.1)$$

$${}_s B_{r_{n-1}, r_n} \geq {}_s B_{r_n, r_{n+1}} \quad \forall s, i \quad (4.2)$$

ここで、式 (4.1) 中および式 (4.2) 中の  $L_i^{\text{out}}, L_i^{\text{in}}$ 、および  ${}_s B_{i,j}$  の定義は、表 4.1 に示すとおりである。また、 $r_0, r_1, r_2, \dots, r_l$  は、ある送信元ノード  $s = r_0$  から任意の受信ノード  $i = r_l$  までのデータの配信経路である。

一般に、ビデオ会議等において、利用者同士が違和感なく会話できるようにするためには、200 ms 以内でデータを相手に到達させる必要があるとされている [40]。

従って、各ノードへの到達遅延の許容値（以降、許容遅延値）を  $D_{\text{max}} = 200(\text{ms})$  と定

表 4.1: 変数定義

| 変数                 | 定義 (および略称)  |
|--------------------|---|
| $L_i^{\text{out}}$ | $i$ のアクセスリンクの上り帯域   |
| $L_i^{\text{in}}$  | $i$ のアクセスリンクの下り帯域   |
| ${}_s D_{i,j}$     | $T_s$ において, データが $i$ からツリー上を經由して $j$ に到達するまでの時間 ( $i$ から $j$ までの総到達遅延)    |
| $d_{i,j}$          | データが $i$ からオーバーレイ上の他のノードを經由しないで直接 $j$ に到達するまでの時間 ( $i$ と $j$ 間の到達遅延)      |
| ${}_s R_i$         | $T_s$ において, $i$ が親ノードからデータを受け取る際のビットレート ( $i$ の受信レート)                     |
| ${}_s B_{j,i}$     | ${}_s R_i$ について, 親ノードを明確に $j$ と表した変数表記                                    |
| ${}_s L_i$         | $T_s$ において, $s$ から $i$ を根とした部分木における各葉ノードまでの総到達遅延の中で最大のもの ( $i$ の最大葉ノード遅延) |

義すると, 式 (4.3) に示す制約条件を得る .

$${}_s D_{s,i} = \sum_{k=0}^{l-1} d_{r_k, r_{k+1}} \geq D_{\max} \quad \forall s, i \quad (4.3)$$

ここで, 式 (4.3) 中の  ${}_s D_{s,i}$  および  $d_{r_k, r_{k+1}}$  の定義は, 表 4.1 に示すとおりである .

なお, トランスコーディングによる遅延は, 次ノードへのデータの転送時にかかる遅延に含まれる .

#### 問題設定

本提案手法におけるツリー構築の目標は, ノード間の受信レートの偏りを軽減しつつ, 全ノードの総受信レートを向上することである . 具体的には, 前述の式 (4.1), (4.2), および (4.3) の制約条件の下で, 以下の (O1) および (O2) を達成するような ALM ツリーを構築することが目標である .

(O1) Maximize  $\min({}_s R_i)$  of all  $\{s, i\}$

(O2) Maximize  $\sum_{s,i} {}_s R_i$

ここで，(O1) および (O2) 中の  ${}_sR_i$  の定義は表 4.1 に示すとおりである．

### 4.3.3 ツリー構成アルゴリズムの詳細

提案手法の中核となる ALM ツリー構成アルゴリズムでは，まず初期状態として，各アプリケーションの通信セッション毎に，そのデータ送信元であるノードをルートとし，そのデータを受信する他の全ノードを子ノードとするようなデータの配信木（ツリー）を構成する．その後，各ノードがツリーの局所的な再構成を行い，これを全てのツリー上で自律分散的に反復して行うことで，段階的にノード間の不公平性を解消し，かつ全体の受信レートを向上する．

図 4.2 に，再構成の 1 サイクルの手順を示す．まずノード  $i$  は，ツリー  $T_s$  において，仮にノード  $i$  がノード  $j$  の子ノードになった場合， $T_s$  における自身の受信レートを向上するかどうかを判定する（図 4.2(a) (1)~ (4)）．この判定には，親ノード候補である  $j$  による後述の各種試算結果を利用する．判定の結果，受信レートを向上すると期待される場合は，ノード  $i$  は自身以下の部分木と共に，親ノード  $j$  の下に移動する（図 4.2(b)）．なお，帯域の試算の際には，後述する帯域分配アルゴリズムを用いて，全 ALM ツリーを考慮し，親ノードの送信レートを公平性を保ちつつ可能な限り最大となるよう分配する．具体的な試算手順については，4.3.7 で述べる．

本提案アルゴリズムでは，各ノードが上述の再構成手順を自律分散的に反復して行う．

なお，本提案アルゴリズムではノード切り替えによるデータの損失は考慮していないが，システムの実装において，新たな子ノードとの接続を完了させて，そのノードへのデータの転送を開始してから古い子ノードとの接続を切断する等の工夫により，ノード切り替えによるデータの損失を抑えることは可能であると考えられる．

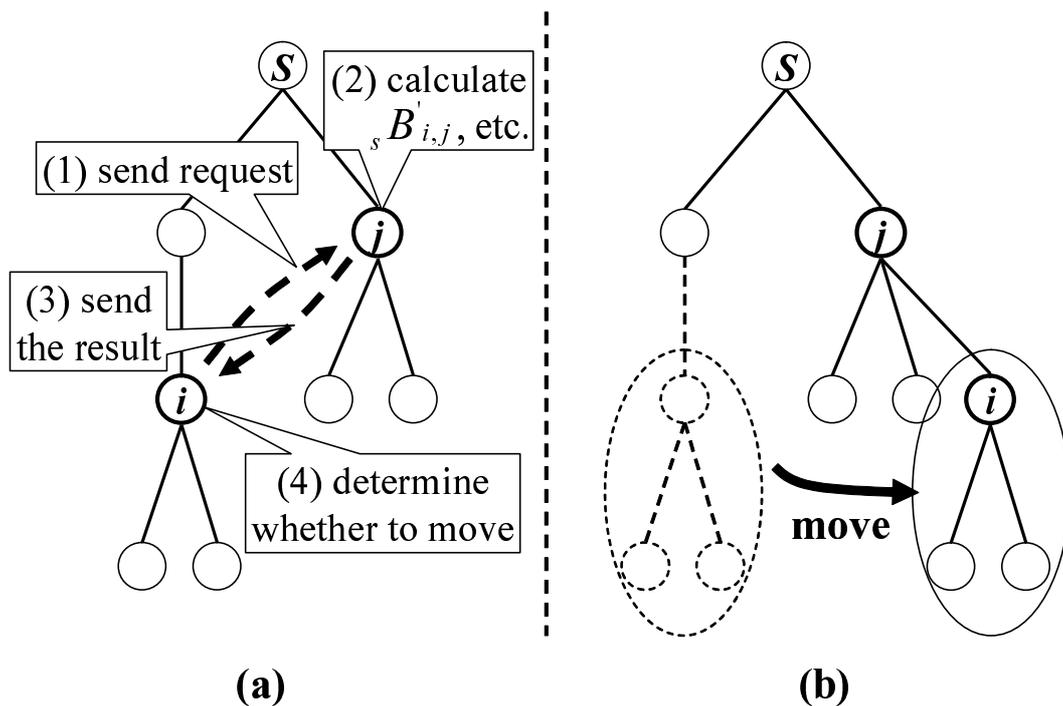


図 4.2: ツリー再構成の手順

#### 4.3.4 各ノードの持つ情報

各ノードは、オーバーレイネットワークの全参加ノードの識別子、および自身のアクセスネットワークの最大下り/上り帯域を、前もって知っているとして仮定する。識別子としては、IP アドレス等を利用できる。また、アクセスネットワークの帯域は、一般的な OS ではネットワークインタフェースの情報として取得可能である。なお、ADSL や FTTH 等のように、アクセスリンクの帯域をネットワークインターフェースから取得できない場合は、手動での設定、もしくは既存手法を用いての自動設定も可能である [41]。表 4.2 に、各ノードが保持する状態情報のうち、ツリーを再構成するかどうかの判断に必要なものを示す。ここで、ルート、親ノード、および子ノードについては、それぞれ対応したノードの識別子を保持する。

表 4.2: ツリー毎に各ノード  $i \in N$  が保持する状態情報

| 表記            | 保持情報 ( の意味 )                          |
|---------------|---------------------------------------|
| $s$           | このツリーのデータの送信元ノード ( ルート )              |
| $p_i$         | このツリーにおける $i$ の親ノード                   |
| $C_i^s$       | このツリーにおける $i$ の子ノード集合                 |
| ${}_sR_i$     | $i$ が現在実際に受信している<br>マルチメディアデータのビットレート |
| ${}_sD_{s,i}$ | ルートから $i$ までの総到達遅延                    |
| $LC_i^s$      | $i$ の全ての子ノードの最大葉ノード遅延集合               |

### 4.3.5 各種状態情報の更新

#### 受信レートおよび送信レート

各ノードは、自身の実際の受信レートや送信レートを最新の状態に保つため、以下の2種類のタイミングで、これらを更新する。

1. 自身の子ノードの数が変化した場合。この場合、現在の受信レートから各子ノードへの送信レートを再計算し、全ての子ノードに通知する。
2. 自身の親ノードから、自分への送信レート値の更新通知を受けた場合。ここで (ア) 「提示された送信レート」が (イ) 「最大下り帯域から他のツリーの受信レート値の総和を引いた値」を超えない場合、そのツリーでの受信レートを (ア) の値に更新する。そうでない場合は、そのツリーでの受信レートを (イ) の値に更新し、そのツリーの親ノードにこの値を通知する (ア) または (イ) のいずれの場合も、更新後の受信レートから各子ノードへの送信レートを再計算し、全ツリーの自身の全ての子ノードにそれを通知する。

## 総到達遅延

ノード  $i$  が親ノード  $j$  との間に  ${}_sD_{s,i} = {}_sD_{s,j} + d_{j,i}$  の関係が成り立つことを利用し、以下の方法で、各ノードが保持している総到達遅延情報を適切に更新する。

- 再構成により自身の子ノードの数が変化したノードは、ルートから自身までの総到達遅延を、そのツリーの全ての子ノードに通知する。
- 各ノードは、親ノードから総到達遅延の更新を受け取った際、その総到達遅延に、親ノードから自身までの遅延を加算し、ルートから自身までの総到達遅延を更新する。

ここで、任意のノード  $i$  と  $j$  の間の到達遅延  $d_{i,j}$  については、ノード間で時間同期をとったうえで、双方向でタイムスタンプを付与するような計測ツールを用いることで、それぞれの方向の到達遅延を求められる。また、本方式を実装したシステムを運用する際に、トランスコーディングによる遅延など、データの転送にかかる遅延が生じる場合、この到達遅延  $d_{i,j}$  に加算する。

## 最大葉ノード遅延 ${}_sL_i$

表 4.1 に示すように、あるツリー  $T_s$  において、 $i$  を根とした部分木における各葉ノードまでの総到達遅延の中で最大のものを、最大葉ノード遅延  ${}_sL_i$  と定義する。4.3.3 で述べた再構成判断時に、最大葉ノード遅延  ${}_sL_i$  の値を用いて、ALM ツリーのルートから全ての葉ノードまで許容遅延値  $D_{\max}$  以内でデータを伝達できるかどうか判断する。もしできない場合は、帯域が向上する場合であっても、親ノードを変更しない。

また、子ノードは、定期的に自身の最大葉ノード遅延を再計算し、その算出結果を自身の親ノードへ通知する。これを再帰的に行うことで、全てのノードにおける最大葉ノード遅延を最新の状態に維持する。

### 4.3.6 帯域分配アルゴリズム

本帯域分配アルゴリズムは、ノードが自身の子ノードへの送信レートを決定する際、および再構成の判断手続きの際に、各子ノードへの分配帯域を算出するアルゴリズムである。

本アルゴリズムの手順は以下の通りである。まず、各ノードは、自身のアクセスリンクの最大上り帯域をもとに、全ての ALM ツリー上での自身の子ノードに、自身の上り帯域（送信レート）を均等に割り当てる。この帯域を  $x$  とする。次に、各 ALM ツリーにおいて、自身の受信レートが  $x$  を下回る場合、余剰分の帯域を他のツリー上の子ノードに再分配する。この再分配処理を、余剰帯域がなくなるまで、繰り返し行う。

### 4.3.7 再構成判断アルゴリズム

本再構成判断アルゴリズムは、各ノードが親ノードを変更するとどのくらいの受信レートを得られるかを予め試算し、親ノードを変更すべきかどうかを判断するアルゴリズムである。

まず、再構成判断が行われるツリーのルートを  $S$ 、試算要求を行うノードを  $A$ 、 $A$  から試算要求を受けるノードを  $C$ 、および  $A$  の親ノードを  $P$  とし、下記の (Step1)~ (Step2) の手順で、試算要求の伝達を示す。

(Step1) ノード  $A$  は、試算要求メッセージをノード  $C$  へ送信する。その際自身の親ノードが  $P$  であることを知らせる。

(Step2) 試算要求メッセージは、ノード  $P$  または  $S$  のいずれかに到達するまで、ツリーのルート方向へと転送される。

次に、図 4.3 に示すように、ノード  $C$  がノード  $P$  を根とする部分木の中に存在する場合を (Case A) とし、図 4.4 に示すように、ノード  $C$  がノード  $P$  を根とする部分木の中に存在しない場合を (Case B) として、場合分けを行う。

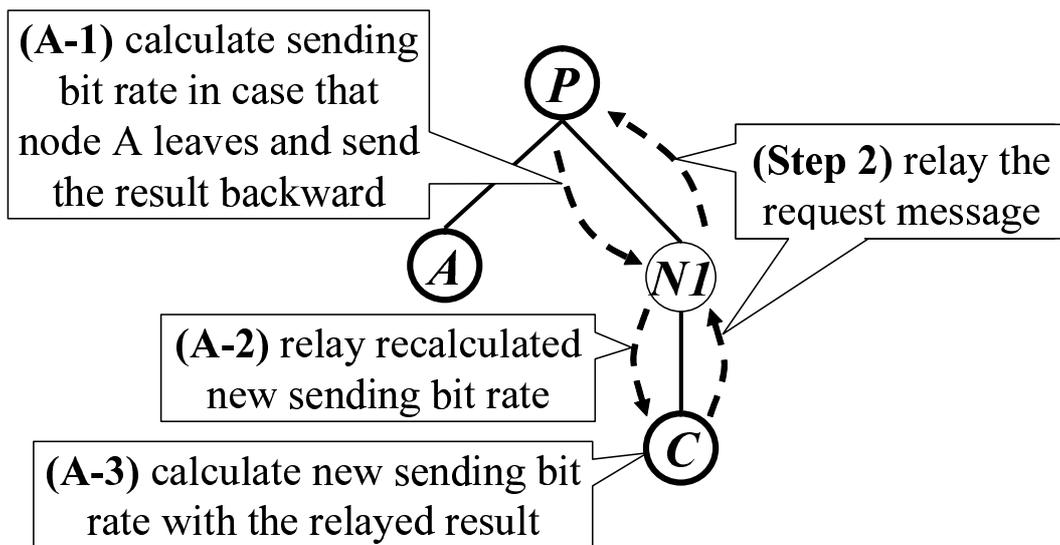


図 4.3: (Case A) の試算手順

ここで、場合分けを行うのは、以下の理由からである。まず、図 4.3 に示す (Case A) の場合、ノード A がノード C の下に移動する際、ノード P の直接の子ノードがノード A の分だけ減り、結果としてノード P からノード C へ向けた送信レートが増加する可能性がある。一方、図 4.4 に示す (Case B) の場合は、ノード A がノード C の下に移動したとしても、ノード S の直接の子ノードの数が変化しないため、ノード S からノード C へ向けた送信レートは増加しない。すなわち、(Case A) の場合のみ、試算要求を受けるノード C は、ノード P まで遡って、再構成後の帯域を試算する必要があるからである。以下、まず (Case A) の場合の試算手順を、図 4.3 の (A-1)・(A-3) に基づき述べる。

(A-1) ノード P は、 $C_p^s \leftarrow C_p^s - \{A\}$  とした場合の各子ノードへの分配帯域を再計算する。

その後、前述の (Step2) によりノード P に試算要求メッセージを転送してきた子ノードへ、算出した分配帯域を返信する。

(A-2) 試算要求メッセージを転送した各ノードは、親ノードから試算結果を受信すると、同様にその算出結果を元に自身の各子ノードへの分配帯域を再計算し、自身の子ノードに送信する。

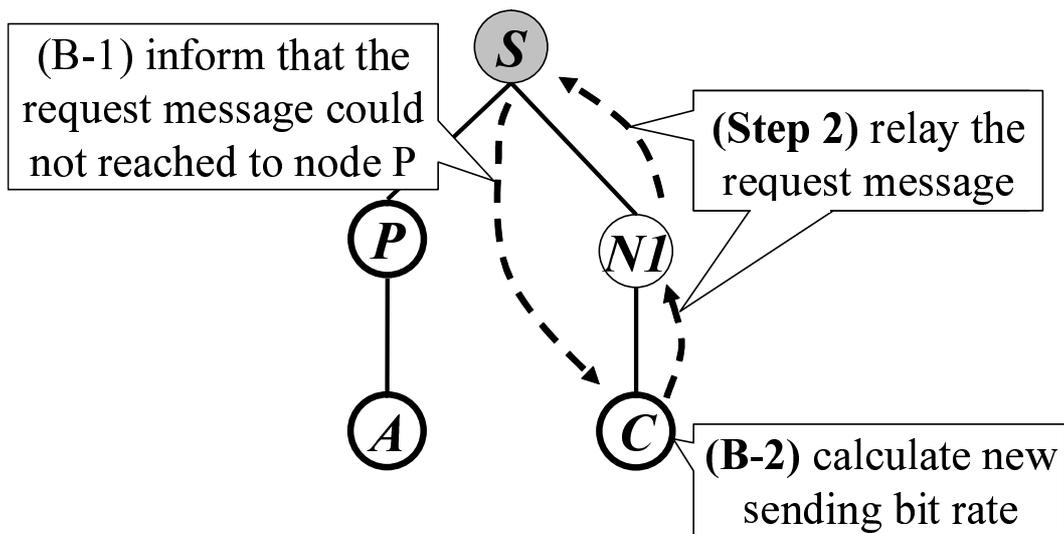


図 4.4: (Case B) の試算手順

(A-3) ノード C は、受け取った算出結果をもとに、 $C_C^s \leftarrow C_C^s + \{A\}$  とした場合の各ノードへの分配帯域を算出する。そして、ノード A への分配帯域、およびルートからノード C までの総到達遅延を、ノード A に通知する。

次に、(Case B) の場合の試算手順を、図 4.4 の (B-1)~ (B-2) に基づき述べる。

(B-1) 前述の (Step2) に基づく試算要求メッセージがノード S に到達したことにより、ノード S は、ノード C とノード S の間にノード P が存在しない、すなわち (Case B) の場合であること判定し、このことをノード C へ通知する。

(B-2) ノード C は、現在保持している受信レートをもとに  $C_C^s \leftarrow C_C^s + \{A\}$  とした場合の各ノードへの分配帯域を算出する。そして、ノード A への分配帯域、およびルートからノード C までの総到達遅延を、ノード A に通知する。

この後、ノード A は、ノード C から通知された自身への分配帯域と現在の自身の受信レートとを比較し、再構成によって自身の受信レートが向上するかを確認する。さらに、通知された遅延情報と現在の自身が保持する最大葉ノード遅延を元に、再構成後の自身の

最大葉ノード遅延を試算し、再構成後に自身を根とする部分木の各葉ノードが、それぞれ許容遅延値以内でデータを受信できるのかを確認する。

上記条件を満たした場合、ノード A はノード C の子ノードになることを決定し、その後、ツリーの再構成を行う。

## 4.4 実験と評価

### 4.4.1 実験概要

本提案手法の有効性を示すために、Java により実装したシミュレータを用いて性能評価実験を行った。本実験では、各受信ノードに割り当てられる帯域の、ALM ツリーの構成・再構成による時間的变化から、本手法の効果を検証する。

本実験では、多対多マルチメディア通信サービスの例としてビデオ会議を想定する。ビデオ会議における参加ノード数を  $n$  ノードとし、全てのノードが送信元ノードで、かつ各送信元ノードに対する受信ノードになる、 $n$  対  $n$  通信として実験を行う。また、アクセスリンクの種類として ADSL と FTTH を想定し、各ノードのアクセスリンクに関する最大下り帯域、最大上り帯域、下り遅延、上り遅延を設定してシミュレーションを試行する。

本シミュレーションにおいては、マルチメディアデータの配信はモデルに含めず、割り当てられた帯域分がデータ配信によって消費されるものと仮定する。また、バックボーンのネットワークは十分に広帯域であることを仮定し、エンドノード間の遅延についてはアクセスリンクでの遅延が支配的であるとするため、アクセスリンクの上り/下り遅延のみを扱うものとしている。試算要求を送信する相手ノードの選択方法は以下のようになる。各ノードは、マルチメディアデータの各送信元ノードをランダムに並べたリスト、および自分を除いた全参加ノードをランダムに並べたリストを保持し、まず送信元ノードからリストの順番に従って 1 ノードを選択する。その後、選択したノードをルートとしたツリーに対する試算要求の相手を、全参加ノードからリストの順番に従って 1 ノード選択する。

表 4.3: 実験に用いた各ノードのパラメータ (パターン 1)

| ノード番号         | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  |
|---------------|----|----|----|----|----|----|----|----|
| 最大下り帯域 (Mbps) | 80 | 80 | 70 | 70 | 70 | 60 | 60 | 60 |
| 最大上り帯域 (Mbps) | 30 | 30 | 20 | 20 | 20 | 10 | 10 | 10 |
| 下り遅延 (ms)     | 10 | 10 | 20 | 20 | 20 | 30 | 30 | 30 |
| 上り遅延 (ms)     | 10 | 10 | 20 | 20 | 20 | 30 | 30 | 30 |

また、シミュレーションの際の時間の単位には「unit」を用いる。unit はシミュレーション内の時間の最小単位であり、ひとつの試算要求発生からノード切り替え処理の完了までが収まる実時間を与える。本実験において 1 unit は 3 秒である。ここで、各ノードが試算要求を行う間隔を 4~ 12 unit のランダム間隔とし、さらに各端末が最大葉ノード遅延を親ノードへ通知する間隔を 5 unit とした。

#### 4.4.2 機能検証実験

まず、本提案手法によって構築される ALM ツリーが、許容遅延値を満たした上で、ノード間の受信レートの偏りの軽減および全体的な受信レートの向上を達成しているかの基本機能の検証実験を行う。本実験では、ビデオ会議における参加ノード数を 8 ノードとし、8 対 8 通信として実験を行う。また、各ノードのアクセスリンクに関する各パラメータを表 4.3 に示す値に設定してシミュレーションを 50 回試行した。

各ノードの受信レートに関して 50 回の試行結果の平均値をまとめたものを図 4.5 に示す。図中の各曲線は、各 unit 時間における、全ノードの受信レートの平均値 (Ave) と標準偏差 (Standard deviation)、および、全ノードの受信レートを昇順にソートした場合の中央値 (Median)、25% 値 (25th percentile)、最小値 (Min) をそれぞれ示している。標準偏差の値については、グラフの右の y 軸に、その他の値についてはグラフの左の y 軸に、それぞれ従う。

図 4.5 より、平均値は変化していないが、最小値や 25% 値は段階的に向上し、かつ標準偏差が小さくなったことから、ノード間の受信レートの偏りが段階的に軽減していることが分かる。また、シミュレーションを開始してから約 150 unit 後に、それぞれの値が収束していることが分かる。この収束時間に関しては、4.4.4 節で詳しく考察する。

以上の結果より、低ビットレートでデータを受信しているノードの受信レートが段階的に向上していることから、本提案手法によってノード間の通信品質の偏りを軽減できることが確認できた。

また、全ツリーの中での最大葉ノード遅延についてトレースした結果、本実験では、全てのツリーにおいて、ルートから各ノードまでの到達遅延が、常に許容遅延値 (200 ms) 以内に収まるようにツリーが構築されていることも確認できた。

なお、提案手法では、各ノード間の受信レートの比率はノード間でのアクセスリンクの帯域の比率に依存して変化する。すなわち、アクセスリンクの帯域の比率のばらつきが同じであれば、各リンクにおける絶対的な帯域に関わらず、各ノード間の受信レートの比率の傾向は変わらない。このことを確認するため、表 4.3 で示されるパターンの上り / 下り帯域を全て 1/10 した場合の実験結果を図 4.6 に示す。図 4.5 と図 4.6 より、両者は同じ傾向を示していることが分かる。

### 4.4.3 比較実験

次に、本提案手法と、一対多通信のための ALM ツリーを作成する既存手法とを比較する実験を行い、単純に複数の ALM ツリーを作成した場合よりも、本手法のほうがノード間の受信レートの偏りの小さい ALM ツリーが作成できることを示す。

本実験では、4.4.2 と同様の実験環境で、ノードのパラメータ設定値についてのみ、表 4.3 の設定に加えて、表 4.4、表 4.5 に示す値を用いた 2 種類のパターンを追加した。ここでノードパラメータの各パターンとしては、アクセスリンクの通信品質について、全ノード

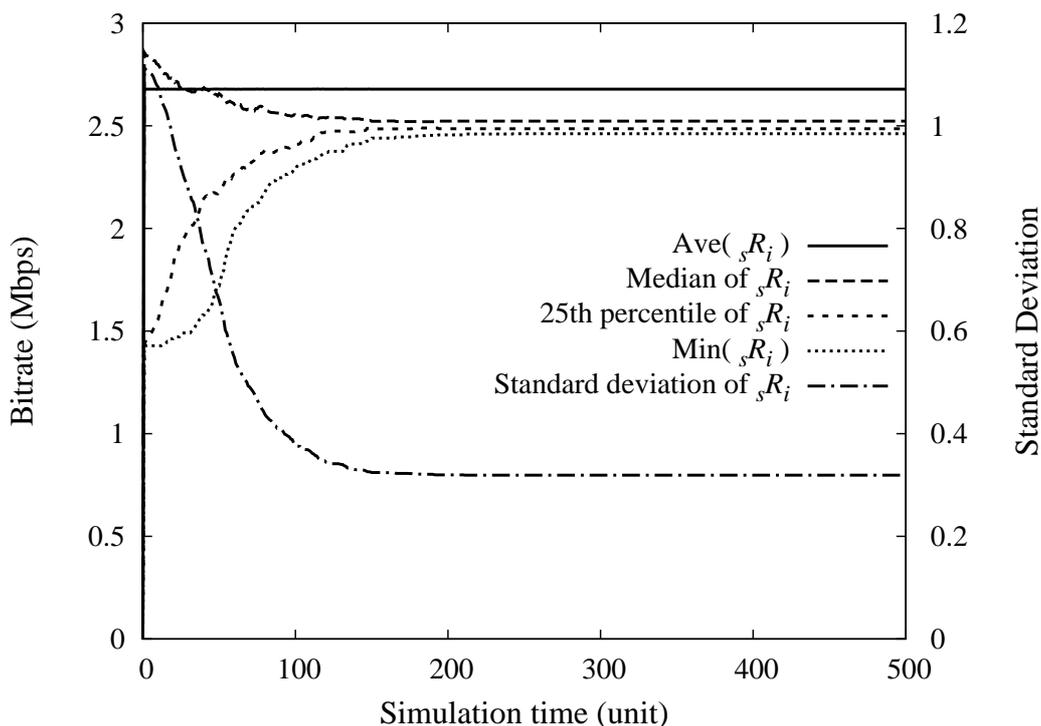


図 4.5: 機能検証実験における受信ビットレートの変化

ド間の相対的な品質を比べた場合に、品質が比較的高い / 中間 / 低いノードの割合が等しい表 4.3 のようなケース (パターン 1)、高い品質のノードがひとつだけ含まれる表 4.4 のようなケース (パターン 2)、高い品質のノードと低い品質のノードの品質の差がパターン 1, 2 より大きい、すなわちノード間でのアクセスリンクの帯域の比率のばらつきが大きいケース (パターン 3) を想定した。以上 3 種類のパターンについてそれぞれ 50 回ずつシミュレーションを行った。

また、比較対象としては文献 [11] の手法を用いる。ただし、文献 [11] の手法の導入に際し、多対多通信の場合であっても帯域に関する制約条件を満たすように、帯域の分配の方法はツリー構築後に本手法と同様の基準で行うこととする。

図 4.7, 図 4.8, 図 4.9 に、それぞれ表 4.3, 表 4.4, 表 4.5 の場合の実験結果を示す。ここで図 4.7(a), 図 4.8(a), および図 4.9(a) では、提案手法 (Our method), 既存手法 (Existing

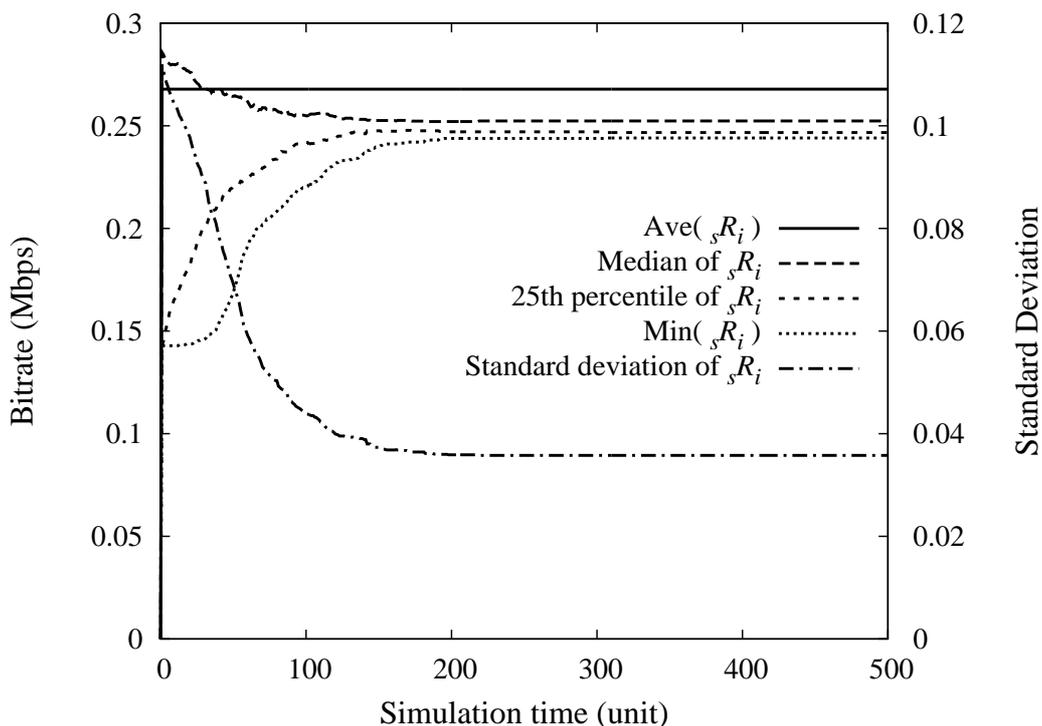


図 4.6: パターン 1 の全てのアクセスリンク帯域を一律 1/10 にした場合の受信ビットレートの変化

method) それぞれの, 各 unit における全ノードの受信レートの平均値および 25% 値を示しており, 図 4.7(b), 図 4.8(b), および図 4.9(b) では, 両手法それぞれの, 各 unit における全ノードの受信レートの最小値および標準偏差を示している. それぞれ, 50 回試行の平均値である.

まず図 4.7 では, 本手法において, 初期状態では受信レートの最小値, 25% 値ともに既存手法に劣っているものの, 再構成を繰り返した結果, どちらの値も既存手法のそれを上回ることが分かる. 最小値については, 最終的に既存手法の値より 1.31 倍程度向上している. また, 標準偏差の値も本手法の方が小さく, 受信レートのノード間の偏りは既存手法より小さいことが分かる. これらについては, 図 4.8 でも同様の傾向が言えるが, 最小値は既存手法の 1.27 倍にとどまるなど, 図 4.7 の場合に比べて本手法の効果は小さくなっている.

これに対し図4.9では、最小値、25%値のともに既存手法に近づいているものの、基本的に既存手法を上回っていない。ただし標準偏差は小さいことから、受信レートのノード間の偏りは既存手法より小さいことが分かる。

また、全ノードの受信レートの平均値については、図4.9(a)のようにパターン3では本手法のほうが若干低いですが、図4.7(a)および図4.8(a)のようにパターン1およびパターン2については両手法とも同等であり、目標のひとつである(O2)、すなわち全ノードの受信レートの総和の最大化に関して大きな差はない。

以上の結果から、本提案手法によって、既存手法に比べノード間の公平性を最大限に保ちつつ、全ノードが受信するマルチメディアデータのビットレートを段階的に向上できることを確認した。ただし、ノードの上り帯域が、他ノードのそれに比べてとても大きい、または小さいノードが多数存在する環境では提案手法の効果が小さいことが分かる。この傾向はノード条件をさらに分散させていくと顕著になり、本手法が既存手法より劣ると考えられる。これは、既存手法では、上り帯域の大きいノードが子ノードをたくさん持つ場合であっても、そのノードの帯域が他に比べてとても大きい場合には十分に子ノードへ帯域を提供することができ、偏りがあまり生じないためである。さらに、既存手法は遅延による制約がないのに対し、提案手法は200 ms以内という制約を設定しているために、受信ビットレートは向上するが遅延制約を満たさなくなるようなツリーへは再構成しない場合が発生することも原因として考えられる。従って提案手法は、ADSLとFTTH、またこれらが混在したとしても極端な上り帯域の差がない環境では特に効果を発揮する一方、無線とFTTHといった極端なアクセスリンク帯域の組み合わせによる環境ではあまり有効ではないと考えられる。

表 4.4: 実験に用いた各ノードのパラメータ (パターン 2)

| 端末番号        | 0   | 1   | 2   | 3  | 4  | 5  | 6  | 7  |
|-------------|-----|-----|-----|----|----|----|----|----|
| 下り帯域 (Mbps) | 100 | 100 | 100 | 80 | 70 | 70 | 60 | 60 |
| 上り帯域 (Mbps) | 80  | 50  | 50  | 30 | 20 | 20 | 10 | 10 |
| 下り遅延 (ms)   | 10  | 10  | 10  | 10 | 20 | 20 | 30 | 30 |
| 上り遅延 (ms)   | 10  | 10  | 10  | 10 | 20 | 20 | 30 | 30 |

表 4.5: 実験に用いた各ノードのパラメータ (パターン 3)

| 端末番号        | 0   | 1   | 2  | 3  | 4  | 5  | 6  | 7  |
|-------------|-----|-----|----|----|----|----|----|----|
| 下り帯域 (Mbps) | 100 | 100 | 70 | 70 | 60 | 60 | 50 | 12 |
| 上り帯域 (Mbps) | 80  | 50  | 20 | 20 | 20 | 10 | 10 | 1  |
| 下り遅延 (ms)   | 10  | 10  | 20 | 20 | 30 | 30 | 35 | 40 |
| 上り遅延 (ms)   | 10  | 10  | 20 | 20 | 30 | 30 | 35 | 40 |

#### 4.4.4 考察

ここで、受信レートの向上や不公平性の軽減と収束状態に至るまでの時間 (収束時間) との関係について考察する。本提案アルゴリズムの性質上、受信レートの最小値は必ず単調増加になるので、基本的に一回の試行中では必ず収束する。しかし、収束するときのステップ数、および最終的に収束したときのその値 (最終値) は、同じノード数、パラメータの組み合わせであってもそれぞれの試行で異なる。

そこで、図 4.7 (pattern 1/best case)、図 4.8 (pattern 2/pretty good case)、および図 4.9 (pattern 3/not good case) のそれぞれの実験結果に対し、一回の試行において、受信レートの最小値が最終値の 95% に到達するまでのステップ数を基準とした場合の、50 回の試行に対する収束ステップ数の分散を表 4.6 にまとめた。

この表から分かるように、既存手法と比べて、達成した受信レートや公平性の観点から効果がある一方で、収束時間が必ずしも短いとは限らない。受信レートの向上に関して同

表 4.6: 各実験における最小値の収束時間

|                            | 平均値    | 標準偏差  |
|----------------------------|--------|-------|
| pattern 1/best case        | 99.04  | 34.95 |
| pattern 2/pretty good case | 227.60 | 82.88 |
| pattern 3/not good case    | 134.92 | 70.18 |

等の性能を保ちつつ、より短いステップ数で収束するようにするために、今後、試算を要求する相手の決定やそのタイミング等に対する効果的な戦略を検討する必要がある。

次にツリー構築の際のネットワーク負荷に関して考察する。本研究中で行った各実験においては、どの場合も 1 unit あたりのメッセージ数は高々 80 個程度であり、実環境に実装した場合、1 unit を 5 秒程度に設定することで、制御メッセージの交換によるネットワークへの負荷はほとんど無視できると考えられる。

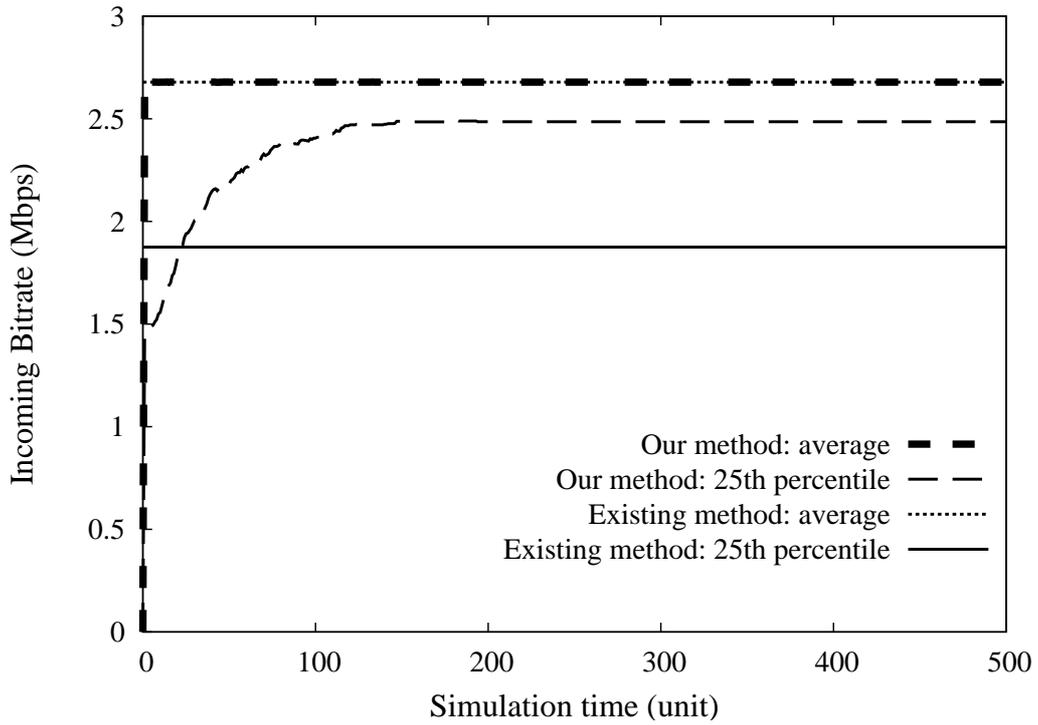
また現状では、ノード全体の状況としては品質向上や公平化がまだ進む可能性を持つはずなのに、ツリーの構成によっては局所的な最適状態に陥ることで、それ以上他の状態に遷移しなくなる場合も見受けられた。本手法に自身と子ノード（親ノード）との入れ替えのためのアルゴリズム等を追加することで、局所的な最適状態を回避することが可能であると考えられる。現在のところ提案手法は、ノードの参加・退出の際のツリーへの初期接続位置や処理手順等に関しては厳密に規定を設けていないため、これらの処理手順の導入については今後検討していく予定である。本手法は、オーバーレイ上のノード構成が変化していなくてもツリーを動的に再構成するアルゴリズムを採用しているため、基本的にはノードの動的な参加・退出にも対応可能である。

## 4.5 要約

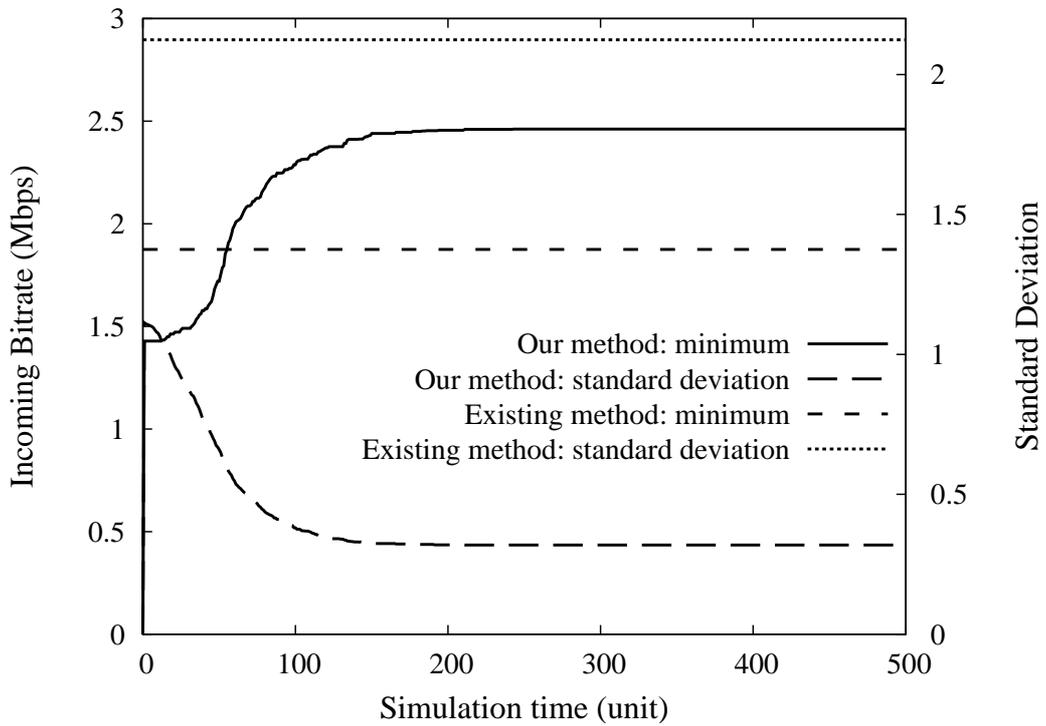
本章では、本研究の目的である、ネットワークレベルのサービス品質を向上させる上での課題 (T2) である、有線ネットワークの技術課題、すなわちエンドシステムマルチキャ

ストの配信経路の制御における帯域予測が困難なことによる、通信品質の偏りの発生（公平性のある多対多通信の実現が困難）であることを解決するために、多対多マルチメディア通信における、利用者端末が受信するマルチメディアデータの通信品質の偏りを段階的に軽減するエンドシステムマルチキャストの配信木の構成法を提案した。また、計算機シミュレーションを通じ、既存手法と比べ、受信ビットレートの最小値を1.31倍に向上し、かつ、その標準偏差を既存手法の1/4以下に抑えられることを示した。これにより、本提案手法が、多様なアクセスリンク帯域が混在する環境においても、各利用者端末へ可能な限り公平な通信品質でマルチメディアデータを提供することが可能であることを確認した。

以上のことから、本提案手法は、有線ネットワークにおけるネットワークレベルのサービス品質を向上させることが可能であるという結論を得た。

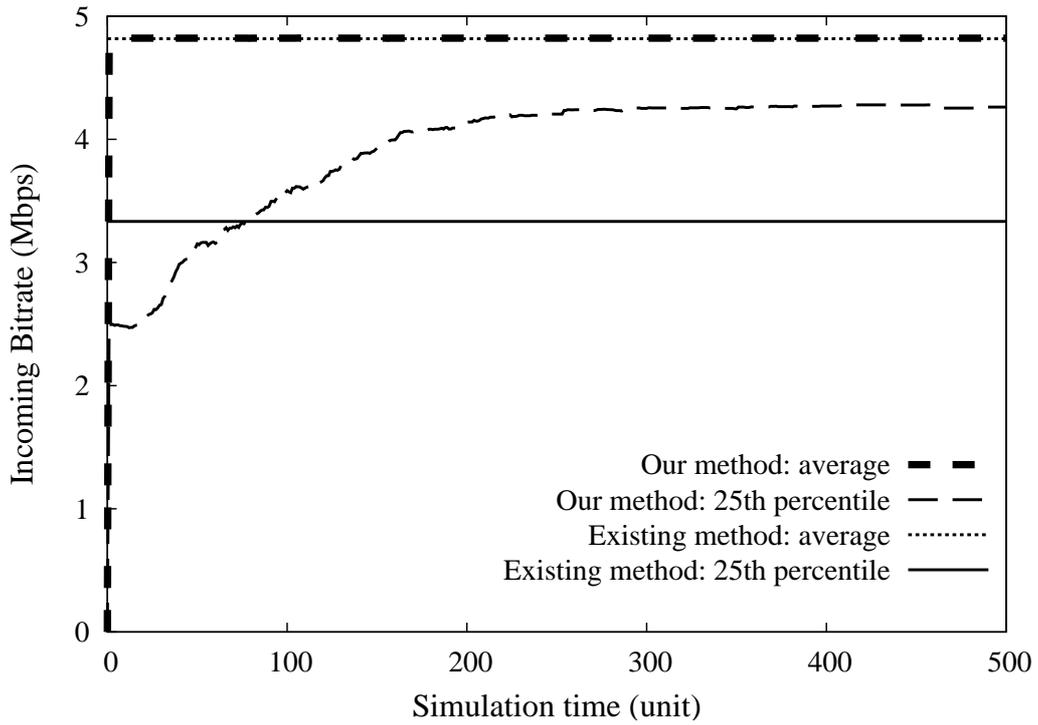


(a) average, 25th percentile

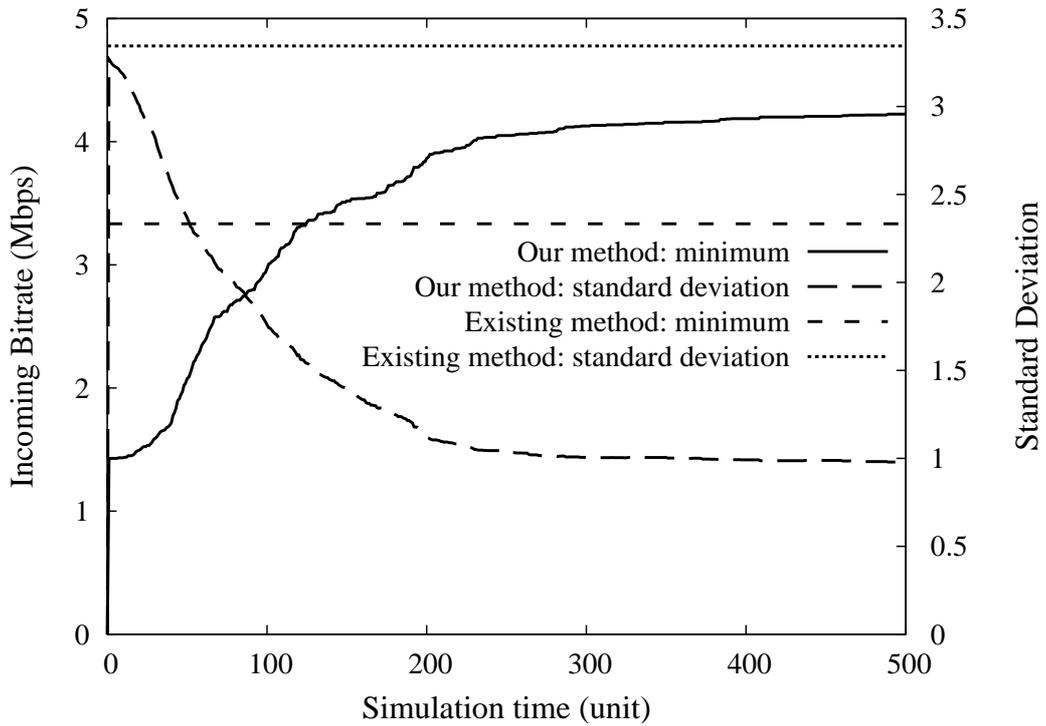


(b) minimum, standard deviation

図 4.7: 受信ビットレートの変化 (パターン 1)

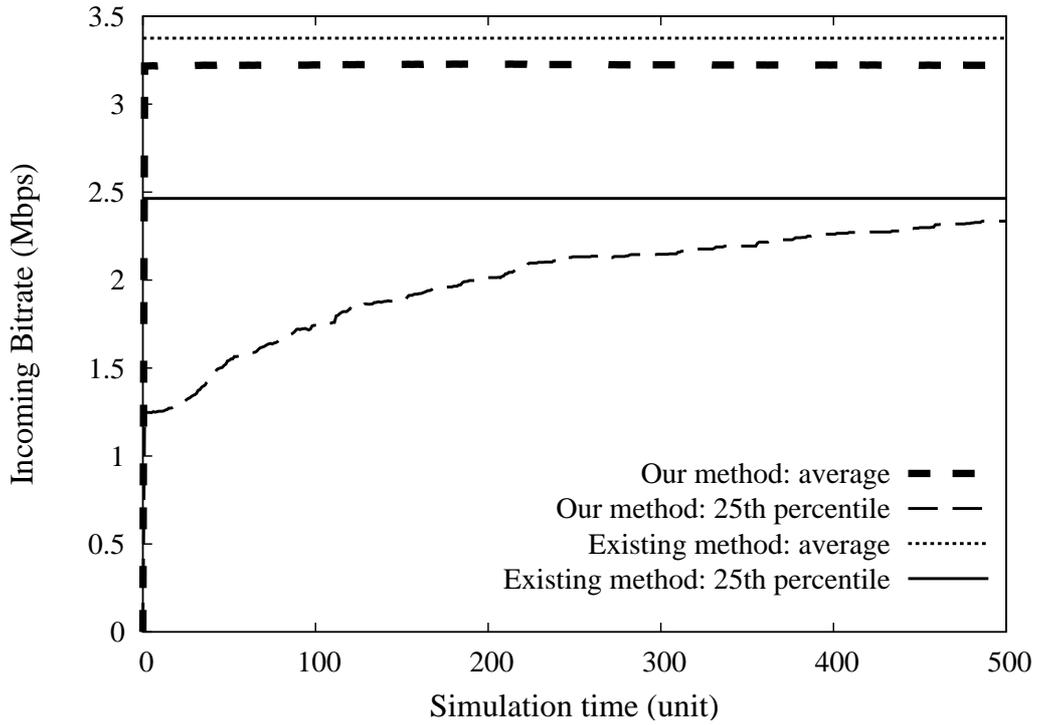


(a) average, 25th percentile

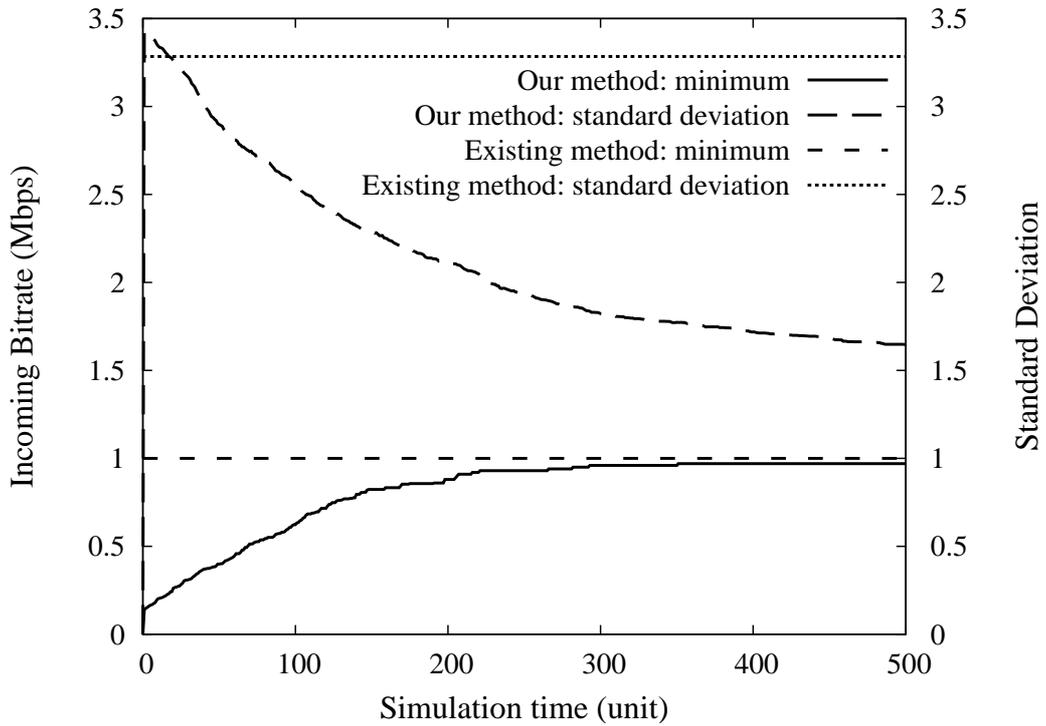


(b) minimum, standard deviation

図 4.8: 受信ビットレートの変化 (パターン 2)



(a) average, 25th percentile



(b) minimum, standard deviation

図 4.9: 受信ビットレートの変化 (パターン 3)

# 第5章 結論

## 5.1 本研究の概要

本研究では，ユビキタス情報環境における利用者同士のコミュニケーション手段として有用であるマルチメディア通信に対して，ユーザレベルのサービス品質を向上させるため，その土台となるネットワークレベルのサービス品質の向上させることを目的とし，ネットワークレベルのサービス品質の向上を実現するためのアプローチとして，ネットワーク QoS 技術の一つである経路制御技術を活用した．ここで，マルチメディア通信のネットワークレベルのサービス品質として，特に重要である通信帯域を本研究では主に取り扱い，ネットワークレベルのサービス品質に基づき，利用可能な帯域を適切に割り当てられる経路を構成する，ネットワークの経路制御法の確立を目指した．

ここで，無線アドホックネットワークにおけるマルチホップ通信，および有線ネットワークにおけるエンドシステムマルチキャストは，利用可能な帯域が変動する情報通信基盤であり，正確な帯域予測に基づく適応型の経路制御技術が重要となる．そこで本研究では，無線ネットワークおよび有線ネットワークの経路制御における，それぞれの帯域予測に関する課題を解決する以下の経路制御法をそれぞれ提案した．

(無線ネットワーク) 電波の干渉範囲内に存在する端末の帯域情報を積極的に活用することによる高精度な帯域予測に基づく無線アドホックネットワークのための経路制御法

(有線ネットワーク) 中継端末を共有する全てのデータ配信木の消費帯域を追従すること

により配信木を反復して改善するエンドシステムマルチキャストの配信経路の制御

## 5.2 各章の要約

### 第1章 序論

研究の背景を示し，本研究の位置付けと目的を設定した．

### 第2章 サービス品質向上のためのネットワーク経路制御に関する課題

ネットワークレベルのサービス品質を向上させる上での従来の経路制御における課題を解析し，その結果，(A) 無線ネットワークの経路制御および(B) 有線ネットワークの経路制御に関して，それぞれ以下のような技術課題があることを明確にした．

- (T1) 無線アドホックネットワークの経路制御において，帯域予測の精度が低いためセッションが所望の通信帯域の充足が困難である
- (T2) エンドシステムマルチキャストのデータ配信木において，多対多通信においては各端末が複数の配信木の間で消費する帯域を予測することが困難であるため，利用者端末間で通信品質に偏りが生じる

### 第3章 無線アドホックネットワークのための高精度な帯域予測に基づく経路制御法

無線ネットワークにおいてネットワークレベルのサービス品質を向上させるためには，従来の無線アドホックネットワークのための経路制御における技術課題である，(T1) 帯域予測の精度が低いため，セッションが必要とする通信帯域の充足が困難であることを解決する必要であった．そこで本章では，各端末自身が把握している帯域情報を端末間で交換し，CSR内の端末(2-hopの隣接端末)が各々把握している帯域情報を収集することで

CSR 内の消費帯域を高精度に予測し，収集した情報を用いたアドミッション制御を行う経路制御手法を提案した．計算機シミュレーションによる性能評価を通し，従来手法に比べて，確立したセッションに対して高いデータ配信率（および低遅延）を実現する，すなわちセッションが必要とする通信帯域の充足が可能である経路を多く構成できることを確認し，本提案手法の有効性を示した．

以上のように，本提案手法によって無線ネットワークにおいてネットワークレベルのサービス品質が向上することから，本提案手法は，無線ネットワークにおいてマルチメディア通信を活用するために効果的な経路制御法であるという結論を得た．

#### 第4章 部分木の帯域予測に基づくエンドシステムによるマルチキャストの配信経路の制御法

有線ネットワークにおいてネットワークレベルのサービス品質を向上させるためには，従来のエンドシステムマルチキャストの配信経路の制御における技術課題である，(T2) 多対多通信において各端末が複数の配信木の間で消費する帯域を予測することが困難であるため，利用者端末間で通信品質に偏りが生じることを解決する必要であった．そこで本章では，1回の配信木の再構成の対象を局所的な部分木にしぼることで複数の配信木の間で消費する帯域の予測を可能にし（容易にし），これに基づき配信木の局所的な再構成を自律分散的に反復して行うことで通信品質の偏りを段階的に軽減するツリー構成法を提案した．シミュレーションによる評価により，不均一なアクセスネットワーク環境においても全ツリー中の全ノードにおける受信ビットレートの中の平均値を維持しつつ最小値を向上し，標準偏差を低減した．すなわち本提案手法が従来手法に比べて利用者端末間の通信品質の偏りを軽減し，多対多通信のサービス全体として通信品質の向上を実現させたことを確認したことで，本手法の有効性を示した．

以上のように，本提案手法によって利用者端末間の通信品質の公平性が向上という形で

有線ネットワークにおいてネットワークレベルのサービス品質を向上させることから、本提案手法は有線ネットワークにおいて多対多のマルチメディア通信を活用するために効果的な経路制御法であるという結論を得た。

### 5.3 本研究の貢献

本研究では、無線アドホックネットワークにおける通信セッションの所望の通信帯域を充足する経路制御法およびエンドシステムのマルチキャストにおいて利用者端末間で公平な通信品質で多対多通信を可能とする配信経路の制御法を提案した。以上の2つの提案手法の導入により、有線ネットワークおよび無線ネットワークそれぞれにおいてネットワークレベルのQoSの向上を実現したため、その結果、その上位に位置するアプリケーションレベルのQoSおよびその上位のユーザレベルのQoSの向上も促進させることが可能となる。以上のことから、本研究はマルチメディア通信のような高度な情報通信サービスをより発展させる上で、高性能な情報通信基盤（すなわちユビキタス情報環境）を効果的に利用するための技術の一端を担うものといえ、これにより図5.1で示すように、ユビキタス情報社会のための情報通信サービスの高度化に貢献したといえる。

### 5.4 今後の課題

今後の課題として、まず、無線アドホックネットワークのための高精度な帯域予測に基づく経路制御法に関して帯域の利用効率を向上する手法の検討を行う。具体的には、本手法はこれまで、確立できるセッションを基本的に早いもの勝ちとしていたが、各セッションの所望の通信帯域や優先度による確立するセッションを調整する仕組みなどを検討する。また、セッション間の通信帯域の公平性の向上のために、確立できるセッションを増加させる機構の検討する。具体的には、通信帯域を充足しない場合の調整のため、アド

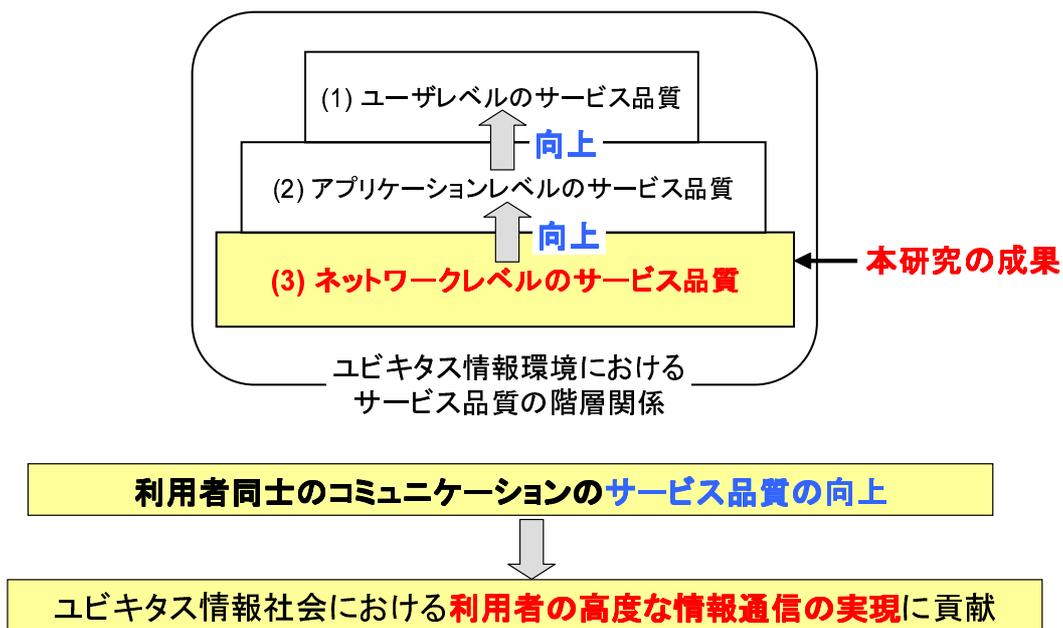


図 5.1: ユビキタス情報社会に対する本研究の貢献

ミッション制御の結果をアプリケーションの所望する帯域の決定にフィードバックすることなどを検討している。

部分木の帯域予測に基づくエンドシステムマルチキャストの配信経路の制御法に関しては、収束時間の短縮などの性能向上の手段を検討している。

以上の個別の提案手法の性能改善に関する課題に加え、全体の課題、すなわちユビキタス情報社会のための技術に関して、エンドシステムマルチキャスト+アクセスネットワークとしてのアドホックネットワークによる、「いつでも・どこでも」な利用者同士のコミュニケーション環境を実現する技術の検討を行う。

# 謝辞

末筆ながら，本研究をおこなうにあたり，多くの方々に御指導・御助言を頂きました．ここに，心より深く感謝の意を表します．

特に，本研究をおこなう機会を与えて頂き，日頃から多くの暖かい御指導を賜りました，東北大学電気通信研究所教授の白鳥則郎先生に心より深謝いたします．

また，本論文の審査をして頂き，有益な御助言をくださいました，東北大学大学院情報科学研究科教授の堀口進先生，並びに，東北大学大学院情報科学研究科教授の橋本和夫先生，東北大学電気通信研究所准教授の北形元先生の各先生方に心より感謝いたします．

そして，数多くの御助言と御指導をくださいました，東北大学サイバーサイエンスセンター教授の木下哲男先生に心より感謝いたします．

さらに，本研究を進めるにあたり終始御指導を賜り，研究室生活においても支えて頂きました，東北大学電気通信研究所准教授の菅沼拓夫先生に心より感謝いたします．

本研究を進めるにあたり，有意義な討論と数多くの御助言を下された，東北大学電気通信研究所客員准教授のDebasish 先生をはじめとする，東北大学電気通信研究所白鳥研究室の皆様へ，心からお礼申し上げます．

最後に，日頃から精神的・経済的に今日まで支えてくださった家族に心から感謝し，本論文を締めくくります．

## 参考文献

- [1] 戸田巖 , ネットワーク QoS 技術 , オーム社 , 2001.
- [2] R. Braden, D. Clark, and S. Shenker, “Integrated Services in the Internet Architecture: an Overview, IETF RFC 1633,” , Jun. 1994.
- [3] R.B. (Ed), L. Zhang, S. Berson, S. Herzog, and S. Jamin, “Resource ReSerVation Protocol (RSVP), IETF RFC 2205,” , Sept. 1997.
- [4] A. Kumar, B. Badrinath, and A. Acharya, “Mrsvp: a resource reservation protocol for an integrated services network with mobile hosts,” Wireless Networks, vol.7, no.1, pp.5–19, Jan. 2001.
- [5] S. Lee, G. Ahn, X. Zhang, and A.T. Campbell, “Insignia: an ip-based quality of service framework for mobile ad hoc networks,” Journal of Parallel and Distributed Computing, vol.60, no.4, pp.374–406, Apr. 2000.
- [6] C. Hedrick, “Routing Information Protocol, IETF RFC 1058,” , June 1988.
- [7] J. Moy, “OSPF Version2, IETF RFC 2328,” , April 1998.
- [8] Y.R. (Ed), T.L. (Ed), and S.H. (Ed), “A Border Gateway Protocol 4 (BGP-4), IETF RFC 4271,” , Jan. 2006.
- [9] E. Rosen, A. Viswanathan, and R. Callon, “Multiprotocol Label Switching Architecture, IETF RFC 3031,” , Jan. 2001.

- [10] Y. hua Chu, S. Rao, S. Seshan, and H. Zhang, "A case for end system multicast," *IEEE J. Selected Areas in Communications*, vol.20, no.8, pp.1456–1471, Oct. 2002.
- [11] M.S. Kim, S. Lam, and D.Y. Lee, "Optimal distribution tree for internet streaming media," *Proc. IEEE ICDCS'03*, pp.116–125, May 2003.
- [12] M. Hosseini, and N.D. Georganas, "End system multicast routing for multi-party video-conferencing applications," *Computer Communications*, vol.29, no.11, pp.2046–2065, 2006.
- [13] J. Jannotti, D.K. Gifford, K.L. Johnson, M.F. Kaashoek, and J.W. O'Toole, Jr., "Overcast: reliable multicasting with an overlay network," *Proc. USENIX OSDI'04*, pp.197–212, Oct. 2000.
- [14] S. Banerjee, C. Kommareddy, K. Kar, B. Bhattacharjee, and S. Khuller, "OMNI: An efficient overlay multicast infrastructure for real-time applications," *Computer Networks*, vol.50, no.6, pp.826–841, Apr. 2006.
- [15] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel, "ALMI: An application level multicast infrastructure," *Proc. USENIX USITS'01*, pp.49–60, Mar. 2001.
- [16] C.E. Perkins, E.M. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing, IETF RFC 3561," , July 2003.
- [17] D.B. Johnson, and D.A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, eds. Imielinski, and Korth, Kluwer Academic Publishers, 1996.
- [18] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, IETF RFC 2474," , Dec. 1998.

- [19] M. Hosseini, D.T. Ahmed, S. Shirmohammadi, and N.D. Georganas, “A survey of application-layer multicast protocols,” *IEEE Communications Surveys & Tutorials*, vol.9, no.3, pp.58–74, 2007.
- [20] V. Padmanabhan, H. Wang, P. Chou, and K. Sripanidkulchai, “Distributing streaming media content using cooperative networking,” *Proc. ACM NOSSDAV’02*, pp.177–186, May 2002.
- [21] D. Tran, K. Hua, and T. Do, “A peer-to-peer architecture for media streaming,” *IEEE J. Selected Areas in Communications*, vol.22, no.1, pp.121–133, Jan. 2004.
- [22] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, “Scalable application layer multicast,” *Proc. ACM SIGCOMM 2002*, vol.32, no.4, pp.205–217, Oct. 2002.
- [23] Y. Sun, E.M. Belding-Royer, and C.E. Perkins, “Internet connectivity for ad hoc mobile networks,” *Int. J. Wireless Information Networks*, vol.9, no.2, pp.75–88, Apr. 2002.
- [24] 間瀬憲一, 大和田泰伯, 前野誉, “モバイルアドホックネットワークのインターネット接続方式,” *信学論*, vol.J90-B, no.4, pp.361–369, Apr. 2007.
- [25] C.E. Perkins, J.T. Malinen, R. Wakikawa, A. Nilsson, and A.J. Tuominen, “Internet connectivity for mobile ad hoc networks,” *Wireless Communications and Mobile Computing*, vol.2, no.5, pp.465–482, Aug. 2002.
- [26] Internet Engineering Task Force (IETF), <http://www.ietf.org/>, MANET WG, .
- [27] T. Clausen, and P. Jacquet, “Optimized Link State Routing Protocol (OLSR), IETF RFC 3626,” , October 2003.
- [28] C. Perkins, and P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers,” *Proc. ACM SIGCOMM’94*, pp.234–244, 1994.

- [29] Z.J. Haas, and M.R. Pearlman, "The performance of query control schemes for the zone routing protocol," *ACM/IEEE Transactions on Networking*, vol.9, no.4, pp.427–438, Aug. 2001.
- [30] Q. Xue, and A. Ganz, "Ad hoc qos on-demand routing (aqor) in mobile ad hoc networks," *Journal of Parallel and Distributed Computing*, vol.63, pp.154–165, 2003.
- [31] C.R. Lin, and J.S. Liu, "Qos routing in ad hoc wireless networks," *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1426–1438, August 1999.
- [32] S. Chen, and K. Nahrstedt, "Distributed quality-of-service routing in ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1488–1505, August 1999.
- [33] C. Zhu, and M.S. Corson, "Qos routing for mobile ad hoc networks," *Proc. IEEE INFOCOM 2002*, vol.2, pp.958–967IEEE, 2002.
- [34] R. Sivakumar, P. Sinha, and V. Bharghavan, "Cedar: A core-extraction distributed ad hoc routing algorithm," *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1454–1465, August 1999.
- [35] , "Ieee standard 802.11 (iso/iec 8802-11: 1999)," .
- [36] "The network simulator – ns-2," .
- [37] J. Broch, D.A. Maltz, 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'98*, pp.85–97, October 1998.

- [38] A. Muqattash, and M. Krunz, “A distributed transmission power control protocol for mobile ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol.3, no.2, pp.113–128, April–June 2004.
- [39] S. Bhattacharyya, C. Diot, and J. Jetcheva, “Pop-level and access-link-level traffic dynamics in a tier-1 pop,” *Proc. 1st ACM SIGCOMM Workshop on Internet Measurement*, pp.39–53, Nov. 2001.
- [40] 栗田孝昭, 井合知, 北脇信彦, “オーディオビジュアル通信における伝搬遅延の影響,” *信学論*, vol.J76-B-1, no.4, pp.331–339, 1993.
- [41] N. Hu, and P. Steenkiste, “Evaluation and characterization of available bandwidth probing techniques,” *IEEE J. Selected Areas in Communications*, vol.21, no.6, pp.879–894, Aug. 2003.

# 発表論文

## 学術論文

- (1) Toshiaki Osada, Gen Kitagata, Debasish Chakraborty, Takuo Suganuma, and Norio Shiratori, “A New QoS Routing Scheme Based on Bandwidth Consumption for MANETs,” 情報処理学会論文誌, Vol.50, No.1, pp.2-13, Jan. 2009.
- (2) 長田俊明, 牧志純, 北形元, 木下哲男, 白鳥則郎, “多対多マルチメディア通信のための反復型アプリケーション層マルチキャストトリー構成法,” 電子情報通信学会論文誌, Vol.J92-B, No.1, pp.98–108, Jan. 2009.

## 国際会議

- (3) Toshiaki Osada, Gen Kitagata, Debasish Chakraborty, Takuo Suganuma, and Norio Shiratori, “An Effective Application-level QoS Routing Scheme for MANETs,” The 2007 International Symposium on Applications and the Internet (SAINT2007), IEEE Computer Society, Jan. 2007.
- (4) Toshiaki Osada, Gen Kitagata, Debasish Chakraborty, Takuo Suganuma, and Norio Shiratori, “A New QoS Routing Scheme for Mobile Ad Hoc Networks,” In Proceedings of the Joint international Conference of the 4th International Symposium on System Construction of Global-Network-Oriented Information Electronics and Student-Organizing

International Mini-Conference on Information Electronics System (the 4th IGNOIE-COE & SOIM-COE06), pp.400–401, Jan. 2007.

- (5) Toshiaki Osada, Jun Makishi, Gen Kitagata, Tetsuo Kinoshita, and Norio Shiratori, “Experiment and Evaluation of Iterative Construction Method of Application-Layer Multicast Tree for Many-to-Many Communication,” In Proceedings of the 1st Student Organizing International Mini-Conference on Information Electronics Systems (SOIM-GCOE08), pp.125–126, Oct. 2008.
- (6) Debasish Chakraborty, Jun Makishi, Toshiaki Osada, Gen Kitagata, Atushi Takeda, Kazuo Hashimoto, and Norio Shiratori, “Construction of Overlay Application Layer Multicast Tree for Efficient Multi-source Multimedia Communication,” 4th International Conference on Intelligent Computing and Information Systems (ICICIS2009), 2009. (submitted)

## 国内ワークショップ・研究会

- 長田俊明，加藤貴司，菅沼拓夫，木下哲男，白鳥則郎，“動的ネットワークにおける無線LAN環境の構築”，長田俊明，加藤貴司，菅沼拓夫，木下哲男，白鳥則郎，第64回情報処理学会全国大会，2002.
- 長田俊明，菅沼拓夫，白鳥則郎，“エージェント指向ミドルウェアによるアドホックネットワークの動的制御”，信学技報 Vol.102, No.352, pp.49–53, 2002.
- 長田俊明，菅沼拓夫，白鳥則郎，“エージェント指向ミドルウェアを用いたアドホックネットワークの制御に関する研究”，マルチメディア，分散，協調とモバイル (DICOMO2003) シンポジウム論文集, pp.781–784, 2003.

- 長田俊明, 菅沼拓夫, 白鳥則郎, “アドホックネットワークにおける End-to-End の帯域を考慮したアプリケーションレベルのルーティング方式”, マルチメディア通信と分散処理 (DPS) ワークショップ論文集, pp.31–36, 2003.
- 長谷川大介, 高野健一郎, 長田俊明, 北形元, 菅沼拓夫, 白鳥則郎, “多人数参加型リアルタイム通信における参加者毎の品質要求に応じた資源割当と中継ツリー構築手法”, 信学技報, Vol.105, No.279, IN2005-84, pp.137–142, 2005.
- 高野健一郎, 長田俊明, 北形元, 木下哲男, 白鳥則郎, “双方向リアルタイムマルチメディア通信のための分散型 ALM ツリー反復構成手法”, 信学技報, Vol.106, No.237, IN2006-53, pp.25-29, 2006.
- Jun Makishi, Toshiaki Osada, Gen Kitagata, Norio Shiratori, “Evaluation System and Its Design of Iterative ALM Tree Construction Method for Many-to-Many Multimedia Communication,” IEICE Technical Report, Information Networks, Vol.107, No.222, pp.1–6, Sept. 2007.
- 牧志純, 長田俊明, 北形元, 木下哲男, 白鳥則郎, “反復型ミドルウェアレベルマルチキャストツリー構成法の実証と評価”, 電子情報通信学会 2008 年総合大会講演論文集, 2008 .
- 牧志純, 長田俊明, 北形元, 木下哲男, 白鳥則郎, “ノード間の通信品質の偏りを軽減する MLM ツリー構成法の評価”, 情報流通基盤分科会ワークショップ, ITRC Technical Report, 2008.
- 牧志純, 長田俊明, 北形元, 木下哲男, 白鳥則郎, “An Iterative Application Layer Multicast Tree Construction for Multi-source Multimedia Communication and its Evaluation,” 電気関係学会東北支部連合大会 ( Student Session ), Aug. 2008.

- 牧志純，長田俊明，北形元，木下哲男，白鳥則郎，“ 多人数参加型マルチメディア通信のための反復型アプリケーションレイヤマルチキャストツリー構成法の実証と実環境評価，”信学技報，IN2008-47(2008-09), pp.21–26, Sept. 2008.