

モバイル環境における一対多通信 —シミュレーションによるFTPとSRMの比較—

佐藤 隆之[†] 柳生 健吾[†] 戸川 望[‡] 大附 辰夫[†]

[†] 早稲田大学 理工学部 電子情報通信学科 〒169-0072 東京都新宿区大久保 3-4-1

[‡] 北九州市立大学国際環境工学部情報メディア工学科 早稲田大学理工学総合研究センター

E-mail: [†] {takayuki.yagyu,to}@ohtsuki.comm.waseda.ac.jp, [‡] togawa@env.kitakyu-u.ac.jp

あらまし 無線端末が存在するネットワーク環境において、マルチキャスト通信プロトコルを用いた一対多通信を実現するための問題点を明らかにする。通信状態を検討できる一つの指標としてT-N点を考案し、一対一通信プロトコルの代表であるFTPとU.C.Berkeleyで開発されたSRMを用いて、nsを用いたシミュレーション結果よりT-N点の有意性を明らかにする。

キーワード マルチキャスト, SRM, ns, トラフィック, T-N点

Multicast for Mobile—Environment —Comparison of FTP and SRM—

Takayuki SATOH[†] Kengo YAGYU[†] Nozomu TOGAWA[‡] and Tatsuo OHTSUKI[†]

[†] Dept. of Electronics and Communication Eng., WASEDA University

ADDRESS: 3-4-1 Ohkubo, shinjuku-ku, Tokyo, 169-0072 Japan

[‡] Department of Information and Media Sciences The University of Kitakyushu

Advanced Research Institute for Science and Engineering WASEDA University

E-mail: [†] {takayuki.yagyu,to}@ohtsuki.comm.waseda.ac.jp, [‡] togawa@env.kitakyu-u.ac.jp

Abstract We show issues for fruition of Multicast Transportation with Multicast Transport Protocol in the network consist of mobile station. We define the T-N POINT for one factor discussing state of transport, and we simulate with “ns” between FTP and SRM. T-N POINT will be one factor for checking network traffic. We show that from simulation.

Keyword Multicast, SRM, ns, Traffic, T-N Point

1. はじめに

現在、モバイル端末が急速に普及しており、ネットワークを用いた情報交換の主な手段は、自宅や会社などの固定端末から、移動するモバイル端末へと推移している。モバイル端末によるネットワークを介した情報交換は、音声通話やメールのみではなく、iモードに代表される情報検索やデータ配信も行われている。通信形態は一対一とは限らず、一対多または多対多の通信サービスを想定する必要性が出てきており、従来の一対一(ユニキャスト)通信やブロードキャスト通信に加え、3人以上の特定のグループによって行われる通信(マルチキャスト通信)の必要性が増してきた。現在モバイル環境におけるマルチキャスト通信プロトコルは研究されている分野である(一対一通信であるTCPを基準として拡張したMoM[7]、放送型通信方式

にエラー修復制御を付加して実現した物[12]、独自の方式により受信端末との同期を取り実現した物[3,11]、各中継ノードに大きなデータ保持機構を想定しハンドオフ時に起こるデータ損失の修復を高速化した物[13]など)。しかしどの方式もある状況に特化したものであり、モバイル環境におけるマルチキャスト通信の最良の方法は確立されていない。

多数の人に重要なデータを配信するサービスを想定する際には、受信データにエラーがないようにするため信頼性のある通信路を確立する必要性が有る。モバイル環境では、通信路は主として無線通信路であるためBER(Bit Error Rate 通信路上でのデータのエラー率)が高い。誤り訂正符号を用いて信頼性を高めようとする付加情報が非常に増える。誤り検出符号を用いて、データの再送を行う方法を用いると、受信不可

であった端末に対しての再送信の技術が必要となる。誤り検出符合を用いた場合、正常に受信した端末からは ACK、受信不可だった端末からは NACK と呼ばれる信号を送信元に返信して再送を行う方法が用いられるが、一対多通信の場合は、送信元付近の回線を ACK や NACK が占有(Ack Implosion)してしまう可能性があるからである。従来のブロードキャスト通信のように到着確認を行わない通信方式を適用すると、相手にデータが到着したかどうかを確認することができず、信頼性のあるデータ送信とは言えない。また課金システムの構築が困難なものとなる。これらの要因により、現在モバイル環境では一対一通信路を多数設けて、疑似的に一対多通信を行う方法が使用されている。しかし一対一通信路を多数設ける方法を用いた場合、同一通信路上に同じデータがある状態ができ、通信路の利用効率の観点から見ると非効率的である。また動画配信と言った大容量データ配信サービスに着目すると、ネットワークにかかる負荷は多大なものとなる。

2. 準備

本章では、本研究を紹介するにあたり、基盤となる情報について紹介する。

2.1. モバイル環境

携帯電話の普及にともない、携帯端末を付随させることによって移動中でもメールのやり取りなどを行いたいというニーズが増えている。人が持ち運んでいるという事で、これらの端末はモバイル端末と呼ばれる。本論文では、モバイル端末を考慮したネットワーク形態を“モバイル環境”としている。モバイル端末に対する通信路は無線通信路となることは言うまでもない。大きな特徴は、ノード(つまりモバイル端末)が移動するという点である。通信中にノードの位置が変わると、通信途中でルートを切り替える作業(ハンドオフ)が必要となる。

2.2. マルチキャスト通信

ユニキャスト通信における‘双方向のインタラクティブな通信であるため信頼性が高い’という特徴と、ブロードキャスト通信における‘同時に多数の人にデータを送信できる’という特徴を合わせ持つのがマルチキャスト通信(図1)である。マルチキャスト通信では、同時に特定の多数の人に信頼性のあるデータを送受信する。送信元は通信路上にデータを1つ流し、各中継点において、次の中継点、及び末端ノードに送出する為に必要な数だけデータをコピーする。ブロードキャストと異なる点は、クライアントが特定されているという点であり、グループ通信とも呼ばれる。

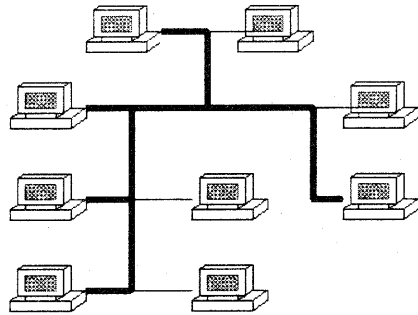


図1：マルチキャスト通信

現状で利用されている、一対一通信路を多数設けてマルチキャストを行う場合は、1つの通信路上に同一のデータが多数存在する事がありえる。同時に多数の人にデータを送受信できるブロードキャスト通信を利用したマルチキャスト通信を行うと、1つの通信路上には同一のデータは複数存在しない。そのため通信路を現在よりも効率的に使用できるようになる。

2.3. 問題点

一般的な信頼性のあるデータ転送は以下のような動作に特徴づけられる。送信元はN個の packets を受信者に送信する。受信者はN個の packets を受信したことを承認すると、送信元は $(N+1)+M$ 個の packets を送信し、受信者は再度承認する繰り返しである。受信者が最初のN個の packets の一部、もしくは全てを受信できなかった場合は、送信元に警告を出し、全てのデータを受信するまで、N個の packets の一部もしくは全てを再送信させるというフィードバックにより再送制御により信頼性のあるデータ転送を実現している。

ユニキャスト通信では、TCPが受信者から送信元に送られるACKによって、信頼性のあるデータ転送を実現している。ACKがなんらかの理由で損失した場合は、送信元はある程度の時間をおいてから再送を開始する。TCPにおいてACK信号は、送信元が情報伝達速度を調節するための情報としても使われる。スループットを妨げることなく、できるだけ早くデータの損失を回復させるため、ACKの送出方式は長年開発されてきた。

これらと同様の、信頼性のあるデータ転送方式を、マルチキャストに当てはめることは困難である。以下にその理由について述べる。

2.3.1. Data Transfer

マルチキャストセッションは一対一の通信ではなく、一つの送信元から多数の受信者に送る一対Nか、多数の送信元から多数の受信者に送るM対Nとなる。各受信者は異なる処理能力を持ち、送信ルートも様々

な種類の帯域を含むため、環境上異種間接続となっている。また、通信アプリケーションがTCP接続で一般化されているユニキャストセッションと異なり、マルチキャストセッションで唯一の共通の要素はグループアドレスのみである。

単一のRMソリューション(リライアブルマルチキャストを実現するための解法、主にマルチキャストプロトコルのこと)では、全ての要求を満たさない。一対Nの信頼性のあるデータ転送は、フィードバックや損失修復の必要性という観点において、リアルタイムのM対Nのグループ共同作業とは異なるからである。それゆえ、各アプリケーションごとに適応するリライアブルマルチキャストの方式を開発することが重要である。

2.3.2. Feedback

マルチキャスト環境の目的は、送信元の処理を最小限にし、再送要求修復パケットによるネットワーク資源の利用を軽減することである。TCPはN個のパケットが正常に到着したかどうかを送信元に連絡するため、ARQを基盤とした方法を用いる。マルチキャスト環境にこの機能を当てはめることは、システム上非常に多くのACK情報が送信元に送られる。TCPの送信元は、どのパケットが送られ、必要なのかについて、各受信者ごとの通信状態を把握する必要がある、またACK情報はすべて送信元に向かって送られるので、ネットワーク資源を占有する(ACK implosion)。

上記の現象を回避するためには、受信者からN個のパケットが受信できなかったことを示すNACKを送出させる事により送信元における処理を軽減し、再送要求による帯域の利用を最小とする方法と、送信元にACKやNACKを全て送るのではなく、中継ノードにそれらの情報を統括させることによって実現する方法がある。前者はリアルタイム性を重視しており、後者は信頼性を上げると同時に、損失修復時間も短くしている。

2.3.3. Loss Recovery

修復パケットの再送信は送信元か、近い受信者によって行われる。後者では、損失を修復するための時間を短縮することができる。近く受信者がデータを保持していない際は、その上位のノードに再送要求を送信する。

また、エラー訂正機能(FEC)を持つハイブリッドARQを利用する方法もある。基本的なFECでは、送信元はある冗長なデータBを、元データであるAに付加することにより、データにエラーが発生した際、受信者が元データになおすことができる。ハイブリッドARQでは、送信元はAパケットを送信し、ACKやNACKが受信者から戻ってくるのを待つ。もし修復パ

ケットが必要なら、送信元はAを修復するのに必要なBのパケットを送出する。Bを受け取ることにより、各受信者は元データにどのようなエラーがあろうとそれを修復することができる。元データAよりも、修復のためのBのパケットの方が小さいため、ネットワーク上のデータ量を減らすことが可能である。

2.3.4. Flow Control

グループ管理とフロー制御も考慮すべき点である。従来のIPマルチキャストモデルでは、送信元がグループメンバ情報を維持する処理をし、受信者は自由にメンバに入るか去ることができる。しかし、送信元や管理機構において、通信中に新しいメンバがグループに参入するか、グループのメンバは終始固定のものなのかを把握する必要がある。この状況により、管理方法が異なるからである。

送信元が大きな異種間接続の環境を扱う場合、フロー制御はより困難となる。解決策は、受信者の状況に応じて、階層的にデータをコード化し、個別のマルチキャストグループアドレスへ各層を送信することである。グループに多くの受信者が参加すればするほど、多くの層が作られる。この方法より、帯域が狭い人はエラーを起こさず、また十分にある人は不自由さを感じることなく通信を行える。ただし、送信元での処理が増大してしまう欠点がある。

2.4. SRM

SRMは、主要な通信手段としてIPマルチキャストを利用したもので、M対Nのグループ作業のためにU.C.BERKELEYで開発された。SRMは異なるプラットフォーム間で稼動可能である。

SRMは、アプリケーションにとって意味のあるデータ単位で信頼性の確保をおこなうべきであるというALF(Application Level Framing)の概念[5]に基づき設計されている。ユニキャスト通信を行なうTCPでは、パケットにシーケンス番号を付加して送出し、その番号により損失の検知が行なわれる。このような方式をマルチキャスト通信に適用した場合、受信側アプリケーションが主体的に再送要求を行ない、意味のある単位で信頼性の確保を行なう事ができない。ALFは、アプリケーションの制御の範囲をトランスポート層まで広げるべきであるという主張である。

SRMはネットワークが通信途中で切れてしまった場合の特別な対処法を持たない。このことから有線ネットワークを見越して作られたプロトコルである。現在キャッシュを用いてネットワーク切断時の特殊な対処法を、SRMに付加する動きがある。しかしネットワークが途切れた場合は、再接続の際に再送要求が発生するので、修復パケットを受信することによって問題を回避できる。

3. マルチキャストシミュレーション

本章では、シミュレーションをするにあたり用いたシミュレータ(ns)について述べ、続いて、T-N点なる新たな評価指標を提案する。

3.1. ネットワークシミュレータ ns

ns は、UCB/LBNL/VINT プロジェクトによるネットワークシミュレータで、以下のような特徴がある。

- 豊富な機能・プロトコル・アプリケーションモデルを標準装備しており、プロトコルについては C++ で拡張可能である。
- シミュレーション記述については OTcl(Tcl のオブジェクト指向拡張言語)で記述する。
- nam(VINT/LBNL Network Animator)によって、アニメーション表示が可能となっている。

ns は、バージョン 2.1b8 が完成版として現在リリースされており、バージョン 2.1b9 が、日々更新されている。無線移動端末をノードとしたシミュレーションは 2.1b6 以降から可能となっているが、2.1b5 と比較すると 2.1b6 にはバグが多く存在している(例えば、動的ルーティングができないと言う事が挙げられる)。本研究においては、2.1b8 を用いている。

3.2. ネットワークトポロジ

本研究におけるネットワークトポロジのターゲットは、携帯電話に代表される PDC-P 網である。トポロジ構成は Tree 状であり、最上位のノード(PGW)が外部ネットワーク(ISP など)と接続している状況とする。シミュレーションを行う上では PC のメモリなどの関係から、実際の日本全国のネットワーク網を構成する事は困難である(700 個ほどのノード間でパケット通信のシミュレーションを行っているが、URTRA SPARK 296MHz×2.1GB 以上のメモリという環境で丸一日かかったという例もある[4])。そこでシミュレーションを行う上ではある程度モデル化されたトポロジが必要となる。本研究におけるシミュレーションについて、表 1 に想定する各ノードの個数、図 2 にその概略図を示す。表 1 に示すように、全体のノード数は 100 個程度としている。

ノード	設置数
PGW	2個
PPM	2個
BS	10個／1PPM
移動端末	全体で80個程度

表 1: シミュレーションで想定するトポロジ

また、ノード間の回線速度は表 2 のように設定している。これらの値については基本的なものとし、シミュレーションの状況に応じて多少の変化はある。

ノードa	ノードb	回線速度
ISP or LAN	PGW	100Mbps
PGW	PPM	10Mbps
PPM	BS	512kbps
BS	mobile 端末	64kbps

表 2: ノード間の通信速度

ネットワークトポロジの生成については Tiers[9] を始めとして、幾つかのアプリケーションが存在し、ns との互換性を持っているが、本研究では Tcl 記述[14]によるシミュレーションスクリプト中で自分で宣言を行なった。

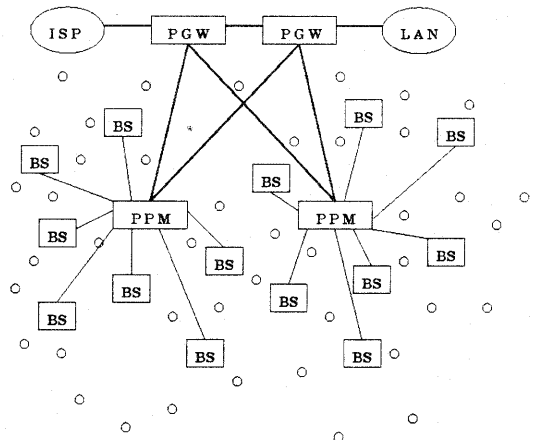


図 2: 想定するネットワーク図

3.3. 検討事項

回線利用率を検討する上で、通信状態を示すために一般的に用いられる尺度を以下にまとめておく[15]。

●トラヒック

ネットワークの混雑具合を表す尺度。ネットワーク内で単位時間に発生した「送出されるべきパケット(各端末から発生した新規のデータパケットと伝送途中に生じた誤りや消失を回復するための再送パケットの両方を含む)」の総量の時間平均を、そのネットワークの公称伝送速度で正規化したものである。トラヒックは $G(0 < G < \infty)$ で示される。

理想的なアクセス制御の下では、トラヒック値が 0~1 の範囲であれば発生したデータは必ず受信側に届く。トラヒック値が 1 を超えた場合はパケットの送達が遅れ、または一部のパケットが不達になる。

●スループット

ネットワークの利用効率を比較する尺度。トラヒックのうちで目的のクライアントまで正常に送ら

れたデータ量の平均をそのネットワークの公称伝送速度で正規化したものである。理想的なアクセス制御の下ではトラヒックが1以下の場合スループット値とトラヒック値は等しくなる。しかし実際はパケットが衝突したり、雑音による誤りが生じたりするので、スループット値はトラヒック値より小さくなる。スループットは $S(0 < S < 1)$ で示される。

● 伝送遅延

スループットと同様にネットワークの利用効率を比較する尺度。パケットの送信側で送信すべきパケットの送信準備が完了してから、それが受信側で受信完了となるまでの所用時間の事である。アクセス制御のために送信側で送信待ちとなる時間や実際の伝送に要する時間などが含まれる。

本研究では、T-N 点と言うグラフ上の交点を元にデータ比較をする。T-N 点とは、任意のノードの全データ受信完了時間(Transport Time)とネットワーク全体の受信ノード数(Number of Node)の関係グラフにおいて、一対一通信路を多数設けて一対多通信を実現する方法(ftp)とマルチキャストプロトコルを用いて一対多通信を実現する方法(SRM)の2者が交差する点である。受信ノード数が少ない時は ftp の方がデータコピーのない分、任意のノードにおける受信完了時間が早い。受信ノード数が多くなると、送信ノード付近のネットワークでデータ溢れが発生しエラー再生のため多大な遅延が発生する。送出データ量やデータ転送速度などについて T-N 点の位置を計測することにより、今後の大規模ネットワークによるデータのやり取りのトレードオフを決定することができると思われる。

4. シミュレーション結果

本章では、3章で述べた方法で行ったシミュレーションによる T-N 点について結果を示し、得られる意義について述べる。

4.1. T-N 点の調査

T-N 点を調査するために、図3にネットワーク全体における受信ノード数と、その内の1つのノード(各値については、受信ノードのうち環境の違う2~8個のノードにおいて時間を調査し、それらの平均値を用いている。ネットワーク全体の受信ノード数が同じ場合は、各ノードの受信完了時間に大きな差異は見受けられなかった)における要求データ受信完了時間の関係グラフを示す。

シミュレーションは、送信元付近のリンクの帯域を2Mbps、データ送信速度を64kbpsとしている。

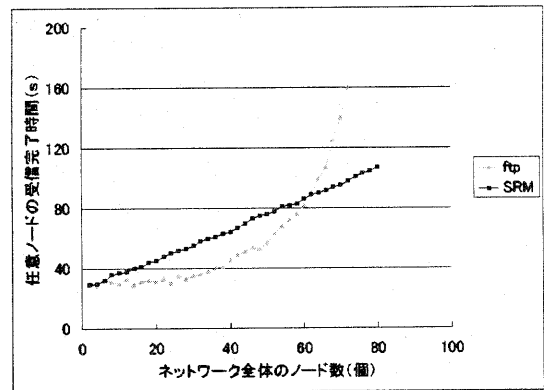


図3：T-N 点の調査

ftp では、受信ノード数が30個程度まで、各ノードの受信完了時間はほとんど変化しない。30個付近を過ぎると、受信完了時間が長くなり、その伸びは急激に増加しているのが見て取れる。本シミュレーションにおける送信元付近のリンクの帯域とデータ送信速度を用いて、

$$\begin{aligned} & 2\text{Mbps} \div 64\text{kbps} \\ &= 2048 \div 64 \\ &= 32 \end{aligned}$$

を得る。上式より ftp を用いた場合、受信ノード数が32個の際に送信元のリンクにおいてトラヒックが $G=1$ となっている。本研究におけるシミュレーションでは、理想的なアクセス制御の元にデータの送受信がされているとしているため、トラヒック $G=1$ となる、受信ノードが32個をこえた時から送信元付近のリンクにてデータ溢れが生じ、パケットロスが発生する。受信完了時間が長くなるのは、再送によりパケットロスのリカバリがされているためである。

SRM では、総受信ノード数に対する受信完了時間は線形に増加する。ftp と異なり送信元付近のリンクにおけるデータの溢れは生じないが、受信ノード数が増えるにつれ、中継ノードにおけるデータコピーに要する時間が顕著に反映しているためである。

4.2. T-N 点の推移

図3では、本論文で掲げる T-N 点は受信ノード数が60弱の際である。つまりトラヒックが $G \geq 2$ となると、SRM の方が ftp よりも受信完了時間が短くなる。上記を元に行ったシミュレーションに対しいくつか操作をし、シミュレーション結果で T-N 点がどのように変動したかを以下に示す。

1. データ送出速度を速くする・・・

T-N 点のノード数減少

2. 送信元付近の帯域を狭くする・・・

T-N 点のノード数減少

3. 受信ノードを集中させる・・・

ほとんど変化なし

4. 中継ノードの処理能力を向上する・・・

T-N 点のノード数減少

3 については、基本的にはネットワーク全体に均等に受信ノードを分散させていたが、それらを 2, 3 の基地局に集中させたトポロジ構成を取る事としている。

上記における各結果について述べる。1. 2. の操作では元シミュレーションに比べ、受信ノード数が少ない段階で送信元のリンク付近でデータ溢れが生じるため、ftp のグラフにおける立上りが早まり、T-N 点のノード数は減少する。この際、T-N 点における送信元付近のトラヒックは、元シミュレーションと同様に $G=2$ であった。3. の操作では T-N 点の位置はほとんど変化しない。中継ノードにおけるコピー量の増加により SRM における受信完了時間が全体的に長くなると同時に、ftp では送信元付近のリンクに比べたら帯域の狭い基地局付近のリンクにて、データ溢れが生じ受信完了時間がこちらも全体的に長くなっているためである。シミュレーションでは T-N 点における送信元付近のトラヒックは $G=1.7$ 程度であったが、基地局付近のリンクのトラヒックは $G=2$ であった。4. の操作により変わる点は、SRM で中継ノードにおけるデータパケットコピーの時間が減少する事である。グラフ中で言うところの、SRM の線形的に増加する関数の傾きが小さくなるため、ftp の関数との交差点である T-N 点は左方向に移動する。この際、T-N 点における送信元付近のトラヒックは $G>2$ である。

以上の事より T-N 点は、ネットワーク構成・データ送出速度・送出データ量・中継ノードの処理能力と言った様々な要素を同時に検討できる一つのファクターであると言える。

5. おわりに

本稿では、無線端末の存在するネットワーク環境におけるマルチキャスト通信についての問題点を明らかにした。また T-N 点を定義し、シミュレーションより T-N 点について、ネットワークを構成する要素や送出するデータの特質などとトラヒックとの関係式を検討する題材ともなりえることを示した。

本稿ではマルチキャストプロトコルに SRM を用いているため、マルチキャスト通信をマルチキャストプロトコルにより実現する際に T-N 点がどの位置に存在するかは不明である。しかし、ネットワーク構成やデータ特質より T-N 点を導出し、実際の運用状況と照らし合わせてネットワーク設計をする事は有効な手段であると考えている。

6. 謝辞

本研究を進めるにあたり、様々な御指導および御助言を頂きました。NTT ドコモ社の村瀬淳様、岡島一郎様、杉山一雄様、久信田雄介様に深く感謝いたします。

文 献

- [1] Christopher Metz, "RELIABLE MULTICAST - When Many Must Absolutely Positively Receive It," IEEE INTERNET COMPUTING, IEEE, JULY-AUGUST 1998..
- [2] Dave Kosiur, 菊田幸男, "マスタリング TCP/IP --IP マルチキャスト編--," オーム社, pp.157-177, 1999.
- [3] Mikio Ikeda, Minoru Okada and Tsutomu Kawai, "Point-to-Multipoint Communication Protocol on Window-Based Network Presentation System," IEICE Trans.on Info.and syst., Vol.E80-D, No2, pp.154-161, 1997.
- [4] Proceedings of IEEE Conference on Local Computer Networks, "Performance Comparison of Reliable Multicast Protocols using the Network Simulator ns-2," <http://www.telematik.informatik.uni-karlsruhe.de/~hofmann/papers/lcn98.ps.gz>.
- [5] S.Floyd et al., "A Reliable Multicast Framework for Light-Weight Sessions and Application-Level Framing," Proc.SIGComm. ACM. Sept.1995.
- [6] S.L Mills, "Network Time Protocon(Version 3)," RFC1305, March 1992.
- [7] Tim G.Harrison, Carey L.Williamson, Wayne L.Mackrell, Richard B.Bunt, "Mobile Multicast (MoM) Protocol : Multicast Support for Mobile Hosts," Advanced Network and Mobile Multimedia Lab, 1997.
- [8] UCB/LBNL/VINT project, "Network Simulator - ns(version2)," <http://www.isi.edu/nsnam/ns/index.html>.
- [9] UCB/LBR/VINT project, "Network Simulator: Topology Generation," <http://www-mash.cs.berkeley.edu/ns/ns-topogen.html>.
- [10] 池田 幹男, 岡田 稔, 河合 励, "一対多同報通信プロトコルとその性能評価," CMN, pp.1-7, 1997.
- [11] 後藤敏, 阪田史郎, "ポイント図解式 モバイル・コンピューティング教科書," 株式会社アスキー, 1999.6.
- [12] 佐藤 文明, 辻 順一郎, 水野 忠則, "放送型ネットワークにおける通信 エラー対処方式の提案," 情報処理学会研究報告 モバイルコンピューティング, pp.71-77, 1999.
- [13] 田島佳武, 森川博之, 青山友紀, "適応型キャッシュを用いたリアルタイムマルチキャストプロトコル," 情報処理学会, マルチメディアと分散処理 92-11.
- [14] 西中芳幸, 石曾根信, "Tcl & Tc ツールキット," ソフトバンク株式会社, 1995.
- [15] 山内雪路, "モバイルコンピュータのデータ通信," 東京電機大学出版局, pp.50-51, 1998.3.