The University of Osaka
Institutional Knowledge Archive

Title	マルチキャスト通信における転送制御に関する研究
Author(s)	野口, 拓
Citation	大阪大学, 2004, 博士論文
Version Type	VoR
URL	https://hdl.handle.net/11094/617
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

マルチキャスト通信における
転送制御に関する研究

野口　拓
大阪大学大学院工学研究科通信工学専攻

2004 年 1 月

謝辞

本論文は，筆者が大阪大学大学院工学研究科通信工学専攻博士後期課程において行った「マルチキャスト通信における転送制御に関する研究」の成果をまとめたものである．本論文をまとめるにあたり，筆者は大阪大学産業科学研究所教授元田浩博士，大阪大学大学院教授北山研一博士，同大学院助教授山本幹博士に有益なる御教示，御助言を賜った．ここに深甚なる感謝の意を表す．

本研究を遂行するにあたり，全過程を通じて懇切丁寧なる御指導，御鞭撻を賜った大阪大学大学院元教授池田博昌博士（現在東京情報大学教授）に深甚なる感謝の意を表す．

筆者の大学院在学中，講義等を通じて通信工学全般，及び本論文をまとめるにあたって多くのご指導を賜った，大阪大学大学院教授小牧省三博士，同教授塩澤俊之博士，同教授河崎善一郎博士，同教授馬場口登博士，同元教授森永規彦博士をはじめとする諸先生方に衷心より御礼申し上げる．

逝去された大阪大学大学院元教授前田肇博士には，御懇篤なる御指導，御鞭撻を賜った．ここに厚く御礼申し上げる．

通信網工学領域に配属以来，大阪大学大学院助手松田崇弘博士ならびに後藤嘉代子技官には，公私にわたりご厚意溢れるご助言とご支援を戴いた．ここに厚くお礼申し上げる．

また，折に触れて熱心な御討論と有益な御助言，御協力を戴いた山口誠氏 (現在富士通株式会社)，山本和徳氏 (現在 NTT ドコモ株式会社)，杉園幸司氏 (現在 NTT 株式会社)，萩原達也氏 (現在松下電器産業株式会社)，三浦浩一氏（現在和歌山大学助手），永田晃氏 (現在富士通研究所株式会社) をはじめとする卒業生，佐野健氏をはじめとする大阪大学大学院工学研究科通信工学専攻通信網工学領域の諸兄に感謝申し上げる．

最後に，寛大なる心をもって惜しみない援助と理解を頂いた家族に感謝を捧げる．

内容梗概

本論文は，筆者が大阪大学大学院工学研究科（通信工学専攻）在学中に行ったマルチキャスト通信における転送制御に関する研究成果をまとめたもので，以下の6章より構成される．

第1章は，序論であり，本論文の背景となる研究分野に関して現状を述べ，本研究の位置づけを明らかにする．

第2章では，一対多通信をサポートするマルチキャスト通信の概要を述べる．マルチキャスト通信は，送信されたデータが途中の中継ノードにおいて複数個に複製され多数の受信者へ転送される通信形態で，特定多数の受信者へデータを配信するのに適している．マルチキャスト通信の実現方法としては，マルチキャスト転送制御をエンドホストに実装するアプリケーションレベルマルチキャストと，ルータに実装するネットワークレベルマルチキャストの2つの方式が検討されている．本章では，これら2つの方式について詳しく述べた後，マルチキャスト通信のインターネットへの普及過程におけるそれぞれの位置づけを明確にする．

第3章では，アプリケーションレベルマルチキャストにおけるツリー構築法を検討する．ここでは，アプリケーションレベルマルチキャストのエンドホスト障害問題に焦点を当てる．アプリケーションレベルマルチキャストでは，エンドホストにマルチキャスト転送制御が実装されるため，頻発するエンドホストの障害が問題となる．そこで本章では，エンドホスト障害時に影響を受ける受信者数を抑制するアプリケーションレベルマルチキャストツリー構築法を提案し，耐障害性の高いアプリケーションレベルマルチキャストによる効率的な一対多通信の実現を図る．また，性能評価により，提案方式の有効性を明らかにする．

第4章では，ネットワークレベルマルチキャストの信頼性保証に着目する．ネットワークレベルマルチキャストの信頼性保証を実現する信頼性マルチキャストでは，再送制御に伴い発生する制御パケットの送信者への集中（Feedback implosion）が最重要課題となる．

そこで本章では，誤り訂正符号 (FEC) を使用した，再送制御の補助をネットワーク内で部分的に行う方式を提案し，効率的な信頼性マルチキャスト通信の実現を図る．また，性能評価により，提案方式の有効性を明らかにする．

第 5 章では，ネットワークレベルマルチキャストにおけるトラヒックの負荷分散に主眼を置く．ここではまず，ネットワークコーディングがトラヒックの負荷分散の点で有効なマルチキャスト転送制御であることを明らかにし，負荷分散効果を得るネットワークコーディングの新たな利用方法を提案する．また，ネットワークコーディングを用いたマルチキャスト通信の性能評価を行い，提案方式の有効性を明らかにする．最後に，ネットワークコーディングを用いたマルチキャスト通信の実装問題について詳細に検討する．

第 6 章は，結論であり，本研究で得られた結果の総括を行う．

目次

第 1 章　序論　1

第 2 章　マルチキャスト通信　5
- 2.1　緒言 5
- 2.2　マルチキャスト通信とは 5
- 2.3　ネットワークレベルマルチキャスト 8
- 2.4　アプリケーションレベルマルチキャスト 10
- 2.5　マルチキャスト通信の普及過程 18
- 2.6　結言 19

第 3 章　アプリケーションレベルマルチキャストにおけるロバストなツリー構築法　21
- 3.1　緒言 21
- 3.2　アプリケーションレベルマルチキャストにおけるエンドホスト障害問題 . 22
- 3.3　エンドホスト障害に対しロバストなアプリケーションレベルマルチキャストツリー 23
- 3.4　性能評価 30
- 3.5　結言 31

第 4 章　信頼性マルチキャストにおけるマルチキャスト再送制御　35
- 4.1　緒言 35
- 4.2　信頼性マルチキャストの概要 36
- 4.3　マルチキャストバックボーンにおけるパケットロスの相関性 38
- 4.4　局所的に FEC を適用した信頼性マルチキャストプロトコル 40
- 4.5　遅延解析 42
- 4.6　シミュレーション 54
- 4.7　結言 55

第 5 章　ネットワークコーディングを用いたマルチキャスト転送制御　61

5.1 緒言 61
5.2 ネットワークコーディングの概要 62
5.3 ネットワークコーディングを用いたマルチキャスト転送制御の有効性 . . . 65
5.4 性能評価 69
5.5 実装問題 76
5.6 結言 79

第 6 章 結論 81

略語表 85

略語表 85

参考文献 87

本論文に関する原著論文 93

図目次

2.1 One-to-many communications by using multiple unicast transmissions. . . . 6
2.2 Multicast communication. 7
2.3 The overview of application-level multicast. 12
2.4 Narada protocol. 14
2.5 Host multicast tree protocol (degree constraint = 3). 15
2.6 Hierarchical arrangement of endhosts in NICE. 16
2.7 Multicast tree in NICE (A_4 is the sender). 16
2.8 Hybrid architecture between IP multicast and application-level multicast. . . 19

3.1 Endhost failure causes tree partitions. 23
3.2 Impact of a single host failure on the connectivity of an application-level multicast tree. 26
3.3 Join process (degree constraint = 3). 27
3.4 Signaling messages between endhosts. 28
3.5 Tree robustness against endhost failure (**Waxman**). 33
3.6 Tree robustness against endhost failure (**BRITE**). 33
3.7 RDP performance (**Waxman**). 34
3.8 RDP performance (**BRITE**). 34

4.1 Sender-initiated protocol and receiver-initiated protocol. 36
4.2 NAK protocol. 38
4.3 Packet loss in multicast backbone. 39
4.4 Reliable multicast protocol applying local FEC. 41
4.5 Network model. 43
4.6 Packet flows under local FEC. 45
4.7 Delay components of local FEC. 48

4.8 Normalized delay performance vs. number of receivers ($p_s = p_d = 5\%, \lambda = 0.125, \tau = 50msec, \tau_s = 15msec$). 57
4.9 Normalized delay performance vs. number of receivers ($p_s = p_d = 5\%, \lambda = 0.125, \tau = 50msec, \tau_s = 15msec, \frac{n}{k}$ is fixed to $\frac{5}{3}$). 57
4.10 Bandwidth usage at the source link vs. FEC parameter n ($p_s = p_d = 5\%, k = 7, R = 300, \lambda = 0.125, \tau = 50msec, \tau_s = 15msec$). 58
4.11 Tiers model. 58
4.12 Normalized delay performance vs. number of receivers ($p_s = 5\%$). 59
4.13 Bandwidth usage reduction for NAK vs. number of receivers ($p_s = 5\%$). . . . 59
4.14 Overall bandwidth usage reduction vs. number of receivers ($p_s = 5\%$). 60

5.1 Max-flow in multicast communication. 64
5.2 Transmission with max-flow. 65
5.3 Comparison with existing approaches. 66
5.4 Multicast capacity performance. 71
5.5 Throughput performance vs. background traffic rate. 73
5.6 Normalized delay performance vs. background traffic rate. 75
5.7 Multicast architecture applying network coding. 77

第 1 章

序論

1960 年代後半，アメリカが軍事目的で運用を開始したパケット交換ネットワーク [1] は，TCP（Transmission Control Protocol）/IP（Internet Protocol）[2, 3] を基盤として多種多様なネットワークを相互接続するインターネットへと発展した．1990 年代に新たなアプリケーションとして WWW（World Wide Web）が登場すると，インターネットは軍事・学術研究機関のみならず一般家庭にも急速に普及し，現在では世界規模のネットワークへと発展を遂げた．

TCP/IP を利用した通信では 1 対 1 通信（ユニキャスト）が基本であり，現在インターネットにおいては，WWW や FTP (File Transfer Protocol)，TELNET (TELecommunication NETwork）などの 1 対 1 通信アプリケーションが主に利用されている．しかし，近年のコンピュータの高性能化やネットワークの高速大容量化などによるインターネットの通信環境の飛躍的向上に伴い，ソフトウェア配布や電子新聞配信，インターネット TV，ビデオ会議など 1 対 1 通信形態に代わり 1 対多あるいは多対多通信形態をとる新たなアプリケーションへの需要が急速に高まりつつある．数百から数千の受信者を相手に同一の情報を配信する 1 対多通信アプリケーションをユニキャストを用いて実現した場合，同一リンクに同じデータが複数流れるため有限であるネットワーク資源を無駄に消費することになり，非効率的な通信となる．これを解決する技術としてマルチキャスト通信がある．これは，送信されたデータを途中の中継ノードにおいて複製しながら多数の受信者へ送る通信形態で，特定多数の受信者へデータを転送するのに適する．インターネットにおいてマルチキャスト通信を実現する方法として，マルチキャスト機能をネットワーク層（ルータ）に実装するネットワークレベルマルチキャストと，アプリケーション層（エンドホスト）に実装するアプリケーションレベルマルチキャストの 2 つの実現方法が考えられる．

エンド・ツー・エンド議論 [4] によると，通信に関する機能は出来るだけ上位層に実装することが望ましく，下位層に実装しても良いのは，性能利得がその実装コストよりも大きい場合だけである．Steve Deering，David Cheriton らは，パケットの複製・転送機能の

みをネットワーク層に実装し，信頼性保証などのアプリケーション毎に最適化方法が異なる機能はトランスポート層以上に実装すれば，エンド・ツー・エンド議論に反しないと主張し，1980年代後半にマルチキャスト通信の実現方法としてネットワークレベルマルチキャスト（IPマルチキャスト）を提案した [5, 6].

IPマルチキャストでは，パケットの複製・転送などのマルチキャスト転送制御はネットワーク層（ルータ）に実装される．複数の受信ノードに対応するマルチキャストグループアドレスと呼ばれる論理的なアドレスを利用し，送信ノードはマルチキャストグループを指定してパケットを送信する．送信ノードが送出したパケットは，送信ノードを根とし全受信ノードを葉，ルータを中間節点とするツリー構造の転送経路 (マルチキャストツリー) 上を，分岐点に相当するルータにおいて必要な分だけ複製されて転送される．このため，ユニキャストを用いて全受信ノードへデータを転送する場合に比べ，ネットワーク資源の効率的運用が可能となる．しかし，IPマルチキャストはそのアーキテクチャに本質的に起因する技術的問題を多く抱えている [7]．例えば，信頼性保証の困難さ，特定リンクへのトラヒックの集中などの問題である．IPマルチキャストでは，マルチキャストグループに属する受信ノードが存在するか否かという情報管理をルータが行い，送信ノードは受信ノードを把握していないため，TCPのように送受信ノード間のコネクションを確立することが技術的に不可能である．このため，再送制御が全くサポートされず信頼性の高い通信を提供することが出来ない．また，IPマルチキャスト配信経路においては，特定リンクにトラヒックが集中する傾向がある．特に，PIM-SM（Protocol Independent Multicast Sparse Mode）[8]，CBT（Core-based Trees）[9] 等の共有木型ルーチングプロトコルによりマルチキャストツリーが形成される場合には，共有木の根に位置するコアルータ近隣のリンクにトラヒックが集中し，ネットワーク輻輳を招く．これらの問題は，未だ完全な解決には至っておらず，IPマルチキャストの本格的な普及を妨げる要因となっている．

一方，近年のCPU（Central Processing Unit）の高速化，メモリの大容量化，ネットワークの広帯域化によって，アプリケーションレベルマルチキャストによるマルチキャスト通信の実現が可能となってきた．ネットワークレベルマルチキャストとは異なりアプリケーションレベルマルチキャストでは，パケットの複製・転送などのマルチキャスト転送制御がアプリケーション層（エンドホスト）に実装される．送信ノードが送出したパケットは，送信ノードを根としエンドホストを中間節点，葉とするツリー構造の転送経路 (アプリケーションレベルマルチキャストツリー) 上を，分岐点に相当するエンドホストにおいて必要な分だけ複製されて転送される．エンドホスト間の通信にはユニキャストが用いられ，ツリーに沿ってユニキャストの連結によってパケットが最終的にエンドホストまで伝送されるため，IPマルチキャスト未対応のネットワークでもマルチキャスト通信が実現可能となる．しかし，アプリケーションレベルマルチキャストでは，冗長転送経路による遅延の増加，重複パケットによる帯域の浪費，安定性の低いエンドホストの障害 (ホストや

接続リンクの故障，アプリケーションの不正終了など) に起因するマルチキャストツリーの分割などが問題となる [10]，[11]．特に，ツリーの分割が発生すると，分割によりアプリケーションレベルマルチキャストセッションから強制的に離脱させられた全エンドホストでバーストロスが起こり，マルチキャスト通信そのものが維持できなくなる．このため，エンドホスト障害問題はアプリケーションレベルマルチキャストの重要な技術課題として認識されており，アプリケーションレベルマルチキャスト普及に向けての大きな課題となっている．

インターネットにおけるマルチキャスト通信の普及過程としては，まず，インフラストラクチャのサポートを必要としないため，迅速なインターネットへの展開が可能であるアプリケーションレベルマルチキャストを用いてマルチキャスト通信を実現し，その後，徐々にネットワークレベルマルチキャストで代替してゆくことが望ましい．しかし，ネットワークレベルマルチキャスト，アプリケーションレベルマルチキャストともに，上述したようなマルチキャスト転送制御の実装方法の違いに起因する様々な技術課題が残されており，効率的な 1 対多通信を実現するには至っていない．

そこで本論文では，マルチキャスト通信の普及の阻害要因となっている，ネットワークレベルマルチキャストおよびアプリケーションレベルマルチキャストの技術課題を検討し，マルチキャスト通信の本来の目的である効率的な一対多通信の実現を図ることを目的とする．なお，インターネットでは現在，IP プロトコルが IPv4（IP version 4）から IPv6（IP version 6）に移行しつつあるが，本論文で検討する主要な技術は，いずれのバージョンにも適用可能であることを前提としている．

本論文の構成を示す．2 章では，マルチキャスト通信の概要を述べる．3 章では，エンドホスト障害に対し堅牢なアプリケーションレベルマルチキャストツリーの構築法を提案する．4 章では，ネットワークレベルマルチキャストの信頼性保証に着目し，効率的な再送制御を行う信頼性マルチキャストプロトコルを提案する．5 章では，ネットワークレベルマルチキャストの負荷分散を実現する，ネットワークコーディングの新たな利用法を提案する．最後に 6 章で，本論文で得られた成果を総括し，結論とする．

第2章

マルチキャスト通信

2.1 緒言

近年，インターネットが急速に発達していく中で，注目を集めている通信形態として特定多数の受信者にデータを配送する1対多通信，もしくは多対多通信がある．インターネット利用者の急速な拡大や，ブロードバンド・アクセス・ネットワークの普及にともなう動画像や音楽等大容量コンテンツの流通量増加により，インターネット上の大幅なトラヒックの増加が予想され，ネットワーク資源の効率的な利用による対応が求められている．こうした状況下で，ネットワーク資源を効率的に利用して1対多通信，もしくは多対多通信を実現する通信方式としてマルチキャスト通信の重要性が認識されている．

本章では，本論文のメインテーマであるマルチキャスト通信について述べる．まず，マルチキャスト通信の概要を述べた後，その実現方法として検討されているネットワークレベルマルチキャストおよびアプリケーションレベルマルチキャストを紹介する．そして，これらの2つの方式によって，マルチキャスト通信がどのようにインターネットへ普及していくかについて述べる．

2.2 マルチキャスト通信とは

コンピュータネットワークの分野は，アメリカ国防総省が出資して構築した共同研究用のネットワークであるARPANETから始まり，現在全世界に広がっているインターネットを代表に飛躍的に発展している．インターネットが開発された当初，転送する情報は主に文書であったが，コンピュータが発達するにつれ，コンピュータ上で個人が簡単に画像や音声等の大容量データを扱えるようになってきた．この流れを受けて，無線で行われていた動画や音声の配信サービスや，CD-ROM等のパッケージメディアで配布されていた大容量ファイルの配信サービスなどをインターネット上で実現することへの要望が高まっ

てきた．これを受けて考え出されたアプリケーションとして，番組を視聴希望者に配信するインターネット TV や，ソフトウェアの配布などが存在する．これらのアプリケーションの特徴として，特定多数の受信ノードがデータを受信することが挙げられる．この通信形態は，1 対多通信，もしくは多対多通信と呼ばれる．1 対多，ならびに多対多通信は同一情報のデータを多数の受信ノードに一斉配送することを目的するものであり，上記の他には，テレビ会議，分散データベースの同期，オンラインゲームなどがアプリケーションとして挙げられる．コンピュータネットワーク上でデータ通信を行う際には，インターネットで使用され現在多くの OS（Operating System）に実装されている TCP/IP をネットワーク層以上で用いるのが一般的となっている．TCP/IP を利用した通信では，1 対 1 通信が基本である．これにより，特定多数の受信ノードにデータを配送する方法として，最初に 1 対 1（ユニキャスト）通信を利用したユニキャスト主体の方式が考案された．その概念図を図 2.1 に示す．

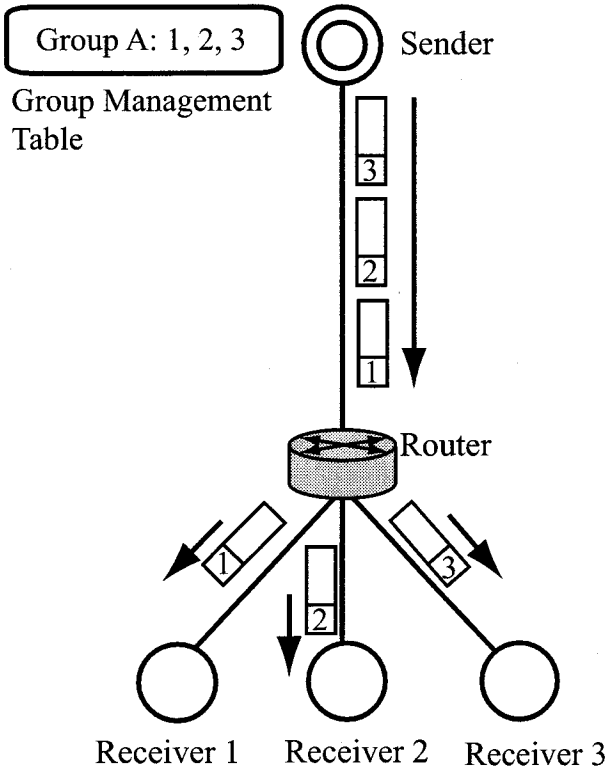


図 2.1: One-to-many communications by using multiple unicast transmissions.

ユニキャスト主体の方式では，送信ノードが受信ノード毎に用意したデータを，直接受信ノードのもとへ送信する．この方式では，送信ノード側で同一データを受信ノードの数だけ複製した後，各受信ノードに向けてデータを転送する．1 人の受信ノードに対しデータを送信する場合と比較して．送信ノード付近の帯域使用率は受信ノードの数だけ増加す

るため，ネットワーク資源を大きく消費する．1 対多通信では，動画や音声などファイルサイズの大きいデータが主に転送されるため，ネットワーク資源の消費量は膨大なものとなる．そこで，送信されるデータ量を削減する方法として，送信ノードと各受信ノードとの間に設定される経路のうち，重複する区間が存在する点に着目した方法が考案された．転送経路の形状は送信ノードを根とした木構造をなすため，受信ノードへのデータ配布は経路の分岐点に相当する中継ノードにおいてデータを複製し，それぞれの経路に対し配送することで実現される．これより，送信ノード側で受信ノード数分データを用意することを避け，ネットワーク帯域使用量の削減を実現する．これがマルチキャスト通信の原理である．

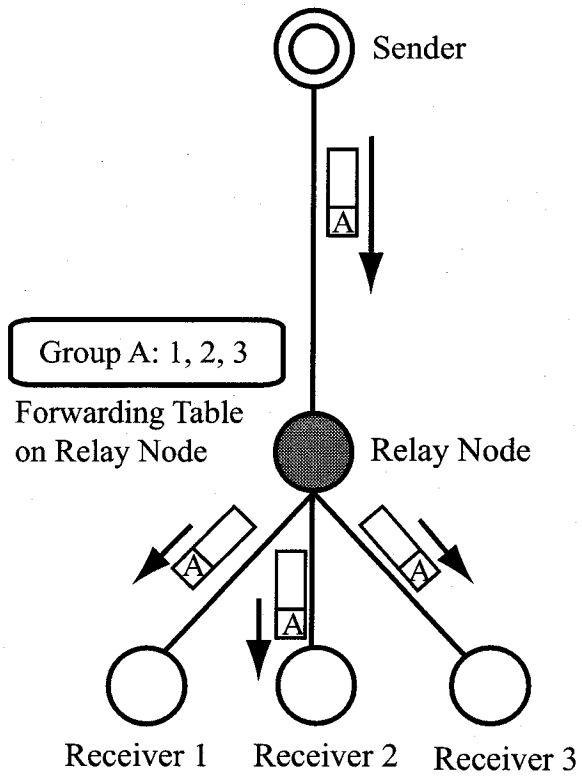


図 2.2: Multicast communication.

マルチキャスト通信の概要図を図 2.2 に示す．マルチキャスト通信では，マルチキャストグループと呼ばれる受信ノードグループにデータを転送する．マルチキャストグループは，マルチキャストアドレスによって一意に識別される．マルチキャストアドレスは，ノードの物理的な位置を示すユニキャストアドレスとは異なり，論理的なアドレスである．送信ノードは，マルチキャストアドレス宛にデータを送信し，データはあらかじめ計算された木構造を持つ転送経路上を流れる．転送経路の分岐点に相当する中継ノードは，それぞれの下流に存在する受信ノードにデータを転送するため，下流にのびる枝の数だけ

データを複製する．これにより，送信ノード側では1人の受信ノードにデータを転送する場合と同様の負荷で特定多数の受信ノードにデータを転送することが可能となる．マルチキャストアドレス宛に送信されるデータの受信を希望するホストは，マルチキャストグループへの参加要求を，マルチキャストメンバの管理を行うノードに伝えることで受信動作を実現する．逆に，データの受信が不要となった場合，受信ノードは該当グループからの離脱要求をマルチキャストメンバの管理を行うノードに伝える．これらの動作により，データの受信を希望するホストのみでマルチキャストグループは構成される．

2.3 ネットワークレベルマルチキャスト

前節で述べたように，マルチキャスト通信を実現するためには，中継ノードにおけるデータの複製と転送，ならびにマルチキャストメンバの管理が必須である．これらのマルチキャスト関連機能を実装する層としては，ネットワーク層（ルータ）とアプリケーション層（エンドホスト）の2つが考えられる．エンド・ツー・エンド議論によると，通信に関する機能は出来るだけ上位層に実装することが望ましく，下位層に実装しても良いのは，性能利得がその実装コストよりも大きい場合だけである．1980年代後半にSteve Deering，David Cheritonらは，マルチキャスト通信の実現方法としてネットワークレベルマルチキャストが最も望ましいと主張し，IPマルチキャストを提案した．IPマルチキャストでは，ルータが中継ノードとなり，データの複製・転送を行う．また，各ルータは配下の受信ノードのメンバ参加状況を検知し，ルータ間でそれらの情報を交換することでメンバ管理を行う．

2.3.1 IPマルチキャストの機構

ここでは，IPマルチキャストにおけるデータの処理手順を各操作毎に分けて述べる．

送信処理

1. IPデータグラムの宛先アドレスとして，クラスD(224.0.0.0～239.255.255.255)のマルチキャストアドレスを設定する．
2. ローカルのネットワークインターフェースに対して，データリンクレベルのマルチキャストを実行する．
3. ローカルのサブネット上のホストと，そのサブネットに接続されているマルチキャスト対応ルータに，そのデータグラムが到達する．
4. ローカルのサブネット外にそのマルチキャストグループに属するホストが存在する

場合，ルータはローカルのサブネット外にデータグラムを送信する．

5. ローカルのサブネット外から送信されたデータグラムを受信したルータは，必要に応じてローカルのサブネットにデータグラムをマルチキャストする．
6. 以下，3，4の操作を繰り返す．このように，次々にルータがそのデータをリレーしていき，最終的にネットワーク全体に送信される．
7. 実際に送出する際には，マルチキャストが伝搬する範囲を適切に指定する必要がある．そのため，IP データグラムが中継されるルータの数を制限する TTL（time to live）を使用し，伝搬する範囲を制限する．

受信処理

- 受信した IP データグラムの終点アドレス (destination adress) が，受信したインターフェースに登録されている受信すべきマルチキャストグループアドレスの場合
  1. 通常の IP データグラムと同様の受信処理を行なう．
  2. 正しく受信された IP データグラムは，ヘッダの上位層プロトコルフィールドが調べられ，それに基づき，IP データグラムのデータ領域の処理を上位層に任せる．
- 受信したインターフェースに登録されている受信すべきマルチキャストグループのアドレスでない場合
  1. 直ちにそのデータグラムを破棄する．

この際，正しくデータを受け取るためには IP 層は少なくとも，そのホストが受信すべきマルチキャストグループのクラス D アドレスのリストを管理しなければならない．このような，マルチキャストグループに関する情報を取り扱うプロトコルは IGMP(Internet Group Management Protocol) として規定されている [6, 12, 13].

2.3.2 IP マルチキャストの問題点

IP マルチキャストは，ユニキャストを用いる場合と比較してネットワーク資源を効率的に運用して 1 対多通信を実現可能であり，有効なネットワークレベルマルチキャストとして広く認識されている．しかし，IP マルチキャストは，ドメイン間ルーチングプロトコル，信頼性保証，トラヒックの集中，インフラストラクチャのサポートの必要性などの点で多くの課題を抱えている．

現在製品化されているマルチキャスト対応ルータの多くは，DVMRP[14, 15], MOSPF[16]，および PIM-SM[8] の 3 種類のマルチキャストルーチングプロトコルを実装

している．これらはドメイン内ルーチングプロトコルとして設計されており，それぞれルーチングプロトコルの異なるドメイン間のルーチングには適していない．インターネット全体でIPマルチキャストを運用するためには，ドメイン間ルーチングプロトコルが必要不可欠であり，現在M-BGP[17]，BGMP[18]をはじめ多くのドメイン間ルーチングプロトコルの研究開発が行われている．しかし，既存のプロトコルとの相互接続性，プロトコルの安定性など様々な技術課題があり，決め手となるドメイン間ルーチングプロトコルが無いのが現状である．

また，上記のマルチキャストルーチングプロトコルでは，主に距離を基準として経路の選択がなされるため，遅延の小さい特定のリンクが転送経路として選択される傾向が強く，マルチキャストトラヒックがその特定リンクへ集中する．ネットワークのトラヒック分布に偏りが発生すると輻輳が引き起こされ，他のトラヒックに多大な悪影響を及ぼす．このため，トラヒックの負荷分散を実現するマルチキャストトラヒックエンジニアリング技術[19]の早急な確立が待たれている．

IPマルチキャストでは送信ノードが全受信ノードを把握していないために，送・受信ノード間のコネクションを確立することが技術的に不可能である．そのため，ユニキャストにおけるTCPのように再送制御や輻輳制御を行い信頼性の高い通信を実現することは不可能である．近年，IPマルチキャストにおいてTCPのような機能を提供するトランスポート層プロトコルとして信頼性マルチキャストプロトコル[20, 21, 22, 23, 24]の開発が進められているが，まだ実用化されるには至っていない．

IPマルチキャストでは，ルータがデータの複製・転送などのマルチキャスト転送制御を行うため，ルータがIPマルチキャストに対応している必要がある．しかし，これらの問題点から実際にIPマルチキャストを運用しているネットワーク事業者は一部分にとどまり[*1]，現状では，すべてのインターネット利用者がIPマルチキャストを利用することはできない．この事実も，マルチキャスト未導入のネットワーク事業者がマルチキャスト対応ルータの導入に踏み切れない理由の一端となっており，IPマルチキャストの普及への足枷となっている．

2.4 アプリケーションレベルマルチキャスト

実用化されているネットワークレベルマルチキャストであるIPマルチキャストには，2.3.2で述べたように，数多くの技術課題が未解決のまま残されているため，近年ではマ

[*1] 現在，製品化されているIPルータの大部分は，IPマルチキャストに対応している．しかし，多くのネットワーク事業者は，IPマルチキャストルーチングプロトコルの機能を停止させているか，もしくはマルチキャストパケットの受信拒否設定をしており，IPマルチキャスト対応ルータとして機能しているIPルータは一部分となっている．

ルチキャスト関連機能をアプリケーション層に実装するアプリケーションレベルマルチキャストへの注目が高まっている．

2.4.1 アプリケーションレベルマルチキャストの概要

アプリケーションレベルマルチキャストでは，エンドホストがマルチキャスト関連機能を提供する．アプリケーションレベルマルチキャストでは，マルチキャストに参加するエンドホストによって論理的なオーバレイネットワークが形成され，このオーバレイネットワーク上でツリー状の転送経路 (マルチキャストツリー) が構築される．転送経路の分岐点に位置するエンドホストがパケットを複製・転送し，エンドホスト間の通信にはユニキャストが用いられる．このため，特定のインフラストラクチャを必要とせず，IP マルチキャスト未対応のネットワークにおいてもマルチキャスト通信を実現することが可能である．また，ユニキャストトランスポートプロトコルの制御機能（信頼性保障，フロー・レート制御など）が利用可能であるなどの利点を備えている.,

アプリケーションレベルマルチキャストとユニキャスト，ネットワークレベルマルチキャストの違いについて例を用いて説明する．図 2.3–(a) のネットワークにおいて R1, R2 はルータ，A，B，C，D はエンドホストであり，各リンク上の数字は遅延を示している．今，A から B，C，D へデータを送ることを考える．

図 2.3–(b) はユニキャストを用いて一対多通信を実現する場合を示している．図のようにユニキャストでは送信ノードに隣接する A-R1 間のリンクにおいて同一パケットの重複が見られる．また，コストのかかる R1-R2 間のリンクにもパケットが重複し，ネットワーク資源を浪費している．

図 2.3–(c) は IP マルチキャストを用いた場合を示している．図のように IP マルチキャストでは送信ノード A はマルチキャストアドレス宛のパケットを送るだけで，ネットワーク中のルータ R1，R2 でそのパケットが適宜複製・転送され全受信ノードへパケットが伝送される．これにより，ユニキャストに見られた同一リンク上の重複パケットは無く，各受信ノードは最短経路でパケットを受けとることができる．

図 2.3–(d) はアプリケーションレベルマルチキャストによる一対多通信の例を示している．受信ノード B，C へは A からユニキャストでパケットが伝送され，受信ノード C がパケットを複製し受信ノード D へ転送する．これにより，ユニキャストに比べ送信ノード近隣のリンクにおける重複パケットは減少し，コストのかかる R1-R2 間のリンクには 1 つのパケットが流れるだけとなっている．

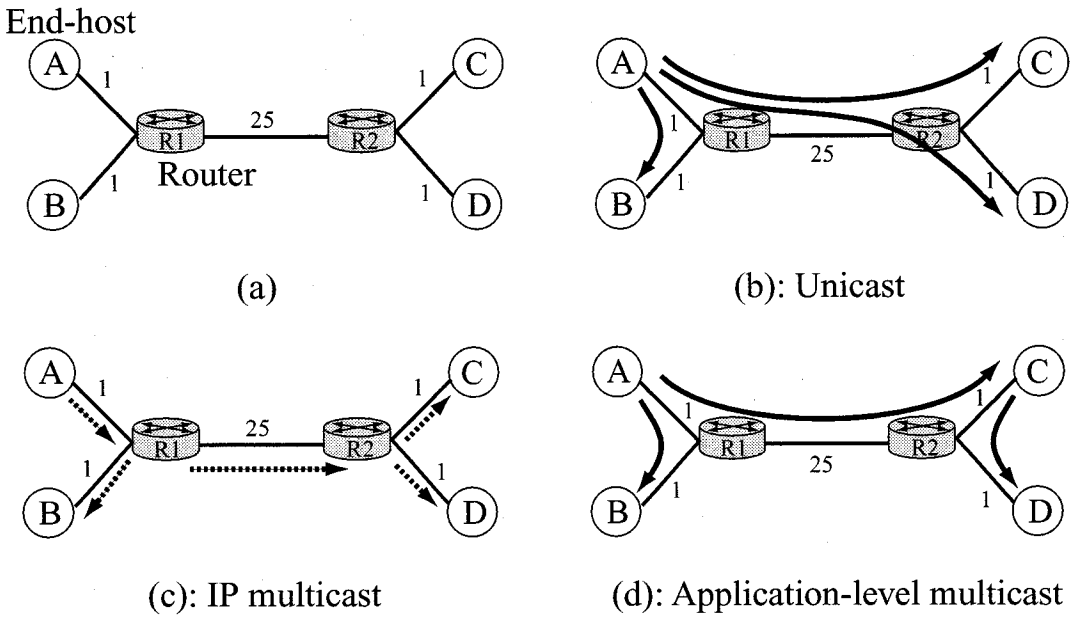


図 2.3: The overview of application-level multicast.

2.4.2 アプリケーションレベルマルチキャストプロトコル

アプリケーションレベルマルチキャストでは，マルチキャストに参加するエンドホストは制御用トポロジと転送用トポロジ（マルチキャストツリー）の2つのトポロジをオーバレイネットワーク上に構築する．各エンドホストは，制御用トポロジ上でグループ管理やマルチキャストツリーの維持・管理に必要な情報を交換し，転送用トポロジを用いて実際にデータの配信を行う．アプリケーションレベルマルチキャストプロトコルはこの制御用トポロジと転送用トポロジの構築方法に応じて，mesh-first 方式，tree-first 方式および implicit 方式の3つに分類される．

mesh-first 方式では，最初にエンドホスト間に論理リンクが張られ mesh と呼ばれるオーバレイネットワークが形成される．ここで形成された mesh が制御用トポロジとして用いられる．次に，mesh 上で既存のマルチキャストルーチングプロトコルを動作させることによりマルチキャストツリーが形成される．mesh-first 方式では，各エンドホストが自分以外の全エンドホストの情報を管理する必要があるため，スケーラビリティの点で問題が残る．しかし，ルーチングプロトコルがループ回避機能を具備しているため，ループ回避機能をアプリケーションレベルマルチキャストプロトコルにおいて提供する必要がない等の利点がある．mesh-first 方式の代表的なプロトコルとして，Narada[25]，scattercast[26] などがある．

tree-first 方式では，mesh-first 方式のように mesh からマルチキャストツリーを構築す

るのではなく，実際のネットワークから送信ノードを根，パケットを中継する受信ノードを中間節点，その他の受信ノードを葉とするマルチキャストツリーを直接構築する．また，各エンドホストは，ツリーの最適化の際に，自身の親ならびに子ノード以外とも通信を行う必要があるため，構築されたマルチキャストツリー上で隣接関係にないエンドホストに対してランダムにリンクを張り，制御用トポロジを構築する．tree-first 方式では，mesh-first 方式と比べ各エンドホストの管理する情報量が減るため[*2]，スケーラビリティが向上する．しかし，送信ノードが複数存在する場合には，各送信ノードごとにオーバレイマルチキャストツリーを維持・管理する必要があるため，多対多通信アプリケーションには適さない．tree-first 方式の代表的なプロトコルとして，HostMulticast[27]，ALMI[28] および Yoid[29] などがある．

implicit 方式では，まず階層構造などの特定の性質を持つ制御用トポロジを構築する．マルチキャストツリーは制御用トポロジ内で潜在的に定義されており，制御用トポロジ上で決められたパケット転送規則に従いパケット転送することでマルチキャストツリーが自動的に構築される．implicit 方式では，階層構造などを利用して各エンドホストの管理する情報量を低減しているため，高いスケーラビリティを実現できる．implicit 方式の代表的なプロトコルとして，NICE[30]，CAN[31] および Bayeux[32] などがある．

次に，これまでに提案されているアプリケーションレベルマルチキャストプロトコルのうち，代表的なものを上の 3 つの方式についてそれぞれ 1 つずつ紹介する．

Narada[25]

Narada はカーネギーメロン大学の Yang-hua Chu らが提案したプロトコルである．広範囲に及ぶグループ通信を行うアプリケーションのうち，特にビデオ会議やネットワークゲームなどの比較的小規模な多対多通信を行うアプリケーションを対象としている．Narada ではマルチキャストツリーは次の 2 ステップで構築される (図 2.4).

ステップ 1(mesh の形成) アプリケーションレベルマルチキャストに新しく参加するメンバーは，まず RP（Rendezvous Point）と呼ばれる特別なサーバからメンバーリストを獲得する．このメンバーリストの中からランダムにホストを選択し，mesh 上の隣接ノードとして論理リンクを張る．その後，全エンドホスト間で定期的にメンバーシップ情報及び互いの距離 (RTT) を交換し，それらの情報から論理リンクを張り直していくことで，物理ネットワークに近付くように mesh を最適化していく．

ステップ 2(tree の構築) ステップ 1 で形成された mesh 上で既存のマルチキャストルーチングプロトコルである DVMRP[14] を用いて，対応する送信ノードを根とするス

[*2] シンプルなプロトコルの場合，マルチキャストツリー上での親子関係のみを管理するだけでよい．

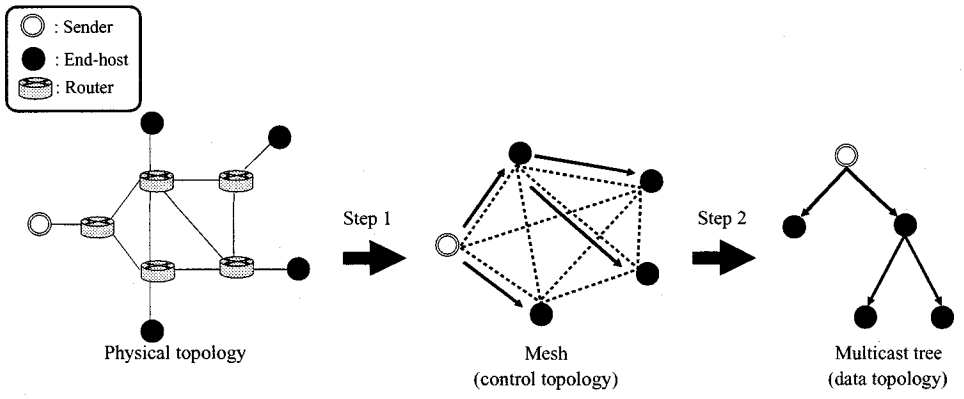


図 2.4: Narada protocol.

パニングツリーを構築する．マルチキャストデータ転送はこのツリー上の隣接ホスト間のユニキャスト転送の連結により実現される．

Narada は，マルチキャストに参加する全エンドホストが自分以外の全エンドホストに関する情報を持つ必要があり，メンバー数が増加するに従って管理する情報量が増加するため大規模マルチキャストには適さない．しかし，mesh 上でルーチングを行いツリーを構築するため，送信ノードが複数存在する場合でも mesh のみを維持・管理するだけでよく，多対多通信に適している．

Host Multicast(HMTP)[27]

Host Multicast はカリフォルニア大学ロサンゼルス校の Beichuan Zhang らが提案したプロトコルである．Host Multicast では IP マルチキャストをサポートしているドメイン (multicast island) 内のメンバーから 1 人 DM（Designated Member）を選び，異なる multicast island 間は DM 同士を UDP(User Datagram Protocol)[33] トンネリングを用いたユニキャストで結び通信を行う．各 multicast island 内の他のメンバーへは DM が IP マルチキャストを用いて情報を転送する．全 DM は Host Multicast Tree Protocol によって双方向共有木を構築する．次に，ツリーの構築方法について説明する．

新しくマルチキャストに参加するメンバー N は，HMRP（Host Multicast Rendezvous Point）と呼ばれる特別なサーバへツリーの Root を問い合わせることで Root を発見する．N はまず Root である A に対して Join Request を送る (図 2.5–(a))．図 2.5 は，各エンドホストの最大隣接ノード数 (次数制約)*3 が 3 である例となっており，この制約により N は

*3 次数制約はプロトコルパラメータであり，大きく設定すると転送ホップ数は減少するが，管理情報量が増加する．

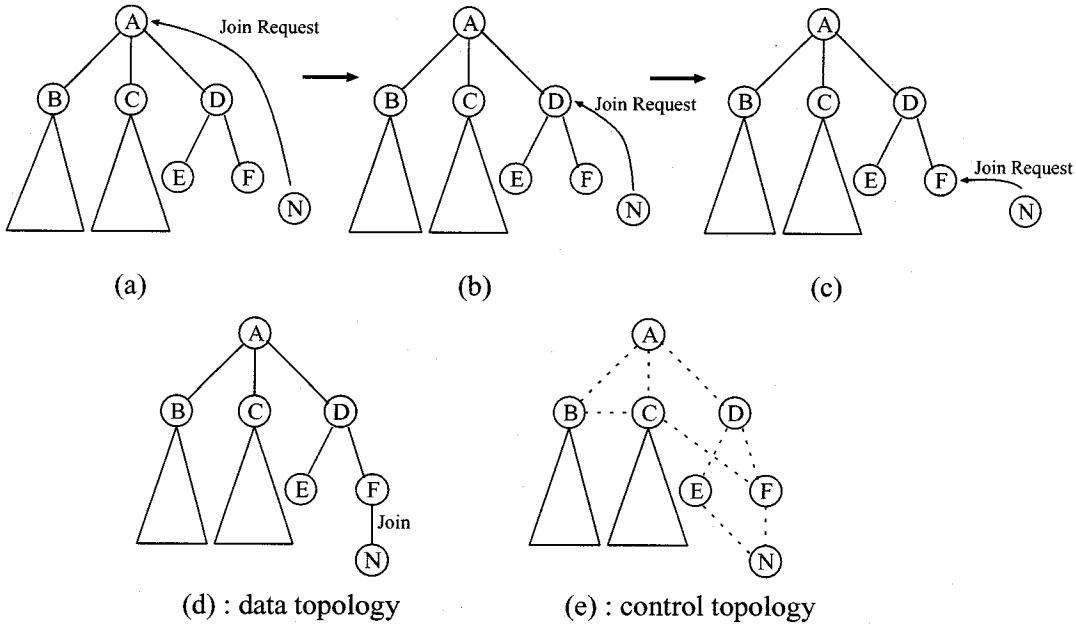


図 2.5: Host multicast tree protocol (degree constraint = 3).

A の子にはなれない．したがって，A の子の中で N に最も近いノード (この例ではノード D) に対して Join Request を送る (図 2.5–(b))．このように次数制約を超えない親が見つかるまで Join Request を繰り返し (図 2.5–(c))，最終的に図 2.5–(d) のような転送用トポロジを構成する．Host Multicast では，各エンドホスト (DM) は自分から Root までの経路上にあるエンドホストの情報を管理する．また，転送用トポロジにおける自分の親，子以外からいくつかのエンドホストを隣接ノードとして選び，転送用トポロジに冗長リンクを付加することで制御用トポロジ (図 2.5–(e)) を構成する．

NICE[30]

NICE はメリーランド大学の Suman Banerjee らが提案したプロトコルである．NICE では図 2.6 に示されるように，マルチキャストに参加するエンドホストはクラスタリング [34] によりいくつかのトポロジクラスタにグループ化され，階層構造を構成する．この階層構造によって形成されたトポロジが制御用トポロジとなる．この制御用トポロジにおいては，各クラスタ内のエンドホストはクラスタ内のエンドホスト同士でメンバーシップ情報を交換する．そのため，最下位層のエンドホストは自身が属しているクラスタ内のエンドホストの情報のみを管理するだけでよく，最上位層のエンドホストでも $O(logN)$(N:メンバ数) のエンドホストの情報を管理するだけでよい．このため，NICE は高いスケーラビリティを実現できる．NICE では，(1) 送信ノードは自身が属しているクラスタ内のエ

ンドホストに対して情報を送信し，(2) 情報を受信したエンドホストのうち，他の層にも属しているエンドホストは，他の層において属しているクラスタ内のエンドホストに対して情報を転送する．(1)，(2) よりマルチキャストツリー (図 2.7) が自動的に構築され情報配送がなされる．

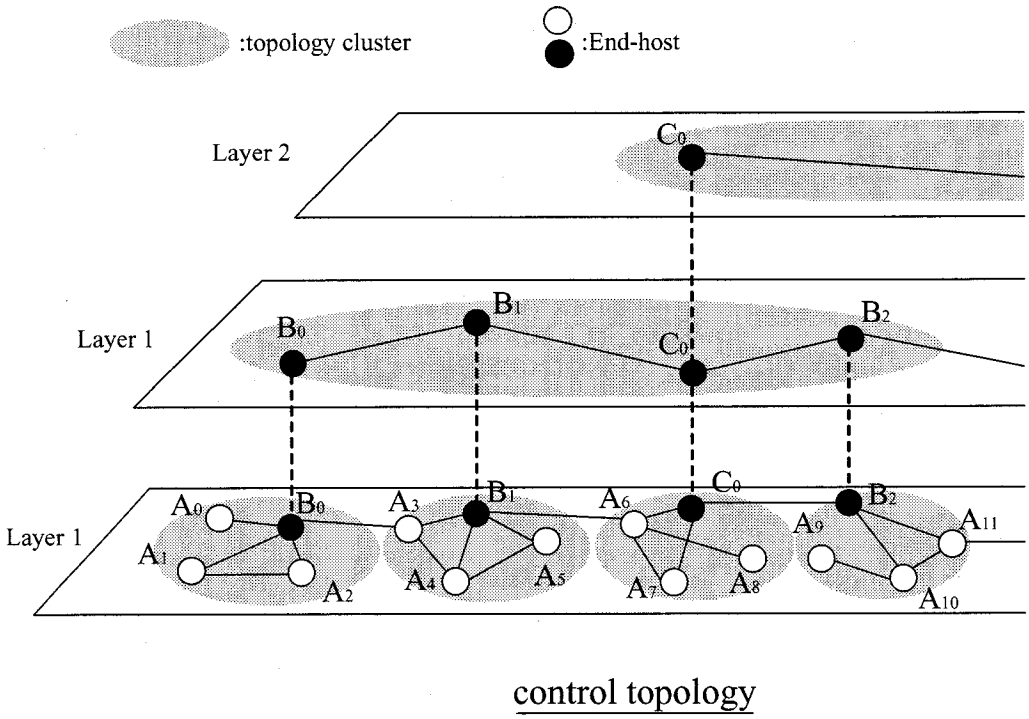


図 2.6: Hierarchical arrangement of endhosts in NICE.

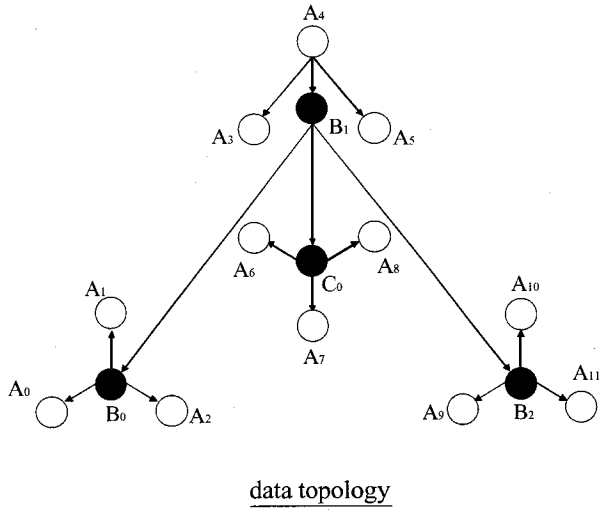


図 2.7: Multicast tree in NICE (A_4 is the sender).

2.4.3 アプリケーションレベルマルチキャストの問題点

アプリケーションレベルマルチキャストではユニキャストに比べネットワーク資源の効率的な運用が可能であり，現在のルータの機能を変更せずにマルチキャスト通信を実現可能である．しかし，エンドホストのみで構成されるオーバレイネットワークを用いて通信が行われるため，次のような 3 つの問題が必然的に発生する．第 1 の問題は，転送経路長の増加に伴う遅延の増大である．アプリケーションレベルマルチキャストでは，複数のエンドホストに中継されてデータが伝送されるため，送受信ノード間の最短経路に比べ必然的に転送経路長が長くなり遅延が増大する．例えば，図 2.3–(d) の図に示されるように，A–D 間のデータ伝送において，最短経路ではない転送経路（A → R1 → R2 → C → R2 → D）が用いられ遅延が増加する．第 2 の問題は，物理リンク上でのパケット重複による使用帯域量の増加である．アプリケーションレベルマルチキャストでは，ユニキャストトンネリングによりエンドホスト間を連結しているため，物理リンク上で同一パケットが重複する場合がある．例えば，図 2.3–(d) の図に示されるように，A-R1 間のリンクに複数の同一パケットが流れネットワーク帯域が浪費される．第 3 の問題は，エンドホスト障害によるマルチキャスト通信の途絶である．アプリケーションレベルマルチキャストでは，専用設計されたルータと比較して安定性の低いエンドホストがマルチキャスト転送制御を行うため，エンドホストの故障やアプリケーションの不正終了などのエンドホスト障害が頻発する．エンドホスト障害に起因してツリーの分割が発生し，送信ノードから切り離された受信ノードにおいてバーストロスが起こり，マルチキャスト通信が途絶する．特に，第 3 の問題はマルチキャスト通信のサービス維持に関わるため，商用利用などの際には深刻な問題となり，重要な技術課題として認識されている．

なお，第 1,2 の問題に関しては，既存のアプリケーションレベルマルチキャストプロトコルにおいて検討がなされており，以下の 2 つの評価関数がアプリケーションレベルマルチキャストの性能を表す指標として用いられている [25, 27, 30] .

RDP (Relative Delay Penalty) ：

送信ノードから任意の受信ノード r_i へユニキャストで直接転送した場合の遅延を d_i，アプリケーションレベルマルチキャストで転送した場合の遅延を d'_i とすると，受信ノード r_i の RDP は次の式で表される．

$$RDP = \frac{d'_i}{d_i}$$

LS (Link Stress) ：

同じ物理リンク上に流れる同一パケットの個数.

RDPは1に近いほどオーバレイネットワークにおける転送経路が最短経路に近いことを表し，ユニキャスト，IPマルチキャストではRDPは1となる．また，LSはIPマルチキャストを用いた場合あらゆるリンクで1であり，ユニキャストでは最悪の場合マルチキャストメンバーの総数となる．

2.5 マルチキャスト通信の普及過程

IPマルチキャストは，2.3.2で述べた諸般の問題から，インターネットへの普及が遅れている．そのため，IPマルチキャストインフラストラクチャを必要とせず，エンドホストのみでマルチキャスト通信を実現可能なアプリケーションレベルマルチキャストに注目が集まっている．しかし，アプリケーションレベルマルチキャストにおいても，2.4.3で述べた問題がありIPマルチキャストを完全に代替する方式とはいえない．特に，アプリケーションレベルマルチキャストは遅延および使用帯域など伝送性能の点でIPマルチキャストに比べ大きく性能が劣るため，IPマルチキャストが普及しインターネット全体で運用されるまでの一時的な技術であると捉えられている [35].

近年，IPマルチキャストをサポートするネットワークが部分的に存在する状況を想定し，IPマルチキャストとアプリケーションレベルマルチキャストを組み合わせることによりインターネット全体でマルチキャスト通信を実現する方式 (図 2.8) の研究が活発に行われている [26, 27, 35, 36, 37]．また，IPマルチキャストに対応した高機能ルータが徐々にインターネットへ導入されてゆく現実的な状況を考慮し，部分的に存在する高機能ルータがアプリケーションレベルマルチキャストを補助する方式 [38, 39, 40] も検討されており，アプリケーションレベルマルチキャストからIPマルチキャストへのシームレスな移行方法についても研究がなされている．

このように，インターネット全体でマルチキャスト通信が運用されるまでには，以下のように3つのステップに分けてマルチキャスト通信の普及が進むと考えられる．

ステップ1 アプリケーションレベルマルチキャスト
ステップ2 アプリケーションレベルマルチキャストとIPマルチキャストの混在
ステップ3 IPマルチキャスト

本論文では，ステップ1およびステップ3に主眼を置き，アプリケーションレベルマルチキャストおよびIPマルチキャストの問題点を検討し，その解決方法の提案を行う．なお，ステップ2に関しては，著者らが文献 [38, 39, 40] において検討を行った．

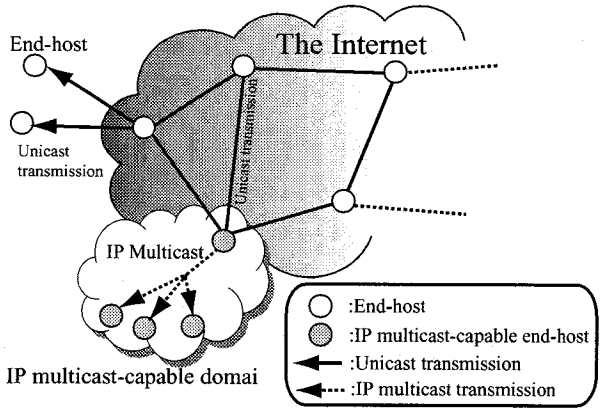


図 2.8: Hybrid architecture between IP multicast and application-level multicast.

2.6 結言

現在インターネットでは，多種多様な 1 対多通信アプリーションが登場しつつある．また，ネットワークの大容量化に勝るスピードでトラヒックが急増しており，効率的な 1 対多通信の実現が望まれている．1 対多通信への需要およびトラヒックの増加は，今後も継続されると考えられるため，効率的な 1 対多通信を実現するマルチキャスト通信は，次世代インターネットの実現に向けて必要不可欠となる．マルチキャスト通信は，マルチキャスト転送制御の実装箇所の違いによってネットワークレベルマルチキャストとアプリケーションレベルマルチキャストに大別される．本章では，まずマルチキャスト通信を概説し，ネットワークレベルマルチキャストおよびアプリケーションレベルマルチキャストについて，その有効性，問題点を述べた．さらに，マルチキャスト通信がインターネット全体で利用可能になるまでの普及過程について述べた．

効率的な 1 対多通信の実現のためには，ネットワークレベルマルチキャストおよびアプリケーションレベルマルチキャスト両方式において技術課題が克服される必要がある．以降の章では，効率的な 1 対多通信の実現を目指し，ネットワークレベルマルチキャストおよびアプリケーションレベルマルチキャストのそれぞれを対象として検討を行う．

第3章

アプリケーションレベルマルチキャストにおけるロバストなツリー構築法

3.1 緒言

効率的な1対多通信への要望から開発されたIPマルチキャストは，信頼性保証などの諸問題により普及が進んでおらず，現状では1対多通信アプリケーションへの適用は容易ではない．そこで，IPマルチキャストに代わる現実解として，広く普及しているユニキャストのみを用いてマルチキャスト通信を実現するアプリケーションレベルマルチキャストが提案され，現在，精力的に研究が進められている．

アプリケーションレベルマルチキャストでは，ルータに代わってエンドホストがマルチキャスト転送制御を行い，データはエンドホスト間をユニキャストの中継によって伝送される．このため，マルチキャスト対応ルータなどのインフラストラクチャのサポートのない状況でマルチキャスト通信を実現できる．しかし，アプリケーションレベルマルチキャストでは，専用設計されたルータと比較して相対的に安定性の低いエンドホストの障害に起因するマルチキャストツリーの分割などが問題となる．ツリーの分割が発生すると，分割によりアプリケーションレベルマルチキャストセッションから強制的に離脱させられた全エンドホストでバーストロスが起こり，マルチキャスト通信そのものが維持できなくなる．このため，エンドホスト障害問題はアプリケーションレベルマルチキャストの重要な技術課題として認識されている．

本章では，エンドホスト障害に対しロバストなアプリケーションレベルマルチキャストツリー構築法を提案し，耐障害性の高いアプリケーションレベルマルチキャストによる効率的な一対多通信の実現を図る．また，性能評価により，提案方式の有効性を明らかに

する．

3.2 アプリケーションレベルマルチキャストにおけるエンドホスト障害問題

アプリケーションレベルマルチキャストでは，これまで，そのアーキテクチャに起因して劣化する2つの伝送性能指標に主眼を置いて研究がなされてきた [25, 26, 27, 28, 29, 30, 31, 35, 37]．第1は，遅延性能である．アプリケーションレベルマルチキャストでは，パケットはエンドホストによって中継されるため，送信ノードから各エンドホストまでのホップ数が増加し，その分遅延が増加する．第2は，ネットワーク使用帯域量である．アプリケーションレベルマルチキャストでは，パケットはホスト間をユニキャストで転送されるため，1つの物理リンクに同一パケットが重複して通過し，ネットワーク帯域が浪費される．

本章では，以下に述べるエンドホスト障害問題 [10, 11] に着目する．アプリケーションレベルマルチキャストでは，ホストの故障，OSのフリーズ，アプリケーションの不正終了，比較的狭帯域なアクセスリンクにおける継続的輻輳などの障害が発生した場合，そのホストが接続される論理リンクが切断され，マルチキャストツリーが再構築される．例えば，図3.1に示されるように送信ノードとエンドホストR1～R9でアプリケーションレベルマルチキャストツリーが形成され，マルチキャスト通信が行われている状況を考える．ここで，エンドホストR3で障害が発生すると，R3の下流に位置するR5～R9がツリーから切り離され，バーストロスが発生しマルチキャスト通信が途絶してしまう．一般に，エンドホストの信頼性は専用設計されたルータより相対的に低いため，参加するホストが増加するほど，マルチキャストツリーの分割がより頻繁に発生してしまう．これまで提案されてきた既存のアプリケーションレベルマルチキャストプロトコルは，ツリーの分割に対してマルチキャストツリーの定期的な再構築で対処しているが，制御メッセージの交換を伴うツリーの再構築には数十秒単位の時間がかかり，インタラクティブなアプリケーションやストリーム配信型アプリケーション等ではこの再構築遅延は許容し難い．そのため，エンドホストの障害発生時にその影響を受けるエンドホスト数は出来る限り小さく抑制することが望ましい．

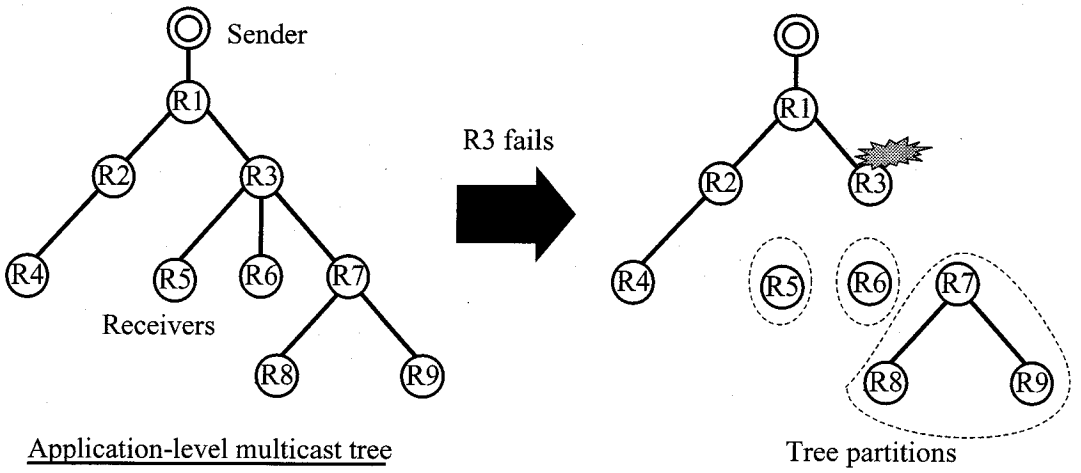


図 3.1: Endhost failure causes tree partitions.

3.3 エンドホスト障害に対しロバストなアプリケーションレベルマルチキャストツリー

アプリケーションレベルマルチキャストでは，エンドホストの障害発生時の対応が重要な技術課題となる．次数の大きい中継エンドホストの故障は，多数のツリー分割を引き起こし，多くのエンドホストに影響を与える．また，どのエンドホストで障害が発生するかはアプリケーションレベルマルチキャストツリー構築時には予測できないため，任意のエンドホストで障害が発生する場合を想定し，障害発生時に影響を受ける平均エンドホスト数を抑えるようなアプリケーションレベルマルチキャストツリーを構築する必要がある．各エンドホストの次数を均衡化させた平衡木型アプリケーションレベルマルチキャストツリーでは，各中継点における下流エンドホスト数が均衡化するため，任意のエンドホストで障害が発生した場合に影響を受ける平均エンドホスト数を減少できると考えられる．

3.3.1 平衡木型アプリケーションレベルマルチキャストツリーの有効性

本節ではまず，数学解析により，次数を均衡化させた平衡木型ツリーを用いた場合に任意のエンドホストの障害によって影響を受ける平均エンドホスト数を検証し，平衡木型ツリーの有効性を明らかにする．

本解析では簡単のため，平衡木型ツリーとしてツリーを構成する全ての節の次数が等しい完全 k 分木を用いる．

送信ノードを根とする高さ D の完全 k 分木における受信エンドホスト数（根を除くツ

リー構成ノード数）N は次式で表される．

$$N = k + k^2 + \cdots + k^D = \frac{k^{D+1} - k}{k - 1} \tag{3.1}$$

また，ツリー上で深さ d の位置にいるエンドホストにおいて障害が発生した場合に，ツリー分割によってマルチキャスト通信の途絶に陥るエンドホスト数 N_d は以下のように表される．

$$N_d = 1 + k + k^2 + \cdots + k^{D-d} = \frac{k^{D-d+1} - 1}{k - 1} \tag{3.2}$$

ここで，任意のエンドホストが深さ d に位置している確率を p_d とすると，p_d は次式で表される．

$$p_d = \frac{k^d}{N} \tag{3.3}$$

したがって，エンドホスト障害発生時にマルチキャスト通信の途絶に陥るエンドホスト数の平均 $E[N_d]$ は，

$$E[N_d] = \sum_{d=1}^{D} p_d \cdot N_d \tag{3.4}$$

で表され，エンドホスト障害発生時に，ツリーから切り離されずにアプリケーションレベルマルチキャストセッションが維持されるエンドホスト数の割合 $Rb(\%)$ は以下のようになる．

$$Rb = \left(1 - \frac{E[N_d]}{N}\right) \times 100 \tag{3.5}$$

以降，本章では，Rb をロバストネス関数と呼び，アプリケーションレベルマルチキャストのエンドホスト障害に対する耐性を表す指標として用いる．

性能比較

ここでは，平衡木型ツリーの有効性を検証するため，既存のアプリケーションレベルマルチキャストプロトコルによって構築されるツリーとの比較を行う．既存のアプリケーションレベルマルチキャストプロトコルの大部分は，マルチキャストツリーとして最短経路木（SPT:Shortest Path Tree）もしくは最小スパニング木（MST:Minimum Spanning Tree）を構築するため，比較対象として最短経路木型ツリーおよび最小スパニング木型ツリーを用いる．最短経路木型ツリーおよび最小スパニング木型ツリーについては，解析が困難であるため，シミュレーションによる評価を行った．シミュレーションでは，ネット

ワークトポロジとして，各ルータの配置位置および接続関係がランダムに与えられるランダムモデル [41] を用い，ルータ数を 1000 とした．また，最短経路木型ツリーおよび最小スパニング木型ツリーに対するロバストネス関数 Rb は以下の式で表される．

$$Rb = \frac{1}{N}\sum_{i=1}^{N}\frac{N_i}{N} \times 100. \tag{3.6}$$

ここで，N は，マルチキャストメンバー数 (全受信エンドホスト数)，N_i はエンドホスト i で障害が発生した場合に，ツリーから切り離されずにアプリケーションレベルマルチキャストセッションが維持されるエンドホスト数を表している．また，上式の $\frac{1}{N}\sum_{i=1}^{N}$ は，全エンドホストに対して和をとり，その平均を取ることを表す．ロバストネス関数が大きいほど，任意のエンドホストで障害が発生した場合にアプリケーションレベルマルチキャストセッションから離脱する平均エンドホスト数が少ないことを表しており，アプリケーションレベルマルチキャストセッション自体のロバスト性が高いことを示している．

図 3.2 は，エンドホスト数に対する各ツリーのロバスト性（ロバストネス関数 Rb）を示している．なお，平衡木型ツリーの次数 k は 4 とした．図より，平衡木型ツリーはエンドホスト数に関係なく，最短経路木型ツリーおよび最小スパニング木型ツリーと比較してエンドホスト障害発生時のアプリケーションレベルマルチキャストセッションのロバスト性が向上している．これは，平衡木型ツリーでは，ツリー上の中継ホストの下流エンドホスト数が均衡化しているため，障害発生時に影響を受ける平均エンドホスト数が減少するためである．なお本評価では，ネットワークトポロジとしてランダムグラフを用いているが，ノードの接続次数の分布が偏り，現実のインターネットによく似たネットワーク構造を持つトポロジ [42] では，エンドホストの接続次数が偏った最短経路木型ツリーが構築されるため，図 3.2 の場合に比べ最短経路木型ツリーの性能が悪化すると考えられる．このインターネットをモデル化したトポロジを用いた評価は，3.4 において行う．

3.3.2 ロバストなツリー構築法

3.3.1 の評価により，ロバスト性の観点で，平衡木型ツリーが有効であることが明らかとなった．そこで本節では，アプリケーションレベルマルチキャストツリーとして次数を均衡化させた平衡木を構築する方式を提案する．

提案方式では，まずアプリケーションレベルマルチキャストセッションへの新規参加ホストは，そのセッションの送信ノードを発見する．次に，アプリケーションレベルマルチキャストツリーを平衡木として構築するために適切な接続先ホストを，送信ノードから順にアプリケーションレベルマルチキャストツリー内を探索する．探索の結果見つかったホストを親，自身をその子として接続することで新規ホストの参加は完了する．以下本節で

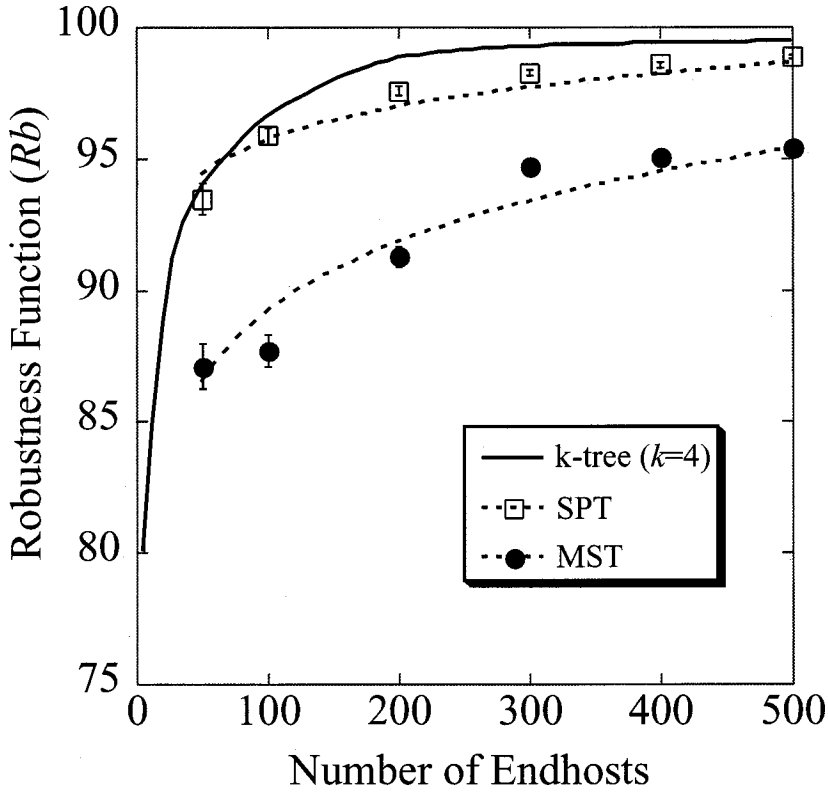


図 3.2: Impact of a single host failure on the connectivity of an application-level multicast tree.

は，提案方式の動作を詳説する．

送信ノードの発見

アプリケーションレベルマルチキャストセッションに新規に参加するホストは，まずネットワーク内のディレクトリサーバに問い合わせを行い，参加するアプリケーションレベルマルチキャストセッションの送信ノードの IP アドレスを取得する．ここで，ディレクトリサーバとは，ネットワークサービスを管理しており，サービス名（マルチキャストセッション名）を解決するサーバである．

平衡木の構築

提案方式における，平衡木構築アルゴリズムの詳細について以下に示す．

1. 新規参加ホストは，最初に送信ノード (アプリケーションレベルマルチキャストの根) を親候補として設定する．

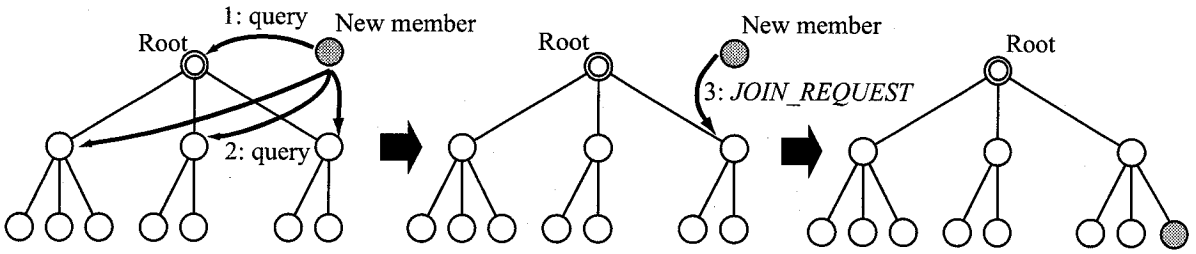


図 3.3: Join process (degree constraint = 3).

2. 親候補に対し query を送り，{ 子リスト, 次数, 高さ (葉からそのホストまでの hop 数)}(表 3.1) を取得する．親候補の次数制約[*1]を超えない場合は接続先とし，*JOIN_REQUEST* を送る．
3. 親候補の次数制約を超える場合は，子リストに記載されている全ての子ホストに query を送り，{ 子リスト, 次数, 高さ } を取得する．子ホストの中で最も次数の小さいホスト[*2]を接続先として選択し，*JOIN_REQUEST* を送る．
4. (3) において，次数制約のため全ての子が接続先となり得ない場合は，最も高さが小さい子ホストを新たに親候補として設定する．
5. 接続先が見つかるまで，(2)–(4) を繰り返す (図 3.3).

また，提案方式において各ホストが管理する必要のある情報を表 3.1 に示す．なお，表 3.1 の，"高さ"および"PATH リスト"はホスト間のシグナリングメッセージの交換により取得する (図 3.4)．アプリケーションレベルマルチキャストツリーの葉に位置するホストは，*HEIGHT* メッセージを用いて，自身の高さ (この場合は 0) を親ホストに通知する．このメッセージを受けた親ホストは，高さを 1 だけインクリメントし，その値を自身の高さとして自分の親ホストに対し *HEIGHT* メッセージを用いて通知する．このように，*HEIGHT* メッセージが葉から根へ向けて順次伝送されることにより，各ホストは自身の高さを取得する．一方，送信ノード (根) は，*PATH* メッセージを用いて，自身のみが記載された PATH リストを子に通知する．子ホストは，PATH リストに自身を追加し，さらに自分の子へ *PATH* メッセージを送信する．このように，*PATH* メッセージが根から葉へ向けて順次伝送されることにより，各ホストは送信ノードから自身までの経路上にあるホストを知ることができる．また，これら *HEIGHT*, *PATH* メッセージの交換は定期的に行い，隣接ホスト間の接続性の確認にも利用する．もしホストが一定期間待機しても，これらメッセージを受信できない場合には，隣接ホストに障害が発生したと判断し，再びセッ

[*1] 次数制約は，プロトコルパラメータである．例えば，次数制約が 4 の場合は，アプリケーションレベルマルチキャストツリーとして平衡 4 分木が構築される．

[*2] 対象が複数ある場合には，自身と対象ホスト間の RTT を測定し，最も近いエンドホストを選択する．

ションに参加し直すことでツリー分割を修復する．

表 3.1: Information kept by end-host.

・**子リスト** (アプリケーションレベルマルチキャスト転送テーブル)：*JOIN_REQUEST* の送信元ホストを子リストに追加する．子リストはデータ転送時には，転送テーブルとして用いられる．
・**次数**：データ中継数 (分岐数) であり，子リストのエントリ数に一致する．
・**高さ**：ホストの新規参加時, ツリー最適化に用いる．
・**PATH リスト**：送信ノードから自身までの経路上にあるホストのリスト．ループ回避，ツリー最適化に用いる．
・**隣接ホスト間の RTT**：ツリー最適化に用いる．

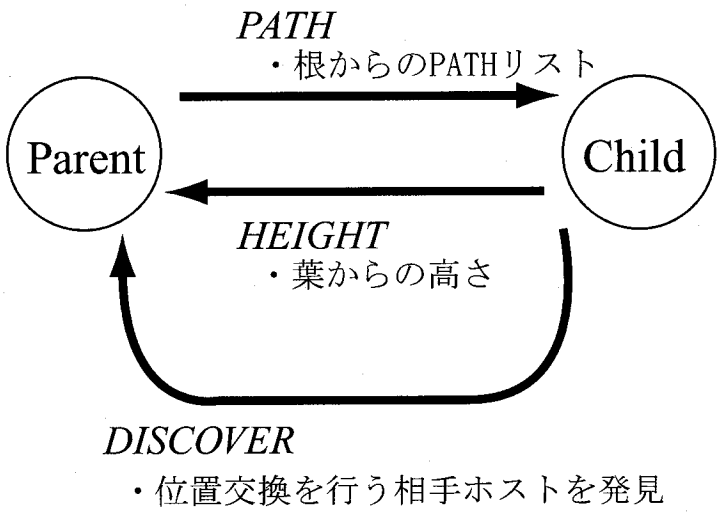


図 3.4: Signaling messages between endhosts.

ツリーの最適化

前節で示したツリー構築アルゴリズムにおいては，新規参加ホストはマルチキャストツリーに対し根から葉までの局所的縦型探索を行うことにより接続先ホストを発見する．新規参加ホストはマルチキャストツリーを構成している全てのエンドホストに対しては探索を行わないため，ツリーの形状が局所的に変形し完全平衡木を形成できない場合がある．また，次数制約を優先してツリー構築を行うため，アプリケーションレベルマルチキャストツリー上での隣接関係とネットワーク距離が対応せず遅延性能が劣化する．そこで提案

方式では，定期的に最適化を行い，ネットワーク距離を考慮した平衡木へとアプリケーションレベルマルチキャストツリーを改善する．

- 完全平衡木への最適化
  各ホストは定期的にツリー上のホストをランダムに選択し，お互いに先祖–子孫の関係になく，かつ互いの位置を入れ替えることによってツリーの高さが小さくなる場合[*3]には，位置を交換する．なお，位置交換の対象ホストの選択には，アプリケーションレベルマルチキャストツリー上でのランダムウォーク [29] を利用する．各ホストは，ランダムに TTL 値を設定した *DISCOVER* メッセージ（図 3.4）を親ノードに送信する．*DISCOVER* メッセージを受信したホストは TTL をデクリメントし隣接ホスト（子リストに記載されているホストおよび親ホスト）の中からフォワード先をランダムに選んで転送する．TTL が 0 になったホストが位置交換の対象ホストとなる．
- ネットワーク距離の最適化
  ランダムに選択されたホストが自身と同じ高さの場合で，なおかつ互いの位置交換によって送信ノードからの受信遅延が改善される場合には，位置交換を実行する．例えば，エンドホスト i が同じ高さのエンドホスト j を位置交換の対象ホストとして選択した場合，まず受信遅延の改善度 Δ を調べる．Δ は以下の式で表される．

$$\Delta = (L'_{k,i} + L'_{k,j}) - (L_{k,i} + L_{k,j}) \tag{3.7}$$

  ここで，k は，i, j が共有する最も近い先祖ホストを表しており，*DISCOVER* メッセージがツリー上を伝搬する際に，上昇から下降に移る点である．また，$L'_{k,i}, L'_{k,j}$ は，位置交換が実行された場合のエンドホスト k からエンドホスト i およびエンドホスト j へのアプリケーションレベルマルチキャストツリーに沿った遅延を表しており，$L_{k,i}, L_{k,j}$ は位置交換が実行される前の遅延を表している．提案方式では，各エンドホストが隣接ホストとの間の RTT を保持して（表 3.1）おり，*DISCOVER* メッセージは中継される際に，各エンドホスト間の RTT を累積加算してゆく．この結果，エンドホスト i, j において，先祖ホスト k から自身の親ホストまでの遅延を知ることが可能となる．また，i, j は，シグナリングによってお互いの親をまでの RTT を測定し，これらの情報から $L'_{k,i}, L'_{k,j}$ および $L_{k,i}, L_{k,j}$ を計算し，最終的に Δ を求める．もし，$\Delta < 0$ の場合には位置交換を実行する．

[*3] 各ホストは，自身の"高さ"と"深さ"(PATH リストのエントリ数) を対象ホストに通知し，この情報を元に位置交換後のツリーの高さが増減するかどうかを計算する．

3.4 性能評価

本節では，計算機シミュレーションを用いて既存のアプリケーションレベルマルチキャストプロトコルと比較することにより，提案方式の基本性能を評価する．

3.4.1 シミュレーションモデル

シミュレーションにおけるネットワークトポロジとして，各ルータの配置位置および接続関係がランダムに与えられるランダムモデル [41] と，2 階層構造をもち，冪乗則[*4]を有するトポロジ [42] を用いた．ランダムモデルでは，ルータ数を 1000 とし，エンドホストはランダムに選ばれたルータに接続するものとした．また，冪乗則を持つトポロジの生成ツールとして BRITE（Boston university Representative Internet Topology gEnerator）[43] を用いた．BRITE では，ネットワーク内のルータが AS（Autonomous System）レベル[*5]とルータレベルの 2 階層に階層化され，AS，ルータ各レベルにおいて冪乗則が成り立つ．このため，BRITE は実際のインターネットのトポロジに非常に近いトポロジとして知られている．シミュレーションにおいては，AS ルータ数 (AS レベル) を 50，各 AS 内のルータ数 (ルータレベル) を 20 とし，ネットワーク内の総ルータ数は 1000 とした．また，エンドホストはルータレベルのルータに接続しているものとした．シミュレーションでは，各エンドホストを順次アプリケーションレベルマルチキャストセッションに参加させ，全マルチキャストメンバーの参加が完了した時点で，ランダムに選んだエンドホストをアプリケーションレベルマルチキャストツリーから離脱させ，ツリー分割を発生させた．

比較対象プロトコルとして，2.4.2 で紹介した Narada プロトコル [25] を用いる．提案方式の次数制約 k を 6 とした場合に，提案方式と Narada プロトコルのエンドホスト管理情報量が等しくなるよう，Narada プロトコルのプロトコルパラメータを設定した．なお，3.3 節で述べた提案方式におけるツリーの最適化は，Narada プロトコルと同頻度で行った．

3.4.2 評価指標

提案方式のロバスト性を評価する指標として，3.3.1 で述べたロバストネス関数 Rb を用いる．また，伝送性能を評価する指標として，2.4.3 で述べた RDP（Relative Delay

---

[*4] ルータの出線数 d とその出現頻度 f_d の間には定数 α を用いて $f_d\, d^{\alpha}$ の関係が成り立つ．

[*5] 統一された運用方針（同一の経路制御プロトコルの使用など）によって管理されたネットワークの集まり．

Penalty）の受信ノード平均を用いる．平均 RDP は送信ノードから任意のエンドホスト i へユニキャストで転送した場合の遅延を d_i，提案方式で転送した場合の遅延を d'_i，受信ノード数を N とすると次式で表される．

$$E[RDP] = \frac{1}{N}\sum_{i=1}^{N}\frac{d'_i}{d_i} \tag{3.8}$$

3.4.3 シミュレーション結果

図 3.5，3.6 は，マルチキャストメンバー数に対する提案方式 (図中では k-tree と表記, 次数制約 k) のエンドホスト障害発生時のロバスト性を 90%の信頼区間とともに示してある．ネットワークモデルは図 3.5 がランダムモデル，図 3.6 が BRITE である．図 3.5，3.6 より，提案方式は次数制約 k 及びメンバー数の大きさに関係なく，Narada プロトコルと比較して，エンドホストの障害発生時のアプリケーションレベルマルチキャストセッションのロバスト性が向上していることがわかる．これは，提案方式ではアプリケーションレベルマルチキャストツリーが平衡木として形成されることによって，ホスト障害の影響を受ける平均エンドホスト数が減少するためである．一方，Narada プロトコルでは，DVMRP 型ルーチング [14] により最短経路木型ツリーが形成されるため中継ホストの次数が不均一となる．このため，ホスト障害によって多数のホストが影響を受ける状況が発生し，影響を受ける平均エンドホスト数が悪化する．

図 3.7，3.8 は，マルチキャストメンバー数に対する RDP 特性を示しており，縦軸は (3.8) 式で表される，マルチキャストメンバー間での平均 RDP を示している．ネットワークモデルは図 3.7 がランダムモデル，図 3.8 が BRITE である．図 3.7，3.8 より，提案方式は次数制約 k を大きく設定すること[*6]により，RDP を改善することが可能である．これは，k の増加に伴ってアプリケーションレベルマルチキャストツリーの深さが浅くなり，送信ノードから各エンドホストまでの論理ホップ数が減少するためである．図 3.5，3.6，3.7，3.8 より，提案方式 ($k = 6$ の時) は，Narada プロトコルと等しいエンドホスト管理情報量で，アプリケーションレベルマルチキャストセッションのロバスト性及び平均 RDP ともに性能が向上する．

3.5 結言

本章では，アプリケーションレベルマルチキャストにおけるエンドホスト障害発生時のアプリケーションレベルマルチキャスト通信途絶問題に注目し，エンドホスト障害に対し

[*6] k の増加に伴い，エンドホスト管理情報量 (隣接ホストに関する情報) は増大する．シミュレーションでは，$k = 6$ の時，エンドホスト管理情報量が Narada プロトコルと等しくなるよう設定した．

ロバストなアプリケーションレベルマルチキャストツリー構築法を提案した．提案方式では，中継エンドホストの次数を均衡化した平衡木型アプリケーションレベルマルチキャストツリーを構築することにより，エンドホストの障害発生時にアプリケーションレベルマルチキャストセッションから分断されるマルチキャストメンバー数を抑えることが可能となる．

また，計算機シミュレーションを用いて提案方式の性能を検証し，既存のアプリケーションレベルマルチキャストプロトコルと比較して，エンドホスト障害発生時のアプリケーションレベルマルチキャストセッションのロバスト性，およびアプリケーションレベルマルチキャスト伝送遅延の点で性能が向上することを明らかにした．これらの結果より，本方式の有効性が実証された．

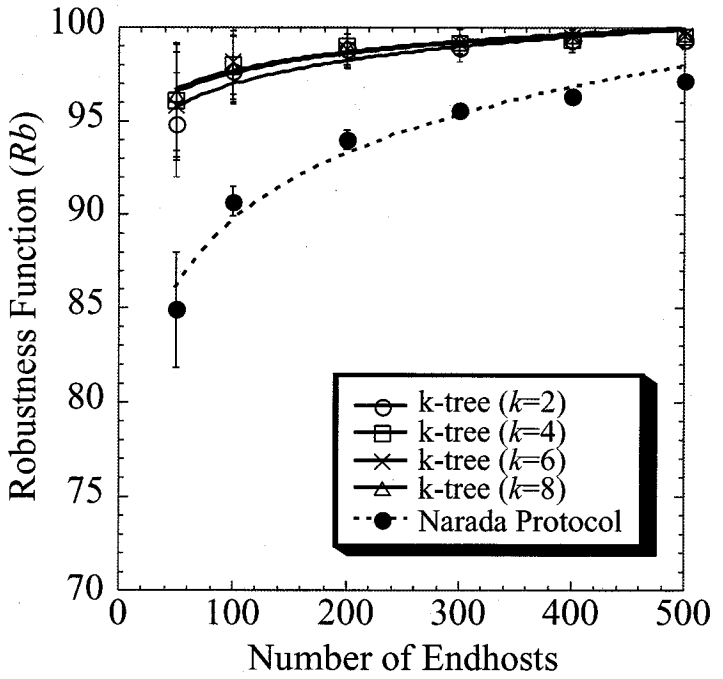


図 3.5: Tree robustness against endhost failure (**Waxman**).

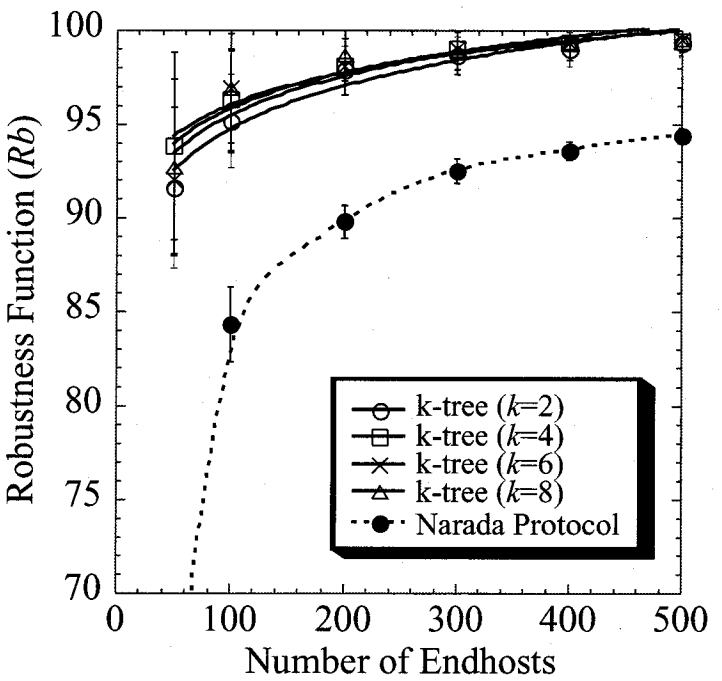


図 3.6: Tree robustness against endhost failure (**BRITE**).

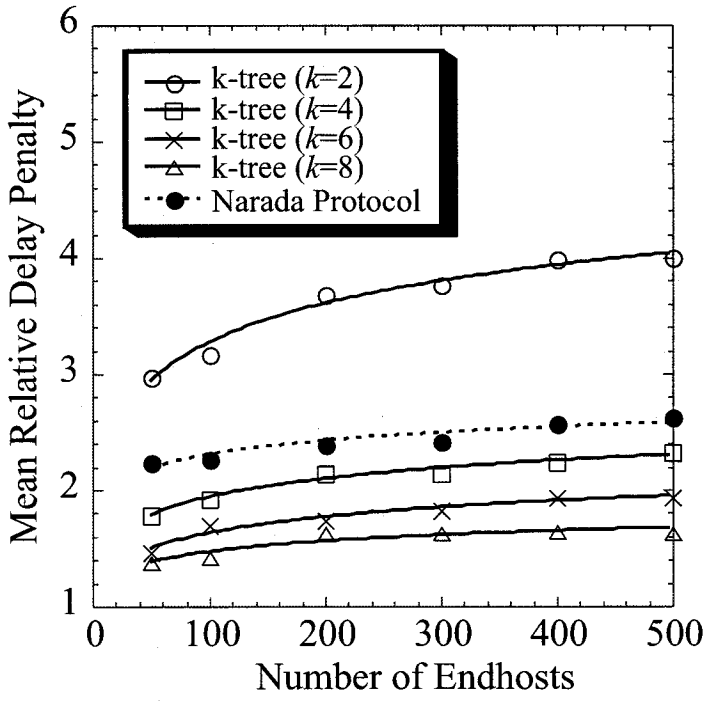


図 3.7: RDP performance (**Waxman**).

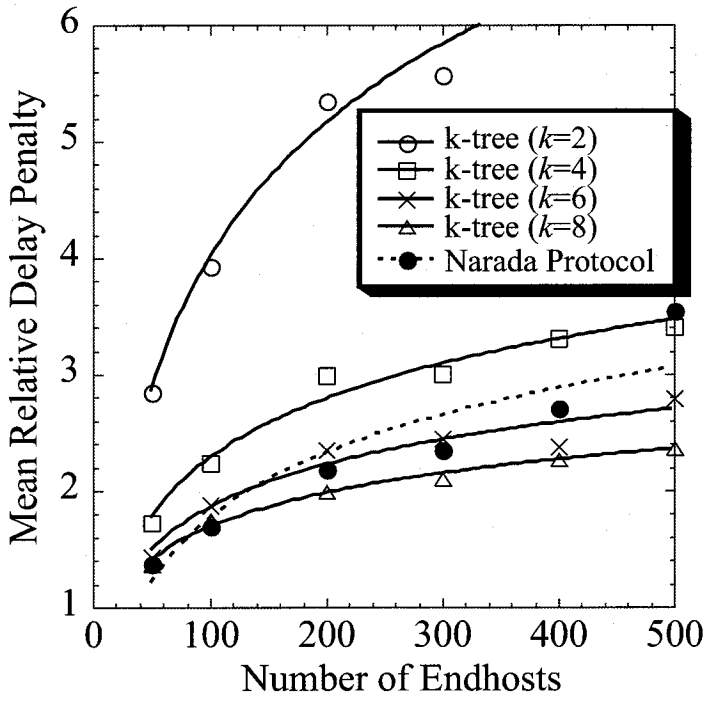


図 3.8: RDP performance (**BRITE**).

第 4 章

信頼性マルチキャストにおける マルチキャスト再送制御

4.1 緒言

ネットワークレベルマルチキャストは，遅延およびネットワーク使用帯域量の点でアプリケーションレベルマルチキャストに比べ遙かに優れた伝送性能を示す．そのため，効率的な 1 対多通信の実現手段として最有力視されているが，信頼性保証などに課題が残り，普及への足枷となっている．本節では，ネットワークレベルマルチキャストの信頼性保証に着目する．ネットワークレベルマルチキャストは best-effort 型サービスであるため，ソフトウェアの配信やデータ共有など情報を誤り無く転送する必要のあるアプリケーションに対しては，上位層であるトランスポート層において信頼性を確保する機能を提供する必要がある．この機能を具備するトランスポートプロトコルとして信頼性マルチキャストプロトコルの開発が進められている．ネットワークレベルマルチキャストでは，より多くのユーザを収容することでネットワークの利用効率が向上することが知られており [7, 44]，ユーザ数増加への対応が重要な技術課題となる [22]．信頼性マルチキャストでは，再送制御に伴い発生する多数の制御パケットが送信ノードへ集中（Feedback implosion）するため，スケーラビリティを考慮して制御パケットの発生をできるだけ抑圧する必要がある．

本章では，誤り訂正符号（FEC:Forward Error Correction）を使用した再送制御の補助をネットワーク内で部分的に行う方式を提案する．提案方式は，ネットワーク内でパケットロスが頻繁に発生する部分にのみ，局所的に FEC を適用することで Feedback implosion の原因となる共有ロスを低減し，効率的な信頼性マルチキャスト通信の実現を目指す．また，性能評価により，提案方式の有効性を明らかにする．

4.2 信頼性マルチキャストの概要

信頼性マルチキャストプロトコル (Reliable Multicast Protocol) は，Best-effort 型プロトコルである IP マルチキャストを利用し，トランスポート層において再送制御を行なうことにより「特定多数の受信ノードに対して情報を同時に誤りなく転送する」通信プロトコルである．信頼性マルチキャストプロトコルは ACK ベースの sender-initiated プロトコルと NAK ベースの receiver-initiated プロトコルの 2 つに分類される [45].

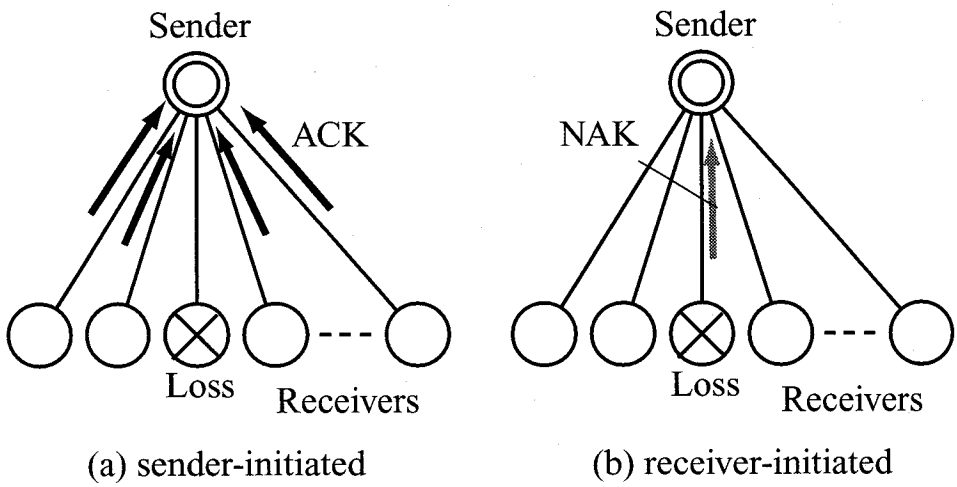


図 4.1: Sender-initiated protocol and receiver-initiated protocol.

sender-initiated プロトコルは，ユニキャスト通信における TCP と同様にパケットを正しく受信した受信ノードが送信ノードに対して ACK(Acknowledgement) をユニキャストで送信することにより送信ノードが再送制御を行う方法である．このプロトコルでは，図 4.1-(a) のように正しくパケットを受け取った受信ノード全てが ACK を送信ノードへ送信するため，送信ノードに ACK が集中し (ACK implosion)，受信ノード数が増加した場合に適用することが困難となる．

receiver-initiated プロトコルでは，パケットのシーケンス番号のギャップによりパケットロスを検出した受信ノードが NAK(Negative Acknowledgement) を送信ノードに対して送信することにより再送制御を行う方法である．

receiver-initiated プロトコルにおいては，図 4.1-(b) のように送信ノードに対する制御パケット数が少ないために ACK implosion のような送信ノードのオーバーヘッドによる問題は生じにくい．しかし，シーケンス番号のギャップによりパケットロスを検出するため，パケット発生率が低い場合には sender-initiated プロトコルに比べロス検出に必要な時間が長くなり [46]，またパケット列の最後のパケットがロスした場合のロス検出法も問

題となる．

receiver-initiated プロトコルは，受信ノードからの NAK の送信方法により 2 種類に分類することができる．

1 つは受信ノードから送信ノードに対して NAK をユニキャストする方法である．この方法ではパケットロスを検出した受信ノードが即座に NAK を送信するため，ロスを検出した全ての受信ノードが NAK を送信する．従って，ロス率の高い場合や受信ノード数が増加した場合には送信ノード周辺において NAK が集中し (NAK implosion)，適用することが困難となる．

もう 1 つは受信ノードから送信ノードおよび同一マルチキャストグループに属する全ての受信ノードに対し NAK をマルチキャストする方法 [24] である．この方法では，パケットロスを検出した受信ノードはまずランダム時間だけ NAK の送信を待機する．待機中に他の受信ノードからの該当パケットに対する NAK を受信した場合，NAK の送信を中止 (NAK suppression) し重複する NAK の発生を抑圧することにより NAK をユニキャストする場合に比べて NAK の発生数を減少することが可能となる．従って NAK をユニキャストする方法と比較してより多数の受信ノードを収容することが可能となる [47]．本章では，マルチキャスト通信において特に重要となるスケーラビリティに焦点を当てているため，スケーラビリティの点で優れている NAK をマルチキャストにより伝送する方法 (NAK プロトコル) を対象として検討を行う．

4.2.1 NAK プロトコルの概要

本研究で対象とする NAK プロトコルは，パケットロス検出時に NAK をマルチキャストすることにより再送制御を行うプロトコルである．NAK プロトコルの基本的な動作を以下に示す．

- 送信ノードはパケットをマルチキャストにより全受信ノードに送信する．
- 受信ノードはパケットロスを検出すると，ランダム時間のランダムタイマーを設定してロスしたパケットに対する NAK 送信をスケジューリングする．
- 該当パケットを受信する前にランダムタイマーがタイムアウトした場合，受信ノードは NAK を送信して受信ノードタイマーを再設定する．またランダムタイマーがタイムアウトする前に他の受信ノードからの該当パケットに対する NAK を受信した場合，NAK の送信を中止 (NAK suppression) して（図 4.2）ランダムタイマーを再設定する．
- 送信ノードは NAK を受信すると，再送制御を行い該当パケットをマルチキャストにより再送する．ただし，NAK カウンタを用いることにより同一パケットに対する NAK を受信した場合は，該当パケットの送信は行わない．

- 受信ノードにおいて NAK 送信後，該当パケットを受信する前に受信ノードタイマーがタイムアウトした場合，NAK を再送して受信ノードタイマーを再設定する．

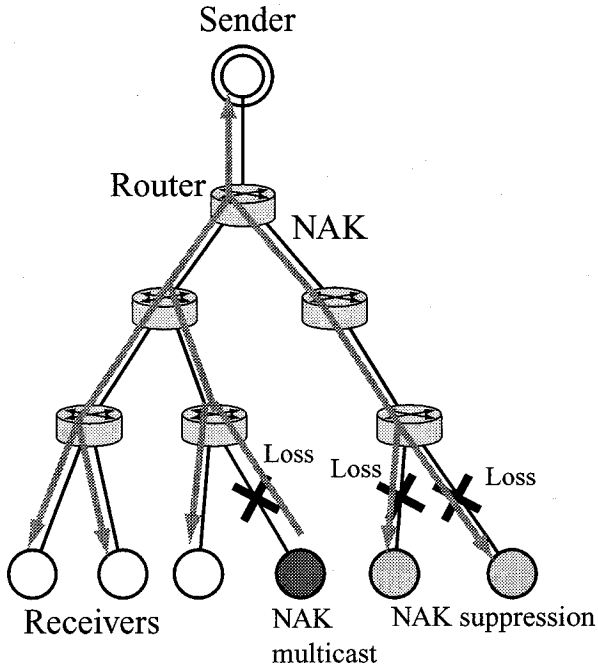


図 4.2: NAK protocol.

4.3 マルチキャストバックボーンにおけるパケットロスの相関性

信頼性マルチキャストにおいては制御パケットを減らすことが重要な技術的課題点であり，特にインターネットにおいて信頼性マルチキャストを適用する場合，送信ノードの近隣リンク（ソースリンク）においてパケットロスが生じると複数の受信ノードが同一パケットのロスを被る可能性があるため，ロスの発生傾向に適合した再送制御を行う必要がある．

University of Massachusetts の Maya Yajnik らは，マルチキャストバックボーン上でのパケットロスの測定を行い，マルチキャストバックボーン上でのパケットロスには地理的・時間的相関があることを報告している [48]．特に多くのマルチキャストプロトコルでは送信ノードだけでなく他のマルチキャストメンバーとも制御パケット等のデータをやりとりするため地理的なロスの相関性が重要となる．文献 [48] では，測定結果の分析によりマルチキャストバックボーンにおけるパケットロスの地理的相関性について以下の傾向があることが示されている (図 4.3).

- バックボーンネットワークパケットロスの発生率は低く，複数の MAN (Metropolitan Area Network) にまたがる shared loss の可能性は小さい．
- 単一 LAN 内のノード間ではロスが発生する確率はほとんど 0%，多くても 0.001% 以下である．
- バックボーンネットワークと受信ノード間のリンクでは一部で 1%程度のロスが発生する．
- 送信ノードに隣接するリンクにおけるロス発生率は 5%程度と他のリンクと比較して高く，ボトルネックとなっている．

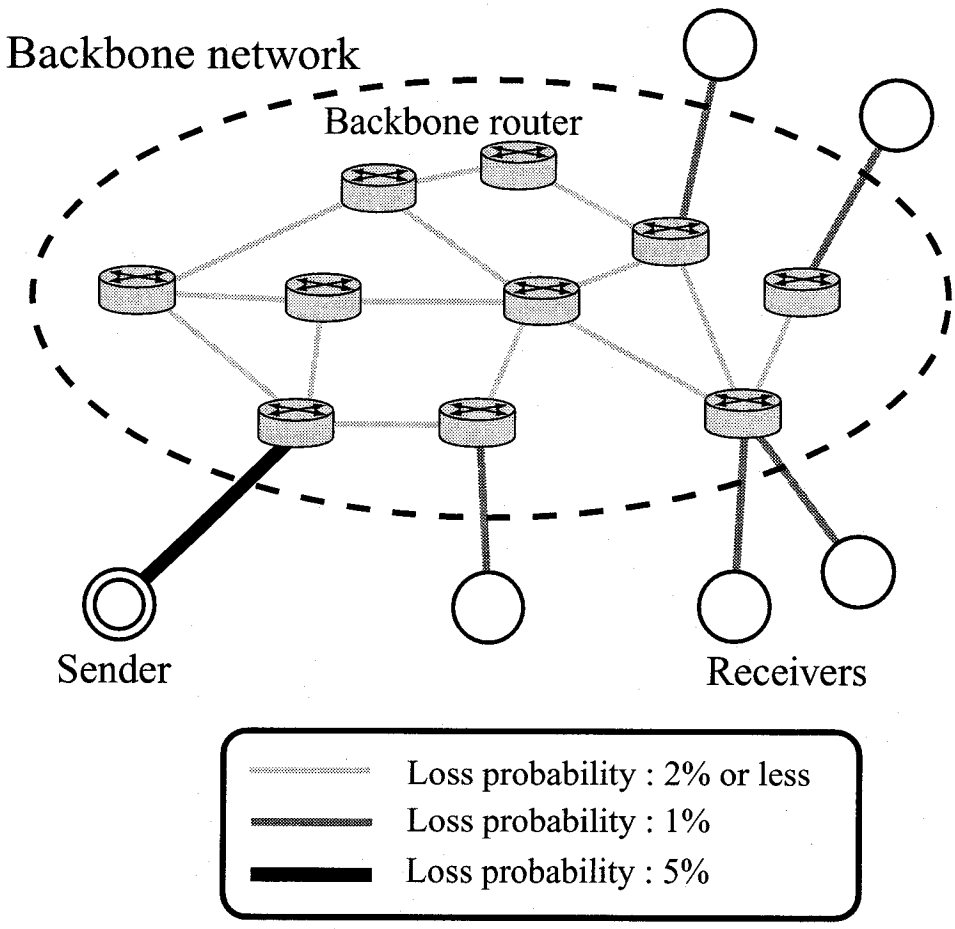


図 4.3: Packet loss in multicast backbone.

マルチキャストバックボーン上ではソースリンクでのロス発生率が高く，バックボーンネットワークや受信ノードの属する LAN 内でのロスの発生率は低くなっており，パケットロスに局所性がある．信頼性マルチキャストプロトコルにおいては，このパケットロスの局所性を考慮してより有効な誤り制御を行う必要がある．しかし，現在提案されている信頼性マルチキャストプロトコルの多くはロスの発生過程については考慮していないか，

もしくはロスの地理的・時間的独立性を仮定している．

NAK プロトコルでは，送信ノードに隣接するソースリンクでパケットロスが生じるとバックボーンネットワークを経由する全ての受信ノードがパケットロスを被る．従って，バックボーンネットワークを経由する全受信ノードが NAK の送信を準備するため，NAK suppression が有効に働かず冗長な NAK の発生を招き，多数の受信ノードを収容可能な効率的信頼性マルチキャスト通信の実現に大きな影響を及ぼす．

そこで，本章ではパケットロスの時間的独立は仮定するもののパケットロスの地理的相関，すなわち送信ノードに隣接するソースリンクでのロスを考慮した信頼性マルチキャストプロトコルを提案する．

4.4 局所的に FEC を適用した信頼性マルチキャストプロトコル

マルチキャスト通信におけるパケットロスの発生傾向を分析した文献 [48] においては，ソースリンクでのロス率がバックボーンネットワークでのロス率に比べて非常に高いことが報告されている．ソースリンクで発生したパケットロスは，下流に位置する全受信ノードでの同一パケットに対するロスを引き起こす．この場合，4.2 で紹介した NAK suppression などの方法では Feedback implosion に有効に対処できない．

一方，経路中のパケットロスを低減させる手段として FEC(Forward Error Correction) を用いる方法がある．信頼性マルチキャストに FEC を適用した例としてはデータパケットと共にパリティパケットを送信し，受信ノードにおいて経路中でロスしたパケットを復元する方式が提案されている [49, 50, 51, 52, 53, 54, 55]．しかし，冗長なパリティパケットの伝搬によりネットワーク帯域が大きく消費されることが問題となる．

そこで本章では，冗長なパリティパケットがネットワーク全体へ伝搬することを回避し，さらに多量の制御パケットの発生の原因となるソースリンクロスを低減させ，スケーラビリティの向上を図る方式として，ソースリンクにのみ FEC を適用する方式を提案する (以下，本方式を Local FEC プロトコルと呼ぶ)．

Local FEC プロトコルの概念図を図 4.4 に示す．

Local FEC プロトコルにおいては，FEC を適用する範囲を送信ノードとバックボーンネットワークのエッジに位置するゲートウェイルータ (以下 GR と呼ぶ) 間とし，GR においてロスパケットの復元を行なう．なお，送信ノードと GR 間の FEC で対応しきれなかったパケットロス，ならびに GR より先で発生したパケットロスに対しては NAK suppression を用いた方式により受信ノードで対応する．ただし，Local FEC プロトコルにおいて，GR で行なうロス検出や FEC の復号化処理等はトランスポート層以上の層で

の処理であるため既存ルータでは実現不可能である．このような処理を実現する方法としてはアクティブネットワーク技術 [56, 57] を適用する方法や FEC 復号化処理機能を持たせた特別なサーバを配置する等の方法が考えられる．

Local FEC プロトコルの詳細な動作を以下に示す．

1. 送信ノードは送信データを分割してデータパケットを生成し k 個毎にグループ化する．次に，グループ毎に計算し生成された $n-k$ 個のパリティパケット*1を付加して全ての受信ノードへマルチキャストする (データパケット k 個とパリティパケット $n-k$ 個を合わせた合計 n 個のパケットグループを以下 FEC グループ*2と呼ぶ).
2. GR はパケットを受信すると，順序通りに正しく受信したデータパケットについてはそのコピーをキャッシングすると同時にフォワードする．ただし，パリティパケットについてはフォワードせずに GR でストップさせる．
3. GR において受信パケットのシーケンス番号のギャップによりロスを検出した場合には，以降到着するパケットをフォワードせずに GR でキャッシングする．
4. GR のキャッシュ内に同一 FEC グループに属するデータパケット及びパリティパケットが合計 k 個揃った時点で，FEC グループのデコードを行いロスパケットを

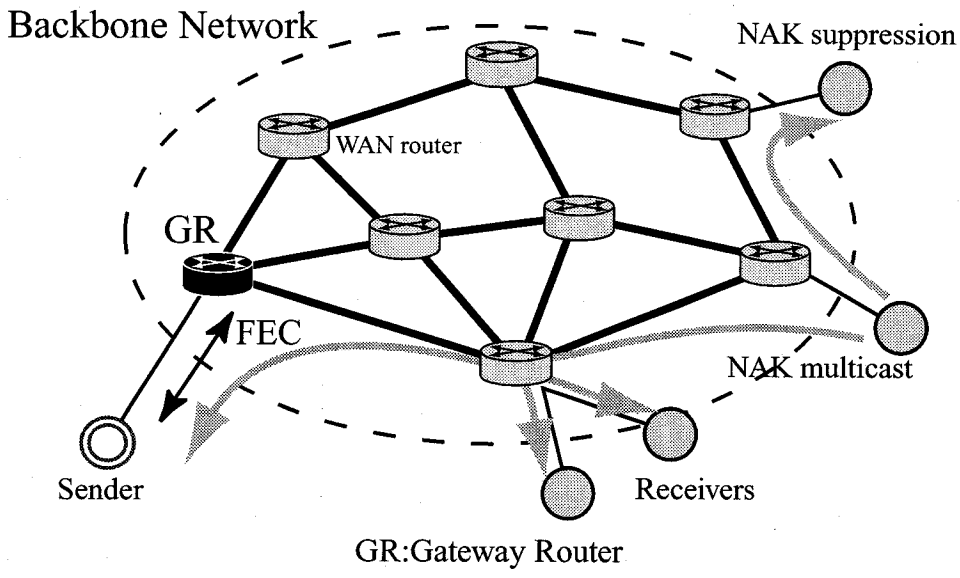


図 4.4: Reliable multicast protocol applying local FEC.

*1 本章で対象とする FEC は，専用ハードウェアに実装される比較的短いビットレベルの誤り訂正ではなく，汎用 CPU 上で動作するソフトウェアによるパケットレベルの誤り訂正を行う FEC である [58].

*2 FEC グループ間での識別のためにグループ番号を導入し，同一 FEC グループに属するパケットには同じグループ番号を割り当てるものとする．また，FEC グループ内でのパケット間の識別のためにパケット番号を導入し，同一 FEC グループに属するデータパケットには $1 \sim k$ のパケット番号を付与し，残りの $n-k$ 個のパリティパケットには $k+1 \sim n$ のパケット番号を付与するものとする．

復元する．その後，復元パケットの中で未送出パケットのみをフォワードし，パリティパケットについてはフォワードせずに廃棄する．

5. GR のキャッシュ内の FEC グループに属するデータパケット及びパリティパケットが合計 k 個揃う前に，次の FEC グループに属するパケットが到着した場合，復元は不可能となりキャッシュ内の未送出パケットをフォワードし，パリティパケットはフォワードせずに廃棄する．
6. 受信ノードでシーケンス番号のギャップによりロスを検出した場合には，前述した NAK プロトコルと同様の動作をする．

ただし，送信ノードは，再送パケットにはパリティパケットを付加しない．また，GR は到着する再送パケットに対してはトランスポート層以上の処理を行わずネットワーク層までの処理を行い既存のルータと同様の動作をする．

4.5 遅延解析

本節では，4.4 で示した Local FEC プロトコルにおいて，送信すべきパケットが送信ノードのトランスポート層に到着してから，ランダムに選ばれた受信ノードのトランスポート層にパケットが正しく到着し処理が完了するまでの平均遅延を求める．なお，本解析では，トランスポート層での処理時間を対象としており，ネットワーク層以下での処理時間はノード間伝搬遅延に含めるものとする．本解析は，以下の 2 つのステップからなる．

1. 送信ノード，GR 及び受信ノードのトランスポート層に到着するパケット (送信/再送データパケット及び NAK パケット) の到着率を求める．このパケット到着率をもとに，送信ノード，GR，受信ノードのトランスポート層でのパケット処理過程を M/G/1 待ち行列でモデル化[*3]し，パケットの平均待ち時間を得る．
2. あるデータパケットに着目し，そのデータパケットが送信ノードのトランスポート層に到着してから，受信ノードのトランスポート層で処理が完了するまでの送・受信ノード間のデータパケット，NAK パケットの流れをもとに，(1) のステップで求めた平均待ち時間を加味して，平均遅延を求める．

[*3] 本解析では，簡単のため各ノードにおけるパケット処理過程を M/G/1 でモデル化した．本解析の目的は，提案方式の絶対的な性能評価ではなく，NAK プロトコルとの相対的な性能比較である．そのため，M[x]/G/1 など他の処理過程を用いたとしても両者の性能比較では同様の結果が得られると考えられる．

4.5.1 ネットワークモデル

1つの送信ノードが，データパケットを k 個毎にグループ化し，そのグループ毎に $n-k$ 個のパリティパケットを付加して，R 個の受信ノードへマルチキャストする状況を考える．ここで，データパケットおよびパリティパケットは送信ノードのトランスポート層へ $\frac{n}{k} \times \lambda$ のポアソン到着する [*4] ものと仮定する．送信ノードから送信されたパケットはソースリンク (送信ノード-GR 間) で確率 p_s でロスし，GR 以降の下流では確率 p_d でロスするものとし，この確率は全受信ノードに対して等しいものとする．さらに，パケットロスは時間的に独立に発生するものとする．ただし，NAK パケットはロスしないものとする．また，ノード間の伝搬遅延は全ノード間で τ とし，送信ノード–GR 間の伝搬遅延は τ_s であると仮定する．以上のネットワークモデルを図 4.5 に示す．

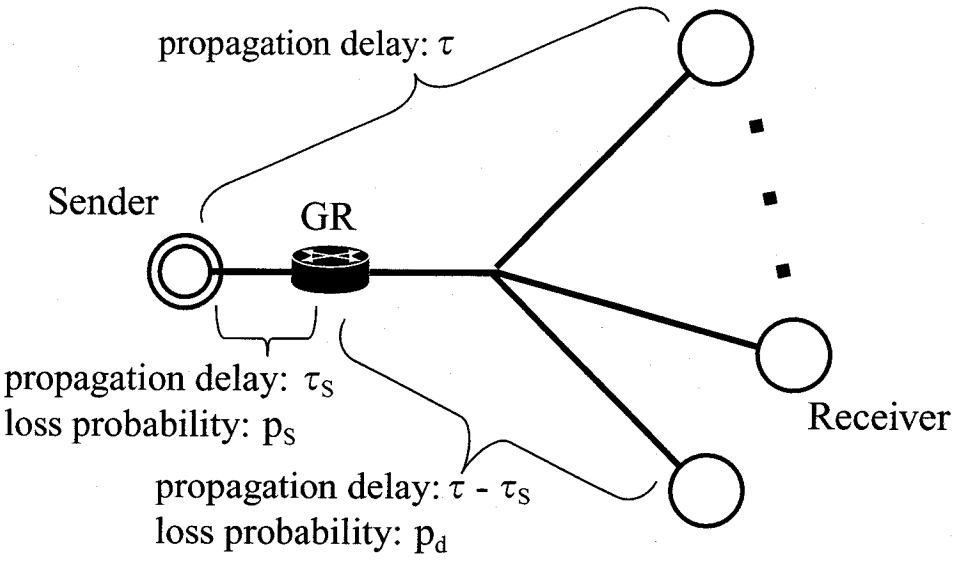


図 4.5: Network model.

4.5.2 NAK 発生数

Local FEC プロトコルでは，GR においてロスパケットの復元を行なうが，ソースリンクにおいて各 FEC グループ当たり $n-k$ 個以上のパケットをロスした場合には GR においてロスパケットの復元を行なうことはできない．ここで，あるデータパケット (i) が GR

[*4] 本来，パリティパケットは，データパケット到着後に FEC エンコーダにおいて生成されるものであるが，本解析では簡単のため，データパケットと同様にトランスポート層へポアソン到着するものと仮定する．なお，FEC のデコード/エンコード時間はノード間伝搬遅延と比較して非常に小さいため [51, 58]，送信ノードにおける FEC のエンコード時間および GR における FEC デコード時間は無視するものとする．

において復元できない確率を p_{GR} とすると，p_{GR} は以下のように求められる．

$$p_{GR} = p_s\left(1 - \sum_{j=0}^{n-k-1}\binom{n-1}{j}p_s^j(1-p_s)^{n-j-1}\right) \tag{4.1}$$

次に，データパケット (i) が全受信ノードに受信されるまでに送信ノードが送信するデータパケット数を表す確率変数を M とすると，その分布関数は以下のように求められる．

$$\begin{aligned}&P[M \le m] = \\ &p_{GR}\sum_{j=0}^{m-2}\binom{m-1}{j}p_s^j(1-p_s)^{m-j-1}(1-p_d^{m-j-1})^R + \\ &(1-p_{GR})\sum_{j=0}^{m-2}\binom{m-1}{j}p_s^j(1-p_s)^{m-j-1}(1-p_d^{m-j})^R\end{aligned} \tag{4.2}$$

したがって，その平均は

$$E[M] = \sum_{m=1}^{\infty}(1 - P[M \le m]) \tag{4.3}$$

となる．

次に，データパケット (i) が送信ノードで発生してから全受信ノードに受信されるまでの間に発生した NAK 数 (確率変数 N) の期待値を求める．この期待値は次の確率変数を用いて以下のように求められる．

M_j データパケット (i) に対する j 回目の再送において，パケット i に対する NAK を送信する受信ノード数

L_j データパケット (i) に対して j 回目の再送を要求する受信ノード数

$$\begin{aligned}E[N] &= \sum_{j=1}^{\infty}E[M_j] \\ &= \sum_{j=1}^{\infty}\sum_{l=1}^{R}\sum_{m=1}^{l}mP[M_j = m, L_j = l] \\ &= \sum_{j=1}^{\infty}\sum_{l=1}^{R}\sum_{m=1}^{l}mP[M_j = m|L_j = l]P[L_j = l]\end{aligned} \tag{4.4}$$

上式の $P[L_j = l]$ は以下のように求められる．

$$\begin{aligned}&P[L_j = l] = \\ &\binom{R}{l}\left\{p_{GR}\sum_{u=0}^{j-1}\binom{j-1}{u}p_s^u(1-p_s)^{j-u-1}p_d^{l(j-u-1)}(1-p_d^{j-u-1})^{R-l} + (1-p_{GR})\cdot\right.\end{aligned}$$

$$\sum_{u=0}^{j-1}\left(\begin{array}{c} j-1 \\ u \end{array}\right) p_s^u(1-p_s)^{j-u-1} p_d^{l(j-u)}(1-p_d^{j-u})^{R-l}\Bigg\} \tag{4.5}$$

$P[M_j = m|L_j = l]$ については，文献 [46] と同様の方法で求められる．

4.5.3 パケット到着率

図 4.6 に Local FEC プロトコルのデータパケット，パリティパケット，NAK の振舞を示す．ここで，データパケットおよびパリティパケットの通信処理に必要な時間を表す確率変数を X とし，NAK の通信処理に必要な時間を表す確率変数を Y とし，それらの期待値および 2 次モーメントはそれぞれ $E[X], E[X^2], E[Y], E[Y^2]$ で表されるとする．

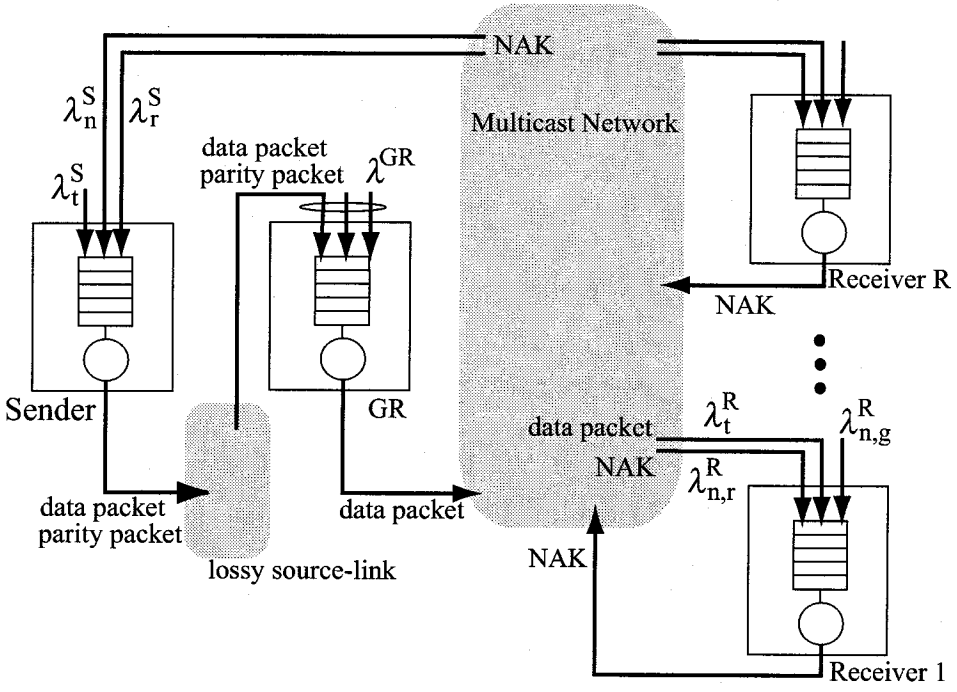


図 4.6: Packet flows under local FEC.

(1) 送信ノード

図 4.6 に示すように，送信ノードのトランスポート層では次の 3 種類のパケットが処理される．

- 本来送信すべきデータパケットとパリティパケット（到着率：λ_t^S）
- 再送を引き起こす NAK（到着率：λ_r^S）
- 再送を引き起こさない重複した NAK（到着率：λ_n^S）

この3種類のパケットの到着率をそれぞれ $\lambda_t^S, \lambda_r^S, \lambda_n^S$ とすると，それらは以下のように求められる．

$$\begin{aligned}\lambda_t^S &= \frac{n}{k} \times \lambda \\ \lambda_r^S &= \lambda(E[M]-1) \\ \lambda_n^S &= \lambda E[N] - \lambda_r^S\end{aligned} \tag{4.6}$$

また，この3種類のパケットの到着に対するサービス時間はそれぞれ $X, X+Y, Y$ で表される．したがって，送信ノードの利用率 ρ_S は，

$$\rho_s = \lambda(\frac{n}{k} + E[M] - 1)E[X] + E[N]E[Y] \tag{4.7}$$

となり，ポラチェック－ヒンチンの平均値公式 ($W = \frac{\lambda \bar{x^2}}{2(1-\rho)}$) を用いて，送信ノードにおける平均待ち時間 $E[W_S]$ は以下のように求められる．

$$E[W_S] = \frac{\lambda_t^S E[X^2] + \lambda_r^S(E[X^2] + E[Y^2] + 2E[X]E[Y]) + \lambda_n^S E[Y^2]}{2(1-\rho_S)} \tag{4.8}$$

(2)GR

GRのトランスポート層では，次の3種類のパケットが処理される[*5]．

- 送信ノードからの到着するデータパケットとパリティパケット
- GRのキャッシュから到着するキャッシングされていたデータパケット
- GRのFECデコーダから到着する復元されたデータパケット

ここで，GRにおいて到着したデータパケットがキャッシングされる確率を求めておく．GRにあるデータパケットが到着した時に，そのデータパケットの属するFECグループのパケットロスが既にGRにおいて検出されており，かつそのロス総数が $n-k$ 個以下の場合に，そのデータパケットはGRでキャッシングされる．よって，データパケットがキャッシングされる確率を $P[cache]$ とすると，$P[cache]$ は以下のように求められる．

$$P[cache] = \frac{1}{k}\left[\sum_{i=1}^{n-k}\{1-(1-p_s)^{i-1}\} + \sum_{i=n-k+1}^{k}\sum_{j=1}^{n-k}\binom{i-1}{j}p_s^j(1-p_s)^{i-j-1}\right] \tag{4.9}$$

また，あるパケット (i) がGRに到着した場合に，パケット (i) の属するFECグループの復元が行われる確率を p_{re} とすると，p_{re} は以下のように求められる．

$$p_{re} = \sum_{j=0}^{n-k-1}\binom{n-1}{j}p_s^j(1-p_s)^{n-j-1} \tag{4.10}$$

[*5] 再送パケット及びNAKはGRのネットワーク層で処理されるため，トランスポート層には到着しない．

よって，3 種類のパケットの到着率の合計を λ^{GR} とすると，$P[cache]$ を用いて λ^{GR} は以下のように求められる．

$$\lambda^{GR} = \frac{n}{k}\lambda(1-p_s) + \lambda(1-p_s)P[cache] + \lambda p_s p_{re} \tag{4.11}$$

また，この 3 種類のパケットの到着に対するサービス時間は X で表される．よって，GR の利用率 ρ_S は，

$$\rho_{GR} = \lambda^{GR}E[X] \tag{4.12}$$

となり，先と同様にして GR における平均待ち時間 $E[W_{GR}]$ を以下のように求められる．

$$E[W_{GR}] = \frac{\lambda^{GR}E[X^2]}{2(1-\rho_{GR})} \tag{4.13}$$

(3) 受信ノード

受信ノードのトランスポート層には次の 3 つのパケットが処理される．

- 送信ノードから到着するデータパケット（到着率：λ_t^R）
- パケットロスの検出によって生成される NAK（到着率：$\lambda_{n,g}^R$）
- 他の受信ノードから到着する NAK（到着率：$\lambda_{n,r}^R$）

この 3 種類のパケットの到着率を求めるために，まず送・受信ノード間のロス率を求める．Local FEC プロトコルでは，1 回目に送信されるデータパケットのみが GR のトランスポート層で処理され，2 回目以降に送信される再送パケットは GR のトランスポート層で処理されないため，1 回目の送信パケットと再送パケットでは送・受信ノード間のロス率が異なる．1 回目に送信されるデータパケットの送・受信ノード間のロス率を p_1，再送パケットの送・受信ノード間ロス率を p とすると，それぞれ

$$p_1 = 1-(1-p_{GR})(1-p_d) \tag{4.14}$$
$$p = 1-(1-p_s)(1-p_d) \tag{4.15}$$

となる．受信ノードで処理される 3 種類のパケットの到着率をそれぞれ $\lambda_t^R, \lambda_{n,g}^R, \lambda_{n,r}^R$ とすると，それらは以下のように求められる．

$$\begin{aligned}\lambda_t^R &= \lambda(1-p_1) + \lambda(E[M]-1)(1-p)\\ \lambda_{n,g}^R &= \frac{1}{R}\lambda E[N]\\ \lambda_{n,r}^R &= \frac{R-1}{R}\lambda E[N]\end{aligned} \tag{4.16}$$

ここで，$\lambda_{n,g}^R, \lambda_{n,r}^R$ のパケットに対するサービス時間は Y，λ_t^R パケットに対するサービス時間は X で表される．よって，受信ノードの利用率 ρ_R は，

$$\rho_R = \lambda[\{(1-p_1) + (E[M]-1)(1-p)\}E[X] + E[N]E[Y]] \tag{4.17}$$

となり，受信ノードにおける平均待ち時間 $E[W_R]$ を以下のように求められる．

$$E[W_R] = \frac{\lambda_l^R E[X^2] + (\lambda_{n,g}^R + \lambda_{n,r}^R)E[Y^2]}{2(1-\rho_R)} \tag{4.18}$$

4.5.4 全遅延解析

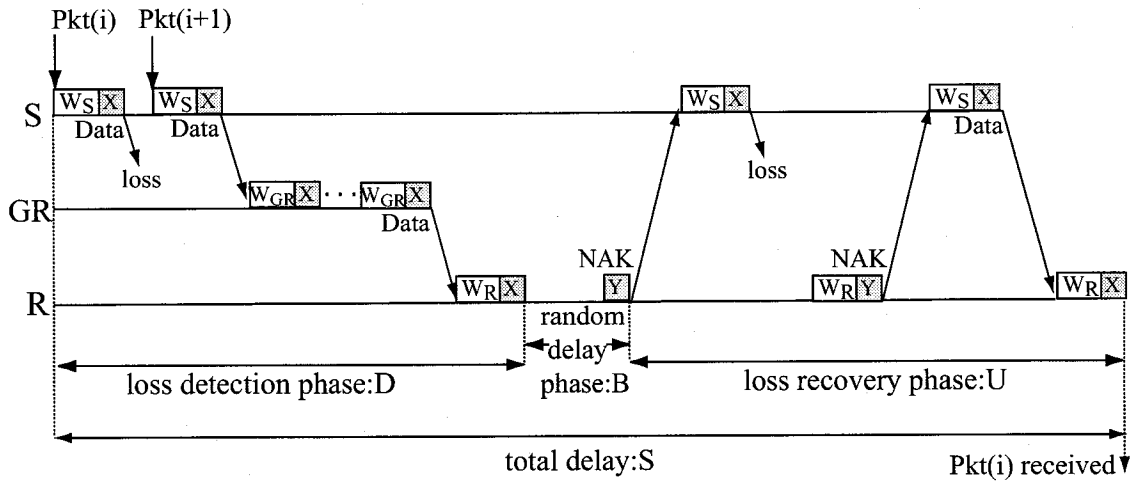


図 4.7: Delay components of local FEC.

図 4.7 に Local FEC プロトコルにおけるパケット伝送の様子を示す．パケット (i) が送信ノードに到着し，通信処理後に送信されたものの，ソースリンクでロスとなり，GR では次のパケット $(i+1)$ を受信する．ここで，GR はパケット $(i+1)$ をキャッシングし，パケット $(i),(i+1)$ の属する FEC グループを復元するまでフォワードを待機する．図 4.7 では，GR がロスパケットの復元が不可能と判断し，キャッシングしていたパケットをフォワードする様子が示されている．受信ノードが GR から送出されたパケット $(i+1)$ を受信すると，パケット (i) のロスを検知できる．パケット (i) が送信ノードに到着し，任意の受信ノードにおいてそのロスが検出される時点までの時間をロス検出フェーズと呼ぶ．その後，パケットロスを検出した受信ノードはランダム時間だけ NAK の送信を待機し，これらの受信ノードのうち最もランダム時間が短く設定されたものが NAK をマルチキャストし，送信ノードへロスパケットの再送を要求する．任意の受信ノードでパケット (i) のロスを検出し，パケット (i) に対する NAK がマルチキャストされる時点までの時間をランダム時間フェーズと呼ぶ．受信ノードよりマルチキャストされた NAK を送信ノードが受信すると再送がマルチキャストにより行われるが，図 4.7 では 2 回目の再送で正しく受信される様子が示されている．受信ノードから NAK が最初にマルチキャストされてから，パケットが正しく受信されるまでの時間をロス回復フェーズと呼ぶ．本節ではロス検

出フェーズ，ランダム時間フェーズ，ロス回復フェーズの平均遅延をそれぞれ求め，最後に，これらを合わせた全遅延を求める．

ロス検出フェーズ

いま着目しているパケット (i) のロス検出フェーズの長さは，i 以降のパケットの振舞いに依存する．このため，i 以降の任意のパケット (j) の振舞いについて考察した後に，i のロス検出フェーズを解析する．その際，パケット (j) の送信ノード－ GR 間遅延を求め，それをもとにロス検出フェーズの遅延を求める．

[送信ノード－ GR 間遅延]

ここでは，パケット (j) が送信ノードに到着し，GR から送出されるまでの時間を求める．パケット (j) がソースリンクでロスしない場合と，ソースリンクでロスしたが GR において復元された場合の 2 つに分けて考える．

(1) パケット (j) がソースリンクでロスしない場合

パケット (j) が GR に到着しキャッシングされた後に，パケット (j) がフォワードされるのは，ロスパケットの復元が行われた場合と GR がロスパケットの復元が不可能だと判断した場合のいずれかである．これらの場合に，パケット (j) が GR において待機する時間を求めるために，以下の確率変数を定義する．

M_{ID} パケット (j) と同一 FEC グループに属するパケットのうち，k 番目に GR に到着するパケットのパケット番号

N_{re} パケット (j) が GR に到着してから，GR においてパケットの復元が行われるまでに GR に到着したパケット数

N_{un} GR でロスパケットの復元が不可能な場合，パケット (j) が GR に到着してから，GR がパケットの復元が不可能だと判断するまでに GR に到着したパケット数

ここで，N_{re}, N_{un} の期待値は以下のように求められる．

$$E[N_{re}] = \sum_{j=2}^{k} \sum_{m=k+1}^{n} (m-j) P[M_{ID} = m] \frac{1}{k-1} \tag{4.19}$$

$$E[N_{un}] = \sum_{h=n-k+2}^{n} \sum_{j=2}^{k} (h-j)_+ \sum_{l=0}^{h-2} \left[\binom{h-l-2}{n-k} \cdot p_s^{n-k} (1-p_s)^{h-l-n+k-2} p_s^l \right] p_{GR}(1-p_{GR}) \frac{1}{k-1} \tag{4.20}$$

ただし，

$$P[M_{ID} = m] = \binom{m-2}{k-2} (1-p_s)^{k-2} p_s^{m-k}$$

$$(h-j)_+ = \begin{cases} h-j & ;h-j \geq 0 \\ 0 & ;h-j < 0 \end{cases}$$

である．

(2) パケット (j) がソースリンクでロスした場合

この場合，パケット (j) が GR から送出される時点は，GR において復元が行われた後となる．パケット (j) が GR において復元されるまでの時間を求めるために以下の確率変数を定義する．

M_{ID}^L パケット (j) と同一 FEC グループに属するパケットのうち，k 番目に GR に到着するパケットのパケット番号

N_{re}^L パケット (j) がロスしてから，パケット (j) が GR において復元されるまでに GR に到着するパケット数

ここで，N_{re}^L の期待値は以下のように求められる．

$$E[N_{re}^L] = \sum_{j=1}^{k} \sum_{m=k+1}^{n} (m-j) P[M_{ID}^L = m] \frac{1}{k} \tag{4.21}$$

ただし，

$$P[M_{ID}^L = m] = \begin{pmatrix} m-2 \\ k-1 \end{pmatrix} (1-p_s)^{k-1} p_s^{m-k-1} (1-p_s)$$

である．

以上より，パケット (j) が送信ノードに到着し，GR から送出されるまでの遅延を表す確率変数を D_{GR} とすると，D_{GR} の期待値は以下のように求められる．

$$\begin{aligned} &E[D_{GR}] = \\ &(1-p_s)\Bigg[P[cache]\Bigg\{p_{re}\Bigg(E[N_{re}]\frac{1}{\lambda_t^S} + \alpha\Bigg) + p_{un}\Bigg(E[N_{un}]\frac{1}{\lambda_t^S} + (1-p_s)\alpha\Bigg)\Bigg\} + \\ &(1-P[cache])(E[W_{GR}] + E[X]) + E[W_S] + E[X] + \tau_{GR}\Bigg] + \\ &p_s p_{re}\Bigg(E[N_{re}^L]\frac{1}{\lambda_t^S} + E[W_S] + E[X] + \tau_{GR} + \alpha\Bigg) \end{aligned} \tag{4.22}$$

ただし，

$$p_{un} = 1 - p_{re}$$

$$\alpha = \left\{\frac{1}{2}(k+1) - \sum_{l=1}^{k} l p_s (1-p_s)^{(l-1)} + 1\right\}(E[W_{GR}] + E[X])$$

である．

任意に選ばれた受信ノードがパケット (i) をロスし，さらに K 個の受信ノードも同様にパケット (i) をロスしたという条件の下で，それら $K+1$ 個の受信ノードがパケット (i) 以降の S パケットを連続してロスする確率分布は，以下のように求められる．

$$P[S = s|K = k] = \sum_{u=0}^{s} \begin{pmatrix} s \\ u \end{pmatrix} p_{GR}^{u}(1-p_{GR})^{(s-u)} p_d^{(k+1)(s-u)} \left[1 - \{p_{GR} + (1-p_{GR})p_d^{k+1}\}\right] \tag{4.23}$$

また，その期待値は以下のように求められる．

$$E[S|K = k] = \sum_{s=0}^{\infty} sP[S = s|K = k] \tag{4.24}$$

先に求めた送信ノード－GR 間遅延を加味し，パケット (i) が送信ノードに到着し，任意の受信ノードにおいてそのロスが検出されるまでの平均遅延を求める．ロス検出フェーズは任意の受信ノードが $S+1$ 番目のパケットを受信した時点で終了する．よってロス検出フェーズは，$S+1$ 番目のパケットの送信ノードにおける処理時間及び伝搬遅延 (送信ノード－GR 間遅延+GR －受信ノード間伝搬遅延) そして受信ノードにおける処理時間により構成される．ロス検出フェーズの遅延を表す確率変数を D とすると，D の期待値は以下のように求められる．

$$\begin{aligned} &E[D] = \\ &\left\{\sum_{k=0}^{R-2} \begin{pmatrix} R-1 \\ k \end{pmatrix} (1-p_{GR})p_d^k(1-p_d)^{R-k-1} E[S|K = k] + p_{GR}E[S|K = R-1]\right\} \frac{1}{\lambda_t^S} + \\ &E[D_{GR}] + E[W_R] + E[X] + \tau - \tau_s \end{aligned} \tag{4.25}$$

ランダム時間フェーズ

ロスを検出した受信ノードのうち，最も短くランダム時間が設定された受信ノードにより NAK がマルチキャストされ，ロス回復フェーズが起動される．パケット (i) をロスした H 個の受信ノードのうち，パケット $(i+1)$ を受信することによりロスを検出する受信ノード数を表す確率変数を G とすすると，H で条件付けられた G の確率分布は以下のように求められる．

$$P[G = g|H = h, H \geq 1] = \frac{\begin{pmatrix} R-1 \\ h-1 \end{pmatrix}(1-p_{GR})(1-p_d)^g p_d^{(h-g)}}{1 - \{p_{GR} + (1-p_{GR})p_d^h\}} \tag{4.26}$$

以上より，ランダム時間フェーズにおける遅延を表す確率変数を B とすると，その期待値は以下のように求められる．

$$
\begin{aligned}
E[B] = & \sum_{h=1}^{R-1} \binom{R-1}{h-1} (1-p_{GR}) p_d^{h-1} (1-p_d)^{R-h} \cdot \\
& \sum_{g=1}^{h} P[G=g|H=h, H \geq 1] E[D^g] + p_{GR}\{(1-p_{GR}) p_d^R\} \cdot \\
& \sum_{g=1}^{R} P[G=g|H=R, H \geq 1] E[D^g] + E[W_R] + E[Y]
\end{aligned} \tag{4.27}
$$

ここで，$E[D^g]$ はスを検出した g 個の受信ノード中で最も短いランダム時間の期待値を表しており，ランダム時間が $(0, T)$ の一様分布に従うとすれば，

$$
E[D^g] = \frac{T}{g+1}
$$

である．

ロス回復フェーズ

ロス回復フェーズは，再送パケットがロスしている段階では受信ノードでのタイムアウト (一定値 T_R とする) と次に送信する NAK のパケット処理時間の両者の組合せが繰り返される．その後，最後の NAK の送信と送信ノードの処理時間及び再送パケットの伝搬遅延そして受信ノードでの処理時間により，最終的に正しく受信される．以上より，ロス回復フェーズの遅延を表す確率変数を U とすると，その期待値は以下のように求められる．

$$
\begin{aligned}
E[U] &= (T_R + E[W_R] + E[Y]) \sum_{j=0}^{\infty} j p^j (1-p) + 2\tau + E[W_S] + E[X] + E[W_R] + E[X] \\
&= \frac{p(T_R + E[W_R] + E[Y])}{1-p} + E[W_S] + E[W_R] + 2(E[X] + \tau)
\end{aligned} \tag{4.28}
$$

全遅延

前節までに求めた，ロス検出フェーズ，ランダム時間フェーズ，ロス回復フェーズの平均遅延をそれぞれの発生確率により重み付けして加えて，全遅延を求める．全遅延を表す確率変数を S とすると，(4.14) 式の結果を用い，その期待値は以下のように求められる．

$$
E[S] = (1-p_1)\{E[D_{GR}] + E[W_R] + E[X] + (\tau - \tau_{GR})\} + p_1(E[D] + E[U] + E[B]) \tag{4.29}
$$

4.5.5 性能評価

図 4.8，4.9 は Local FEC プロトコル及び NAK プロトコルにおける受信ノード数－平均遅延特性である．ここで，平均遅延は送受信ノード間伝搬遅延 τ で正規化されている．なお，NAK プロトコルの特性は文献 [46] の解析結果を用いて求めている．図 4.8 より Local FEC プロトコルの場合，NAK プロトコルと比較して多数の受信ノードを収容可能であることが分かる．これは，多数の制御パケットの発生原因であるソースリンクでのロスに対し，GR でロスしたパケットを復元することにより受信ノードにおける NAK の発生が抑圧されるためである．したがって，マルチキャストグループ内でのロスの大半がマルチキャストツリーの枝にあたるエッジリンクでのロスとなり，NAK suppression によって NAK 数を大幅に減らし送信ノードの負荷を軽減することが可能となる．また，ソースリンクでのロスを低減させることによで再送回数が減少し，再送に要する時間が短縮されることにより遅延が短縮される．また，$(n,k) = (5,3)$ の場合に遅延が最も小さくなっているが，これは冗長度 $(\frac{n}{k})$ の増加によって GR でのロスパケットの復元確率が高くなっていることに加え，k が小さいほど GR におけるフォワード待機時間が短くなるためである．実際，図 4.9 のように冗長度を $\frac{n}{k} = \frac{5}{3}$ に固定した場合でも k が小さくなるほど，平均遅延は短縮されている．

図 4.10 は，ソースリンクにおけるネットワーク使用帯域特性であり，Local FEC プロトコルにおけるオーバーヘッドパケット（再送パケットおよびパリティパケット），再送パケット，パリティパケット，および NAK プロトコルにおける再送パケットの平均送信回数を示している．Local FEC プロトコルでは，オーバーヘッドパケットの送信回数が，オーバーヘッドパケットによるソースリンクのアップリンク使用帯域[*6]を表している．なお，図 4.10 では，横軸に FEC パラメータ n をとっており，k に関しては 7 に固定している．図より，再送パケットとパリティパケットは n に依存して変化しているが，オーバーヘッドパケットは n の値に関係なく一定である．また，オーバーヘッドパケットの送信回数は，NAK プロトコルにおける再送パケットの送信回数より少なくなっており，アップリンク使用帯域の点で Local FEC プロトコルは性能が向上している．この結果より，Local FEC プロトコルにおいて冗長なパリティパケットがソースリンクを伝送される場合でも，NAK プロトコルと比べソースリンクの輻輳を引き起こしにくいといえる．

[*6] NAK プロトコルにおいては，再送パケットの送信回数がソースリンクのアップリンク使用帯域を表す．

表 4.1: Parameters and assumption for simulation.

パケット到着率 λ	0.125
送信ノード－GR 間ロス率	5.0 %
WAN ルータロス率	0.1 %
MAN ルータロス率	1.0 %
LAN ルータロス率	0.0 %
平均プロトコル処理時間 (データパケット)	500μsec
平均プロトコル処理時間 (NAK)	100μsec

4.6 シミュレーション

4.5 では解析を簡単にするため，ノード間伝搬遅延が一定で均質である仮定を設けた．本節では，シミュレーションにより不均質ネットワークにおける Local FEC プロトコルの性能評価を行い，さらに前節で評価できなかったネットワーク全体での使用帯域量を評価する．

4.6.1 ネットワークモデル

シミュレーションで用いたネットワークトポロジーは図 4.11 に示す Tiers モデル [59] により生成し，WAN (Wide Area Network)・MAN・LAN により構成されるインターネットの階層構造をモデル化している．なお，送信ノードが属する LAN および MAN にはエンドホストとしては送信ノードのみが存在し，GR を経由しない受信ノードは存在しないものとする．また，FEC の符号化，復号化時間は，送受信ノード間の伝搬遅延と比べ非常に小さいため [51, 58]，本シミュレーションでは考慮しない．その他のシミュレーション条件を表 4.1 に示す．

4.6.2 性能評価

図 4.12 に Local FEC プロトコル及び NAK プロトコルにおける受信ノード数－平均遅延特性を示す．図 4.12 より，不均質ネットワークモデルにおいても，Local FEC プロトコルが NAK プロトコルと比較して多数の受信ノードを収容可能であることがわかる．また，図 4.8 と比較して，図 4.12 の場合は NAK プロトコルの平均遅延が短縮されている．これは，不均質ネットワークにおいてソースリンクでロスが発生した場合には，全ての受信ノードがロスを被るために送信ノードの近隣に位置する受信ノードが早く NAK をマル

チキャストする確率が高く，その結果再送が早く行われるためである [47]．しかし，Local FEC プロトコルでは，ソースリンクでのロスを FEC により低減させているため，不均質ネットワークによる影響は薄くなっている．

図 4.13 は，受信ノード数－ NAK 使用帯域特性を示している．ここで，NAK 使用帯域とは，データパケット 1 個あたりに発生する NAK パケットの総ホップ数を表しており，図 4.13 の縦軸は Local FEC プロトコルの NAK 使用帯域を NAK プロトコルの NAK 使用帯域で正規化している．図 4.13 より，Local FEC プロトコルは NAK プロトコルと比較して NAK によるネットワーク帯域の消費を 30% 程度軽減していることが分かる．これは，Local FEC プロトコルではソースリンクでのロスが低減することにより NAK suppression が有効に働き冗長な NAK の発生が抑圧されるためである．

図 4.14 は，受信ノード数－ネットワーク使用帯域特性を示している．ここで，ネットワーク使用帯域とは，データパケット 1 個あたりに発生するデータパケット (再送パケットを含む)，パリティパケット，NAK パケットの総ホップ数を表しており，図 4.14 の縦軸は Local FEC プロトコルのネットワーク使用帯域を NAK プロトコルの NAK によるネットワーク使用帯域で正規化している．図 4.14 より，パリティパケットを含む全てのパケットによるネットワーク帯域の消費を考慮しても，図 4.13 と同様に NAK の発生数が NAK プロトコルより減少しているため，Local FEC プロトコルはネットワーク帯域の消費が少なく抑えられている．Local FEC プロトコルでは NAK プロトコルには存在しないパリティパケットのホップ数が加算されるが，パリティパケットの伝搬する部分はネットワークのソースリンク部分のみであり，NAK によるネットワーク帯域の使用が大きく影響しているため，ネットワーク全体を対象とした場合，Local FEC プロトコルがネットワーク帯域の点でも優れていることが分かる．

4.7 結言

本章では，ネットワークレベルマルチキャストの重要な技術課題である信頼性保証に着目し，誤り訂正符号 (FEC) を使用して再送制御の補助をネットワーク内で部分的に行う方式を提案した．提案方式は，信頼性マルチキャストにおいてスケーラビリティに大きく影響を及ぼすソースリンクでのロスを考慮しソースリンクにのみ FEC を適用することで，feedback implosion の原因となる NAK の発生をを抑圧することが可能となる．

また，提案方式に対し，パケットが送信ノードに到着してからランダムに選択された受信ノードに正しく受信されるまでの平均遅延を数学解析により求めた．均質ネットワークモデルに対する遅延解析による性能評価，および不均質ネットワークモデルに対するシミュレーションによる性能評価を行い，NAK プロトコルと比較して提案方式の性能がスケーラビリティおよびネットワーク使用帯域量の面で向上していることを明らかにした．

これらの結果より，本方式の有効性が実証された．

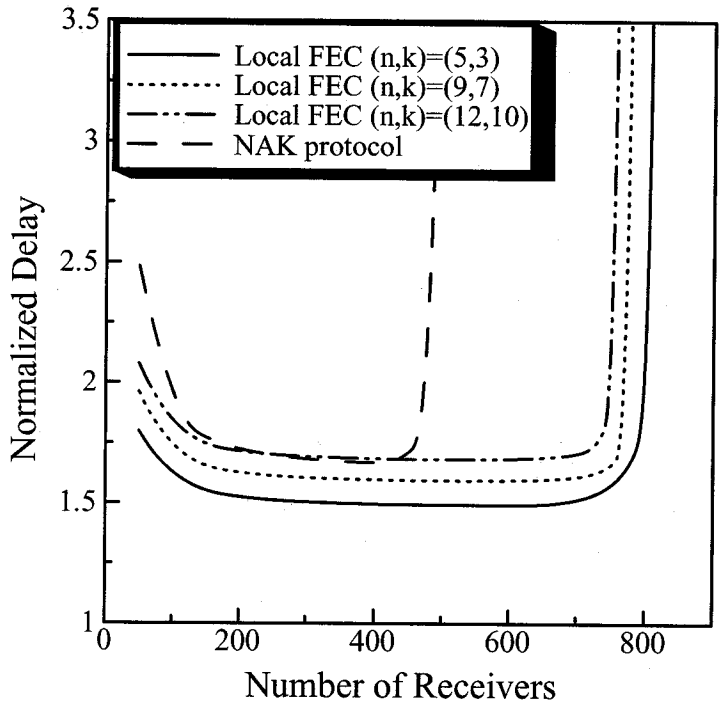


図 4.8: Normalized delay performance vs. number of receivers ($p_s = p_d = 5\%, \lambda = 0.125, \tau = 50msec, \tau_s = 15msec$).

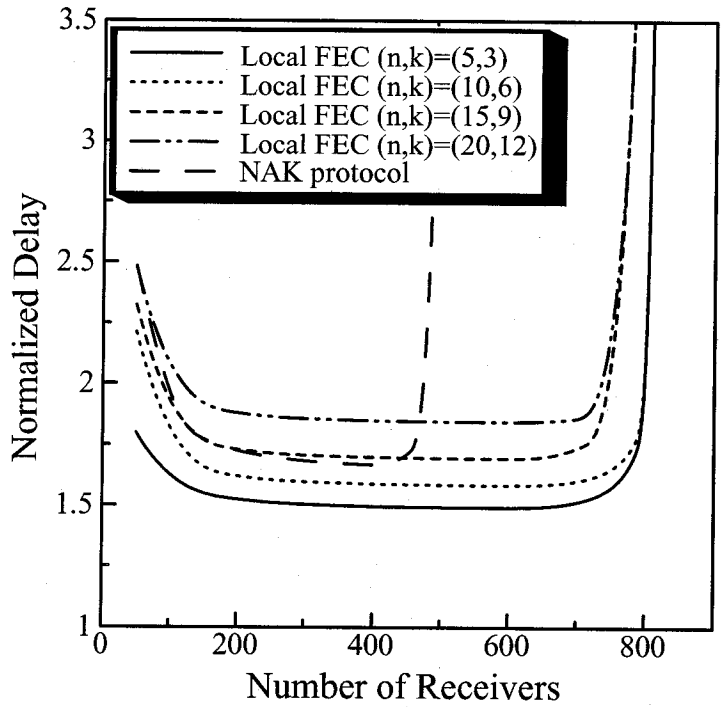


図 4.9: Normalized delay performance vs. number of receivers ($p_s = p_d = 5\%, \lambda = 0.125, \tau = 50msec, \tau_s = 15msec, \frac{n}{k}$ is fixed to $\frac{5}{3}$).

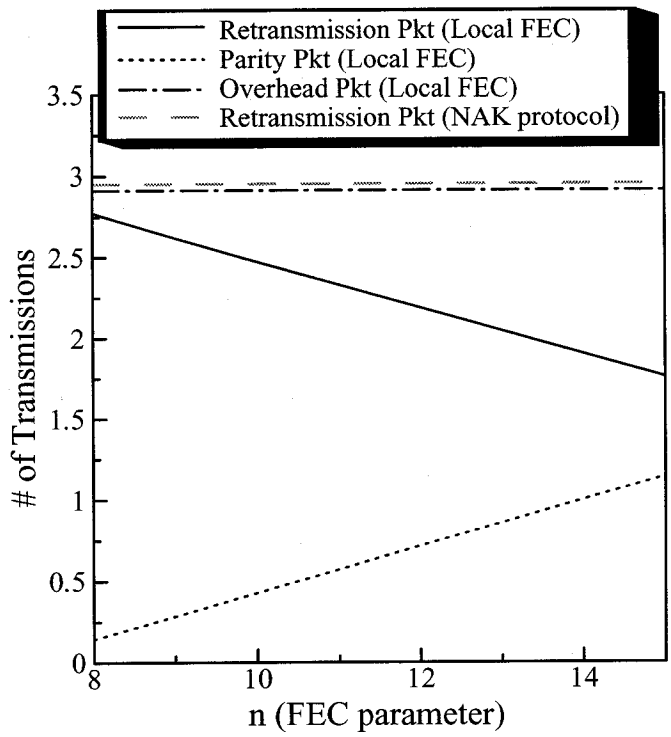


図 4.10: Bandwidth usage at the source link vs. FEC parameter n ($p_s = p_d = 5\%, k = 7, R = 300, \lambda = 0.125, \tau = 50msec, \tau_s = 15msec$).

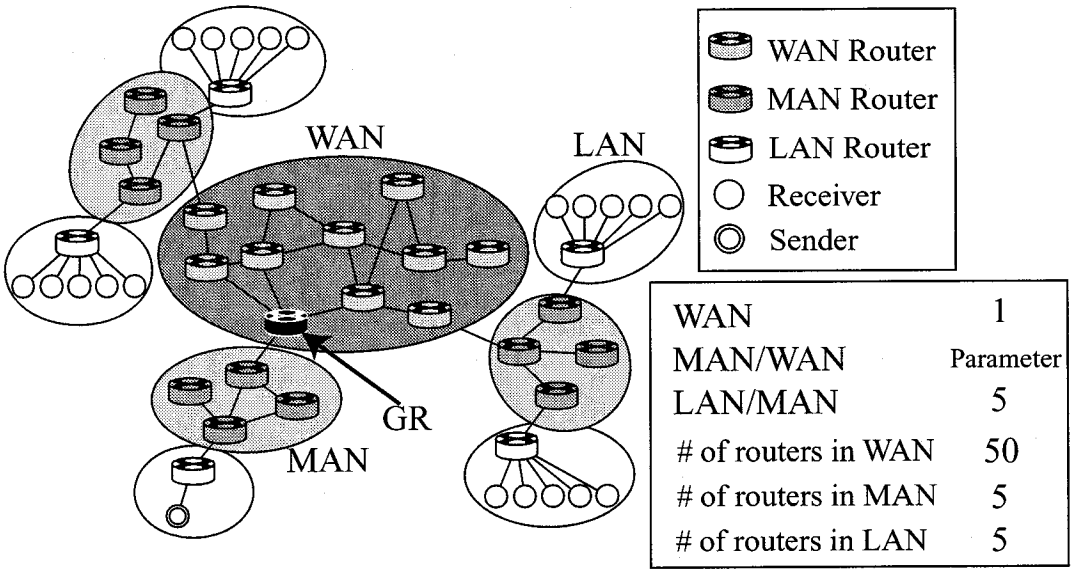


図 4.11: Tiers model.

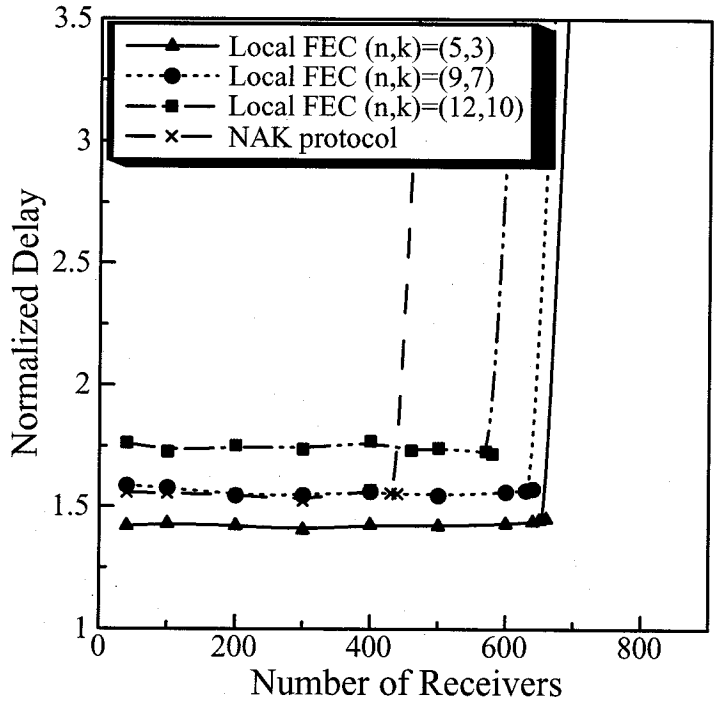


図 4.12: Normalized delay performance vs. number of receivers (p_s = 5%).

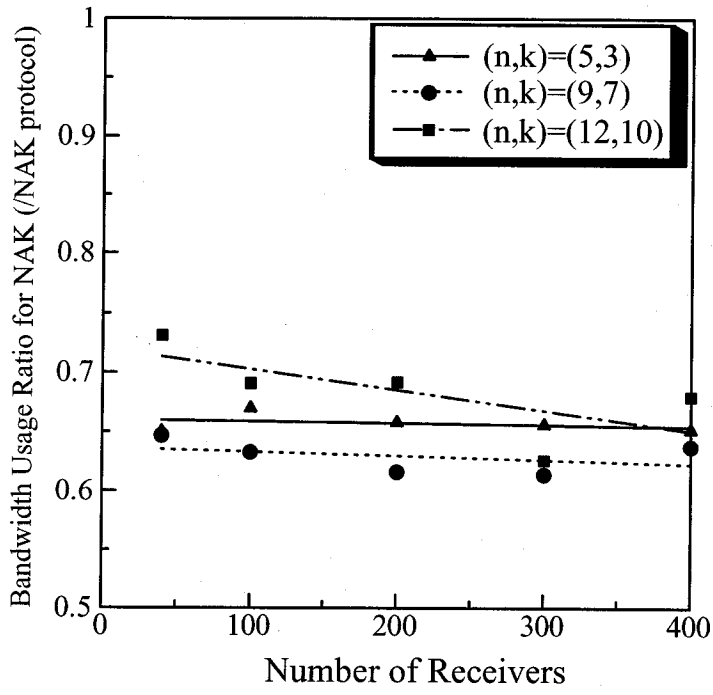


図 4.13: Bandwidth usage reduction for NAK vs. number of receivers (p_s = 5%).

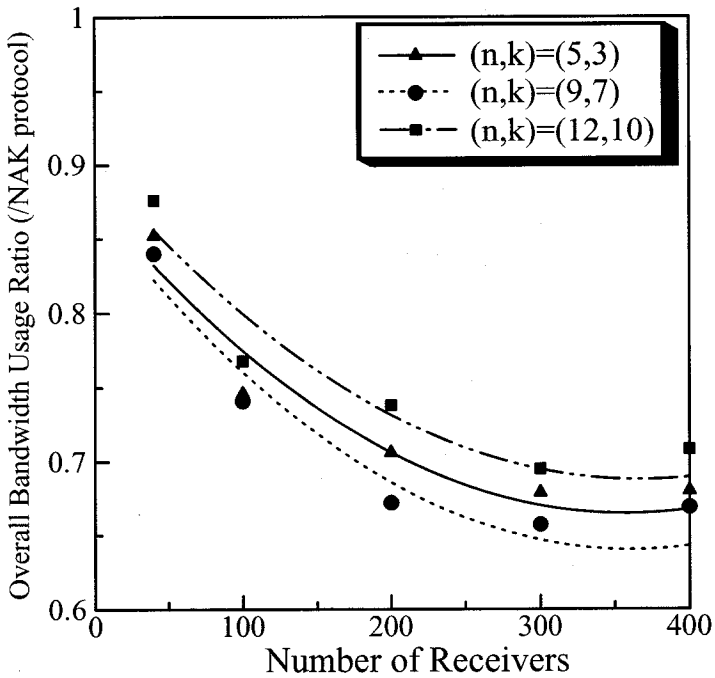


図 4.14: Overall bandwidth usage reduction vs. number of receivers (p_s = 5%).

第5章

ネットワークコーディングを用いたマルチキャスト転送制御

5.1 緒言

4章では，ネットワークレベルマルチキャストの問題点である，信頼性保証に主眼を置いた．本章では，ネットワークレベルマルチキャストのもう1つの重要な技術課題である，マルチキャストトラヒックの負荷分散について検討を行う．

現在のIPマルチキャストルーチングプロトコルでは，主に距離を基準として経路の選択がなされ，遅延の小さい特定のリンクが転送経路として選択される傾向が強い．このため，マルチキャストトラヒックが特定リンクに集中し，ネットワーク輻輳が引き起こされる．このトラヒックの集中問題は，IPマルチキャストの，送受信ノード間を単一の経路で情報伝送がなされるという伝送形態自体に起因する問題であるため，現状では有効な解決方法が見つかっていない．

一方，IPマルチキャストの伝送形態に疑問を抱いたRudolf Ahlswedeらは，ネットワーク内で情報の符号化・再構成を行いつつデータ伝送を行うことで，IPマルチキャストより効率的に1対多通信を実現することが可能であることを，ネットワークコーディング理論により証明した[60]．ネットワークコーディングを適用したネットワークレベルマルチキャストでは，ネットワーク伝送効率の理論的限界値である最大フローまで伝送効率を高めることが可能となる．

本章では，ネットワークレベルマルチキャストのトラヒック集中問題に主眼を置き，マルチキャストトラヒックの負荷分散について検討する．まず，ネットワークコーディングがトラヒックの負荷分散の点で有効なマルチキャスト転送制御であることを明らかにする．次に，負荷分散効果を得るネットワークコーディングの新たな利用方法を提案し，ネットワークレベルマルチキャストにおけるトラヒックの負荷分散の実現を目指す．

5.2 ネットワークコーディングの概要

IP マルチキャストでは，ルーチングプロトコルによって決定されたツリー構造の転送経路を用いて各受信ノードに対して単一経路を用いて情報が配送される．このため，送信ノードの送信レートは転送経路上のリンクの最小利用可能帯域の制約を受ける．また，ネットワーク内のリンクのうち，転送経路を構成するリンクへ特にトラヒックが集中し，複数のマルチキャストセッションにおいて同一のリンクが転送経路として選択されている場合には，そのリンクで輻輳が発生する．これらの問題は，単一転送経路を用いる IP マルチキャストの伝送形態に起因するものであるため，解決のためには IP マルチキャストとは異なる伝送形態をとる必要がある．

情報理論の分野で提案されたネットワークコーディング理論 [60] では，IP マルチキャストにおいて，ルータに対し，マルチキャスト転送（パケットの複製・転送）以外に，新たに転送情報を符号化・再構成する機能を付与することで，伝送効率が向上し，最大フローでのマルチキャスト通信が可能であることが示されている．また，ネットワークコーディングでは，情報の符号化に付随して複数経路を用いた情報配送がなされるため，特定リンクへのトラヒックの集中を抑制する効果も期待できる．

本節では，まず，ネットワークコーディングにおいてルータが転送する情報に対しどのように符号化・再構成を行うかを述べ，理論的背景を明らかにする．

5.2.1 最大フロー

ネットワークコーディングでは，ネットワーク内のルータにおいて符号化を行うことにより，送信ノードはネットワークの利用効率の理論上の最大値である最大フローの送信レートで情報の送信を実現することができる．ここではまず，1 対 1 通信における最大フローを定義し，続いてマルチキャスト通信における最大フローを定義する．

定義

頂点集合 V と辺集合 E からなるグラフ $G = (V, E)$ において，G の各辺 e の容量[*1]が $c(e)$ で表されるものとする．また，頂点 s 及び頂点 t をグラフ G の特定の 2 頂点とする．ここで，以下の 2 つの条件を満足する非負の実数値を割り当てる関数 f を頂点 s から頂点 t へのフローと呼ぶ．

[*1] インターネットのような通信ネットワークでいえば，各リンクのリンク帯域に相当する．

1. $0 \leq f(e) \leq c(e) \ \ for \ \forall e \in E$
2. s と t を除く全ての頂点 $v \in V$ において次の式が成り立つ．

$$\sum_{e \in out(v)} f(e) - \sum_{e \in in(v)} f(e) = 0$$

ここで，$out(v)$ は頂点 v を始点とする辺の集合で，$in(v)$ は頂点 v を終点とする辺の集合である．また，次式で与えられる F をフロー f の値といい，F を最大にするフロー f を最大フローという．

$$F = \sum_{e \in out(s)} f(e) - \sum_{e \in in(s)} f(e) \tag{5.1}$$

上記の最大フローは，送信ノードと受信ノードの 2 頂点間に対して定義されており，1 つの送信ノードと複数の受信ノード間で通信が行われるマルチキャスト通信にそのまま適用することはできない．一般的に，リンク容量や接続構造が不均質な現実のネットワークにおいてマルチキャスト通信を行う場合，受信ノードによって最大フローが異なる．また，各受信ノード間で受信能力が異なる場合に損失なく情報伝送を行うためには，伝送レートを最も受信能力の低い受信ノードに合わせる必要がある．したがって，本論文では，マルチキャスト通信に対する最大フローを以下の式で定義する．

$$F_{mcast} = \min_{t \in M} F(t). \tag{5.2}$$

ここで，$F(t)$ は送信ノード s から受信ノード t への最大フローであり，M はマルチキャストメンバーを表す集合である．

図 5.1 に示されるネットワークを用いてマルチキャスト通信の最大フローの例を示す．このネットワークを用いて，送信ノード S が受信ノード R_1, R_2, R_3 へ情報をマルチキャストする場合，S から R_1, R_2, R_3 への最大フローはそれぞれ 4，7，5 となる[*2]．したがって，最も小さい最大フローは 4 となり，定義よりこのマルチキャスト通信の最大フローは 4 となる．以降，本論文においては，「最大フロー」はこのマルチキャスト通信の最大フローを意味するものとする．

ネットワークコーディング

ルータの本来の機能は，受信した情報に対し経路選択を行ない，適切な経路上へ転送することである．また，ネットワークレベルマルチキャストでは，転送情報の複製機能がルータに追加される．一方，これら既存機能に加え，ルータに符号化機能を付加し，ルータにおいて受信情報を符号化してから転送する，新たな転送制御がネットワークコーディ

[*2] 最小カット最大フローの定理 [61] を用いることにより簡単に確かめられる．

ングである．このネットワークコーディングをマルチキャストに適用した場合，ルータにおいて複数経路から受信した情報を符号化し，複数の受信ノードにとって有効な別の情報に変化させ転送することにより，ネットワーク伝送効率の理論的限界値である最大フローまで伝送効率を高めてマルチキャスト伝送を行なうことが可能である．

ネットワークコーディングによる最大フロー伝送の例を図 5.2 に示す．図 5.2–(a) は，各リンクの容量を示しており，単位としてリンクの転送速度 (bits/unit time) を用いるものとする．よって，(5.2) 式より送信ノード S から頂点 R_1, R_2 へマルチキャスト通信を行う場合の最大フローは 2[bits/unit time] となる．このネットワークにおいて，送信ノード S が受信ノード R_1, R_2 へ情報をマルチキャストする場合，(5.2) 式より，最大フローは 2 となる．図 5.2–(b) より，IP マルチキャストでは，ルーチングプロトコルによって決定されたマルチキャストツリーに沿って単位時間当り 1 ビットを送信することができる．しかし，IP マルチキャストではリンク $(1-3)$, $(2-3)$, $(3-4)$, $(4-R_1)$, $(4-R_2)$ を使用しておらず，最大フロー 2[bits/unit time] を実現することはできない．一方，図 5.2–(c) では，送信ノード S が異なる情報 a, b を別々のリンクに送出しノード 3 において a, b を $a+b$ へ符号化して転送している[*3]．受信ノード R_1, R_2 はそれぞれ受信した a, $a+b$ 及び b, $a+b$ から送信ノードから送信された a, b の 2 つの情報を復号化することができる．このようにネットワークコーディングでは，複数経路を用いて情報を送信し，ルータに符号化機能を持たせることによって送信ノードは最大フローと等しい情報量をマルチキャスト伝送することができる．ここでは，図 5.2–(a) のネットワークに対するネットワークコーディングの一例を示した．しかし，ネットワークコーディングは図 5.2–(a) のネットワークにのみ適用できる技術ではない．文献 [60] において，あらゆるネットワークにおいて最大フローと等しい送信レートでマルチキャストができるネットワークコーディングの符合化方法が

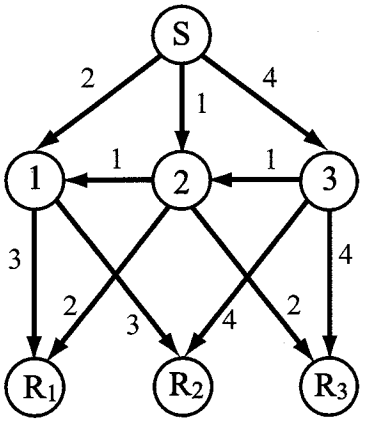


図 5.1: Max-flow in multicast communication.

[*3] ただし，+ は $GF(2)$ 上での加算を表している．

必ず存在することがネットワークコーディング定理として理論的に定式化されている．

5.3 ネットワークコーディングを用いたマルチキャスト転送制御の有効性

5.3.1 最大フロー伝送

ネットワークコーディングは，ネットワーク資源を効率的に利用し，1 対多通信における最大フロー伝送を実現する．本節では，ネットワークコーディングによる最大フロー伝送の有効性を検証するため，次の 3 つのマルチキャスト転送方式を比較する．

- ネットワークコーディングを適用したマルチキャスト
- 単一セッション IP マルチキャスト
- 複数セッション IP マルチキャスト

ネットワークコーディングを適用したマルチキャストでは，ネットワークコーディングにより最大フロー伝送が実現される．単一セッション IP マルチキャストは，既存の IP マルチキャストを表しており，送信ノードから受信ノードへのデータ配送には単一のマルチキャストツリーが利用される．一方，複数セッション IP マルチキャストでは，送信ノードから受信ノードへのデータ配送に，複数の異なるマルチキャストツリーが利用される．このため，複数セッション IP マルチキャストでは，単一セッション IP マルチキャストと比較してネットワーク資源の利用効率が向上する．

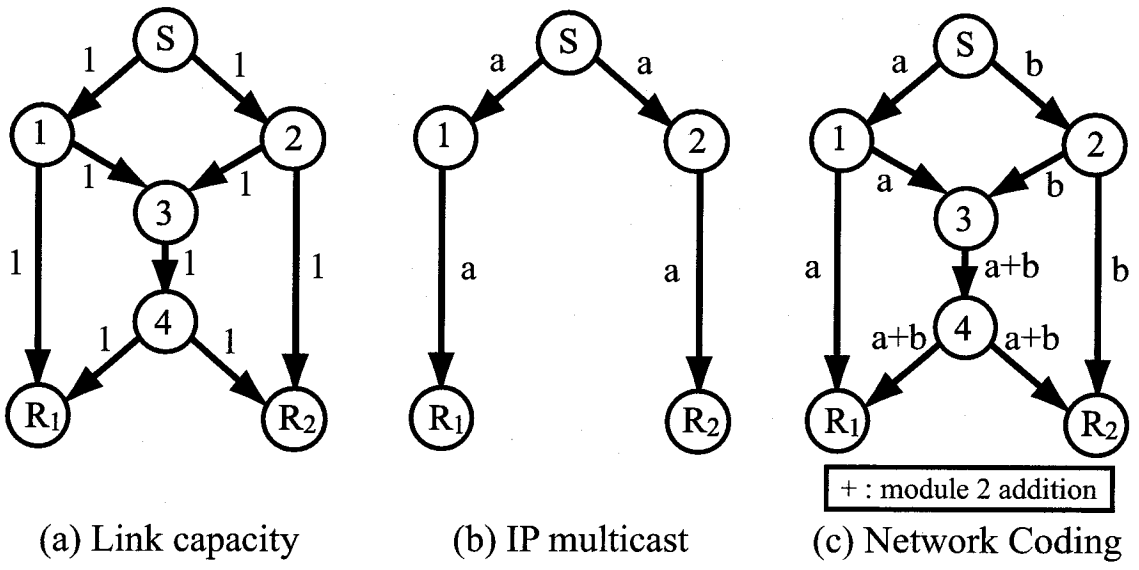
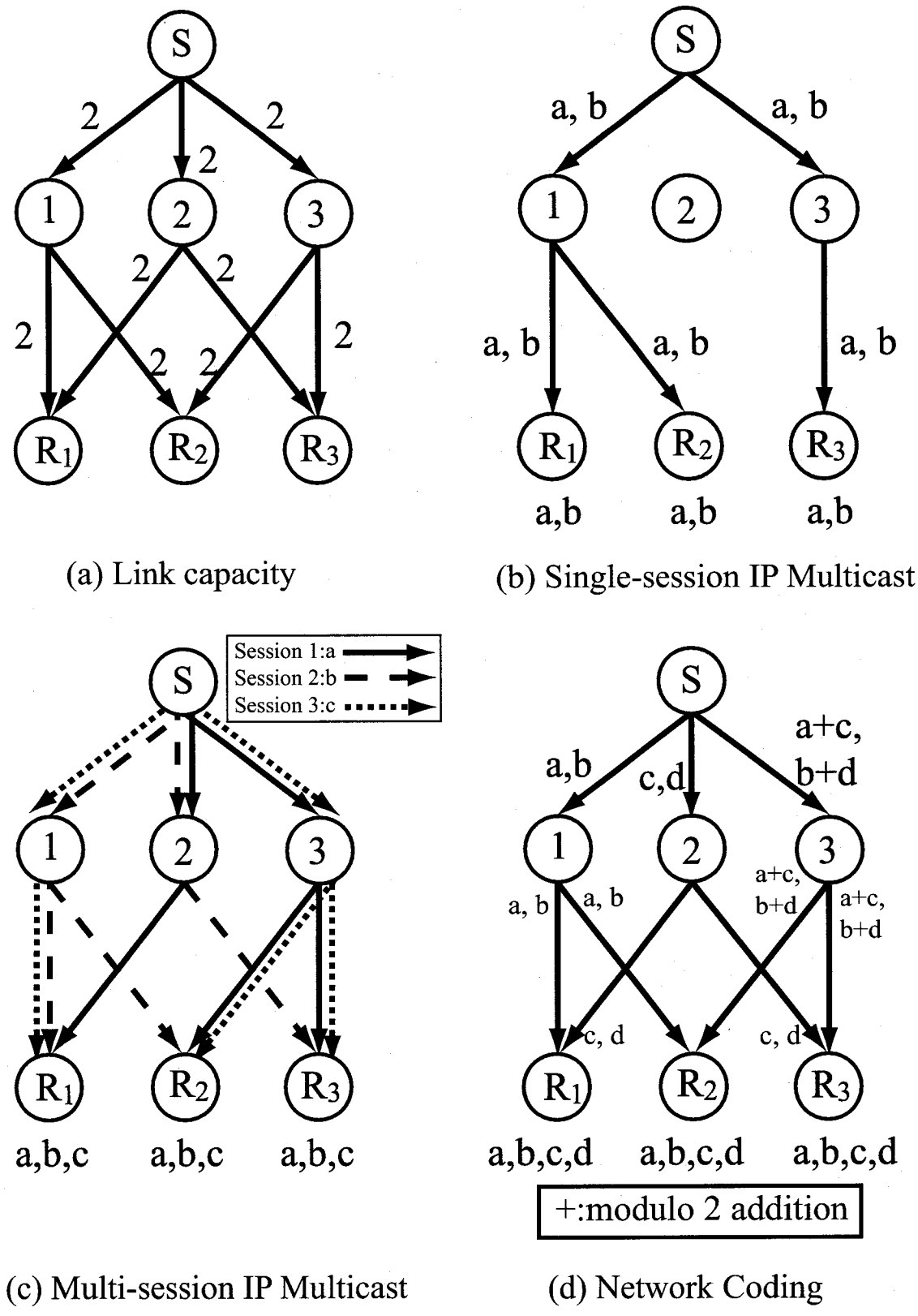


図 5.2: Transmission with max-flow.



(a) Link capacity

(b) Single-session IP Multicast

(c) Multi-session IP Multicast

(d) Network Coding

図 5.3: Comparison with existing approaches.

例として図 5.3–(a) で示されるネットワークを用いて，単一セッション IP マルチキャスト，複数セッション IP マルチキャストおよびネットワークコーディングを適用したマルチキャストの 3 つの方式を比較し有効性を検証する．図 5.3–(a) は，各リンクの容量（単位は転送速度 [bits/unit time]）を示しており，最大フローの値は 4[bits/unit time] となる．単一セッション IP マルチキャストでは，図 5.3–(b) で示されるように各受信ノードに対し単一経路を用いて情報を送信しているため，受信ノードにおけるスループットは各リンク容量と等しい 2[bits/unit time] であり，最大フローを実現できない．図 5.3–(c) に示される複数セッション IP マルチキャストでは，3 つのマルチキャストセッション 1, 2 および 3 が，a, b および c をそれぞれ伝送しており，合計で 3[bits/unit time] のスループットを実現している．しかし，単一セッション IP マルチキャストとは異なりネットワークを構成する全リンクを使用しているため，スループットが向上するのは定性的に見て当然である．一方，ネットワークコーディングは，図 5.3–(c) の複数セッション IP マルチキャストと同様に全てのリンクを使用し，さらに符号化を導入して情報を効率的に送信しており，最大フローと等しい 4[bits/unit time] のスループットを実現できている．このように，受信ノードにおけるスループットの点で，ネットワークコーディングは性能の向上が可能である．

5.3.2 トラヒックの負荷分散

ネットワークコーディング本来の目的は最大フロー伝送である．しかし，ネットワークコーディングを用いて最大フロー伝送を行う場合，転送経路上のいくつかのリンクにおいて利用可能なネットワーク帯域が完全に消費される．待ち行列理論 [62] の観点から眺めると，この状況はネットワーク輻輳を招くため，許容し難い．そのため，インターネットにおいて，最大フロー伝送を目的としてネットワークコーディングを利用することは現実的ではない．そこで本論文では，最大フロー伝送に代わるネットワークコーディングの新たな利用法を検討する．その際，ネットワークコーディングにおいて複数経路で情報伝送が行われる点に着目する．ネットワークコーディングでは，送受信ノード間の最短経路の他に，複数の経路を用いて情報が伝送されるため，トラヒックが多数のリンクに分散し，トラヒックの負荷分散効果が期待できる．

現在の IP マルチキャストルーチングプロトコルは，ネットワーク内で偏ったトラヒック分布を発生させると言われており [63]，トラヒックの集中に起因するネットワーク輻輳が問題となっている．したがって，トラヒックの負荷分散は，IP マルチキャストにおける最も重要な技術課題として認識されている．

マルチキャストルーチングプロトコルは，PIM-SM[8] や CBT[9] に代表される共有木型プロトコルと，DVMRP[14]，MOSPF[16] および PIM-DM[64] に代表される始点木型プ

ロトコルに大別される．共有木型プロトコルでは，複数の送信ノードが単一のマルチキャストツリーを共有し，データパケットや制御パケットなどのマルチキャスト通信に関連する全パケットがRP（Rendezvous Point）と呼ばれる中央ノードに送信されるため，RP近隣にトラヒックが集中する．一方，始点木型プロトコルではこのような問題は発生しない．しかし，近年の研究[65, 66]によって，インターネットトポロジには，べき乗則と呼ばれる，ルータの接続次数が偏る性質があることが明らかとなり，この偏りがトラヒックの特定リンクへの集中を招くことが指摘されている．これは，接続次数の大きいルータの接続リンクがマルチキャストツリーの一部分として選択されやすく，たとえ始点木型プロトコルを用いた場合でもトラヒックの集中問題が発生することを意味している．

ここでは，ネットワークコーディングを適用したマルチキャストとIPマルチキャストを比較することにより，ネットワークコーディングの負荷分散効果を検証する．図5.3–(d)のように各リンクの容量を使い切らずに，各リンクに1ビットのみを流すことによりIPマルチキャストと等しいスループットを実現する場合を考える．この場合，ネットワークコーディングと単一セッションIPマルチキャスト（図5.3–(b)）は同一のスループット，2[bits/unit time]となる．単一セッションIPマルチキャストでは図5.3-(b)に示されるように，ネットワークを構成する9本のリンクのうち5本のみに容量の限界値である2ビットが流れ（合計で10ビットのネットワーク資源が使用されたことになる），ネットワークが偏って利用されている．一方，ネットワークコーディングでは，9本のリンク全てに容量の半分である1ビットが流れ（合計で9ビットのネットワーク資源が使用されたことになる），ネットワーク全体を均等に利用している．よって，ネットワークコーディングを適用すれば，複数経路に分散させて情報を送信することによりトラヒックの負荷分散を実現できる．

また，ネットワークコーディングでは，ルータの入力リンクから入力された複数の情報（ビットもしくはパケット）が，符号化によって単一の情報（ビットもしくはパケット）に変換されて出力リンクへ出力されるため，ルータにおける符号化が情報の圧縮を意味する．このため，ネットワークコーディングを適用することで，ネットワーク使用帯域量の節約効果が得られる．上記の例では，単一セッションIPマルチキャストでは，合計10ビットのネットワーク資源が消費されるのに対し，ネットワークコーディングでの消費量は9ビットとなる．このように，この例では簡単な符号化を行うことにより，ネットワーク帯域使用量が10%の節約されていることがわかる．

以上のように，ネットワークコーディングでは，最大フロー伝送ではなくIPマルチキャストと同等のマルチキャスト伝送を行う場合に，トラヒックの負荷分散およびネットワーク使用帯域量の節約を実現しつつ効率的なマルチキャスト通信が可能となる．本論文では，最大フロー伝送に代わるネットワークコーディングの新たな利用法として，ここで示したトラヒックの負荷分散を目的とした利用法（以降，本利用法をネットワークコーディ

ングの負荷分散利用と呼ぶ）を提案する．

5.3.3 ネットワークコーディングの問題点

ネットワークコーディングを適用したマルチキャストでは，ネットワークコーディングにより生じる本質的な問題点がある．ここでは，その問題点について例を用いて説明する．図 5.3–(d) において，受信ノード R_2 は c, d を復号するためにそれぞれ a, $a+c$ と b, $b+d$ の 2 ビットの組が必要である．もし，a がネットワーク中でロスした場合，受信ノード R_2 は $a+c$ を正しく受信できたとしても c を復号することができないため，a, c のロスを被ることになる．すなわち，ネットワークコーディングでは，受信した符号化情報は単体では利用できないため，ロスによって復号が不可能となる場合には受信できた符号化情報もロスしたものと見なされる．

次に，受信ノード R_2 が先に $c+d$ を受信し，遅れて a を受信した場合を考える．この場合，受信ノード R_2 は a を受信するまで c を復元することができないため，c の受信は a を受信するまで遅延することになる．ネットワークコーディングでは復号に必要な情報が全て受信されるまで復号できないため，復号された情報の受信時刻は復号に必要な情報の受信時刻の影響を受ける．以上のように，ネットワークコーディングでは符号化を行うためにロスおよび遅延の影響を大きく受けると考えられる．

5.4 性能評価

文献 [60] は，ネットワークコーディングの基本概念を提案し，ネットワークコーディングを用いることで最大フロー伝送が可能であることを示した．しかし，ネットワークコーディングの既存 IP マルチキャストに対する，伝送容量の点での有効性の定量的評価はなされていない．また，インターネットにおいてネットワークコーディングを適用したマルチキャスト通信を実現するためには，ロスや遅延の発生する現実的なネットワークモデルを用いて性能評価を行い，伝送特性を明らかにする必要がある．そこで本節では，ネットワークコーディングを適用したマルチキャストの性能評価を，マルチキャスト伝送容量および伝送性能の 2 つの観点から行う．まず，パケットロスの発生しない理想的なネットワークモデルを用い，ネットワークコーディングを適用したマルチキャストによって実現されるマルチキャスト伝送容量を評価する．この際，比較対象として，単一セッション IP マルチキャストより潜在的に大きい伝送容量を有する複数セッション IP マルチキャストを用いる．次に，パケットロスおよび遅延の生じる現実的なネットワークモデルを用いて，ネットワークコーディングの本来の利用法である最大フロー伝送，および本論文で提案する新利用法であるトラヒックの負荷分散の両面からネットワークコーディングの性能

評価を行う．

5.4.1 マルチキャスト伝送容量

ネットワークコーディングのマルチキャスト伝送容量の評価の目的は，伝送容量の点でのネットワークコーディングの複数セッション IP マルチキャストに対する有効性の検証である．ネットワークコーディングを適用した場合のマルチキャスト伝送容量は最大フローであり，複数セッション IP マルチキャストのマルチキャスト伝送容量は，マルチキャストセッションの数，すなわち，同時に構築できる経路の異なるマルチキャストツリーの数が反映される．また，最大フローは，ネットワークの接続構造およびリンク容量に大きく依存する．そのため，ネットワーク構造にとらわれない一般的な評価を行うには，接続構造が異なる様々なネットワークを用いてシミュレーションを行う必要がある．本研究では，ネットワークトポロジとしてランダムグラフ [41] を採用する．その他のシミュレーションモデルを以下に示す．

- ネットワークのノード数は 50 ノードとし，受信ノード数は 10 ノードとする．
- リンク容量として次の 2 つのモデルを用いる．
  - ネットワークを構成する全リンクのリンク容量を 5 とする均質モデル
  - 各リンクのリンク容量を 1～10 の一様分布で与える不均質モデル

また，比較対象である複数セッション IP マルチキャストのツリー構築アルゴリズムを以下に示す．

複数マルチキャストツリー構築アルゴリズム

1. 対象とするネットワーク (G, s) に対し Dijkstra アルゴリズムを適用し，送信ノード s から各受信ノードへ最短経路を求め，マルチキャストツリー $T = (V_T, E_T)$ を作成する．
2. $$R_{min} = \min_{(i,j)\in E_T} R_{ij}$$
   とし，リンク $(i, j) \in E_T$ の容量 R_{ij} を $R_{ij} = R_{ij} - R_{min}$ に更新する．
3. 2 の操作で容量 $R_{ij} = 0$ となるリンクをネットワーク (G, s) から削除し，削除後のネットワークを新たに (G, s) とする．
4. 1～3 の操作を，マルチキャストツリーが作成できなくなるまで繰り返す．

ネットワークコーディングおよび複数セッション IP マルチキャストにおけるマルチキャスト伝送容量は以下の通りである．

マルチキャスト伝送容量

- ネットワークコーディング：最大フロー
- 複数セッション IP マルチキャスト：送信ノードから任意の受信ノードに対する各セッションでのフローの値を全てのセッションにわたり和をとったもの．

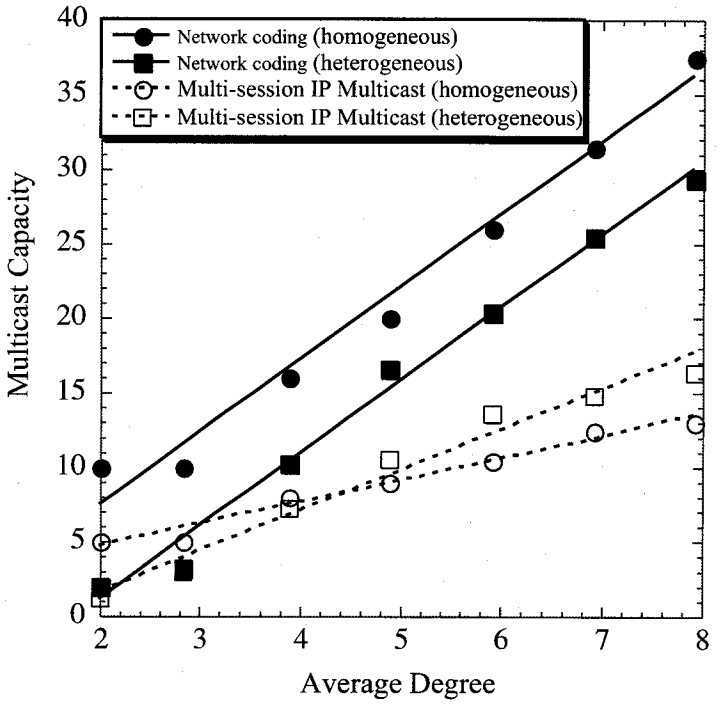


図 5.4: Multicast capacity performance.

図 5.4 は，縦軸にマルチキャスト伝送容量，横軸にランダムグラフを構成するノードの平均次数をとり，複数セッション IP マルチキャストとネットワークコーディングにおいて実現可能な伝送容量の違いを示している．なお，次数とはノードに接続しているリンク数を示す値である．図 5.4 のプロット点は，送受信ノードが異なる 10 個のトポロジにおけるマルチキャスト伝送容量の平均値である．図 5.4 より，リンク帯域が均質モデル，不均質モデルともにネットワークコーディングが複数セッションマルチキャストと比較して高いマルチキャスト伝送容量を示している．これは，ネットワークコーディングではマルチキャスト伝送容量の理論上の最大値である最大フローを実現できるためである．また，ノードの平均次数が大きくなるにしたがい送受信ノード間を結ぶ有向路の数が増え，最大フローの値も大きくなる．したがって，ノードの平均次数が大きい領域においてネットワークコーディングと複数セッション IP マルチキャストの性能差が大きくなる．一方，複数セッション IP マルチキャストに注目すると，ノードの平均次数が大きい領域では，均質モデルよりも不均質モデルの方が高い性能を示している．平均次数が小さい領域では，構築できるマルチキャストツリー数が少ないため，マルチキャストツリー当りの

フローの値が大きくなる均質モデルの性能が高くなる．しかし，平均次数が大きい領域では，不均質モデルの方が構築できるマルチキャストツリー数が多くなるため，不均質モデルの方が高い性能を示すと考えられる．最大フローは，送受信ノード間に設定可能な経路数によって決定されるため，マルチキャスト伝送容量の性能はネットワークサイズではなく，ノードの次数とリンク容量に依存するものと考えられる．したがって，本節ではノード数 50，受信ノード数 10 の特定のネットワークトポロジを用いて性能評価を行ったが，ネットワークサイズの異なる他のネットワークトポロジにおいても同様の結果が得られると考えられる．このように，ネットワークコーディングを適用すれば複数セッション IP マルチキャストと比較してより高いスループットを実現することができる．

5.4.2 伝送性能

5.3.3 で述べたように，ネットワークコーディングを適用したマルチキャスト通信は，パケットロス及びネットワーク遅延により，その伝送性能が劣化する可能性がある．そこで，本節では，パケットロス，伝搬遅延，ルータにおける待ち時間等が発生する現実的なネットワークモデルを用いて，ネットワークコーディングの伝送性能を評価する．なお，比較対象として，単一セッション IP マルチキャストおよび複数セッション IP マルチキャストを用いる．

モデル

シミュレーションで用いたネットワークトポロジは，図 5.3–(a) で示したものを用いた．ネットワークコーディングの符号化方法としては，5.3–(d) 示したものを用いた．ここで，各リンクの容量，すなわちリンク帯域は全てのリンクで 1.5[Mbps] とする．また，各リンクの伝搬遅延は，全リンクで伝搬遅延が 10[msec] とした．パケットロスはルータの出力バッファのバッファ溢れによってのみ生じるものとする．ここで，送信パケットのパケットサイズ 1024[bytes] とし，各ルータの出力バッファサイズは 50[Kbytes] とした．また，各リンクに平均到着率 λ のポアソントラヒックをバックグラウンドトラヒックとして加えた．送信ノードは，一定のデータレート R で情報を送信する．符号化処理時間，送受信ノード及びルータにおけるパケット処理時間はリンクへの伝送遅延および伝搬遅延と比較して小さく無視できるものとする．

最大フロー伝送時の伝送性能

図 5.5 は，λ に対するネットワークコーディングと IP マルチキャストの受信ノードにおける平均スループットである．ここで，受信ノードにおけるスループットは図 5.3–(a) で示される 3 つの受信ノード間での平均を表している．また，送信ノードの送信レート R は，ネットワークコーディングでは R = 2.0 [Mbps]，複数セッション IP マルチキャストでは R = 1.5 [Mbps]，単一セッション IP マルチキャストでは R = 1.0 [Mbps] である．本シミュレーションでは，送信ノードから送出された情報は，図 5.3–(b)(c)(d) で示されるように転送されるため，受信ノードへ至る各リンクに流れるデータレートは 1.0 [Mbps] となる．この状況では，バックグラウンドトラヒックレート λ が 0.5 [Mbps] の時，各リンクの利用率が 1 となり，ネットワークコーディングにおいて最大フロー伝送が実現される．図 5.5 より，ネットワークコーディングでは最大フロー伝送が可能となるため，IP マルチキャストと比較して高いスループットを実現できる．しかし，λ が 0.5[Mbps] を越えた時点で，IP マルチキャストに比べ急激にスループットが減少している．これは，ネットワークコーディングでは，5.3.3 で述べたように，符号化されていないパケットのロスが発生した場合，既に受信したパケットもロスしたものと見なされるためである．したがって，ネットワークコーディングは，IP マルチキャストと比較してパケットロスに敏感な伝

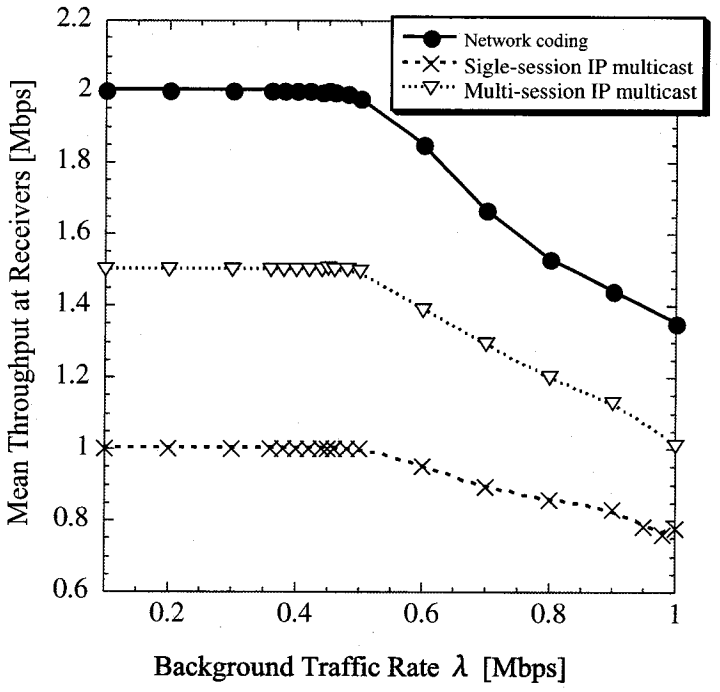


図 5.5: Throughput performance vs. background traffic rate.

送方式であり，ネットワークコーディングを適用したマルチキャストアーキテクチャを設計する際には，パケットロスへの対処が重要な技術課題となる．

負荷分散利用時の伝送性能

本節では，データフローが各リンクで消費する帯域の分散を用いて，ネットワークコーディングの負荷分散利用において，どの程度トラヒックの負荷が分散されているかを評価する．リンク e で消費される帯域を X_e とすると，リンクの使用帯域の平均 $\bar{X}$ 及び分散 σ^2 は以下の式で表される．

$$\bar{X} = \frac{1}{n}\sum_{e\in E} X_e, \quad \sigma^2 = \frac{1}{n}\sum_{e\in E} X^2 - \bar{X}^2$$

ただし，X_e はマルチキャストフローの帯域を表しておりバックグラウンドトラヒックは含まない．また，上式の $\sum_{e\in E}$ はネットワーク中のリンク全てに対して和をとることを表し，n はリンクの総数を表している．σ^2 が小さい場合，ネットワーク全体のリンクを偏りなく使用し，トラヒックの負荷分散を実現していることを示している．

表 5.1: Bandwidth utilization

	Mean bandwidth consumption $\bar{X}$ [Mbps]	Variance σ^2 [Mbps^2]
Network coding (R=2.0[Mbps])	0.998	0
Network coding (R=1.5[Mbps])	0.750	0
Network coding (R=1.0[Mbps])	0.500	0
Single-session IP multicast (R=1.0[Mbps])	0.556	0.247
Multi-session IP multicast (R=1.5[Mbps])	0.833	0.056

表 5.1 に，λ = 0.5 [Mbps] の場合の，平均使用帯域 $\bar{X}$ と使用帯域の分散 σ^2 を示す．R=2.0[Mbps] のネットワークコーディングは，ネットワークコーディングの最大フロー伝送利用を表しており，R=1.5, 1.0 [Mbps] のネットワークコーディングは，本論文で提案するネットワークコーディングの負荷分散利用を表しており，IP マルチキャストと同等の送信レートで伝送が行われている．表 5.1 より，ネットワークコーディングは全ての R において，σ^2 が 0 となっており，トラヒックの負荷分散を実現している．これは，ネットワークコーディングでは複数経路を用いてデータの配送が行われており，ネットワークを構成する全リンクが均等に利用されているためである．また，単一セッション IP マルチキャストおよび複数セッション IP マルチキャストと，送信レートのそれぞれ等しい

ネットワークコーディングを比較すると，ネットワークコーディングの方が $\bar{X}$ が小さく，同じスループットであっても使用帯域を節約可能なことがわかる．これは，ネットワークコーディングではネットワーク内で符号化を行い効率的にマルチキャストを行っているためである．ただし，ネットワークコーディングを用いた場合に，トラヒックの負荷分散と使用帯域の節約がどの程度実現されるかは，ネットワークトポロジに大きく依存する．しかし，ネットワークコーディングの負荷分散利用時には，同等量のデータ配送を IP マルチキャストよりも多くの経路を用いて行うため，あらゆるネットワークトポロジにおいても IP マルチキャストより高い負荷分散効果を得ることができる．

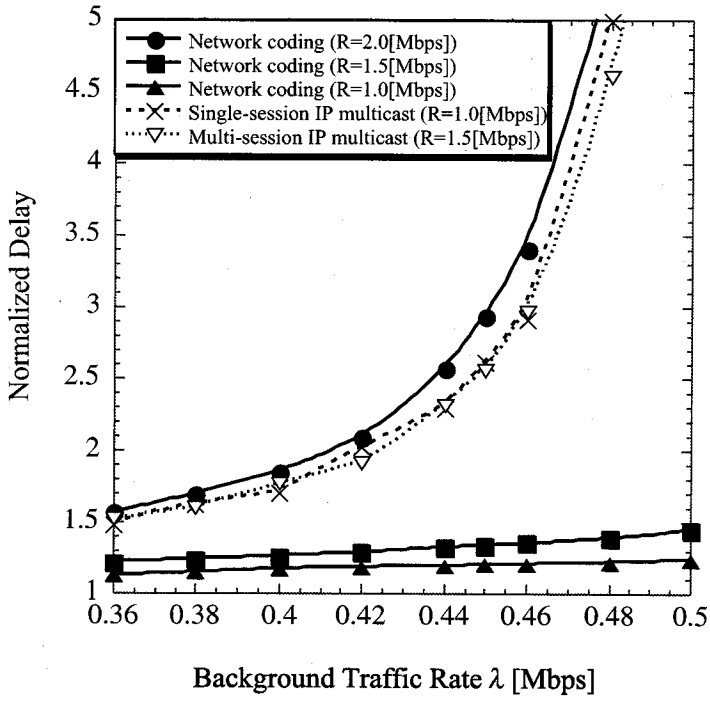


図 5.6: Normalized delay performance vs. background traffic rate.

ネットワークコーディングを用いて最大フロー伝送を行う場合，転送経路上のいくつかのリンクにおいて，利用可能なネットワーク帯域が完全に消費される（リンク利用率が 1 となる）．したがって，各リンクに流れるバックグラウンドトラヒックレートが増加するにつれて，ルータでの待ち時間が増大し，遅延性能が著しく劣化する．一方，ネットワークコーディングの負荷分散利用時には，全てのリンクにおいて利用率が 1 以下となり，遅延性能の改善が期待できる．図 5.6 は，IP マルチキャストおよびネットワークコーディングの λ に対する平均遅延を示している．ここで，平均遅延は，送信ノードがパケットを送信してから受信ノードにおいて正しく受信されるまでのパケットの平均遅延時間であり，送受信ノード間伝搬遅延およびリンク伝送遅延の和で正規化している．ネットワークコーディングの最大フロー伝送利用（R=2.0[Mbps]）および IP マルチキャスト（R=1.5,

1.0 [Mbps]）では，転送経路上のリンク（これらのリンクのデータレートは R=1.0 [Mbps]）のネットワーク資源がほぼ完全に利用されている．一方，ネットワークコーディングの負荷分散利用（R=1.5, 1.0 [Mbps]）では，トラヒックがネットワーク全体に分散し，転送経路上の各リンク（これらのリンクのデータレートは，それぞれ 0.75, 0.5 [Mbps]）に残余帯域が存在する．したがって，ネットワークコーディングの負荷分散利用では，IP マルチキャストおよびネットワークコーディングの最大フロー伝送利用と比較して遅延の性能が大きく向上している．ネットワークコーディングの最大フロー伝送利用および IP マルチキャストにおいて，各リンクのトラヒック負荷は等しいが，ネットワークコーディングの最大フロー伝送利用が遅延の点でもっとも性能が低い．これは，5.3.3 で示したように，ネットワークコーディングでは復号に必要なパケットが全て揃うまでの遅延が加算されるためである．したがって，ネットワークコーディングを適用したマルチキャストアーキテクチャを設計する際には，パケットロスだけでなくリンク遅延への対処も重要な技術課題となる．

5.5 実装問題

本節では，ネットワークコーディングを適用した次世代マルチキャストアーキテクチャを提案する．また，本アーキテクチャの実現に必要な要素技術および発生すると考えられる問題点を明らかにする．

5.5.1 次世代マルチキャストアーキテクチャ

ネットワークコーディングを，現在の IP ネットワークに適用するためには，2 つの問題点が存在する．第 1 に，ネットワークコーディングの機能を提供するルータは，適切な経路選択と符号化を行うために，ネットワーク全体の構成に関する知識をもつ必要がある．第 2 に，ネットワークコーディング対応ルータはアプリケーション層までの処理を必要とするために，全てのルータにネットワークコーディング機能を実装することは困難である．これらの理由から，ルーチングドメインや ISP（Internet Service Provider）ドメインなどの限られた範囲でネットワークコーディングのサービスが提供されることが望ましい．図 5.7 に，次世代マルチキャストアーキテクチャの構成をに示す．本アーキテクチャでは，ネットワークコーディングを適用するネットワークコーディングドメイン内にネットワークコーディング対応ルータを配置し，このドメイン内でネットワークコーディングサービスを提供する．送信ノードが情報をマルチキャストする場合，ネットワークコーディングドメイン内に流入するフローに対してはネットワークコーディングが適用され，ネットワークコーディングドメインを通過しないフローに対しては既存のマルチキャスト

が適用される．図 5.3–(d) のネットワークをネットワークコーディングドメインと考えた場合，ノード S がネットワークコーディングドメインの入口ルータ (図 5.7 の Ingress) に相当し，ノード R_1, R_2, R_3 は出口ルータ (図 5.7 の Egress 1，Egress 2，Egress 3) に相当する．

5.5.2 必須技術

本節の目的は，次世代マルチキャストアーキテクチャを実現するために必要な要素技術を明らかにし，ネットワークコーディングに関する今後の研究課題を提示することである．次世代マルチキャストアーキテクチャを実現するためには次のような要素技術が必要である．

(1) 最大フローに基づいた複数転送経路の決定法

ネットワークコーディングドメイン内で，あるマルチキャストフローに対しネットワークコーディングを適用する場合，ルーチングアルゴリズムは，そのマルチキャストフローに対応するネットワークコーディングドメインの入口ルータおよび出口ルータを決定し，それらのエッジルータ間の経路を最大フローに基づいて決定する必要がある．このような

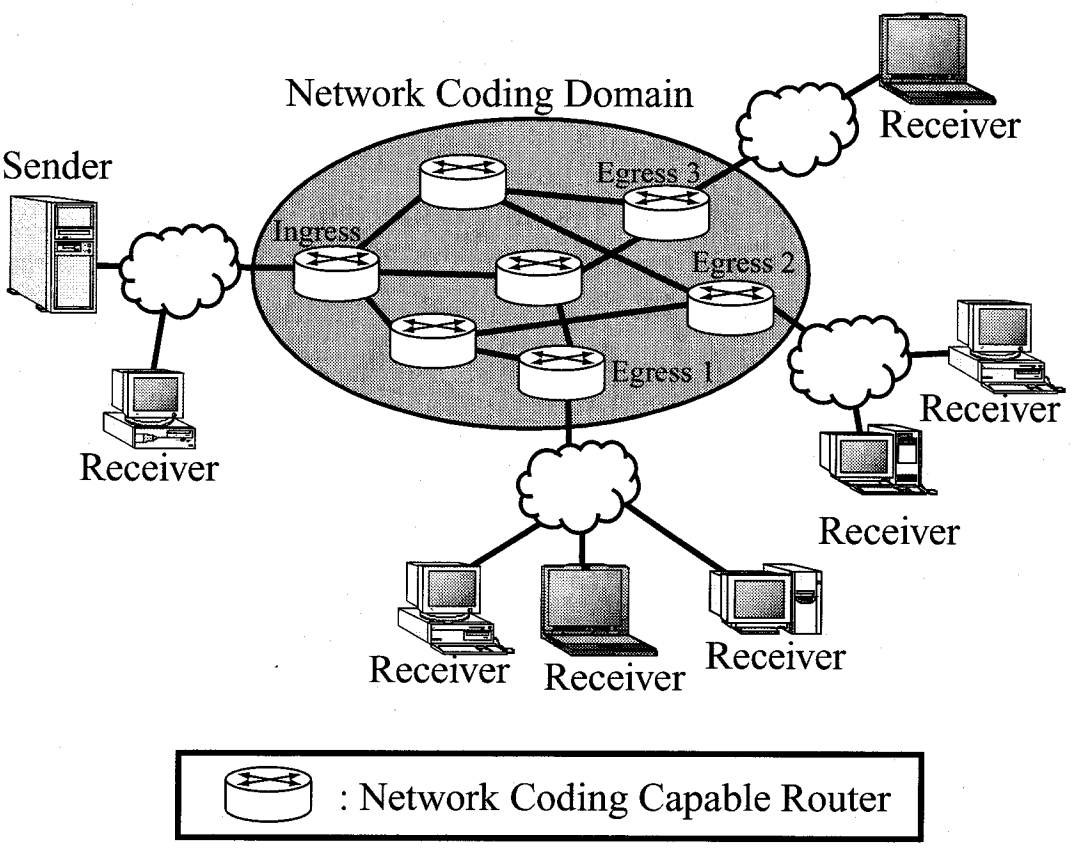


図 5.7: Multicast architecture applying network coding.

ルーチングアルゴリズムとして，Ford-Fullkerson のラベリングアルゴリズム [67] などの既存の最大フロー発見アルゴリズムを利用する方法が考えられる．

(2) 各ルータのトポロジー情報獲得及び交換方法

経路決定のためには，各ルータにおいてネットワークのトポロジ情報 (接続構造及び各リンクの容量，すなわち利用可能帯域[*4]) を知る必要がある．各ルータで接続するリンクの利用可能帯域を調べる方法としては，ルータにおいてトラヒック量を定期的に測定し見積もる方法などが考えられる．また，各ルータでのリンク利用可能帯域が変化した場合には，それに伴い最大フローも変化するため最大フローの再計算が必要となり，QOSPF[68] で用いられているリンク状態メッセージのような方法で各ルータ間で定期的にトポロジ情報を交換する仕組みが必須である．

(3) 明示的な経路指定法

ネットワークコーディングでは情報の転送経路と各ルータでの符号化が密接に関係しているため配送される全ての情報は決定された転送経路を通るように明示的に指示する必要がある．これを実現する仕組みとして，各ルータ間で経路情報をシグナリングによって交換する方法や，ネットワークコーディングドメインの入口ルータがソースルーチング [69] を用いて明示的に経路を指定する方法などが考えられる．

(4) 符号化決定手法

様々なマルチキャストフローに対してネットワークコーディングドメイン内でネットワークコーディングを適用するためには，最大フローに基づいた転送経路上で行う適切な符号化方法を決定する仕組みが必要である．符号化方法は，転送経路及び最大フローの変化に応じて変わるため，動的に符号化方法を決定する方式が必須である．Shuo-yen Robert Li らは，文献 [70] において，非環状ネットワークにおいて，線形符号を構成するためのグリーディアルゴリズムを提案している．この符号化アルゴリズムを用いれば動的に符号化方法を決定することが可能であるが，他のルータの符号化関数を知る必要があるため，実際のネットワークに適用するためには各ルータの符号化関数の交換方法を確立することが課題となる．

(5) ルータにおける符号化機能

現在のルータでは符号化処理等の高機能な処理を実現することはできないため，ネットワークコーディング対応ルータにはアクティブネットワーク技術 [56, 57] を適用する方法などが考えられる．

[*4] ネットワークコーディングを適用するマルチキャストフローが利用できる帯域であり，物理的なリンク帯域ではない．

5.6 結言

本章では，ネットワークコーディングのトラヒックの負荷分散効果に着目し，最大フロー伝送に代わるネットワークコーディングの利用法として負荷分散効果を得る利用法を提案した．また，ネットワークコーディングを適用した次世代マルチキャストアーキテクチャを提案し，その実現のために必要な要素技術および問題点を明らかにした．

ネットワークコーディングのトラヒックの負荷分散利用法では，複数の経路でデータ配送を行うことにより，トラヒックの負荷分散を実現しつつ IP マルチキャストと同等量のデータ伝送が可能である．

計算機シミュレーションを用いた性能評価により，ネットワークコーディングの負荷分散利用では，IP マルチキャストと比較して高いトラヒックの負荷分散を実現し，ネットワーク使用帯域量の節約が可能となることを示した．また，トラヒックの負荷分散に伴いネットワーク負荷が低減するため，遅延性能も向上することを示した．

ネットワークコーディングの負荷分散利用法は，ネットワークコーディングの現実的な利用法であり，トラヒックの集中問題を解決できるため，IP マルチキャストと並んで有力なネットワークレベルマルチキャストとなり得る．これを用いることにより，伝送性能およびネットワーク資源の利用効率の点で性能が向上した効率的な 1 対多通信を実現することが可能となる．

第 6 章

結論

本論文は，著者が大阪大学大学院工学研究科 (通信工学専攻) 在学中に行ったマルチキャスト通信における転送制御に関する研究成果をまとめたものである．以下では，本研究で得られた成果を総括して述べる．

近年，インターネットが急速に普及する中，アプリケーションに対するニーズは多様化しており，なかでも 1 対多，多対多通信を用いたマルチキャストアプリケーションが注目を集めている．マルチキャスト通信の魅力は，ソフトウェアの一斉配布や，株式情報の配信，インターネット TV，ビデオ会議などの新しいアプリケーションの提供が可能となることに加え，従来のユニキャスト通信に比べ，ネットワーク資源の利用効率を格段に向上させることにある．インターネットにおいてマルチキャスト通信を実現する方法として，マルチキャスト転送制御機能をネットワーク層（ルータ）に実装するネットワークレベルマルチキャストと，アプリケーション層（エンドホスト）に実装するアプリケーションレベルマルチキャストの 2 つの方式が検討されており，現在も並行して研究が進められている．

ネットワークレベルマルチキャストは，ネットワーク資源の利用効率や遅延など伝送性能の点で高い性能を示すため，マルチキャスト通信の最終解として期待されている．しかし，信頼性保証が困難であることや，特定リンクへのトラヒックの集中により輻輳が引き起こされるなどの問題があり本格的な普及には至っていない．一方，アプリケーションレベルマルチキャストは，インターネットへの迅速な展開が可能なため，ネットワークレベルマルチキャストが完全に普及するまでの現実解として注目されている．しかし，不安定なエンドホストがマルチキャスト転送制御を行うことから，エンドホスト障害への対処が課題として残されており，利用できるアプリケーションが限られる．本論文では，ネットワークレベルマルチキャストおよびアプリケーションレベルマルチキャスト両方式の技術課題について検討し，それぞれの課題を克服する方式を提案し，その有効性を示した．

アプリケーションレベルマルチキャストにおけるエンドホスト障害問題に対しては，本論文で提案したロバストなアプリケーションマルチキャストツリー構築法が有効である．なぜならば，本方式は，アプリケーションレベルマルチキャスト通信において重要とされる2つの要求条件を満たすからである．第1に，平衡木型ツリーを構築するため，任意のエンドホストで障害が発生した場合に，アプリケーションレベルマルチキャストセッションから切り離され通信途絶に陥る平均エンドホスト数を低減でき，エンドホスト障害に対するロバスト性が高い．第2に，アプリケーションレベルマルチキャストツリーの最適化動作により，遅延性能を向上している．したがって，本方式により，耐障害性の高い効率的なアプリケーションレベルマルチキャスト通信の実現が可能となる．

マルチキャスト通信の本来の目的である効率的な1対多通信の観点から眺めると，アプリケーションレベルマルチキャストの利用は，あくまでネットワークレベルマルチキャストが完全に普及するまでの一時的なものであり，ネットワークレベルマルチキャストの技術課題を克服し，ネットワークレベルマルチキャストの普及を促すことが重要である．

ネットワークレベルマルチキャストに対し信頼性のある通信を提供する信頼性マルチキャストプロトコルの確立は，ネットワークレベルマルチキャストの重要な技術課題の1つである．本論文では，部分的にFECを適用する信頼性マルチキャストプロトコルを提案した．本方式では，NAKの発生数を大幅に削減することにより，スケーラビリティ，遅延，ネットワーク使用帯域量の3つの点で，既存の信頼性マルチキャストプロトコルより性能が向上する．本方式により，ネットワークレベルマルチキャストを用いた効率的な信頼性マルチキャスト通信の実現が可能となる．

ネットワークレベルマルチキャストのもう1つの技術課題として，トラヒックの集中問題が挙げられる．本論文では，トラヒックの負荷分散を実現する方式として，ネットワークコーディングに着目した．本論文で提案したネットワークコーディングの新利用法では，既存のIPマルチキャストでは不可能であったトラヒックの負荷分散を実現できる．また，伝送性能では，遅延およびネットワーク使用帯域の点でIPマルチキャストより性能が向上することを示した．本方式を用いることにより，伝送性能が向上した効率的なマルチキャスト通信が実現可能となる．

以上より，本論文は，アプリケーションレベルマルチキャストおよびネットワークレベルマルチキャスト両方式を用いて，マルチキャスト通信の本来の目的である効率的な転送制御を実現したと結論づける．なお，ネットワークレベルマルチキャストでは，遅延及びネットワーク帯域使用量の性能利得が大きいため，ネットワーク内部にマルチキャスト転送制御機能を実装しても，エンド・ツー・エンド議論のシステム設計の原則に反しない．また，ネットワーク内部で様々な機能を提供することは，ネットワークの柔軟性の増大に

つながり，多様化するネットワークアプリケーションへの対応を容易にさせる．したがって，マルチキャスト通信の最終解としては，アプリケーションレベルマルチキャストではなくネットワークレベルマルチキャストが適切であり，特にネットワークレベルマルチキャストに関する本論文の成果は，ネットワークレベルマルチキャストの普及に大いに貢献するものと考える．

略語表

ACK	Acknowledgement
ALM	Application Level Multicast
AS	Autonomous System
BGMP	Border Gateway Multicast Protocol
BRITE	Boston university Representative Internet Topology gEnerator
CAN	Content-Addressable Network
CBT	Core-Based Trees
CPU	Central Processing Unit
DR	Designated Receiver
DVMRP	Distance Vector Multicast Routing Protocol
FEC	Forward Error Correction
GR	Gateway Router
HMRP	Host Multicast Rendezvous Point
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IP	Internet Protocol
IPv4	IP version 4
IPv6	IP version 6
ISP	Internet Service Provider
LAN	Local Area Network
MAN	Metropolitan Area Network
MBGP	Multiprotocol extensions for Border Gateway Protocol
MOSPF	Multicast extensions to Open Shortest Path
MST	Minimum Spanning Tree
NAK	Negative Acknowledgement

OS	Operating System
PIM-SM	Protocol Independent Multicast Sparse Mode
PIM-DM	Protocol Independent Multicast Dense Mode
RDP	Relative Delay Penalty
RTT	Round Trip Time
SPT	Shortest Path Tree
TCP	Transmission Control Protocol
TELNET	TELecommunication NETwork
TTL	Time To Live
UDP	User Datagram Protocol
WAN	Wide Area Network

参考文献

[1] P. Baran, "The beginnings of packet switching: some underlying concepts", IEEE Commun. Mag., vol.40, no.7, pp.42-48, July 2002.

[2] J. Postel, "Internet Protocol", RFC 791, Sept. 1981.

[3] J. Postel, "Transmission Control Protocol", RFC 793, Sept. 1981.

[4] J. H. Saltzer, D. P. Reed, and D. D. Clark, "End-to-end arguments in system design", ACM Trans. Comp. Syst., Vol. 2, No.4, pp. 277–288, Nov. 1984.

[5] S. Deering and D. Cheriton, "Host Groups: A Multicast Extension to the Internet Protocol", RFC 966, Dec. 1985.

[6] S. Deering, "Host Extensions for IP Multicasting", RFC 1112, Aug. 1989.

[7] C. Diot, B. N. Levine, B. Lyles, H. Kassem, and D. Balensiefen, "Deployment Issues for the IP Multicast Service and Architecture", IEEE Network Mag., Vol.1, No.14, pp.78-88, Jan. 2000.

[8] B. Fenner, M. Handley, H. Holbrook, and J. Kouvelas, "Protocol independent multicast sparse mode (PIM-SM):Protocol specification (revised)", Internet draft, draft-ietf-pim-sm-new-v2-01.txt, Mar. 2002.

[9] A. Ballardie, "Core-based trees (CBT version 2) multicast routing", RFC 2201, Sept. 1997.

[10] A. El-Sayed, V. Roca, and L. Mathy, "A survey of Proposals for an Alternative Group Communication Service", IEEE Network Mag., Vol.17, No.1, pp.46-51, Jan./Feb. 2003.

[11] V. Roca, and A. El-Sayed, "A Host-Based Multicast (HBM) Solution for Group Communications", IEEE International Conference on Networking 2001 (ICN'01), pp.610-619, Colmar, France, July 2001.

[12] W. Fenner, "Internet Group Management Protocol, Version 2", RFC 2236, Nov. 1997.

[13] B. Fenner, I Kouvelas, A. Thyagarajan, "Internet Group Management Protocol Version 3", RFC 2933, Jan. 2002.

[14] D. Waitzman, C. Partridge and S. Deering, "Distance Vector Multicast Routing Protocol

(DVMRP)", RFC 1075, Nov. 1988.

[15] T. Pusateri, "Distance Vector Multicast Routing Protocol", Internet Draft, draft-ietf-idmrdvmrp-v3-10.txt, Aug. 2000.

[16] J. Moy, "Multicast extensions to OSPF", RFC 1584, Mar. 1994.

[17] T. Bates, R. Chandra, D. Katz, and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 2858, June 2000.

[18] D. Thaler, "Border Gateway Multicast Protocol (BGMP)", Internet Draft, draft-ietf-bgmp-spec05.txt, June 2003.

[19] B. Wang and J.C. Hou, "Multicast routing and its QoS extension: problems, algorithms, and protocols", IEEE Network Mag., vol.14, pp.22-36, Jan.2000.

[20] IRTF Reliable Multicast Research Group Homepage, http://www.east.isi.edu/RMRG

[21] IETF Reliable Multicast Transport Working Group Homepage, http://www.ietf.org/html.charters/rmt-charter.html

[22] M. Handley et al., "The Reliable Multicast Design Space for Bulk Data Transfer", RFC 2887, Aug. 2000.

[23] J. C. Lin and S. Paul, "RMTP: A Reliable Multicast Protocol", IEEE INFOCOM'96, vol.3, pp.1414-1424, San Francisco, California, USA, Mar. 1996.

[24] S. Floyd, V. Jacobson, S. McCanne, C. G. Liu, and L. Zhang, "A Reliable Multicast Framework for Light-Weight Sessions and Application Level Framing",IEEE/ACM Trans. on Net., vol.5, no.6, pp.784-803, Dec. 1997.

[25] Y.-H. Chu, S. G. Rao and H. Zhang, "A Case for End System Multicast", IEEE J. Select. Areas Commun., vol.20, pp.1456-1471, Oct. 2002.

[26] Y. Chawathe, "Scattercast: An Architecture for Internet Broadcast Distribution as an Infrastructure Service", Ph.D.Thesis, University of California, Berkeley, Dec. 2000.

[27] B. Zhang, S. Jamin, and L. Zhang, "Host Multicast: A Framework for Delivering Multicast To End Users", IEEE INFOCOM'02, New York, NY, June 2002.

[28] D. Pendarakis, S. Shi, D. Verma and M. Waldvogel, "ALMI: An Application Level Multicast Infrastructure", 3rd USENIX Symposium on Internet Technologies and Systems (USITS '01), pp.49-60, San Francisco, CA, Mar. 2001.

[29] P. Francis, "Yoid:Extending the Multicast Internet Architecture", 1999, white paper, http://www.icir.org/yoid/.

[30] S. Banerjee, B. Bhattacharjee and C. Kommareddy, "Scalable Application Layer Multicast", ACM SIGCOMM'02, pp.205-220, Pittsburgh, PA, Aug. 2002.

[31] S. Ratnasamy, M. Handley, R. Karp and S. Shenker, "Application-level Multicast using Content-Addressable Networks", 3rd International Workshop on Networked Group

Communication (NGC'01), pp.14-29, London, UK, Nov. 2001.

[32] H. Q. Zhuang, B. Y. Zhao, A. D. Joseph, R. H. Katz and J. Kubiatowicz, "Bayeux: An Architecture for Scalable and Fault-tolerant Wide-Area Data Dissemination", ACM NOSSDAV'01, pp.11-20, Port Jefferson, NY, June 2001.

[33] J. Postel, "User Datagram Protocol", RFC 768, AUg. 1980.

[34] S. Banerjee and B. Bhattacharjee, "Scalable Secure Group Communication over IP Multicast", IEEE International Conference on Network Protocols 2001 (ICNP'01), Riverside, CA, Nov. 2001.

[35] B. Zhang, S. Jamin, and L. Zhang, "Universal IP multicast delivery", 4th International Workshop on Networked Group Communication (NGC'02), Boston, MA, Oct. 2002.

[36] Y. Chawathe, S. McCanne, and E. A. Brewer, "RMX: Reliable Multicast for Heterogeneous Networks", IEEE INFOCOM'00, pp.795-804, TelAviv, Israel, Mar. 2000.

[37] S. Banerjee et al., "Construction of an Efficient Overlay Multicast Infrastructure for Real-time Applications", IEEE INFOCOM'03, San Francisco, USA, Mar. 2003.

[38] T. Sano, T. Noguchi, M. Yamamoto, "Improving Efficiency of Application-level Multicast with Network Support", IEICE Trans. on Commun., Mar. 2004. (to be published).

[39] T. Sano, T. Noguchi, M. Yamamoto, "Improving Efficiency of Application-level Multicast with Network Support", 5th International Workshop on Networked Group Communication (NGC'03), pp.13-22, Munich, Germany, Sept. 2003.

[40] 佐野健，野口拓，山本幹，"アプリケーションレベルマルチキャストにおけるネットワーク支援を用いた冗長経路削減方式"，2003 年電子情報通信学会総合大会，B-6-260，2003 年 3 月.

[41] B. Waxman, "Routing of Multiple Connections", IEEE J. Select. Areas Commun., vol.6, no.9, pp.1617-1622, Dec. 1988.

[42] A. Medina, I. Matta, and J. Byers, "On the Origin of Power Lows in Internet Topologies", ACM Comp. Commun. Rev., vol.30, no.2, pp.18-28, April 2000.

[43] A. Medina, A. Lakhina, I. Matta, and J. Byers. "BRITE: An Approach to Universal Topology Generation", IEEE Modeling, Analysis and Simulation of Computer and Telecommunications Systems, 2001 (MASCOTS '01), Cincinnati, OH, Aug. 2001.

[44] P. V. Mieghem, G. Hooghiemstra and R. Hofstad, "On the Efficiency of Multicast," IEEE/ACM Trans. on Net., Vol.9, No.6, pp.719-732, Dec. 2001.

[45] D. Towsley, J. Kurose, and S. Pingali, "A Comparison of Sender-initiated and Receiver-initiated Reliable Multicast Protocols", IEEE J. Select. Areas Commun., vol.15, no.3, pp.398–406, April 1997.

[46] M. Yamamoto, J. Kurose, D. Towsley, and H. Ikeda, "A Delay Analysis of Sender-

initiated and Receiver-initiated Reliable Multicast Protocols", IEEE INFOCOM'97, vol.2, pp.480-488, Kobe, Japan, April 1997.

[47] T. Hashimoto, M. Yamamoto, H. Ikeda and J. Kurose, "Performance Evaluation of Reliable Multicast Communication Protocols under Heterogeneous Transmission Delay Circumstances", IEICE Trans. on Commun., vol.E82-B, no.10, pp.1609-1617, Oct. 1999.

[48] M. Yajnik, J. Kurose, and D. Towsley, "Packet Loss Correlation in MBone Multicast Network", IEEE Global Internet mini Conference 1996 (GIC'96), London, UK, Nov. 1996.

[49] J. J. Metzner, "An improved broadcast retransmission protocol", IEEE Commun. Mag., vol.COM-32, no.6, pp.679-683, June 1984.

[50] L. Rizzo, and L. Vicisano, "RMDP: an FEC-based Reliable Multicast protocol for wireless environments", ACM Mob. Comp. and Commun. Rev., vol.2, no.2, pp.23-31, Apr. 1998.

[51] J. Nonnenmacher, E. Biersack, and D. Towsley, "Parity-Based Loss Recovery for Reliable Multicast Transmission", ACM SIGCOMM '97, pp. 289-300, Cannes, France, Sept. 1997.

[52] D. Rubenstein, J. Kurose, D. Towsley, "Real-Time Reliable Multicast Using Proactive Forward Error Correction", NOSSDAV'98, Cambridge, UK, July 1998.

[53] D. Rubenstein, S. Kasera, D. Towsley, and J. Kurose, "Improving Reliable Multicast Using Active Parity Encoding Services (APES) ", IEEE INFOCOM'99, pp.1248-1255, New York, NY, Apr. 1999.

[54] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data", ACM SIGCOMM '98, pp.56-67, Vancouver, Canada, Sept. 1998.

[55] J. Gemmell, E. Schooler, and J. Gray, "Fcast multicast file distribution", IEEE Network Mag., Vol.14, No.1, pp.58-68, Jan./Feb. 2000.

[56] D.J. Wetherall, U. Legedza, and J. Guttag, "Introducing New Internet Services: Why and How", IEEE Network Mag., vol.12, no.3, pp.12-19, July/August 1998.

[57] M. Yamamoto, "A Survey of Active Network Technology", IEICE Trans. on Commun., vol.J-84-B, no.8, pp.1401-1412, Aug. 2001.

[58] L. Rizzo, "Effective erasure codes for reliable computer communication protocols", ACM Comp. Commun. Rev., vol.27, no.2, pp.24-36, Apr. 1997.

[59] M. B. Doar, "A Better Model for Generating Test Network", IEEE GLOBECOM'96, pp.86-93, Nov. 1996.

[60] R. Ahlswede, N. Cai, S. -Y. R. Li, and R. W. Yeung, "Network Information Flow", IEEE

Trans. on Info. Theo., Vol.46, No.4, pp.1204-1216, July 2000.

[61] B. Bollobas, "Graph Theory, An Introductory Course", New York:Springer-Verlag, 1979.

[62] L. Kleinrock, "Queueing Systems, Volume 1: Theory", John Wiley & Sons; ISBN: 0471491101, 1975.

[63] X. Xiao and L. M. Ni, "Internet QoS: A Big Picture", IEEE Network Mag., vol.13, no.2, pp.8-18, Mar. 1999.

[64] A. Adams, J. Nicholas and W. Siadak, "Protocol independent multicast dense mode (PIM-DM):Protocol specification (revised)", Internet draft, draft-ietf-pim-dm-new-v2-01.txt, Feb. 2002.

[65] M. Faloutsos and P. Faloutsos and C. Faloutsos, "On Power-law Relationships of the Internet Topology", ACM SIGCOMM'99, pp.251-262, Aug./Sept. 1999.

[66] A. Medina, I. Matta and J. Byers, "On the Origin of Power Laws in Internet Topologies", ACM Comp. Commun. Rev., vol.30, no.2, pp.18-28, Apr. 2000.

[67] L. R. Ford, Jr. and D.R. Fulkerson, "Maximal Flow Through a Network", Canadian Journal of Mathematics, 8:399-404, 1956.

[68] R. Guerin, S. Kamat, A. Orda, T. Przygienda, and D. Williams, "QoS Routing Mechanisms and OSPF extensions", Internet draft, draft-guerin-QoS-routing-ospf-03.txt, Jan. 1998.

[69] W. R. Stevens "TCP/IP Illustrated, Volume 1", Addison-Wesley, ISBN 0-201-63346-9, 1999.

[70] S. -Y. R. Li,R. W. Yeung, and N. Cai, "Linear Network Coding", IEEE Trans. on Info. Theo., vol.49, no.2, pp.371-381, Feb. 2003.

本論文に関する原著論文

A. 論文

1. Taku Noguchi, Miki Yamamoto, "Reliable Multicast Protocol Applying Local FEC", IEICE Transactions on Communications, Special Issue on Internet Technology Series (3), Vol.E86-B, No.2, pp.690-698, Feb. 2003.
2. Taku Noguchi, Takahiro Matsuda, Miki Yamamoto, "Performance Evaluation of New Multicast Architecture with Network Coding", IEICE Transactions on Communications, Special Issue on Content Delivery Networks, Vol.E86-B, No.6, pp.1788-1795, June 2003.
3. Takeshi Sano, Taku Noguchi, Miki Yamamoto, "Improving Efficiency of Application-level Multicast with Network Support", IEICE Transactions on Communications, Special Issue on Internet Technology Series (4), Mar. 2004. (採録決定済み：2004 年 3 月掲載予定)

B. 国際会議

1. Taku Noguchi, Miki Yamamoto, Hiromasa Ikeda, "Reliable Multicast Protocol Applied Local FEC", IEEE International Conference on Communications (ICC 2001), vol.8, pp.2348-2353, Helsinki, Finland, June 2001.
2. Taku Noguchi, Miki Yamamoto, "Analysis and Performance Evaluation of Reliable Multicast Protocol Applying Local FEC", 2002 International Symposium on Performance Evaluation of Computer and Telecommunication System(SPECTS 2002), pp. 843-851, San Diego, USA, July 2002.
3. Taku Noguchi, Takahiro Matsuda, Miki Yamamoto, "Performance Evaluation of New Multicast Architecture with Network Coding", International Working Conference on Active Network (IWAN2002), Poster Session, Zurich, Switzerland, Dec. 2002.
4. Takeshi Sano, Taku Noguchi, Miki Yamamoto, "Improving Efficiency of Application-level Multicast with Network Support", Fifth International Workshop on Networked

Group Communications (NGC 2003), pp. 13-22, Munich, Germany, Sept. 2003.

C. 全国大会

1. 野口拓，山口誠，山本幹，池田博昌，“局所的に FEC を適用した信頼性マルチキャストプロトコル”，2000 年電子情報通信学会通信総合大会，B-7-49，2000 年 3 月.
2. 佐野健，野口拓，山本幹，“アプリケーションレベルマルチキャストにおけるネットワーク支援を用いた冗長経路削減方式”，2003 年電子情報通信学会総合大会，B-6-260，2003 年 3 月.

D. 研究会発表

1. 野口 拓，山本 幹，池田博昌，“局所的に FEC を適用した信頼性マルチキャストプロトコル”，信学技報 SSE2000-277，2001 年 3 月.
2. 野口 拓，松田 崇弘，山本 幹，“Network Coding を適用した次世代マルチキャストアーキテクチャ”, 信学技報 NS2001-118, 2001 年 9 月.
3. 野口 拓，山本 幹，“アプリケーションレベルマルチキャストにおけるロバスト なツリー構築法”，信学技報，2003 年 12 月.