The University of Osaka
Institutional Knowledge Archive

Title	マルチメディアネットワークにおける自律分散型サービス制御アーキテクチャに関する研究
Author(s)	木下, 和彦
Citation	大阪大学, 2003, 博士論文
Version Type	VoR
URL	https://hdl.handle.net/11094/452
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

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

The University of Osaka

工乙 8604

マルチメディアネットワークにおける自律分散型サービス制御アーキテクチャに関する研究

2002年10月

木下 和彦

マルチメディアネットワークにおける自律分散型サービス制御アーキテクチャに関する研究

2002年10月

木下 和彦

内容梗概

本論文は，筆者が平成7年から現在までに，大阪大学工学部情報システム工学科・大阪大学大学院工学研究科情報システム工学専攻在学中及び在職中に行ってきた，マルチメディアネットワークにおける自律分散型サービス制御アーキテクチャに関する研究をまとめたものである．

近年，ネットワークの高速化に伴い，インターネット上で動画像や音声などを含めたマルチメディアアプリケーションが提供されつつある．しかし，現在のインターネットはパケット交換による情報伝達処理のみを行っており，各種サービスは主に WWW(World Wide Web) サーバによって提供されている．このため，ネットワーク全体での効率的なサービス制御を行うことができず，負荷集中による応答速度低下や回線の輻輳による通信品質の劣化を招いている．

そこで，サービス制御をネットワーク主導で行う概念が注目を集めつつある．また，大規模ネットワークにおいては，スケーラビリティの観点から，この制御を分散処理環境下で行うことが重要である．しかし，これまでに，TINA(Telecommunication Information Network Architecture) に代表されるいくつかの分散ネットワークアーキテクチャが提案されているが，実用化には至っていない．

これらの背景から，本研究では，大規模なネットワークにおいて，マルチメディアサービスをユーザの要求する QoS(Quality of Service) を満たしつつ効率的に提供する制御アーキテクチャについて，多角的に検討している．

まず，アーキテクチャ全体を支えるプラットフォームとして，モバイルエージェントネットワークを導入する．この時，各制御ノードが分散的にエージェントの位置を記録し，各エージェントが特定のサーバに問い合わせることなく，他のエージェントと通信可能とすることで，大規模ネットワークにおける処理効率を高めている．次に，このプラットフォーム上で高度なサービス制御を行うため，ネットワーク全体の経路情報とサービス情報を一元的に管理する機能的データベースとして，ネットワークマップを構築する．ここで，大規模なネットワークにおいて変動の激しい情報を正確に把握することは難しいため，ネットワークマップを分散配置し，相互に情報を交換して全体を把握する手法を提案する．更に，このネットワークマップに蓄積された情報を基にした呼制御方式について，サービス検索と経路制御の両面で QoS を考慮した方式を提案する．最後に，多対多通信も含めたマルチキャスト通信について，経路を分散的に生成することで，通信参加者数に比例しない処理時間で効率的な制御を行う手法を提案する．また，提案アーキテクチャの性能と有効性について，計算機シミュレーションとプロトタイピングを通して評価している．

研究業績

I. 学術論文誌掲載論文

[1] K. Kinoshita, T. Takine, K. Murakami, and H. Terada: "Holonic Network: A New Network Architecture for Personalized Multimedia Communications Based on Autonomous Routing," *IEICE Transactions on Communications*, Vol. E80-B, No.2, pp. 282–288, Feb. 1997.

[2] 木下和彦，添田純一郎，山井成良，滝根哲哉，村上孝三："多対多通信のための経路制御とネットワーク設計法，" 電子情報通信学会論文誌，Vol.J82-B，No.10，pp. 1773–1781，Oct. 1999.

[3] 谷岡秀昭，木下和彦，滝根哲哉，村上孝三："複数の QoS 要求を満たす経路選択アルゴリズム Fallback+ の提案，" 電子情報通信学会論文誌，Vol.J83-B，No.5，pp. 690–696，May 2000.

[4] 河野圭太，益田智員，木下和彦，村上孝三："複数の QoS を考慮したサービス情報検索方式，" 電子情報通信学会論文誌，Vol.J84-B，No.3，pp. 443–451，Mar. 2001.

[5] K. Kinoshita, T. Masuda, K. Kawano, H. Tanioka, T. Takine, and K. Murakami: "Network Map Architecture using Inter-Agent Communication Techniques for Multiple QoS Routing," *IEICE Transactions on Communications*, Vol. E85-B, No.7, pp. 1333–1341, July 2002.

[6] 木下和彦，神原正義，谷岡秀昭，村上孝三："複数の QoS 要求を満たすマルチキャストツリーを生成するリンク状態アルゴリズム，" 電子情報通信学会論文誌，Vol.J85-B, No.10, pp. 1738–1747, Oct. 2002.

[7] K. Kawano, K. Kinoshita, K. Murakami: "A Mobility-Based Terminal Managemant in IPv6 Networks," *IEICE Transactions on Communications,* Vol. E85-B, No.10, pp. 2090–2099, Oct. 2002.

II. 国際会議発表論文

[1] K. Murakami, S. Uno, K. Kinoshita, M. Higashida, and T. Takine: "Distributed Cooperative Networking Architecture for Multimedia Communications Based on Autonomous Routing," in *Proceedings of the 1996 International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC '96)*, pp. 83–86, Seoul, Korea, July 1996.

[2] K. Kinoshita, H. Tanioka, N. Yamai, T. Takine, and K. Murakami: "Network Map Design for Holonic Routing," in *Proceedings of the 1997 International Seminar on Teletraffic and Network (ISTN '97)*, pp. 189–192, Xi'an, China, Nov. 1997.

[3] K. Kinoshita, J. Soeda, N. Yamai, T. Takine and K. Murakami: "Network Design and Control for Multipoint-to-Multipoint Communications," in *Proceedings of the 1998 IEEE International Symposium on Circuits and Systems (ISCAS '98)*, vol. 6, pp. 529–532, Monterey, CA, May 31–June 3 1998.

[4] H. Tanioka, K. Kinoshita, T. Takine, and K. Murakami: "Fallback+: A Routing Algorithm Subject to Multiple QoS Constraints," in *Proceedings of the ITC-CSCC '98*, vol. I, pp. 319–322, Sokcho, Korea, July 1998.

[5] K. Kinoshita, M. Fujisawa, T. Takine, and K. Murakami: "Performance Evaluation Method of Cellular Networks with Retrial," *Proceedings of the 1999 IEEE International Conference on Personal Wireless Communications (ICPWC '99)*, pp. 389–393, Jaipur, India, Feb. 1999.

[6] M. Kuwada, K. Watanabe, K. Kinoshita, N. Yamai, and K. Murakami: "A Design of Mobile Agent System with Network Transparency," in *Proceedings of the EURO-MEDIA '99*, pp. 170–174, Munich, Germany, Apr. 1999.

[7] K. Watanabe, K. Kinoshita, N. Yamai, M. Kuwada, and K. Murakami: "Agent Gateway: A Location Transparent Agent Platform on Multi-Plane Logical Network," in *Proceedings of the 1999 International Conference on Software, Telecomuunications and Computer Networks (SoftCOM '99)*, pp. 325–334, Split, Croatia, Oct. 1999.

[8] T. Masuda, K. Kawano, H. Tanioka, K. Kinoshita, T. Takine, and K. Murakami: "Network Map Design for QoS Routing on WAN," in *Proceedings of the 5th Asia-Pacific Conference on Communications (APCC '99)*, pp. 196–199, Beijing, China, Oct. 1999.

[9] K. Kawano, T. Masuda, K. Kinoshita, and K. Murakami: "An Efficient Method to Search for the Location of Network Services with Multiple QoSs Guarantee," in *Proceedings of the SoftCOM 2000*, pp. 665–674, Split, Croatia, Oct. 2000.

[10] T. Nagamatsu, K. Kinoshita, N. Yamai, and K. Murakami: "A Fast and Efficient Routing Method for Inter-Agent Communications,"in *Proceedings of the 2000 IASTED International Conference on Software Engineering and Applications (SEA 2000)*, pp. 140–145, Las Vegas, USA, Nov. 2000.

[11] K. Kinoshita, M. Takase, N. Yamai, and K. Murakami: "Remote Program Execution Mechanism with Network Transparency on Multi-Platform Environment," in *Proceedings of the ISTN 2000*, pp. 336–339, Hangzhou, China, Nov. 2000.

[12] K. Kinoshita, H. Tode, and K. Murakami: "Future Personalized Multimedia Communication Network Composed of Optical CDM Switching and Cooperative Network

Maps," in *Proceedings of the 2001 IASTED International Conference on Advances in Communications (AIC2001),* pp. 172–177, Rhodes, Greece, July 2001.

[13] H. Tokumaru, K. Kinoshita, N. Yamai, and K. Murakami: "A Fast and Efficient Dynamic Routing for Inter-Agent Communications," in *Proceedings of the 2001 International Conferences on Info-tech & Info-net,* Conference D, pp. 73–78, Beijing, China, Oct. 29 – Nov. 1 2001.

[14] H. Tanioka, K. Kinoshita, and K. Murakami: "Multipoint-to-Multipoint Routing for Multimedia Communication Service," in *Proceedings of the 2002 IEEE International Conference on Communications (ICC 2002),* C13-3, New York, USA, Apr. 28 – May 2 2002.

[15] M. Kanbara, H. Tanioka, K. Kinoshita, and K. Murakami: "A Multicast Routing Algorithm for Multiple QoS Requirements," in *Proceedings of the 3rd International Network Conference (INC 2002),* pp. 253–260, Plymouth, UK, July 2002.

[16] K. Kawano, K. Kinoshita, and K. Murakami: "A Multilevel Hierarchical Distributed IP Mobility Management Scheme for Wide Area Networks," in *Proceedings of the 2002 International Conference on Computer Communications and Networks (ICCCN 2002),* pp. 480–484, Miami, USA, Oct. 2002.

III. 国内研究会等発表論文

[1] 木下和彦，滝根哲哉，村上孝三，寺田浩詔："ホロニックネットワークを指向した網制御アーキテクチャ，" 電子情報通信学会第2種研究会・ネットワーキングアーキテクチャワークショップ, 4-1-1〜4-1-6, Dec. 1996.

[2] 木下和彦，谷岡秀昭，山井成良，滝根哲哉，村上孝三："ホロニックルーチングのためのネットワークマップ構成法，" 電子情報通信学会技術研究報告, SSE97-62, Aug. 1997.

[3] 木下和彦，藤嶋貴志，滝根哲哉，村上孝三："マルチレート衛星通信における帯域割り当て法，" 電子情報通信学会総合大会，B-3-16，Mar. 1998.

[4] 谷岡秀昭，木下和彦，滝根哲哉，村上孝三："Fallback+: 複数の QoS 要求を満たす経路選択アルゴリズム，" 電子情報通信学会技術研究報告, SSE98-28, May 1998.

[5] 桑田倫彦，木下和彦，山井成良，村上孝三："ネットワーク透過性を考慮したエージェントシステム，" 電子情報通信学会技術研究報告, IN98-129, Nov. 1998.

[6] 益田智員，河野圭太，木下 和彦，滝根哲哉，村上孝三："QoS ルーチングのためのネットワークマップ構成法，" 電子情報通信学会総合大会，B-7-151, Mar. 1999.

[7] 谷岡秀昭，木下 和彦，滝根哲哉，村上孝三："複数 QoS を考慮した経路選択アルゴリズム，" 電子情報通信学会総合大会，B-7-155, Mar. 1999.

[8] 渡辺君人，木下和彦，山井成良，村上孝三："Agent Gateway：多面論理ネットワークによる位置透過性を考慮したモバイルエージェントプラットフォーム，" 電子情報通信学会技術研究報告，SSE99-124，Dec. 1999.

[9] 永松孝之，渡辺君人，木下和彦，山井成良，村上孝三："エージェント間通信のための広域ルーチング方式，" 電子情報通信学会第2種研究会・ネットワーキングアーキテクチャワークショップ，pp. 77–82, Feb. 2000.

[10] 河野圭太，益田智員，木下 和彦，村上孝三："ネットワークマップにおける経路情報を利用したサービス情報検索，" 電子情報通信学会総合大会，B-7-25, Mar. 2000.

[11] 渡辺君人，木下 和彦，山井成良，村上孝三："モバイルエージェント復元のための分散オブジェクトデータベース，" 電子情報通信学会総合大会，B-7-54, Mar. 2000.

[12] 河野圭太，益田智員，木下和彦，村上孝三："経路状況を考慮したマルチメディアサービス検索方式，" 電子情報通信学会技術研究報告，SSE2000-162，Oct. 2000.

[13] 渡辺君人，永松孝之，木下和彦，山井成良，村上孝三："多面的論理ネットワークによる位置透過性を考慮したエージェントシステム Agent Gateway の提案，" 電子情報通信学会ソフトウェアエージェントとその応用ワークショップ（SAA2000），pp. 53–60, Nov. 2000.

[14] 谷岡秀昭，木下和彦，村上孝三："マルチメディア通信のための多対多ルーチング方式，" 電子情報通信学会技術研究報告，SSE2000-276，Mar. 2001.

[15] 徳丸裕章，木下和彦，山井成良，村上孝三："エージェント間通信における迂回ルーチング方式，" 電子情報通信学会総合大会，B-7-60，Mar. 2001.

[16] 木下和彦，益田智員，村上孝三："大規模ネットワークにおける QoS 保証を考慮したフロー管理方式，" 電子情報通信学会総合大会，B-7-119，Mar. 2001.

[17] 神原正義，谷岡秀昭，木下和彦，村上孝三："複数の QoS を考慮したマルチキャストルーチング方式，" 電子情報通信学会総合大会，B-7-129，Mar. 2001.

[18] 神原正義，谷岡秀昭，木下和彦，村上孝三："マルチキャスト通信における複数の QoS 要求を考慮したルーチング方式，" 電子情報通信学会技術研究報告，NS2001-182，Dec. 2001.

[19] 細川晃，木下和彦，山井成良，村上孝三："端末の移動を考慮したユーザ／エージェント間通信方式，" 電子情報通信学会総合大会，B-6-137，Mar. 2002.

[20] 松井知章，河野圭太，木下和彦，村上孝三："ホロニックネットワークにおける端末の移動を考慮した経路切り替え方式，" 電子情報通信学会総合大会，B-6-196，Mar. 2002.

[21] 河野圭太，木下和彦，村上孝三："端末の移動速度に着目した多階層分散型 IP モビリティ制御方式，" 電子情報通信学会技術研究報告，NS2002-44，June 2002.

[22] 河野圭太，木下和彦，村上孝三："多階層分散型 IP モビリティ制御方式のための端末移動特性類推手法に関する検討，" 電子情報通信学会技術研究報告，CQ2002-77,MoMuC2002-25，Sep. 2002.

目 次

第1章 序論 **1**

第2章 自律分散型サービス制御アーキテクチャの基本構成 **5**
2.1 緒言 5
2.2 マルチメディアネットワークの構成 5
2.2.1 マルチメディアネットワークの特徴 5
2.2.2 既存のサービス制御アーキテクチャ 6
2.3 ホロニックネットワーク 8
2.3.1 ホロニックネットワークの概要 8
2.3.2 全光伝達網 9
2.3.3 エージェント通信を基本としたサービス制御網 10
2.4 Agent Gateway 11
2.4.1 位置透過性 11
2.4.2 Agent Gateway の構成 12
2.4.3 エージェントの移動方法 14
2.4.4 ユーザ／エージェント間通信 15
2.4.5 エージェント／エージェント間通信 16
2.4.6 論理ネットワークの設計 17
2.5 実用化に関する考察 18
2.6 結言 19

第3章 ネットワーク情報を管理する機能的データベース **21**
3.1 緒言 21
3.2 経路情報管理方式 21
3.3 コネクション管理方式 24
3.4 サービス情報管理方式 25
3.5 性能評価 27
3.5.1 記憶容量 28
3.5.2 更新頻度と性能の関係 28
3.6 結言 29

第4章 サービス品質を考慮した呼制御方式 **31**
4.1 緒言 31
4.2 サーバ検索方式 31
4.2.1 既存のサーバ検索方式 31
4.2.2 経路情報を利用したサーバ検索方式 32
4.3 複数のQoSを満たす経路選択アルゴリズム 38
4.3.1 既存のアルゴリズム 38
4.3.2 Fallback+アルゴリズム 40
4.4 性能評価 44
4.4.1 検索速度の評価 44
4.4.2 経路選択アルゴリズムの評価 48
4.5 結言 51

第5章 分散型マルチキャスト経路制御方式 **53**
5.1 緒言 53
5.2 1対多通信経路制御方式 53
5.2.1 問題の定式化 53
5.2.2 ユニキャスト経路による実現 54
5.2.3 提案方式 57
5.3 多対多通信経路制御方式 62
5.3.1 既存方式 62
5.3.2 提案方式 64
5.4 性能評価 70
5.4.1 1対多通信経路制御方式に関する評価 70
5.4.2 多対多通信経路制御方式に関する評価 73
5.5 結言 75

第6章 結論 **77**

謝辞 **79**

参考文献 **81**

第1章
序論

近年，光通信技術の発展によるバックボーンネットワークの大容量化と，ADSL (Asymmetric Digital Subscriber Line) やFTTH (Fiber To The Home) の普及によるアクセスネットワークの高速化に伴い，インターネット上で動画像や音声などを含めたマルチメディアアプリケーションが提供されつつある．しかし，現在のインターネットはパケット交換による情報伝達処理のみを行っており，各種サービスは主にWWW (World Wide Web) サーバによって提供されている．ユーザがインターネット上で提供されているサービスを得る際の典型的な手続きは，サーチエンジンで目的とするサービスを提供するサイトを検索し，見つかったサイトに個別にアクセスするというものになるが，この時，回線状況やサーバの利用状況などは全く考慮されていない．このため，効率的なサービス制御を行うことができず，負荷集中による応答速度低下や回線の輻輳による通信品質の劣化を招いている．最近では，CDN (Contents Delivery Network) など，サービス制御をネットワーク主導で行う概念が注目を集めつつあるが，現状ではミラーサーバに対してラウンドロビンDNSを利用した負荷分散を行う程度の機能しかなく，ユーザの要求する品質を提供できる状況にはない [1].

このように，回線系の技術向上に比較して，制御系の整備が遅れているため，VoD (Video on Demand) サービスやVR (Virtual Reality) 技術を応用した臨場感テレビ会議システム等の普及には至っていない．本格的なマルチメディアサービスの普及のためには，単なる負荷分散ではなく，ユーザの要求するサービス品質や，その時点でのネットワークの利用状況などを総合的に考慮した上でのサービス制御が必要となる．

これらの背景から，本研究では，大規模なネットワークにおいて，マルチメディアサービスをユーザの要求するQoS（Quality of Service）を満たしつつ，効率的に提供する制御アーキテクチャについて多角的に検討する．

大規模ネットワークにおいては，スケーラビリティの観点から，全ての制御を分散処理環境下で行うことが重要であり，これまでに，TINA (Telecommunication Information Network Architecture) に代表されるいくつかの分散ネットワークアーキテクチャが提案されているが，実用化には至っていない [2–4]．そこでまず，アーキテクチャ全体を支えるプラットフォームとして，モバイルエージェントネットワークを導入することを提案する [5,6].

モバイルエージェントとは，ネットワーク上を移動しながら，与えられた処理を自律的に行うソフトウェアのことである．また，このようなエージェントが同時に複数稼働し，互いに協

調しながら全体として共通の目的を達成するシステムはマルチエージェントシステムと呼ばれる．マルチエージェントシステムにおいては，エージェントはネットワーク上を移動するだけでなく，必要に応じて他のエージェントと通信し，情報を交換しながら処理を進める [7]．エージェントはそれぞれが自律的に動作するため，ノード間通信で同期をとる必要がなく，分散処理に適している [8,9]．

一方，エージェントは，ユーザの代理人として動作させることで，大規模ネットワーク上に散在する多種多量な情報を容易に利用するための手法としても注目されている．この場合，ユーザはエージェントに適切な仕事を与えることにより，ネットワーク上に存在する情報をその位置を知ることなく利用できることになる．

このように，エージェントシステムは管理者とユーザの双方にとって有用なシステムである．そこで，ユーザがパーソナルエージェントと呼ばれるエージェントをネットワークに送出し，ネットワーク上に存在する各エージェントと協調しながら処理を進める，つまり，エージェントを全ての処理について共通のプラットフォームとするようなネットワークアーキテクチャを提案する [10–12]．

しかし，ここで，大規模ネットワークにおいてマルチエージェントシステムを効果的に利用するためには，ネットワーク上での位置透過性を実現することが必要不可欠である．ここでの位置透過性とは，ユーザ及びエージェントが，必要とするリソースや協調相手となるエージェントの位置を意識する必要がないということである [13]．

この位置透過性を実現する技術として，CORBA (Common Object Request Broker Architecture) が挙げられる [15,16]．CORBA で定義されているネーミングサービスを用いれば，各種リソースやエージェントをオブジェクトとして登録することによって，その位置を意識せずにそれらを利用可能となる．しかし，現在の実装では，これらの登録は1つのデータベースによって集中管理されることになっており，大規模ネットワークでの利用はスケーラビリティの点で現実的でない．

そこで，位置透過性をスケーラブルに実現するため，各制御ノードが分散的にエージェントの位置を記録し，各エージェントが特定のサーバに問い合わせることなく，他のエージェントと通信可能とするプラットフォームを提案する．これによって，自律分散的なサービス制御を行うための基盤となるネットワークが構築できる．

次に，このエージェントプラットフォーム上でのサービス制御方式について検討する．高度なサービス制御のためには，ネットワーク情報の管理とそれを利用した呼制御が必要となる．

まず，ネットワーク情報を管理するため，本研究ではネットワークマップと呼ばれる機能的データベースを導入する [17–19]．このネットワークマップは，ネットワーク全体における接続構造や故障箇所，回線状況などを常に把握し，かつサービスプロバイダに関する情報も蓄積する．しかし，大規模なネットワークにおいて回線状況のような変動の激しい情報を，正確かつリアルタイムに把握することは難しい．

そこで，ネットワークをいくつかのサブネットに分割して管理し，そのサブネット単位で整理された情報を相互に交換してネットワークマップを構成する手法を提案する．各ネットワークマップは当該サブネット内のスイッチングノード（SN: Switching Node）に関する情報を正確に把握し，そこからサブネット全体としての回線状況情報を抽出し，他のネットワークマップとの間で交換する．

具体的には，各ネットワークマップを管理するエージェント（NMMA: Network Map Manager Agent）は当該サブネット内のSNに関しては，コネクションが張られる度にそれを記録し，完全な状況把握を行う．そこから，他のサブネットに隣接するSNについてのみ，任意の2つのSN間の回線状況を他のNMMAに通知する．また，そのSNと隣接する他のサブネットのSNとの間の回線状況についても同様に情報を送る．

このとき，回線状況を抽出するためには複数のQoSを考慮した経路選択アルゴリズムが必須である．この問題は，複数の加法性QoS制約を満たす経路を発見する数理計画問題として定式化できるが [20,21]，NP完全問題であることが知られている [22]．そのため，これまでに多くの発見的アルゴリズムが提案されてきたが，実用的な速度で目的経路を十分高い確率で選択するものは存在しない [20–26]．

そこで本研究においては，Dijkstraのアルゴリズム及びFallbackアルゴリズムを拡張した，高速かつ精度の良い経路選択アルゴリズム *Fallback+*を提案する [27–30]．このアルゴリズムは，実際にサービスを提供する際の経路選択にも用いられる．

ここで，従来の研究では，単に利用者が要求する複数のQoSを満たす経路を選択することのみが考慮されてきたが，Fallback+では，利用者が要求する複数のQoSを満たしつつ，ネットワーク資源を有効に利用する経路を選択することができる．つまり，利用者が要求する複数のQoS制約に加えて，ネットワーク管理者の視点からみた指標を最小化する経路を探索している．

また，高速かつ柔軟なサービス提供をするためには，ネットワークマップがネットワーク全体のサービスプロバイダの位置とその内容を把握しておくことも必要となる．ここでも，負荷集中を回避するため，分散管理を行う．特に，検索の高速性が要求されることから，木構造を利用した論理的な階層管理を用いる [31–34]．

具体的には，各ネットワークマップにサービスコンテンツ，サービス提供位置情報及びサービス提供に最低限必要となるQoS条件を保持させるとともに，仮想的に，階層化されたネットワークマップを構成し，この上位層に設置されたネットワークマップには，あるサービスに対してどの子ノードに情報が存在するかどうかをナビゲートする程度の情報を蓄積する．このように，上位層に存在するネットワークマップの保持する情報を簡易なものにとどめることにより，下位層のたネットワークマップが保持する複数の同一コンテンツの情報をひとまとめに管理することが可能となり，サービス情報の追加，削除，修正に伴う上位層でのデータベース更新頻度を軽減できる．例えば，既に存在するサービスコンテンツの情報が追加された場合，この追加処理の前後で，そのノードにコンテンツが存在するかどうかという情報は変化しないため，親ノードに設置されたネットワークマップの保持する情報は更新されない．今後のマルチメディアネットワークでは，一般に，ネットワーク中に複数の同一コンテンツを提供するノードが存在することが予想されるため，この方式は有効であると考えられる．

次に，このネットワークマップに蓄積された情報を基にした呼制御方式について検討する．一般に，マルチメディア通信サービスに要求されるQoSは，伝送遅延や遅延の揺れ等のネットワーク状況に起因するQoSと，映像品質等のサービスプロバイダによって決められるQoSの2つに分類できる．マルチメディアネットワークにおける呼制御では，これらのQoSを満足するため，コンテンツサーバの選択と経路制御が重要な要素となるが，これらは相関のある処理であるため，統一的に扱うことが重要である．

サービスの利用に際して，ユーザはパーソナルエージェントに目的コンテンツと要求するQoS

を与えて，ネットワークに送出する．このエージェントはまず最も近いネットワークマップに移動し，そこで保持されているサービス情報と経路情報を用いて，ユーザの要求するコンテンツをユーザの要求を満たす範囲で提供可能かどうかを判断する．ここでの情報検索において要求を満たすサービスが発見できなければ，前述の階層構造に従って上位層のネットワークマップへ移動し，検索を続ける．

ここで，本研究で対象とするマルチメディアネットワークはユーザに対して複数のQoSを保証することを前提としているが，これらのQoSの中で，例えば遅延等の経路状況に依存するQoSについては，ネットワーク中のノード間距離が近いほどその要求が満たされやすいことが予想される．しかしながら，物理的に隣接しているノードであっても，単純に論理的な木構造を利用した情報検索では数ホップ離れたノードとなる可能性がある．つまり，比較的QoSを満たしやすいことが予想される隣接ノードの情報検索が遅れることになる．そこで，本研究では，物理的に近距離のノードを優先した検索を行う．これにより，比較的QoSを満たしやすいことが予想されるノードを優先的に検索することが可能となり，効率的なサービス情報検索を行える．

以上の手続きと前述したFallback+アルゴリズムを用いた経路選択によって，高速にユーザの要求するQoSを満たすコンテンツサーバの位置と，そのサーバの存在するノードからユーザの存在するノードまでの経路が決定される．

ところで，今後のネットワークにおいては，1対多及び多対多のマルチキャスト通信に対する考慮も重要である．そこで，ユニキャスト経路を求めるアルゴリズムであるFallback+を，計算量を増やすことなくマルチキャスト経路に対応させることで，高速にQoSを考慮した1対多通信を実現する方法を提案する [35–37].

更に，多対多通信は，単一の1対多通信の経路を用いて実現することも可能であるが，この場合には全てのトラヒックが1つの経路上に集中することとなるため，エージェントの自律性を利用して，複数の1対多経路を並列に生成することで高速化を図りつつ，相互に協調して負荷集中を避ける方式を提案する [38,39].

以降，第2章では，エージェント通信を基本とした自律分散型サービス制御アーキテクチャを提案し，その基本構成と動作例を示す．また，実用化に関しても考察する．第3章では，提案アーキテクチャ上でネットワーク情報を管理するデータベースであるネットワークマップの機能と情報管理方式について説明し，更に，その性能を計算機シミュレーションとプロトタイプ実装を通じて評価した結果を示す．第4章では，ネットワークマップを基盤とした呼制御方式について，サービス検索と経路制御の両面から検討するとともに，複数のQoS指標を考慮した経路選択アルゴリズムを提案する．また，その性能を処理時間と負荷分散性の面から評価した結果を示す．第5章では，多対多通信を含めたマルチキャスト通信について検討し，複数のQoS指標を考慮した上でネットワークにかかる負荷を軽減・分散し，かつ処理時間が通信参加者数に比例しない制御方式を提案する．最後に第6章では，結論として，本研究を総括し，提案アーキテクチャ全体の評価と実現可能性に関して今後に残された課題などについて述べる．

第2章

自律分散型サービス制御アーキテクチャの基本構成

2.1 緒言

本章では，まず，マルチメディアネットワークの特徴と既存のサービス制御アーキテクチャについて述べ，その問題点を明らかにした上で，マルチメディアネットワークにおけるサービス制御アーキテクチャの基本構成を提案する．

マルチメディアネットワークにおいては，大規模なネットワークでも高速かつ柔軟に動作することと，耐故障性の観点から，全てのノードが自律分散的に動作しながら互いに協調して処理を進めることが重要である．そこで，本研究ではモバイルエージェントを統合的なプラットフォームとし，これを基盤とした自律分散協調制御を行うホロニックネットワークを提案する [10–12]．

また，このアーキテクチャのためのエージェントシステムとして，*Agent Gateway* を提案する [5,6]．これは，ネットワーク透過性を分散的に実現するシステムであり，大規模ネットワークにおけるエージェントの効率的な協調動作を支援する．

2.2 マルチメディアネットワークの構成

2.2.1 マルチメディアネットワークの特徴

これまでの通信網は，電話や無線に代表される音声通信やケーブル TV 網による映像放送等，単一メディアの情報を伝送するものであった．また，運用開始当初のインターネットも，データ通信のみを扱うシングルメディアネットワークだったと言える．しかし，近年，光通信技術の発展や通信端末の高度化により，1つの網で音声・映像・テキストなど複数のメディアを利用した情報伝達が可能となってきた．このような網をマルチメディアネットワークと呼ぶ [40,41]．

マルチメディアネットワークにおいては，性質の異なるトラヒックが混在するため，その品質保証が困難となる．そこで，これまでに多くの研究がなされてきた [42,43]．また，多種多様なサービスを提供することから，その制御に柔軟性が要求される．

2.2.2 既存のサービス制御アーキテクチャ

【インテリジェントネットワーク】

インテリジェントネットワーク（IN: Intelligent Network）及び高度インテリジェントネットワーク（AIN: Advanced IN）は，ネットワークを物理的な回線接続や情報伝送を行う伝達層とそれに高度なサービスを付加する高機能層の2層に機能分割し，これらを論理的に独立させて扱うアーキテクチャである [44]．このINアーキテクチャの概念図を図2.1に示す．

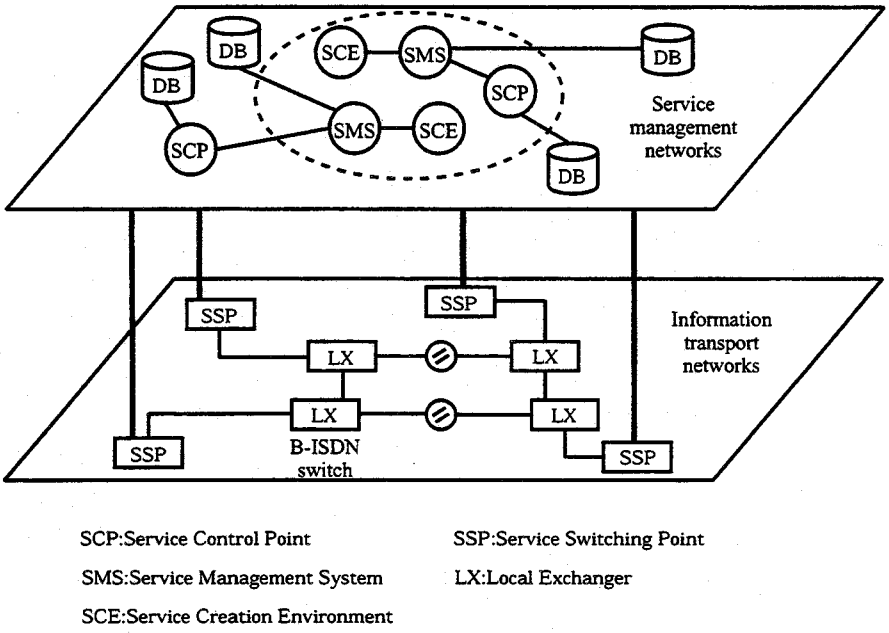


図 2.1: インテリジェントネットワークの概念図

ここで各部の役割は以下のようなものになっている．

- LX：交換器．物理的な回線交換を行う．
- SSP：サービス交換点．伝達層においてサービス信号を検知し，高機能層に転送する．
- SCP：サービス制御点．SSPからの要求に応じて処理を特定し，サービスデータを扱う．
- SMS：サービス管理システム．SCPが保有するデータベースへの情報の追加と更新を行いつつ，網全体を管理する．
- SCE：サービス生成環境．サービスの内容を規定する．

INおよびAINはITU-Tによって標準化され，フリーダイヤルなどのサービスが実用に供されている [45]．しかし，近年，インターネットとの連携も検討されているが，現時点での応用は電話網への適用のみである [46,47]．

【TINA】

TINAは，ネットワークノード上に仮想的に分散処理環境を構築し，その上で各種アプリケーションを制御・管理するアーキテクチャである [2]．このTINAの概念図を図2.2に示す．

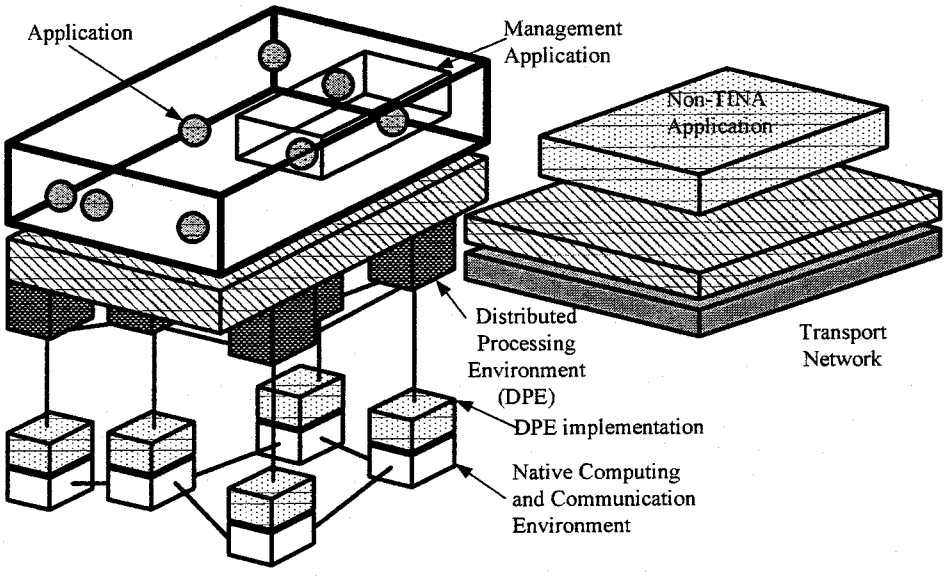


図 2.2: TINAの概念図

各ノードにはDPE（Distributed Processing Environment） implementationが搭載され，これが全体としてDPEを構成する．TINA対応のアプリケーションはこのDPE上で実行され，それ以外のアプリケーションはこれまで通り，一般伝達網上で提供される．

このように，TINAの仕様は分散性に優れており，ITU-Tによって標準化もなされているが，その実装方法の検討に課題を残している [3,4]．

【CDN】

現在のインターネットでは，各ルータはパケット交換による情報伝達処理のみを行っており，各種サービスは主にWWWサーバによって提供されている．このため，効率的なサービス制御を行うことができず，負荷集中による応答速度低下や回線の輻輳による通信品質の劣化を招いている．そこで，あるコンテンツを提供するサーバを複数設置し，ユーザからのサービス要求をそれらのサーバに対して適切に割り振るCDNが注目を集めている [1]．

図2.3にCDNの概念図を示す．CDNでは，ユーザはまず要求するコンテンツの提供位置をBrokerに問い合わせる．これを受けたBrokerはクライアントの位置や，サーバの負荷状況などから最適なサーバを選択し，URLを回答する．そして，クライアントはそのURLへアクセスし，サービスを受けるという手順になる．

このように，CDNは現在のIP網においてネットワーク主導でサービスを提供できる枠組みであるが，現状では，ミラーサーバに対してラウンドロビンDNSを利用した負荷分散を行う程度の機能しかなく，ユーザの要求する品質を提供できる状況にはない．

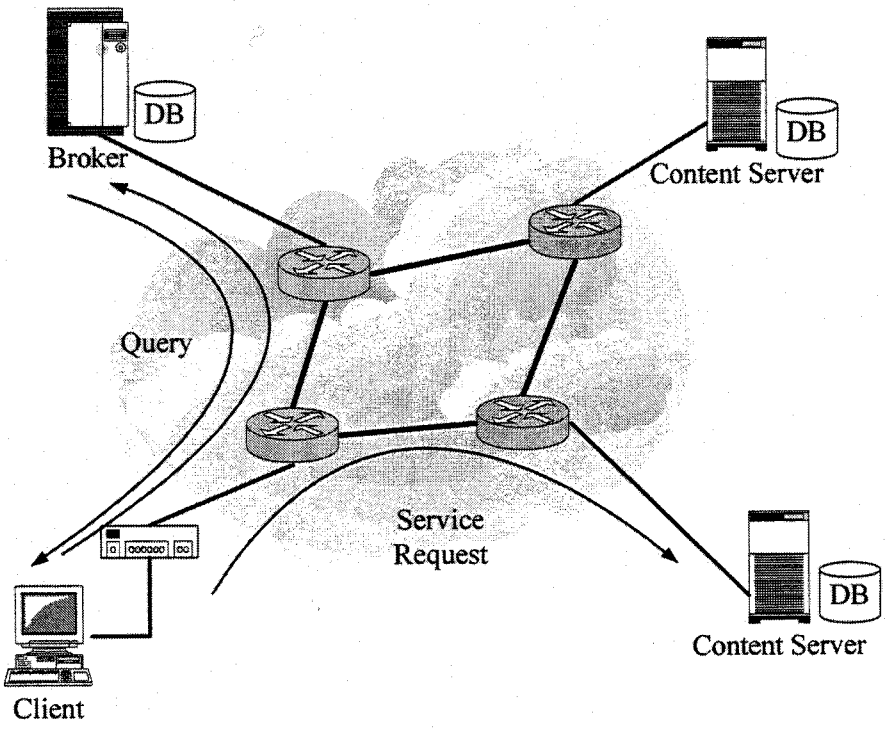


図 2.3: CDN の概念図

2.3 ホロニックネットワーク

2.3.1 ホロニックネットワークの概要

本研究では，新たなマルチメディアネットワークアーキテクチャとしてホロニックネットワークを提案する．これは全光伝達と自律分散制御を特徴とし，パーソナル・マルチメディア通信に対応したネットワークである [10–12].

将来のマルチメディア通信サービスの実現のためには，高速な伝達網と柔軟なサービス制御網が必要となる．特に，伝達網の高速化が実現されつつある現況では，その高速性を損なわずに高度なサービスを提供できる枠組みの考案が急務となっている．そこで，ホロニックネットワークでは，図 2.4 に示すように，高度サービス制御機能と情報伝達機能とを論理的に分割し，更に光伝送と電子制御を完全に分離している．

OTN（Optical Transport Network）は全光の伝達網であり，各 HN（Holonic optical Node）は伝送される情報に付与されたヘッダを基に自律的にスイッチングを行う．このとき，光ファイバを用いた超高速伝送網においては，交換器において電子的制御が介入すると，その部分が伝送速度上のボトルネックとなるため，OTN では O/E・E/O 変換を廃し，全光での情報伝送を行う [48].

一方，PNN（Personal Navigation Network）は自律分散，エージェント通信を基本とした電子制御網であり，ユーザによるサービスカスタマイズを実現すると共に，分散処理環境下でのサービス提供を目指している．つまり，各ノードが自律的に動作しながら，ネットワーク全体

としては共通の目的を達成することとなり，処理性能やスケーラビリティ，安定性等，これからの社会基盤としてのネットワークに対する要求を全て満たすことのできるアーキテクチャとなっている．

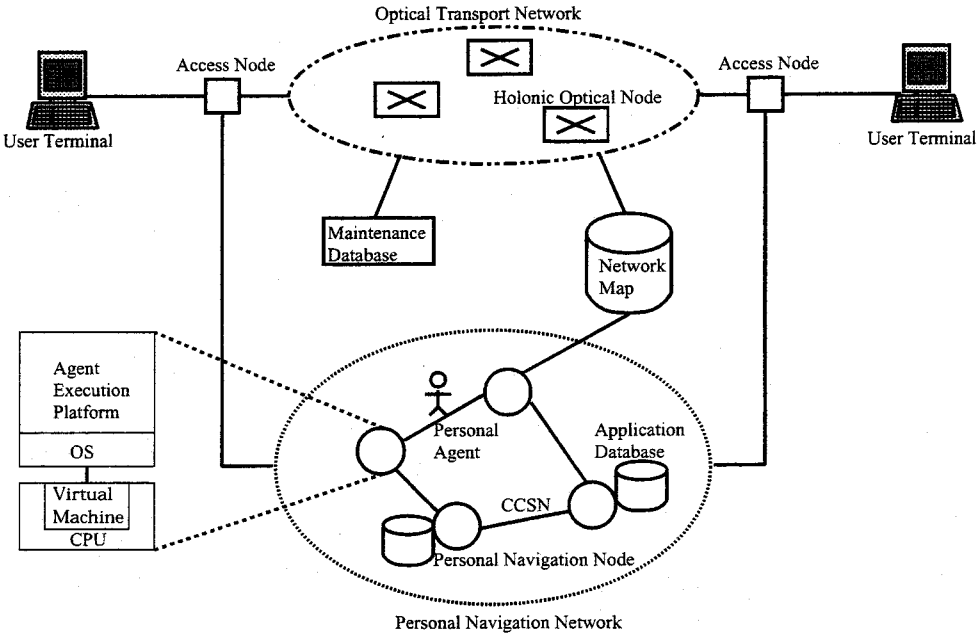


図 2.4: ホロニックネットワークの概念図

また，ホロニックネットワークでは，ネットワークの可視化という概念を取り入れている．従来の，ユーザに対して内部が隠蔽されたネットワークは，ユーザが伝送過程を意識せずにすむという点で，回線サービスのみを提供するには有効であった．しかし，種々の情報サービスを融合した将来のパーソナル・マルチメディア通信サービスを利用するには，ユーザ自身がネットワークサービスをカスタマイズし，使いこなすことができること，そのためにネットワークを直視できることの方がより重要になると考えられる．そこで，このネットワークでは，ネットワーク内部の情報をユーザに対して公開する．これによって各ユーザは，個人の要求とネットワーク状況に応じたサービスを独自に生成し，享受することが可能となる．

ホロニックネットワークでは，ユーザは要求するサービスを個人の嗜好に応じてカスタマイズし，代理人となるパーソナルエージェントに記述して，ネットワークに送出する．パーソナルエージェントは PNN 上で他のエージェントと協調しながら，サーバの選択，サービス品質の交渉，経路の決定等を行い，最終的にサーバからユーザ端末までのコネクションを確立する．これらの制御方式については後の章で詳しく述べる．

2.3.2 全光伝達網

本節では，全光伝達網である OTN について説明する．

高度なマルチメディアネットワークサービスの提供のためには，Tbps 級の超高速・大容量

ネットワークが必要となるため，光の高速性を最大限活用する全光型ネットワークの利用が必須である．

この時，光多重化方式としては，これまで時分割多重（TDM: Time Division Multiplexing）方式と波長分割多重（WDM: Wave-length Division Multiplexing）方式が主として研究されてきた．しかしながら，両方式とも多重度の向上には限界があることが指摘されている [49]．一方，無線分野で実用化されている符号分割多重 (CDM: Code Division Multiplexing) 方式が光多重化方式としても注目を集めており，これを利用することによって更に通信容量を増加できる可能性がある [50]．

また，全光型ネットワークにおいて，光素子で実現可能な論理機能では，従来のような複雑な機能を持つスイッチング機能を実現するのは困難であるため，自律ルーチング方式が適していると考えられる．

そこで，ホロニックネットワークでは，光 CDM を用いて多重度を上げつつ，符合をラベルとしたラベルスイッチングを用いる [51]．この時，全体の経路は PNN によってソースルーチングを用いて決定され，送出されるパケットは各スイッチにおける交換先に応じたラベルで多重符号化される．

図 2.5 にこの光 CDM を用いたスイッチングの例を示す．ソースノードで符合 A,B,C で多重化されたパケットは，中継ノードで復号されながら，その符合に該当する出力先に転送される．この時，電子的な制御は全く介入しないため，全光での高速なスイッチングが実現される．

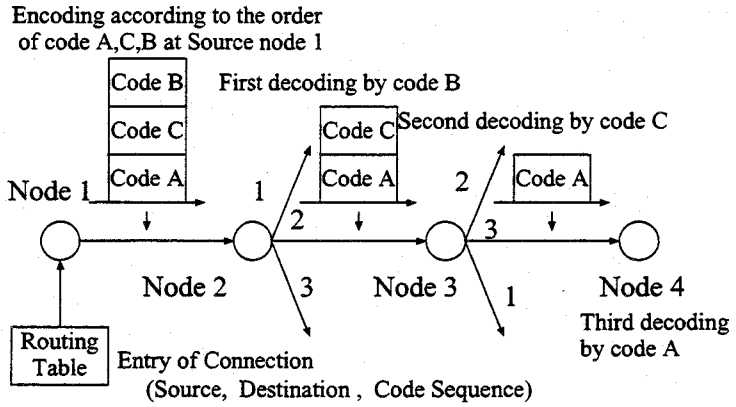


図 2.5: 光 CDM スイッチング

また，今後のマルチメディアネットワークにおいては，無線通信によるアクセスも重要である．本稿での詳細は記述は割愛するが，ホロニックネットワークでは，この点についても対応している [52–55]．

2.3.3 エージェント通信を基本としたサービス制御網

次に，電子的制御網である PNN について説明する．

近年，ユーザの要求を満たすネットワークサービスを提供するためにエージェント技術を応用する研究が注目されている [8,9]．エージェントとは，ユーザやサービスプロバイダの代理人

として，与えられた仕事を自律的に処理するソフトウェアのことである．特に，ネットワーク上のノードを移動しながら処理を行うエージェントはモバイルエージェントと呼ばれ，多数の研究事例がある [56–59]．本稿では，以後,「エージェント」と記せばこのモバイルエージェントを指すものとする．

このようなエージェントが同時に複数稼働し，互いに協調しながら全体として共通の目的を達成するシステムはマルチエージェントシステムと呼ばれる．マルチエージェントシステムにおいては，エージェントはネットワーク上を移動するだけでなく，必要に応じて他のエージェントと通信し，情報を交換しながら処理を進める [7]．

一方，このマルチエージェントシステムを大規模ネットワークにおける網制御に利用することを考えると，エージェントはそれぞれが自律的に動作するため，ノード間通信で同期をとる必要がなく，分散管理に適していると言える [10,60,61]．また，プログラムの更新も新しいエージェントを送り込むことによって実現可能であるため，管理負荷が軽減されるという利点もある．

このように，エージェントシステムはユーザと管理者の双方にとって有用なシステムである．そのため，ホロニックネットワークでは，ユーザがパーソナルエージェントと呼ばれるエージェントをネットワークに送出し，ネットワーク上に存在する各エージェントと協調しながら処理を進める，つまり，エージェントを全ての処理について共通のプラットフォームとするようなネットワークアーキテクチャを採用する．

しかし，ここで，大規模ネットワークにおいてマルチエージェントシステムを効果的に利用するためには，ネットワーク上での位置透過性を実現することが必要不可欠である．ここでの位置透過性とは，ユーザ及びエージェントが，必要とするリソースや協調相手となるエージェントの位置を意識する必要がないということである [13,14]．

この位置透過性を実現する技術として，CORBA が挙げられる [15,16]．CORBA で定義されているネーミングサービスを用いれば，各種リソースやエージェントをオブジェクトとして登録することによって，その位置を意識せずにそれらを利用可能となる．しかし，現在の実装では，これらの登録は 1 つのデータベースによって集中管理されることになっており，大規模ネットワークでの利用はスケーラビリティの点で現実的でない．

そこで本研究では，大規模ネットワークにおける網制御プラットフォームへの適用を目的とし，位置透過性をスケーラブルに実現するエージェントシステム *Agent Gateway* を提案する．

2.4 Agent Gateway

2.4.1 位置透過性

前述のように，大規模ネットワークにおいてマルチエージェントシステムを有効に利用するためには，位置透過性の実現が重要である．本研究では，具体的に以下の点に関する位置透過性について考慮する．

【リソースの指定に関する位置透過性】

従来のエージェントシステムにおいては，ユーザは処理に必要とするリソースをその格納ノードの位置と共に明示的に記述する必要がある．また，複数のノードに存在するリソースを利用する際には，格納ノードへの移動順も事前に設定しなければならない．よって，対象となるリソースの増加に伴い，ユーザの負担も大きくなる．なお，ここでリソースとは，そのノードで提供されているサービスや格納されているデータなど，エージェントがアクセスする対象一般を指すものとする．

そこで，リソースに関して一意に定められたIDを指定すれば，その物理位置についてはシステム側で管理し，エージェントをそのリソースの存在するノードに移動させる機能が求められる．また，必要リソースが多数に渡る場合，それらをグループ化し，そのグループ名を指定することで，全てのリソースにアクセスできる仕組みが実現されることが望ましい．

【ユーザ／エージェント間通信に関する位置透過性】

前述のように，エージェントはネットワーク内を自律的に移動する．また，マルチエージェントシステムでは，必要に応じてエージェントが複製されることもある．そのため，任意の時点においてユーザがエージェントの現在位置及び総数を把握することは難しい．よって，活動中のエージェントに対してユーザが仕事内容の変更や中止などの命令を送ることは極めて困難となる．

そこで，エージェントの位置及び数をシステムが管理し，ユーザはエージェントに一意に与えられたIDを指定するだけで，そのエージェントにメッセージを送信できる機能が必要となる．

【エージェント／エージェント間通信に関する位置透過性】

マルチエージェントシステムにおいては，エージェント間通信によってその処理効率を高めることができるが，前項で述べた点から，エージェントが他のエージェントの位置及び数を把握することは難しい．

そこで，同様にシステムが全てのエージェントの位置及び数を管理し，エージェントは他のエージェントのIDを指定することで通信可能とすることが求められる．

2.4.2 Agent Gatewayの構成

本研究では，前節で示した点を考慮したエージェントシステムAgent Gatewayを提案する[62–64]．Agent Gatewayでは，システムがリソース及びエージェントの存在位置を管理することにより，ユーザ及びエージェントはそれらを意識することなく利用できる．また，大規模ネットワークにおける利用を前提として，スケーラビリティを考慮した設計となっている．

Agent Gatewayの基本構成を図2.6に示す．

本システムでは，エージェントが活動する全てのノードに，Agent Gateway Core System(AG)がインストールされる．AGは，エージェントの基本活動である生成・実行・移動・終了など

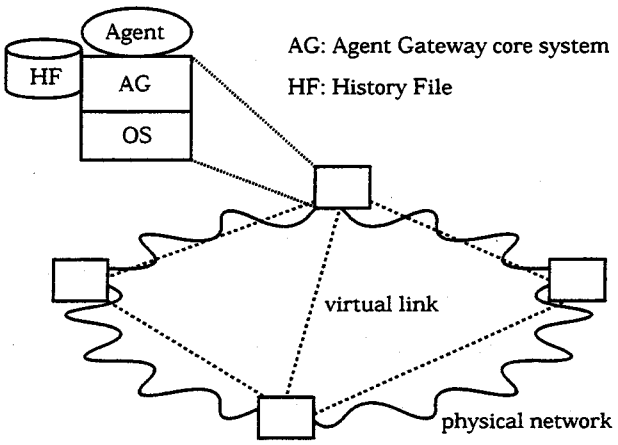


図 2.6: Agent Gateway の構成

をサポートするミドルウェアである．

AG 間には仮想的なリンク（仮想リンク）を設定し，全体として論理的なネットワーク（論理ネットワーク）を構成する．この論理ネットワークは，AG の管理者が物理ネットワークとは独立に設定できるものである．エージェントの移動はこの論理ネットワーク上の仮想リンクに沿って行われる．また，この論理ネットワークは多重に設定することができる．つまり，1つのノードは複数の論理ネットワークに属することが可能である．例えば，図 2.7 では 6 つのノードからなる物理ネットワーク上に 2 つの論理ネットワーク A，B が設定されているが，AG2 はその両方に属している．

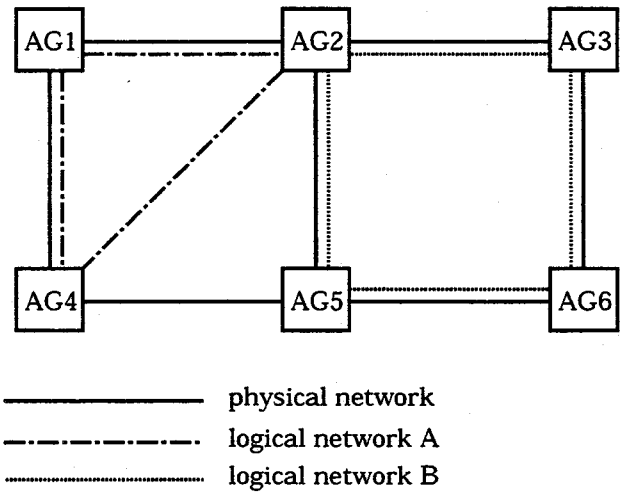


図 2.7: 論理ネットワークの設定例

ユーザはこの論理ネットワークの 1 つを指定することで，その論理ネットワークに属する全てのノードに対して，その物理位置及び総数を意識せずにエージェントを移動させることができる．

また，上記の例ではノードの集合として論理ネットワークを設定しているが，これをリソースの集合として定義することも可能である．これによって，例えばコンテンツ配信サービスを提供する際に，エージェントを用いて負荷分散制御を行うような応用を考えた場合，同一コンテンツを提供するサーバの集合として論理ネットワークを規定すれば，サーバの移動や増減が生じたとしてもシステム側で論理ネットワークの設定を変更するだけで，エージェント側からは透過的にリソースにアクセス可能となる．

また，各AGはそのノードにおけるエージェントの活動結果を履歴ファイル（History File: HF）に記録する．具体的には，そのノードでエージェントが新たに生成されたり，活動を終了したりした場合にはその旨を記録し，隣接ノードからエージェントが移動してきたり，隣接ノードへエージェントを移動させたりした場合には，そのノードIDを記録する．

2.4.3 エージェントの移動方法

次に，本システムにおけるエージェントの具体的な移動方法を述べる．

まず，ユーザが明示的に移動先及びその順序を指定した場合には，それに従った経路でエージェントを移動させる．

一方，ユーザが前節で述べた論理ネットワークを指定することで複数のノードあるいはリソース全てへのアクセスを要求した場合には，その順序をシステム側が決定する．この時に考慮すべき指標としては，各ノードへの移動に要する時間とネットワークにかかるトラヒック量が考えられる．例えば，最初にエージェントを生成したノードで，移動先ノード数に等しい数のエージェントを複製し，同時に移動させれば最も早く全てのリソースにアクセスできるが，それによって発生するトラヒック量は膨大なものとなる．逆に，ノードの移動順を巡回セールスマン問題として解き，得られた順序に従って1つのエージェントを移動させていくと，トラヒックは最小になるが，全てのリソースに到達するまでに要する時間は長くなる．

そこで，この経路決定方法としてフラッディングアルゴリズムを用いることを提案する．これは，全てのノードに対して最短の経路でエージェントが到達することが保証されているため，高速な処理が可能であり，かつ，必要以上のトラヒックを発生させないという利点がある [65]．

例えば，図2.8において，AG1，AG2，AG3，AG4から構成された論理ネットワーク上の全てのノードに，AG1にいるエージェントが移動する場合を考える．

初めに，AG1で生成されたエージェントがこのノードでの処理を終えると，システムはAG1に隣接しているAG2，AG3を移動先の候補とし，これらに既に移動したことがあるかどうかを調べるためにAG2，AG3のHFを参照する（step1）．AG2，AG3のHFには，このエージェントの到着記録は存在しないので，AG1ではクローンを作成し，AG2とAG3の両方にエージェントを移動させる（step2）．AG1のHFには，このエージェントが生成されたことと，AG2及びAG3に移動したことがその時刻と共に記録される．

次に，AG2及びAG3でのエージェントの処理が終わると同様の手順が行われる．ここでは，AG2に移動したエージェントが，より早く処理を終えたと仮定する．この場合，AG2はエージェントをAG4へ移動させようとする．なぜなら，仮想リンク接続先はAG1とAG4であるが，AG1は移動元であるため新たな移動先とはしないからである．AG2はAG4のHFを参照し（step3），到着記録がないことを確認してエージェントを移動させる（step4）．

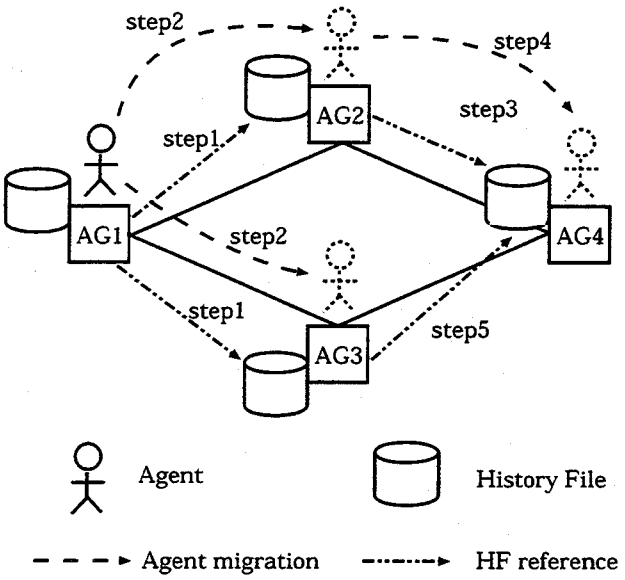


図 2.8: エージェント移動の例

AG3 でも同様に，次の移動先候補である AG4 の HF を参照するが，既に同じエージェントの到着記録があるため，エージェントを移動させることはない（step5）．

また，エージェントが与えられた仕事を全て終了し消滅する場合には，AG はその旨を HF に記録すると共に移動元のノードに通知する．通知を受け取った AG は同様に HF への記録と移動元ノードへの通知を行う．

このようにユーザは，AG が稼動する論理ネットワーク内の目的ノードを明示的に示すことなく移動させることができ，また，複数個のノードを巡回させる際に個々のノードを意識することなく位置透過的に移動させることができる．

2.4.4 ユーザ／エージェント間通信

Agent Gateway におけるユーザ／エージェント間通信には，HF を利用する [66–70]．前節で述べたように，HF にはそのノードからのエージェントの移動先が記録されているため，AG がエージェント宛てのメッセージを受け取った場合，HF を参照して，そのノードに当該エージェントが存在すればそのエージェントにメッセージを伝達し，そうでなければ移動記録に従って隣接ノードにメッセージを中継する．ただし，そのエージェントが既に活動を終えていることが記録されているならば，メッセージはそこで破棄される．

図 2.9 は，AG1 で生成されたエージェントが図 2.8 で示された手順に従って移動し，AG3 と AG4 に存在している状況を示している．ここで，ユーザがこれらのエージェントに対して，メッセージを送ることを考える．ただし，ユーザにはその時点でのエージェントの存在位置は分からないことに注意する．

この時，ユーザはまずそのメッセージをエージェントの生成ノードである AG1 に送る．AG1 はそこでの当該エージェントの活動履歴を知るため，HF を参照する（step1）．HF には当該

エージェントがAG2とAG3に移動したことが記されているため，AG1はこのメッセージをAG2とAG3に転送する（step2）．このメッセージを受け取ったAG2では，同様にHFを参照し（step3），AG4に転送する（step4）．AG4には当該エージェントが存在するため，メッセージがそのエージェントに渡される．また，HFを参照しても（step5）これ以上の移動記録はないため，転送は終了する．一方，AG3でも同様に当該エージェントにメッセージが渡され，通信は終了する．

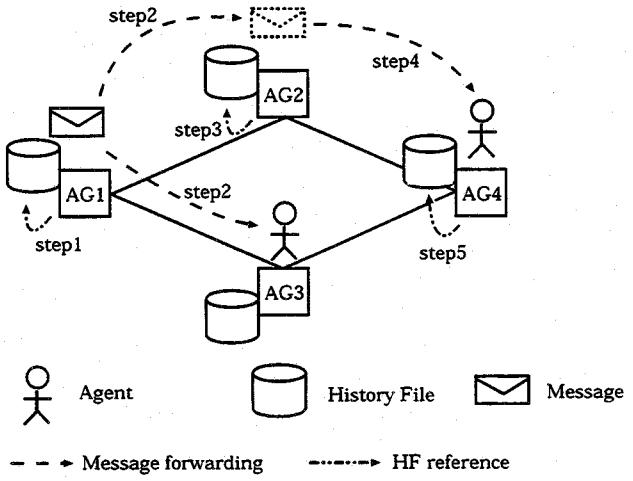


図 2.9: ユーザ／エージェント間通信の例

これにより，ユーザはメッセージを送りたいエージェントのIDを指定するだけで，その現在位置や数を意識することなく通信可能となる．また，前節で述べたように，HFにはエージェントの移動先が記録されていると共に，既に活動を終了したかどうかも記されているため，不必要なメッセージが送信されることもない．

2.4.5 エージェント／エージェント間通信

エージェント／エージェント間通信においても，HFを利用してエージェントの移動経路を辿ってメッセージを伝達することで，位置透過的な通信が可能である．この場合，エージェントの送ったメッセージはそのノードから，HFを参照しながら移動履歴を遡ることになる．

図2.10は，図2.9と同様にAG3とAG4にエージェントが存在する状況を示している．ここで，AG4にいるエージェントが他のエージェントに対してメッセージを送ることを考える．

まず，AG4はメッセージを送出したエージェントがどのノードから到着したかを知るために，HFを参照する（step1）．その結果，このエージェントはAG2から移動してきたことが分かるため，AG2にメッセージを転送する（step2）．AG2でも同様にHFを参照し（step3），AG1にメッセージを転送する（step4）．AG1でHFを参照すると（step5），当該エージェントはここで生成され，AG2とAG3に移動したことが分かる．ここでAG2はこのメッセージの転送元であるため，AG3にメッセージを転送する（step6）．AG3には目的のエージェントが存在するため，このメッセージを渡す．また，HFを参照しても（step7）これ以上の移動先は存在し

ないため，通信は完了する．

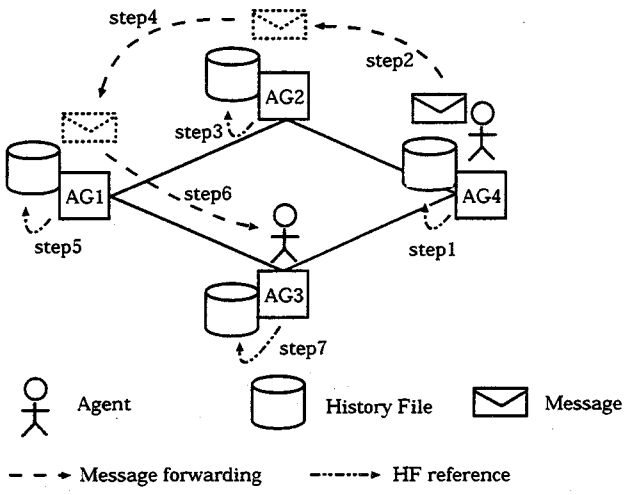


図 2.10: エージェント／エージェント間通信の例

これにより，エージェント／エージェント間通信においても位置透過性が実現されていることが分かる．なお，図 2.10 では，簡単のために通信先となるエージェントが 1 つしかない状況を示したが，対象エージェントが複数であっても提案システムは問題なく動作することは明らかである．

2.4.6 論理ネットワークの設計

上記のように，Agent Gateway において論理ネットワークはリソースの指定に関する位置透過性を実現すると共に，エージェントの移動及びメッセージの伝達の経路を規定する役割を果たす．特に後者の点で，論理ネットワークの形状によってエージェントの移動速度や，物理ネットワークに課せられるトラヒック量が決定されるため，その設計法は非常に重要である．

論理ネットワークの設計は，物理ネットワーク上の特定のノードにインストールされた AG を相互に接続する問題に帰着されるため，マルチキャストツリーの生成と同様の考え方を適用可能である．

ただし，ここで，エージェントは全てのノードで生成される可能性があるため，単なるマルチキャストではなく，多対多のマルチキャストと同様に扱わなければならない点に注意する．この場合，どのノードを始点とした時にも，他のノードへの到達ホップ数に大きな偏りがないことと，トラヒックが特定の経路に集中しないことが重要である．

そこで，本研究では，[97] で提案されている 3 ノード接続を基本としたネットワークを利用することを考える．3 ノード接続は，図 2.11 に示すように，全てのノード間について 1 ホップの直接経路と 2 ホップの迂回経路が存在するため，経路選択の多様性が高くなる．そのため，各ノード間の到達ホップ数の分散を小さくし，かつトラヒックの集中を避けることができる．

また，論理ネットワークの設計については，大規模ネットワーク上での運用を考慮すると，分散的な管理が可能であることが望ましい．この点でも，本方式は，基本的に新たに論理ネット

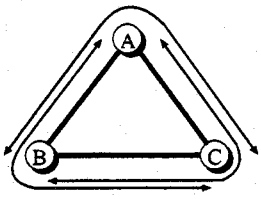


図 2.11: 3ノード接続を基本とした論理ネットワーク

ワークに加わるノードを，最も近い2つのノードに接続するだけでよいため，スケーラブルであると言える．

更に，このような構成にすることで，あるAGが故障した場合にも，他のノード間での通信が不可能になることはないため，耐故障性にも優れていると言える．

2.5 実用化に関する考察

本研究で提案するホロニックネットワークは将来のマルチメディアネットワークに要求される機能・性能を満足するアーキテクチャであるが，既存のIPネットワークとの差異が大きいため，即時の移行は難しい．しかし，以下のように既存技術を用いることによって，無理のない移行が可能になると考えられる．

まず，伝達網については，MPLSやMPλS，GMPLSのようなラベルスイッチング技術を用いることによって，ホロニックネットワークと同様の通信環境を構築可能である [71,72]．ただし，ラベルに対応する経路を動的に変更する機能が必要となる．

一方，制御網については，アクティブネットワーク（AN: Active Network）技術の適用が有効であると考えられる．ANは，パケットに埋め込まれたプログラムをルータが処理することを可能とする枠組みであり，これによってプロトコル変更などに要するコストを削減することを狙いとしている [73,74]．

現在のインターネットでは今後，GMPLSを用いた高速通信やANによる柔軟な網制御が達成されると考えられるが，IPを基本とする限り，伝達網と制御網が同一であるため，高度なサービス制御を実現することは難しい．

そこで，ANの導入されたIPネットワークをホロニックネットワークにおけるPNNに適用し，エージェントなどの制御用情報のみを流すようにし，マルチメディアトラヒックにはGMPLSを用いた専用網をOTNとして利用することが有効であると考えられる．この時，ANでGMPLSにおけるラベルと経路の対応を動的に設定できるルータが開発されれば，移行は容易である．

この場合，図2.12に示すように，ネットワーク全体は，物理的にはアクティブルータで構成されるが，論理的には，IPネットワークを基盤とした制御網 (PNN) と，GMPLSネットワークによる全光伝達網 (OTN) とに分割される．そして，アクティブパケットがパーソナルエージェントとして，制御網上で送受信され，ホロニックネットワークとしての高度かつ柔軟なサービス制御が実現されることとなる．

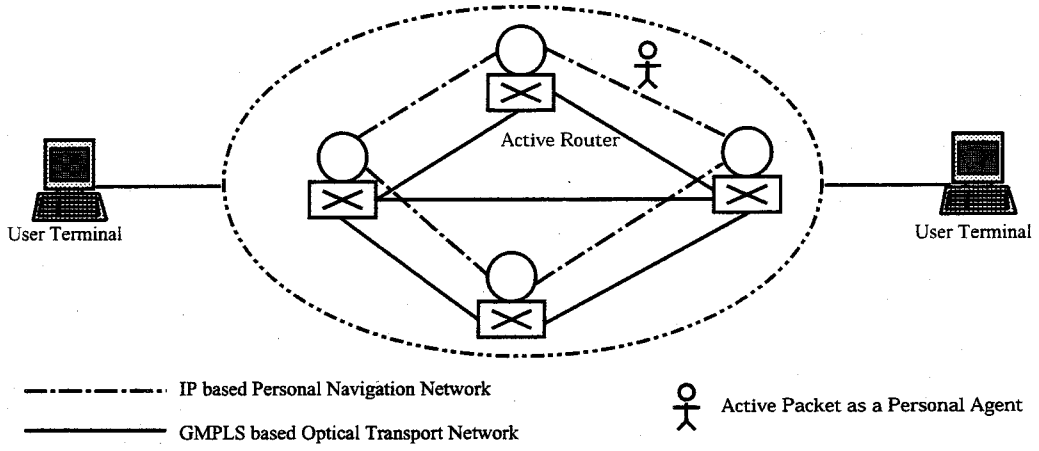


図 2.12: IP ネットワークからホロニックネットワークへの移行

2.6 結言

本章では，まず，既存のサービス制御アーキテクチャについて調査し，その問題点として，柔軟で厳密な品質保証ができないことと，大規模ネットワークにおけるスケーラビリティの欠如を指摘した．

次に，それを踏まえて，新たなネットワークアーキテクチャとして，ホロニックネットワークを提案した．これは全光伝達と自律分散制御の論理的分割を特徴とし，高度なサービスを高速かつ柔軟に提供するものである．

また，サービス制御アーキテクチャのためのプラットフォームとして，エージェント通信システムが有用であることを示し，しかし，現在のエージェントシステムではスケーラビリティの観点から，大規模ネットワークへの適用は難しいことを指摘した．

そこで，ホロニックネットワークにおけるサービス制御プラットフォームとなるエージェントシステムとして，Agent Gateway を提案した．これは，ネットワーク透過性を分散的に実現したシステムであり，大規模ネットワークにおけるエージェントの効率的な協調動作を支援するものである．

最後に，提案アーキテクチャを実用化するにあたって考慮するべき点について考察し，現在の IP ネットワークからの移行方法について述べた．

第3章

ネットワーク情報を管理する機能的データベース

3.1 緒言

マルチメディアネットワークにおいて，高度なサービスを，その品質を維持しつつ効率的に提供するためには，ネットワーク情報の管理が必要不可欠である．しかしながら，大規模ネットワークにおいて時間的に変動する情報を詳細かつリアルタイムに管理することは難しく，これまでに実用化された技術はない．

そこで本章では，ネットワーク情報を管理する機能的データベースとして，ネットワークマップを提案する [18,19]．ネットワークマップは経路情報とサービス情報を一元的に管理し，ホロニックネットワークにおける高度で柔軟な網制御及び全てのサービス提供の基盤となる．

また，全てのネットワーク情報は分散管理され，そこからエージェントが抽出した情報を相互に交換することにより，協調的に全体を管理する．これによって，ネットワークマップは高いスケーラビリティを有し，大規模ネットワークでも利用可能となる．

この時問題となるのは，抽出する情報の精度と実際のサービス提供への影響との関係及び処理量とのトレードオフである．本章では，この点についても検討し，性能と実用性の両面から最適なトレードオフポイントを明らかにする [17]．

3.2 経路情報管理方式

ホロニックネットワークでは，ネットワークマップと呼ばれるデータベースを導入してネットワーク全体の情報を管理しており，これを基にネットワークの構成及び性能を管理し，更にネットワークの可視化を図っている．

その目的から，ネットワークマップは少なくとも HN の接続構造や故障箇所，回線状況について動的に把握する必要がある．大規模なネットワークにおいて，時間的に変動する情報を管理する際に問題となるのが，その更新のためのトラヒック量と処理負荷である．ネットワークマップに関しても，この点に注意して構成する必要がある [75–77]．上記の情報の中では，HN の接続構造の変化及び故障の発生については，その頻度から考えて，発生の度にネットワークマップにその情報を送ったとしても，その量は問題とならない．しかし，回線状況に関してリ

アルタイムに全情報を管理するためには，大量の情報をほぼ連続的に送信する必要があることから，実現は難しいと考えられる．

そこで，ネットワークマップをPN毎に設置し，分散管理することを提案する．図3.1は，この点を踏まえたホロニックネットワークの構成図である．各PNはいくつかのHNで構成されるサブネットを管理しつつ，他のPNと協調して各種サービスを提供する．また，PN間は共通線信号網によって結ばれる．ここで，ネットワークマップは各PNに付帯する形で構築され，NMMAによって管理される．

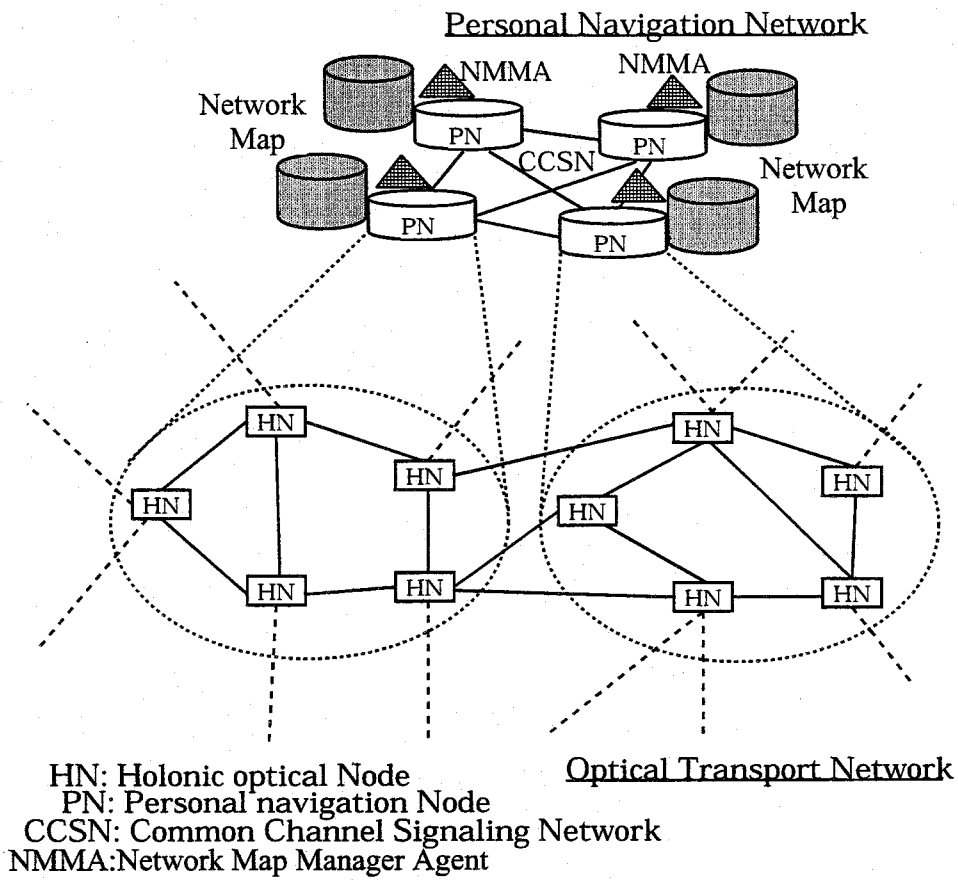


図 3.1: ホロニックネットワークの構成図

各ネットワークマップは当該サブネット内のHNに関する情報を正確に把握し，そこからサブネット全体としての回線状況情報を抽出し，他のネットワークマップとの間でNMMAによるエージェント間通信を用いて交換する．この手法はインターネットにおけるOSPF（Open Shortest Path Fast）プロトコル [78] 等にも見られるが，ホロニックネットワーク上での通信が基本的にコネクション指向であること，ホロニックネットワークのコンセプトから，より詳細な状況把握が必要とされる一方で，共通線信号網によってネットワークマップ間での情報交換が高速に行えることなどの点で異なる．

具体的には，各ネットワークマップは当該サブネット内のHNに関しては，コネクションが張られる度にそれを記録し，完全な状況把握を行う．そこから，他のサブネットに隣接するHN

についてのみ，任意の2つのHN間の経路情報を他のネットワークマップに対してブロードキャストする．また，そのHNと隣接する他のサブネットのHNとの間の経路情報についても同様に告知する．例えば図3.2において，A–B, B–C, C–A間のパスとA, B, Cの各ノードから他のサブネットに接続している各リンクに関する情報を発信する．

ここで，この交換される情報の内容とその発信頻度について検討する必要がある．内容が詳細であれば，それに比例して変化も激しくなることから，その更新頻度も高くする必要が生じ，その結果CCSNにおけるトラヒック量と，各PNでの処理量の増大を招くことになる．よって，ここで発信される情報は数ビット程度のフラグとし，他の処理に影響を与えないようにする．図3.3では，それがローマ数字で示されている．

また，将来のマルチメディアネットワークにおいては，複数のQoS指標を考慮する必要があることに注意する [43]．そのため，ここでの経路情報は全てのQoS指標に対して個別に導出する．

なお，このとき各HNの状況から経路に沿った情報を導く際の時間計算量は，HNの数をnとしたときに$O(n^2)$で実行できることが知られており，実用上の問題はない [79]．

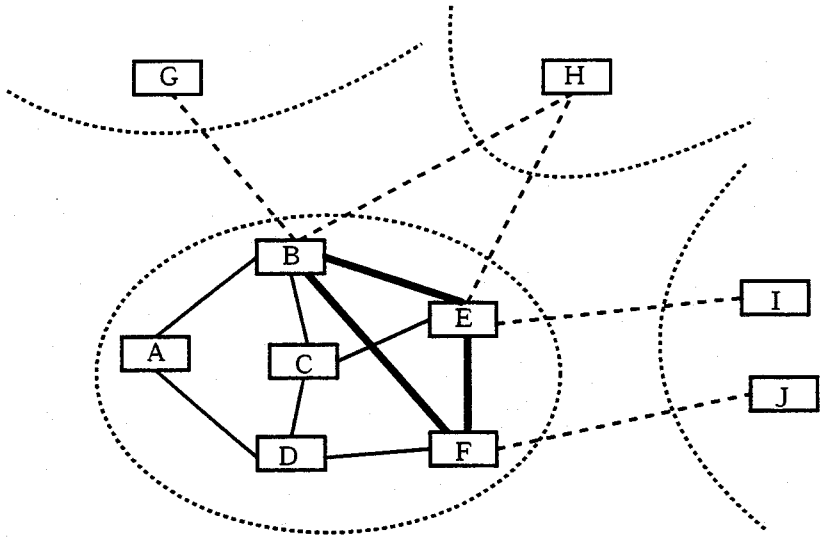


図 3.2: 交換される情報

図3.3で他のネットワークマップと交換する情報の具体例を示す．この例では，QoS指標をα, βとし，αに関する最適経路（以下，α優先経路）の情報，β優先経路の情報をそれぞれ導出している．図の括弧内の第一要素をα，第二要素をβとする．図3.3のサブネットではA, B, Cが他のサブネットにリンクしているHNである．αについては，A–B間の最適路はA→E→Bで値は(18,9)となる．同様に，A–C間はA→D→G→Cで(39,38)，B–C間はB→Cで(32,22)となる．次に，βについて考えると，A–B間はA→E→Bで値は(18,9)，A–C間はA→D→F→Cで(52,23)，B–C間はB→F→Cで(41,14)となる．

しかし，このままでは隣接する2つのHN間に複数の直接経路が存在することになる．Dijkstraのアルゴリズムを基本とした経路決定アルゴリズムは，このような場合には正常に動作しないため，ネットワークマップの情報をそのまま適用することができない [81]．そこで，ネットワー

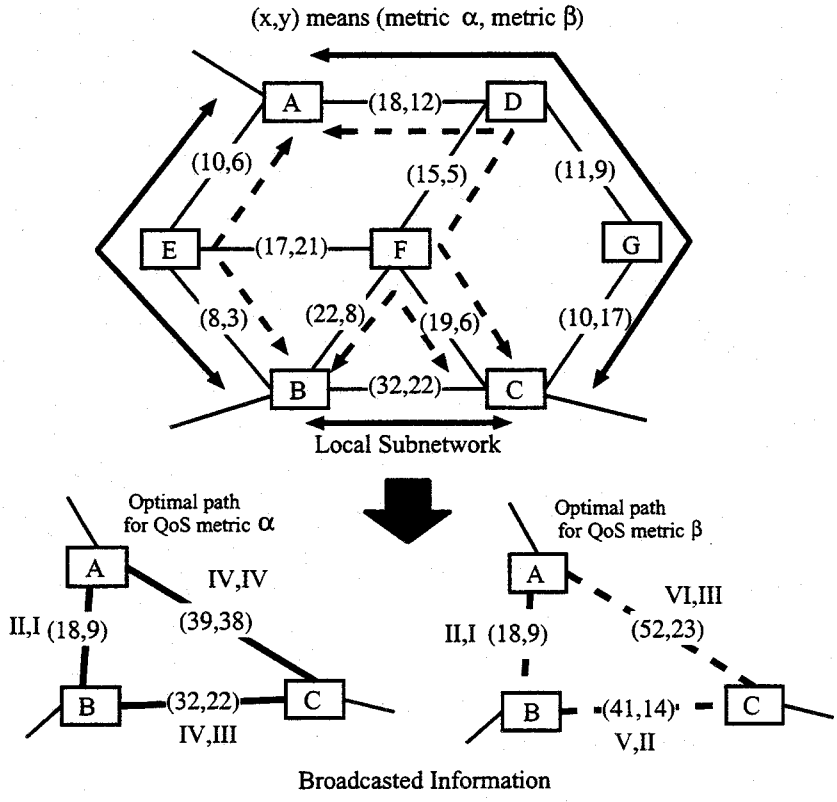


図 3.3: 他のネットワークマップに伝送する情報の内容

クマップ内部では，QoS 指標毎に HN を仮想的に生成し，それらの HN 間を全ての QoS 指標の値が 0 の回線で接続する．

図 3.4 にこの点を考慮したデータ構造の例を示す．ここでは，2 つの QoS 指標 α，β のうち，β 優先時の HN を図のように別の HN とみなし，仮想的にその HN 間を α，β とも値が 0 の枝で接続している．図 3.4 では，α 優先時の HN をそのまま A，B，C とし，β 優先時の HN を A'，B'，C' として A-A'，B-B'，C-C' 間を α，β とも値が 0 の枝で結合する．この情報にそのまま経路決定アルゴリズムを適用することにより最適路を決定できる．

3.3 コネクション管理方式

次に，前節で述べた経路情報を用いたコネクションの確立とその管理方法について説明する．なお，サービスを提供するサーバの決定や，経路選択アルゴリズムを含めた経路制御方式全体については次章で詳細に述べることとし，ここでは，ネットワークマップを用いたコネクション確立手順についてのみ言及する．

前節で述べたように，ネットワークマップはサブネット毎に分散配置され，それぞれのネットワークマップには，サブネット単位での縮退された情報と，当該サブネット内に関する詳細な情報が蓄積されている．そこで，コネクションを確立する際には，まずサブネット単位での経

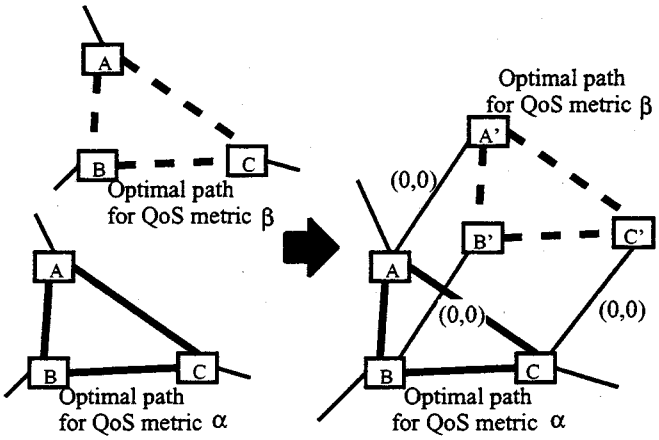


図 3.4: 経路決定アルゴリズムを適用するためのデータ構造

路を決定し，次に各サブネット内で詳細な経路を確定するという階層的な制御を行う．これは，ATM（Asynchronous Transfer Mode）ネットワークにおける PNNI（Private Network-Network Interface）[82] にも見られる方式であるが，最初に全体の経路を決定した上で，エージェント間通信を用いて各サブネットにおける経路確保を同時並列に行うことで，高速性と可用性を両立している．

例えば，図 3.5 において，ノード A–E 間を 2 つの QoS 指標 α, β に対して $\alpha \leq$ V, $\beta \leq$ VII を満たすコネクションを確立する場合を考える．

まず，複数の QoS を考慮した経路選択アルゴリズムによって，経路 A–A'–B'–B–D–E が選択される．これは，サブネット単位での経路である．次に，選択された経路上のサブネットに対して，エージェントが送られ，そのサブネット内での詳細な経路選択とコネクション確立が行われる．この図においては，サブネット Y 内でノード D–E 間の経路が決定される様子が示されている．

また，確立されたコネクションに関する情報はソースノードの所属するサブネットに設置されたネットワークマップによって管理され，中継サブネットにおいてはそのサブネットについての送受信ノードによって束ねられ，フローとして扱われる．これによって，コネクション単位のきめ細かい QoS 管理や課金を実現しつつ，管理すべき情報量を削減する [80]．

このように，階層的な経路制御を行うことにより，速度の向上と負荷分散を実現している．

3.4 サービス情報管理方式

もう 1 つの重要なネットワーク情報として，サービス情報がある．これには，サーバの位置や，そこで提供されるコンテンツの QoS 及びそのサーバの負荷状況などが挙げられる．以下，これらをプロバイダ QoS と呼ぶ．ユーザからのサービス要求が発生した際に，要求を満たすサービス提供位置を高速に検索するためには，ネットワークマップがネットワーク全体のサービス情報を把握することが重要となる．そこで，本節では，ネットワークマップを用いたサービス情報管理方式を提案する．ここで，サービス情報とは，あるコンテンツサーバが提供する

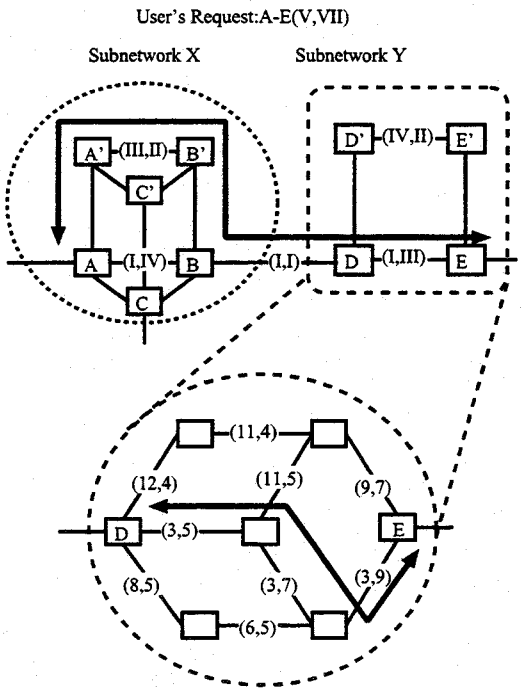


図 3.5: ネットワークマップを用いた経路制御

コンテンツのサービス内容と，付加されたプロバイダ QoS 値から構成される情報である．表 3.1 に，VoD サービスを想定した場合のサービス情報の具体例を示す．

また，広域網への対応を図るため，論理的な階層管理を行う．本方式では，階層構造として 2 分探索木を利用する．図 3.6 はこの階層構造の具体例である．

図に示すように，PN 毎に設置されたネットワークマップを最下層とし，2 つの PN 毎（例えば，ノード A，B）に親ノード（ノード E）を設置し，さらに，2 つの親ノード毎（例えば，ノード E，F）にその親ノード（ノード G）を設置する．ここで，これらの親ノードは論理的なノードであり，実際には最下層の PN がこれを兼任している．ただし，負荷集中を回避するため，各 PN は高々1 つの親ノードのみを兼任する．

このような階層構造を用いた場合，上位層のネットワークマップほど，より広範囲のサービス情報を保持することになるが，それぞれについて詳細な情報を保持することは，情報量およ

表 3.1: サービス情報の例

Service Contents	Title	*title*
	Category	*romance*
	...	...
Provider QoSs	Movie Quality	*640×400*
	Frame Rate	*30fps*
	...	...

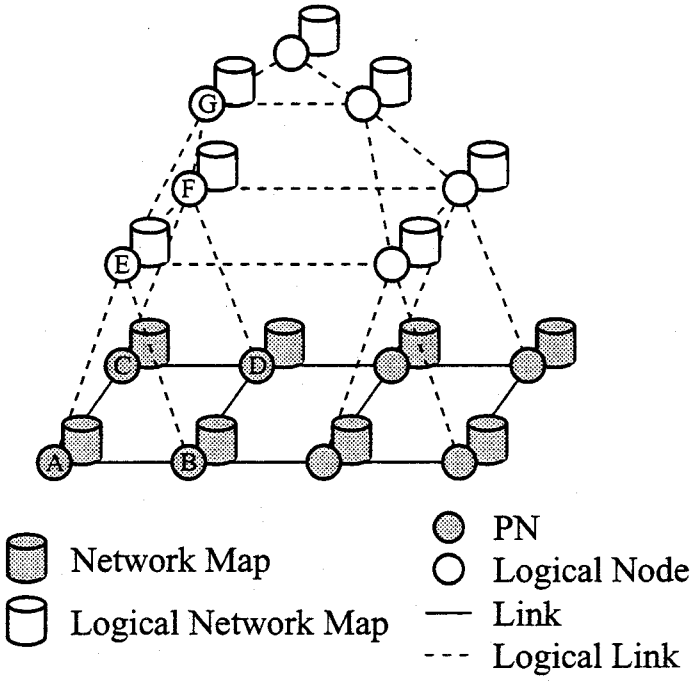


図 3.6: 論理的な階層管理

び情報更新頻度の増大の問題から，現実的ではない．そこで，本方式では，上位層のネットワークマップの保持する情報を，その子ノードに，あるサービスコンテンツを提供するサーバが存在するかどうかといった簡易なものにとどめている．

例えば，図3.6において，ノードA，Bに，同一のサービスコンテンツ α を提供するサーバがあるとする．この場合，図3.7に示すように，ノードA，Bのネットワークマップは，サービスコンテンツ α に関するプロバイダQoSを含む，全ての情報を保持している．しかしながら，ノードEのネットワークマップは，ノードA，Bのネットワークマップがコンテンツ α に関する情報を持っているという程度の情報しか保持しない．同様に，ノードGのネットワークマップも，ノードEのネットワークマップがコンテンツ α に関する情報を持っているという程度の情報しか保持しない．これによって，上位層のネットワークマップの保持する情報量を削減すると共に，情報の更新頻度を下げることが可能となる．

3.5 性能評価

本研究では，ネットワークマップの有効性と性能特性を評価するため，要求される記憶容量を解析的に導出し，また，更新頻度と性能の関係を明らかにする計算機シミュレーションを行った．本章ではこれらの結果について述べる．

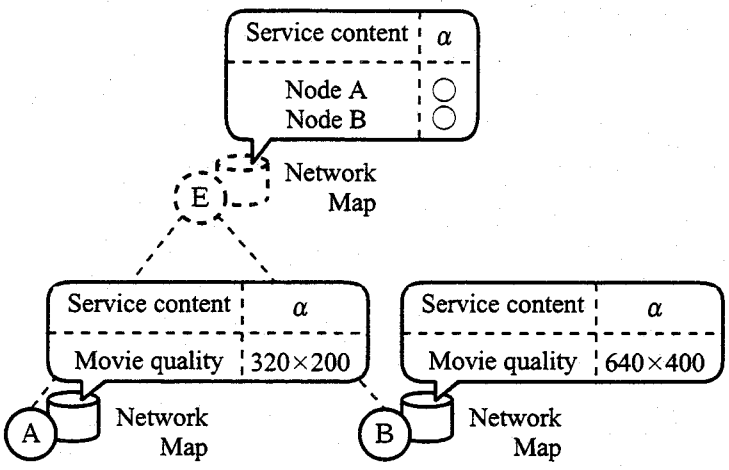


図 3.7: サービス情報

3.5.1 記憶容量

まず，記憶容量について考える．まず，当該サブネット内の情報の記憶容量 L は，当該サブネット内の HN 数を h，回線状況の詳細な情報の大きさを d，QoS 指標数を q とすると

$$L = h^2 \times d \times q$$

で表される．また，他のサブネットの情報の記憶容量を M とすると，M は PN 数を p，他のサブネットにリンクしている HN 数を l，他のサブネットに伝送する簡略化された情報の大きさを s とすると

$$M = (p \times l)^2 \times s \times q$$

である．

今，国内の各市町村区に PN を一つ割り当てると仮定し，PN の総数を約 1000 個とする．また，各 PN のサブネット内に 1000 の HN が存在するものとする．取り扱う QoS 指標数は 5 とし，それぞれ，当該サブネット内の回線状況の情報に 10byte，PN 間で伝送される情報に 1byte を要するものとする．他のサブネットにリンクしている HN 数を平均 10 とすると，L，M とも約 5Mbyte となり，一つのネットワークマップが管理する情報量は約 10Mbyte となるため，十分実現可能である．

3.5.2 更新頻度と性能の関係

先に述べたように，ネットワークマップは当該サブネットに関する経路情報を他のネットワークマップに定期的に送信し，それを基にネットワーク全体の情報を管理している．そのため，その更新頻度が性能に大きく影響する．

更新頻度が高ければ，より精度の高い情報を蓄積できるが，そのための処理負荷及びトラヒック量が増加する．一方，更新頻度を下げると，情報が最新の状況を反映しなくなり，経路制御の失敗を引き起こす．

そこで本研究では，この更新頻度をパラメタとして，ネットワークマップの性能特性を評価した．まず，1024 ノードで構成されるサブネットを 16 接続したネットワークを用意した．これらのノードは格子状に接続され，各リンクの容量は 6[Gbps] とした．呼は，各サブネットで平均して 1 日に 2.0×10^4 回のポアソン過程に従って発生し，6[Mbps] の帯域を 1 時間要求するものとした．発生した呼に対して，ネットワークマップ上の情報を基に，各リンクの空き帯域の逆数を目的関数として Dijkstra のアルゴリズムを実行し，経路を決定した．ここで，ネットワークマップ上の情報は，その時点でのネットワーク状況を正確に反映しているとは限らないため，選択された経路が実際には利用できない可能性がある．そこで，発生した呼に対して，このような事象が起こる確率を評価指標とした．以下，これを経路確保失敗率と呼ぶ．

以上の条件で，$1.0 \times 10^6 (sec)$ のシミュレーションを行った結果を図 3.8 に示す．図 3.8 は，横軸にネットワークマップの更新間隔の底を 10 とした対数，縦軸に経路確保失敗率をとったグラフである．

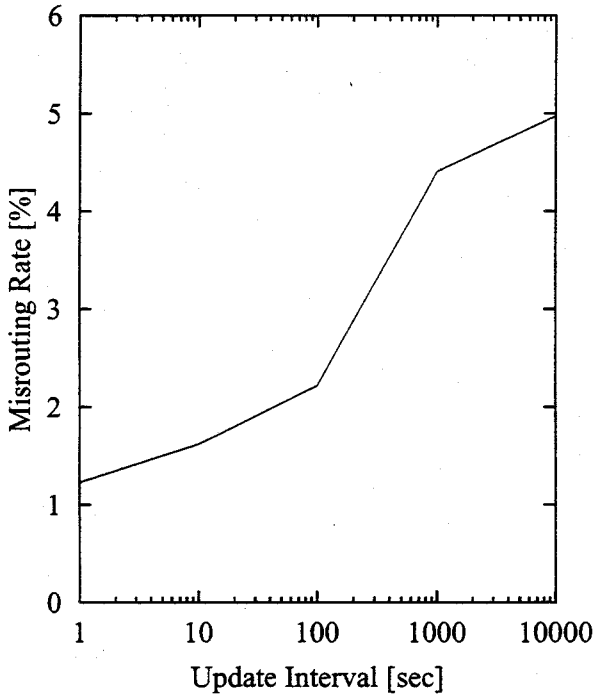


図 3.8: シミュレーション結果

この図から明らかなように，ネットワークマップの更新頻度を 10^2 秒毎程度にすることで十分に実用的な性能を得ることができる．

3.6 結言

本章では，ネットワーク情報を管理する機能的データベースとして，ネットワークマップを提案した．ネットワークマップは経路情報とサービス情報を一元的に管理し，高度で柔軟な網

制御の基盤となるデータベースである.

本章では，このネットワークマップを各サブネットに分散配置し，当該ネットワーク内に関する詳細な情報を分散管理しつつ，そこからエージェントが抽出した情報を相互に交換することにより，協調的に全体を管理する手法を提案した．これによって，ネットワークマップは高いスケーラビリティを有し，大規模ネットワークでも利用可能となる.

また，この時問題となる，抽出する情報の精度と実際のサービス提供への影響とのトレードオフについて検討し，理論解析とシミュレーション実験の結果から，最適なトレードオフポイントを明らかにした.

第4章
サービス品質を考慮した呼制御方式

4.1 緒言

本章では，前章までに提案したアーキテクチャ上での呼制御方式について述べる．マルチメディアネットワークにおける呼制御に関しては，サービス品質を考慮したサーバ検索と経路選択が重要であるが，これまで，これらの要素は独立に扱われてきた．そのため，選択されたサーバとクライアント間のトラヒック状況によっては，ユーザの享受するサービスの品質が著しく劣化する場合がある．

また，経路選択については，これまで多数の研究事例があるが，選択される経路の最適性と実行速度の点で実用的な方式は存在しない．

そこで本研究では，経路情報を利用したマルチメディア通信のためのサービス情報検索方式と，複数のQoS要求を満たすしつつ最適性の高い経路を高速に生成するアルゴリズムを提案する [27,28,31,32].

4.2 サーバ検索方式

4.2.1 既存のサーバ検索方式

現在，マルチメディア通信サービスに関する研究は発展途上であり，そのため，これらのサービスをユーザの要求に沿った形で提供するためのサーバ検索に関する研究はなされていないのが現状であるが，以下のような既存の検索方式を適用可能である．

全探索に基づくサーバ検索方式 全探索に基づく情報検索は最も単純な方式であり，WWWにおける巡回ロボットによる検索等に用いられている．この方式では，ネットワーク中の全ノードを訪ねて検索を行うため，全てのサービス情報の中から最適な情報を検索するために，必ず全てのノードから情報を収集する必要がある場合等に有効である．

階層管理によるサーバ検索方式 階層管理による情報検索は，データを何らかのまとまりでグループ化することにより，膨大な情報から特定の情報を効率的に検索することを可能とする．例えば，ネットワークを幾つかのサブネットワークに分割し階層管理することで，広域ネットワークにおける効率的な情報検索が可能となる．

しかし，このようなサーバ検索方式をマルチメディア通信のためのサーバ検索に適用した場合，以下のような問題が生じる．

まず，全探索に基づく情報検索を適用した場合，前述したように全ノードから情報を収集する必要がある場合には有効な手段であると言える．しかし，全ノードを訪ねるための処理時間が膨大なものとなり，広域ネットワークにおいて高速な応答時間が要求される場合には適していない．

また，階層管理による情報検索を適用した場合，全探索に基づく情報検索で問題となる広域ネットワークにおける情報検索の効率性の問題は解決される．しかし，いくつかのサブネットワークから構成されるネットワークを論理的に階層管理した場合，検索に必要なノード間の移動は，各ノードの物理的なノード位置を考慮していない移動となる．本研究で対象とするマルチメディア通信のためのサーバ検索においては，要求するサービス情報の有無だけでなく，サービス提供ノードと要求が生じたノードとの間にユーザのQoS要求を満たす経路が存在することも重要となる．これらの経路状況に依存するQoSは，物理的なノード位置の影響を受けやすいため，階層管理によるサーバ検索も効率的なものであるとは言えない．

4.2.2 経路情報を利用したサーバ検索方式

前節で述べたように，サービス提供ノードとユーザの要求が生じたノードとの間にQoSを保証した経路が存在することが重要となるマルチメディア通信のためのサーバ検索において，従来の情報検索方式を適用することは効率的であるとは言えない．

そこで本節では，木構造を利用した階層管理を行うことにより，広域網に対応可能なサービス情報管理方式を提案し，それにより管理されるサービス情報と，ネットワークマップの経路情報を利用することにより，QoS保証を考慮したサーバ検索方式を提案する．具体的には，まず，物理的なノード位置を考慮することにより，ユーザの要求するQoSを満たす可能性の高いノードの検索を優先するサーバ検索方式を提案する．更に，経路情報を利用することにより，ユーザの要求を満たしにくいと判断されるノードへの移動を遅らせ，検索効率の向上を図るサーバ検索方式を提案する．この方式は，特に，ユーザのQoS要求が厳しく，要求を満たすサービスを提供可能なノードがネットワーク中に全く存在しない場合に，不必要な検索を大幅に削減する効果がある．

【木構造を利用した階層管理によるサービス情報管理方式】

前章で述べたように，将来のマルチメディア通信サービスの実現のためには，高速かつ柔軟なサービス提供を行うことが重要となる．そのため，ネットワークマップがネットワーク全体の情報を把握しておくことが必要となる．しかし，大規模なネットワークにおいてネットワーク全体の情報を集中管理することは，データベース更新頻度増大の要因となり，それに伴うトラヒック量及び処理負荷の増加が問題となる．

そこで本研究では，PNに設置されたネットワークマップの保持するサービス情報，つまりサブネットワーク単位のサービス情報を最下層とした木構造を利用した論理的な階層管理を行うことにより，データベース更新頻度を軽減し，広域網への適応を図る．図4.1は2分木を用い

た場合の木構造の構成図である．この図に示すように，いくつかのPNに対して1つの親ノードを設置し，更にいくつかの親ノードに対してその親ノードを設置する．また，PN間は物理的なリンクで，階層間及び上位階層内は論理的なリンクで接続する．ここで，論理的なリンクは，情報検索の際の階層間及び階層内の移動に用いられる．ただし，前章で述べたように，PNは幾つかのHNで構成されるサブネットワークの情報を管理している．

なお，親ノード及び親ノードに設置されたネットワークマップは論理的なものであり，実際にはPN及びPNに設置されたネットワークマップがこれを兼任する．ただし，負荷分散のため，各PNは単一の親ノードを兼任するものとする．

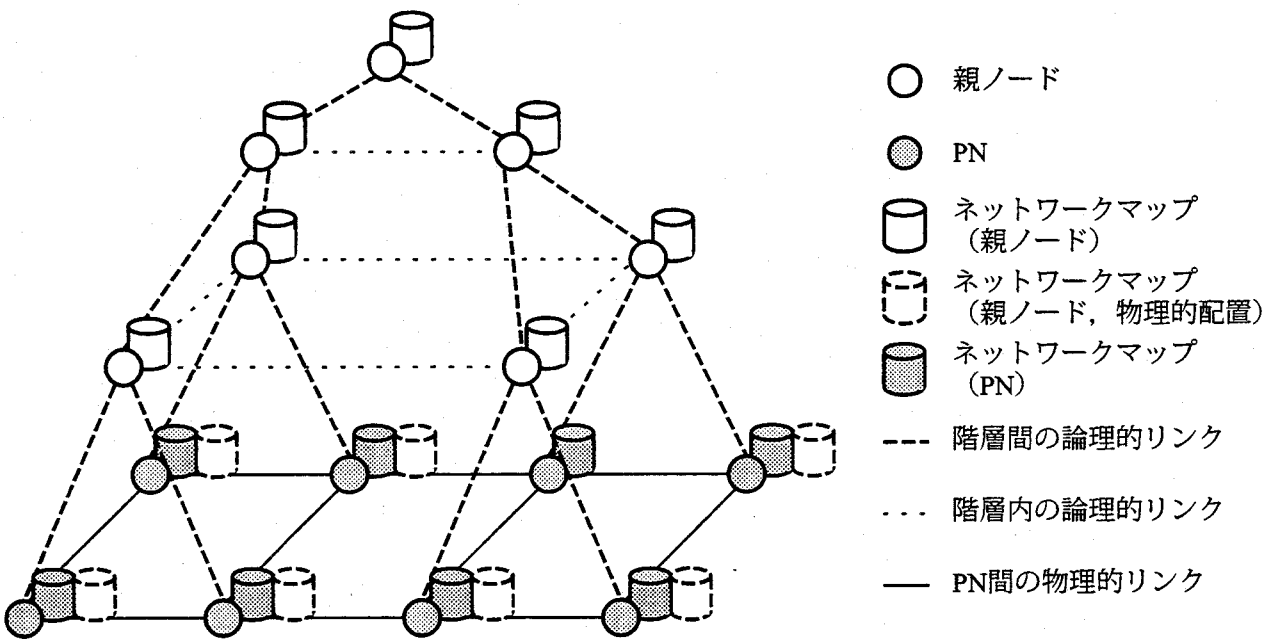


図 4.1: 木構造を利用した論理的な階層管理

最下層に設置されたネットワークマップは，当該サブネットワーク内の各サービスに対するサービス提供位置情報及びサービス提供に最低限必要となるQoS条件を管理する．これは，サービスコンテンツ毎に異なる詳細な情報である．サービス情報の検索の際には，ユーザの依頼を受けたエージェントが，これらの情報と経路情報を用いて，ユーザの要求するサービスを提供可能かどうかを判断する．つまり，ユーザの要求が生じたノードとユーザの要求するサービスを提供するノードの間に，ユーザの要求及びサービス提供に最低限必要となるQoS条件を満たす経路が存在するかどうかを調べ，経路が存在すればサービス提供が可能であると判断する．

しかし，階層化の上位層に設置されたネットワークマップにおいても詳細な情報を保持させると，下位層に設置されたネットワークマップの保持する情報の変化が上位層へ伝播するため，上位層でのデータベース更新頻度の増加が問題となる．そこで，上位層に設置されたネットワークマップでは，子ノードにおける情報の有無を保持させる程度にとどめる．つまり，ユーザからの要求に対して，どの子ノードへ行けば目的とするサービスが存在するのかをナビゲートする程度とする．上位層に設置されたネットワークマップの保持する情報をこの程度にとどめることにより，子ノードに設置されたネットワークマップの保持する複数の同一コンテンツに対

する情報を1つの情報として扱うことが可能となり，サービス情報の追加，削除，修正に伴う上位層でのデータベース更新頻度を軽減できる．今後のマルチメディア通信サービス網では，一般に，同一コンテンツを提供するノードが複数存在することが予想されるため，この方式は有効であると考えられる．

図4.2を用いて簡単な例を示す．まず，2つのPNに対して親ノードを設置し，各ノードに設置されたネットワークマップにノード右下に併記したサービス情報を保持させる．ここで，PNに設置されたネットワークマップは複数のHNに関する情報を保持している．PNの情報は,「コンテンツ」が,「提供位置」でサービス提供されており，提供に最低限必要となるQoS条件が「帯域幅，遅延，…」であることを示している．また，親ノードの情報は,「コンテンツ」がノード1，ノード2で提供されているかどうかを示している．

このような状況で，ノード1に新たにコンテンツ1の情報が追加される場合，現在ノード1には既にコンテンツ1が存在しているため，親ノードにおける情報は更新されない．また，ノード1からコンテンツ1の情報が削除される場合も，削除した後でもノード1にはコンテンツ1が存在するため，親ノードにおける情報は更新されない．更に，情報の修正はPNのみの問題であるため，同様に親ノードにおける情報は更新されない．このように，親ノードでの情報を簡易化することにより，データベース更新頻度を軽減できる．より上位の層においても同様の処理を繰り返すと，上位層になるほど同一のコンテンツが存在割合が高くなるため，データベース更新頻度は大幅に軽減される．

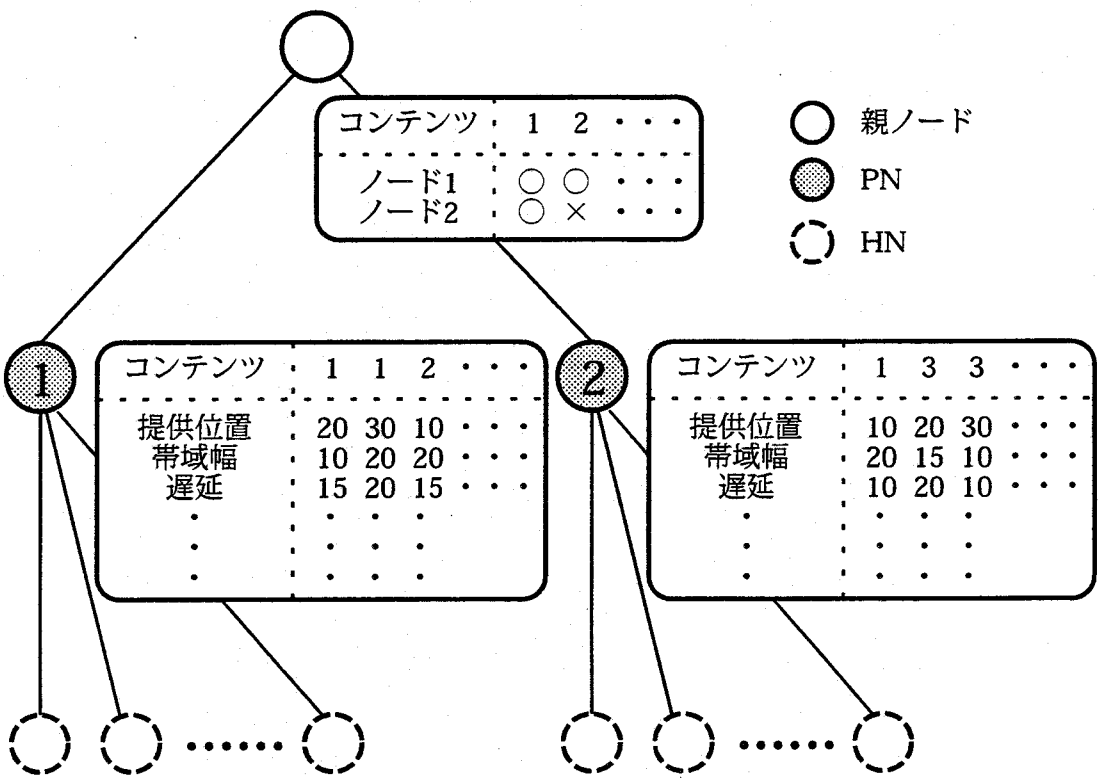


図 4.2: ノードの保持するサービス情報

【物理的なノード位置を考慮したサーバ検索方式】

ユーザがマルチメディア通信サービスを快適に享受するためには，ネットワークが複数のQoSを保証することが重要となる．すなわち，サービス情報の検索においても，あるサービスがネットワーク中に存在するかどうかという点のみでなく，要求の生じたノードとサービスを提供するノードの間にQoS要求を満たす経路が存在するかどうかという点も問題となる．

ここで，ネットワークが保証すべき複数のQoSの中で経路状況に依存するQoSを，以下，ネットワークQoSと呼ぶ．具体的なネットワークQoSとしては，帯域幅や遅延等が挙げられる．これらのネットワークQoSについては，ユーザの要求が生じたノードとユーザの要求するサービスを提供するノードのネットワーク距離が近いほどその要求が満たされる可能性が高いことが予想される．しかし，前項で述べたような木構造に沿った単純な情報検索では，物理的に隣接しているノードであっても，論理的に数ホップ離れたノードとなってしまう可能性がある．つまり，比較的QoS要求を満たす可能性が高いことが予想される隣接ノードの情報検索が遅れることになる．これは，同一コンテンツを提供する複数のノードからユーザの要求を満たすサービス提供が可能なノードを検索する場合には効率的であるとは言えない．

例えば，隣接ノードの情報検索の前に検索対象となるノードにおいて，ユーザのQoS要求を満たす経路は存在しないがユーザの要求するコンテンツを提供しているノードが複数あった場合，これらのノードに対する不必要な移動及び情報検索が増大し，検索速度が劣化することになる．

そこで本研究では，1つのノードに対して仮想的に複数の親ノードを設置することにより，物理的なノード位置を考慮した情報検索を行うことを提案する．具体的には，1つのノードに対して，単純な木構造での親ノードの他に，そのノードに対する物理的な隣接ノードの親ノードも仮想的な親ノードとして扱う．これにより，木構造に沿った広域網における効率的な情報検索とQoS保証を考慮した情報検索を同時に行うことが可能となる．

図4.3（a）は，総数16（4×4）のPNを格子状に接続したネットワークに対する仮想的な階層構造を示しており，図4.3（b）はこれを木構造を用いて表現したものである．これらの図を用いて例を示す．ここで，図中の番号は各PNのノード番号である．なお，図4.3（a）において親ノードに設置されたネットワークマップはその領域内の子ノードに設置されたネットワークマップの保持するサービス情報を保持しているものとする．

PN6で要求が生じたとすると，まず，このノードにおける情報検索が行われる．この検索において，ユーザの要求を満たすサービスが発見できなければ，PN5，6の親ノード（網掛け部の情報を保持）へ移動し，情報検索を行う．単純に階層管理に利用される木構造に沿った情報検索の場合，このノードでの検索においても要求を満たすサービスを発見できなければ，さらにPN1，2，5，6の親ノードへの移動を行う．以後，この繰り返しとなる．しかし，この場合，PN6の右及び下のリンクに接続されている隣接ノード，つまりPN7及びPN10への移動が遅れる．特にPN10への移動は非常に遅くなり，PN10への移動と比べて，比較的QoS要求を満たす可能性が低いことが予想される遠方のノード，例えばPN4の親ノードへの移動が先に行われることになる．この時，仮にPN4にユーザの要求するコンテンツが存在しているとすると，PN4への移動が行われる．ここで，PN4と要求の生じたPN6との間に，QoS要求を満たす経路が存在しなければ，PN10の前に検索対象となったPN4への移動は不必要なものとなる．

そこで，隣接ノードの親ノード（斜線部の情報を保持）を仮想的な親ノードとして扱うことにより，比較的QoS要求を満たす可能性が高いことが予想されるノードを優先的に検索する．つまり，親ノードでの情報検索において，要求を満たすサービスの発見ができなければ，仮想的な親ノードへ移動し，情報検索を行う．以後，同様に，親ノード及び仮想的な親ノードへの移動及び情報検索を繰り返す．この場合の移動例を図4.3（b）に矢印で示した．ここで，実線は単純な木構造を利用した検索の場合の移動，波線は提案方式により付加される移動を示している．ただし，実際には，PN1，2，5，6の情報を保持する親ノードへは，既に全ての子ノードに関する情報を調べているため，移動を行わない．これは，エージェントの移動記録と木構造の接続構造から，各ノードで容易に判断できる．

【経路情報の利用による検索効率の向上】

前項で述べたサーバ検索では，要求ノードと仮想的な親ノードの間にQoSを満たす経路が存在しない場合，仮想的な親ノードへの移動により無駄な検索が多くなるため，検索終了までに要する時間，つまり応答速度が劣化する．そこで，QoS保証の必要性を考慮し，仮想的な親ノードへの移動に伴い検索範囲に追加されるPN群の中で実際にそれまでの検索範囲と隣接する部分のPNとユーザの要求が生じたノードの間にQoSを満たす経路が存在するかどうかを判断し，QoSを満たす経路が存在しない仮想的な親ノードへは，この段階では移動させないことにより，検索効率の向上を図ることを提案する．

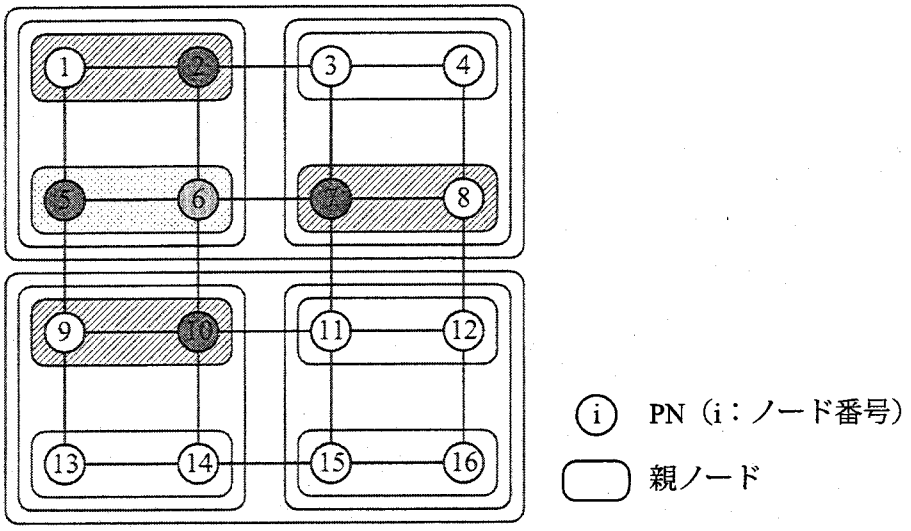
また，同様の処理を親ノードへの移動にも適用することが可能である．これらの処理は，ユーザの要求が厳しく，ネットワーク中のいずれのノードからもユーザの要求を満たすサービス提供が不可能であることの早期検出を可能とする．

具体的には，あるノードとその隣接ノードが物理的に隣接している部分，つまり最下層で隣接しているノードに対して，要求の生じたノードとの間にQoSを満たす経路が存在するかどうかを調べる．この操作により，経路が存在しなければその隣接ノードに対応する仮想的な親ノードへの移動は，この段階では行わない．一方，最下層で隣接しているノードに対して1つでも経路が存在すれば，移動及び検索を行う．親ノードへの移動も，親ノードを隣接ノードに対応する仮想的な親ノードとみなすことにより，同様の処理が可能となる．また，どの隣接ノードにもQoSを満たす経路が存在しなくなれば，それ以上の検索は無駄であることが分かるため，検索を終了する．これらの手続きによって，情報検索の効率を向上させることができる．

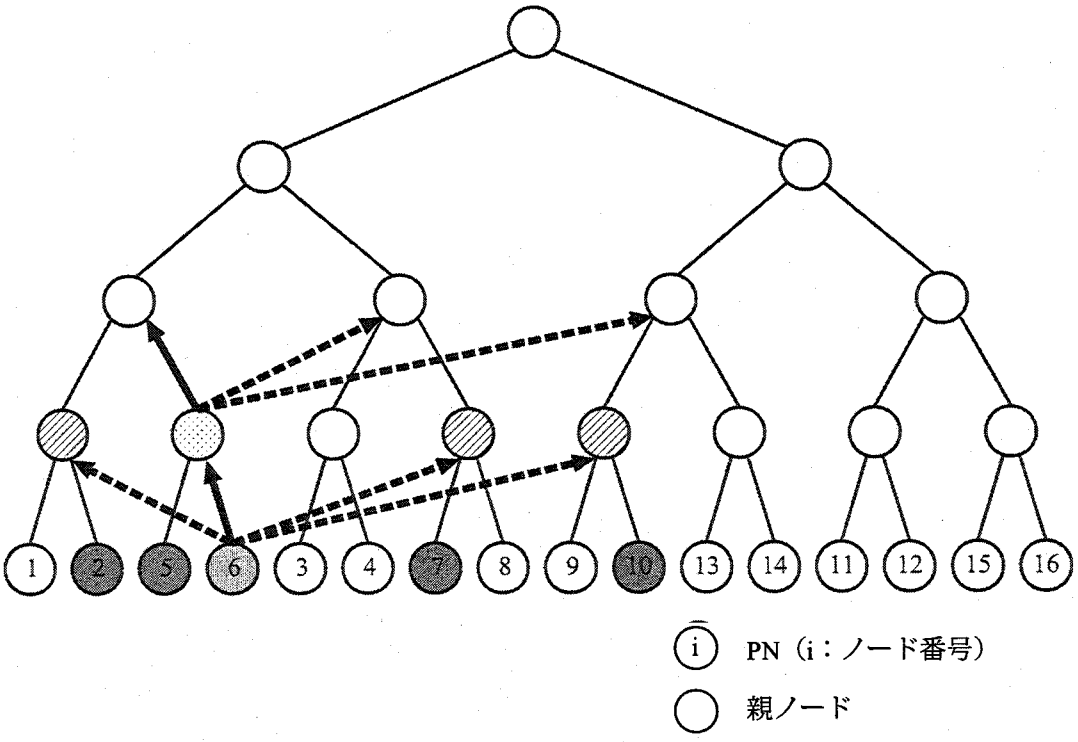
なお，親ノードへの移動はさらに上位層の検索を行う際に必要となるので，仮想的な親ノードへの移動が行われる場合には，親ノードへの移動も行う．ここで，隣接ノードとの間にQoSを満たす経路が存在するかどうかを調べる際に，それにかかる処理時間が問題となる．しかし，本研究で用いる経路選択アルゴリズムは1対多の経路選択に応用でき，その処理時間もノード間の移動及びデータベース検索時間と比べ非常に短時間であるため，これに要する時間は無視できるものと考えられる．

図4.4を用いて例を示す．なお，図4.4（a）の括弧内の数字は，各リンク間のネットワークQoSの値である．ここでは，単純化のためQoS指標数は1とし，検索途中でQoSの値は変わらないものとする．

PN6において要求として「QoS値5以下でのサービス提供」が生じたとすると，まず，この



(a) 格子状ネットワーク



(b) 木構造

図 4.3: 仮想的な親ノードの設置

ノードにおける情報検索が行われる．この検索において，ユーザの要求を満たすサービスが発見できなければ，PN5–6 間には QoS 要求を満たす経路（PN6–5）が存在するので，親ノードへ移動し，情報検索を行う．次に，隣接ノードの中で PN7，10 には QoS 要求を満たす経路（PN6–7，PN6–10）が存在するので，仮想的な親ノードへ移動し，情報検索を行う．ここで，PN2 には QoS 要求を満たす経路が存在しないので移動は行わない．

仮想的な親ノードの検索においても，ユーザの要求を満たすサービスが発見できなければ，更に上位の親ノードへ移動する．まず，PN1，2，5，6 の親ノードへ移動しようとするが，新たに追加される部分の中で PN5，6 の親ノードに隣接する部分，すなわち PN1，2 と PN6 の間に QoS 要求を満たす経路が存在しないので移動は行わない．

次に，隣接ノードの中で PN7，8 の親ノードと PN9，10 の親ノードにおいて PN5，6 の親ノードに隣接する部分，すなわち PN7 と PN9，10 と PN6 の間には QoS 要求を満たす経路（PN6–7，PN6–5–9，PN6–10）が存在するので，仮想的な親ノードへ移動し，情報検索を行う．ここで，仮想的な親ノードへの移動が行われたため，先程行わなかった親ノードへの移動も行う．

以後，同様の操作を繰り返す．図 4.4（b）に，これらの移動を木構造上に表したものを太線で示す．

4.3 複数の QoS を満たす経路選択アルゴリズム

4.3.1 既存のアルゴリズム

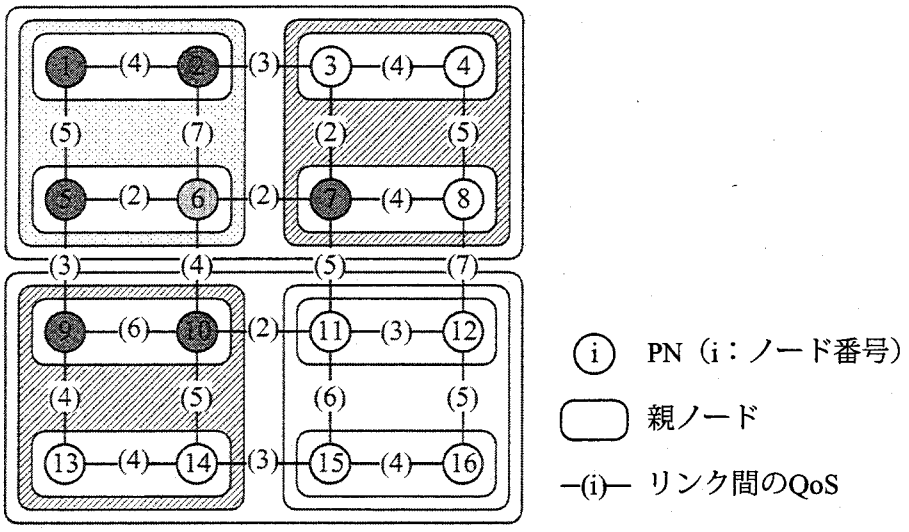
マルチメディアネットワークにおいて利用者の満足する通信サービス品質を提供するためには，帯域や遅延など複数の QoS を考慮した経路選択が必要不可欠である．

利用者が要求する QoS は，以下のように分類できる [21]．特定の経路の QoS に対する指標が，(i) その経路を構成する全リンクの QoS の値の和となる加法性 QoS(コスト，遅延など)，(ii) その経路を構成する全リンクの QoS の値の積となる乗法性 QoS(誤り率など)，ならびに (iii) その経路を構成する全リンクの QoS の値の中の最小値をとる最小性 QoS (帯域など) である．

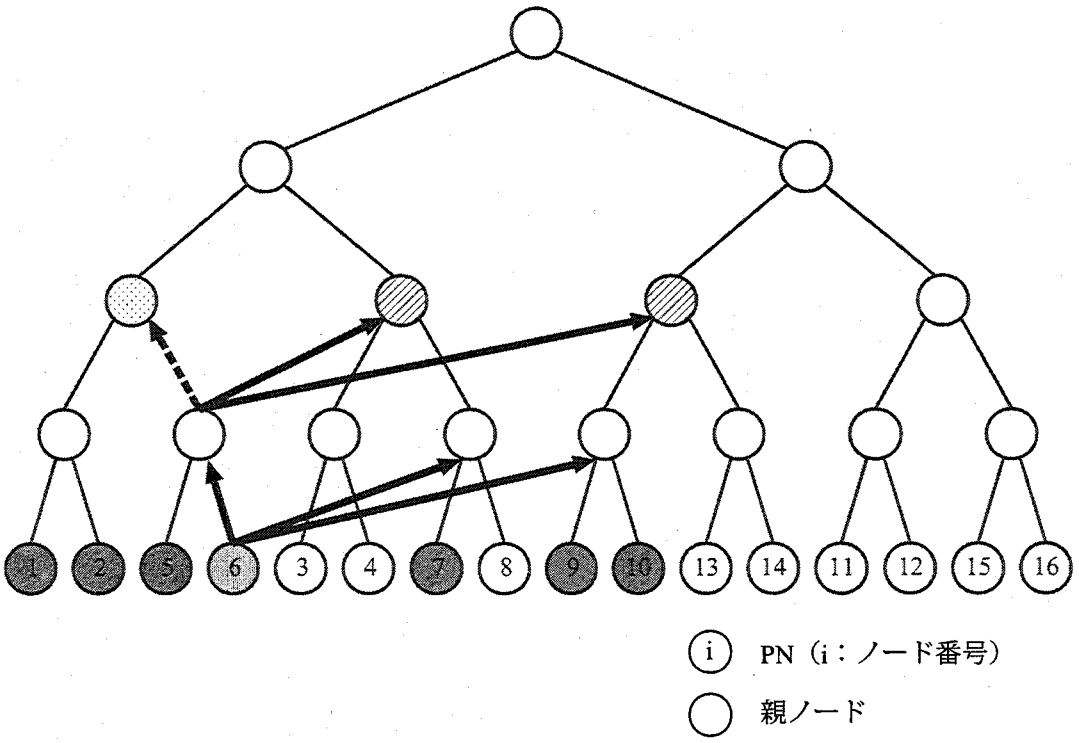
一般に，利用者はこれら性質の異なる QoS を同時に要求するものと考えられが，最小性 QoS に関しては，経路選択を行う前にあらかじめ QoS 制約を満たさないリンクを経路選択の対象外とすればよく，また，乗法性 QoS に関しては，各リンクの QoS の値を対数値に置き変えることで，加法性 QoS に変換できる．よって，この問題は，複数の加法性 QoS 制約を満たす経路を発見する数理計画問題として定式化できるが [20,21]，NP 完全問題であることが知られている [22]．そのため，これまでに多くの発見的アルゴリズムが提案されてきた [20–26]．これらは，いかに利用者が要求する複数の QoS を満たす経路を実時間で選択するかを目的としている．

中でも FallBack（FB）アルゴリズムは，複数の QoS 制約を満たす経路を生成する最も基本的なアルゴリズムである [24]．FB アルゴリズムでは，制約条件のない最短路問題を実時間で解くことができる Dijkstra のアルゴリズムを用いて，1 つの QoS 指標に関して最適な経路を生成し，その経路が全ての QoS 制約を満足するかどうかを調べ，満足しない場合には他の QoS 指標を目的関数として選択し，同様の操作を行うという手続きを逐次的に繰り返すものである．

FB の最大計算量は，R を QoS 制約数，V をネットワーク中の全ノード数としたとき，$O(RV^2)$ であるが，$R \ll V$ であるため Dijkstra のアルゴリズムの最大計算量 $O(V^2)$ と同等であると考



(a) 格子状ネットワーク



(b) 木構造

図 4.4: 提案方式を用いたサーバ検索例

えられることから，実用的な計算量と言える．

本研究では，従来の研究で検討されてきた，単にユーザが要求する複数の QoS を満たす経路を選択するという点だけでなく，ユーザが要求する複数の QoS を満たしつつ，ネットワーク資源を有効に利用する経路を選択する問題を考える．これは，利用者が要求する複数の QoS 制約に加えて，ネットワーク管理者の視点からみた指標（コスト）を導入し，複数の QoS 制約をもつコスト最小の経路を発見する数理計画問題として定式化されるが，明らかに NP 完全である．そこで本研究では，複数の QoS 制約を満たす経路を選択する FB アルゴリズムを拡張した Fallback+アルゴリズムを提案する．

4.3.2 Fallback+アルゴリズム

【アルゴリズムの概要】

通常の Dijkstra のアルゴリズムでは，コスト最小経路を図 4.5–(A) に示すように，段階的により目的関数の値が小さくなる経路を探索し，経路の書き換えを行う．この操作を繰り返し実行し，最終的にコスト最小経路を決定する．つまり，アルゴリズムを実行する過程において，送信ノードから受信ノードへ至る経路が暫定解として生成され，より良い経路が生成される度に廃棄されている．

一方，本研究で提案する Fallback+アルゴリズムでは，図 4.5–(B) に示すように，Dijkstra のアルゴリズムを適用して最小経路を生成する過程において生成される暫定経路を廃棄せず，一時的にスタックに保管する．これにより，選択の候補となる対象経路の増加を図る．

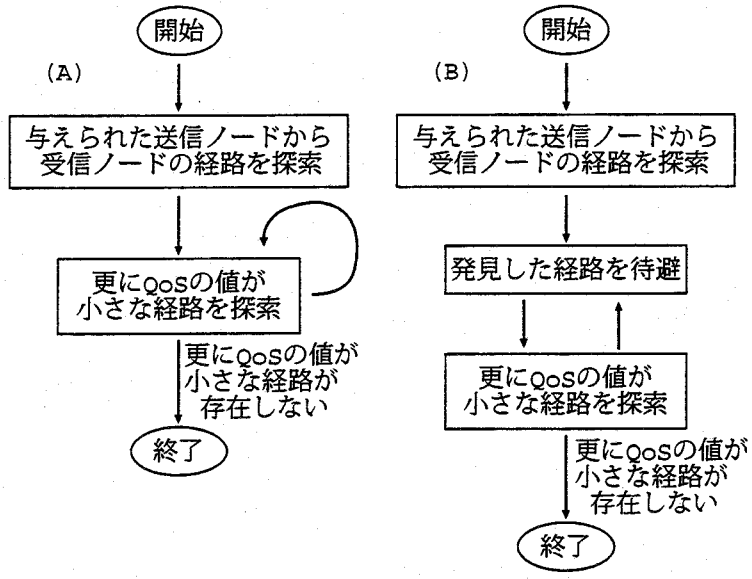


図 4.5: 経路生成アルゴリズムの比較

【Fallback+の手続き】

図 4.6 に Fallback+の手続きをフローチャートに示す.

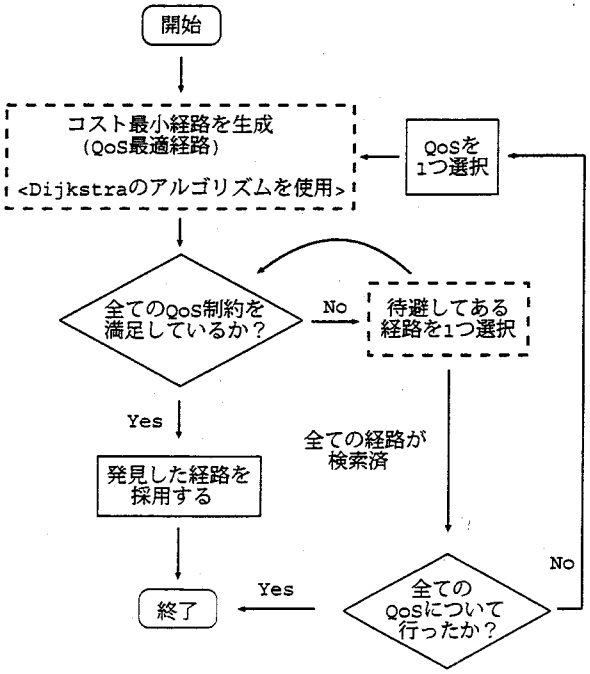


図 4.6: Fallback+のフローチャート

まず，全ての QoS 制約を無視し，前項で記したように拡張された Dijkstra のアルゴリズムを適用してコスト最小経路を生成する．この時，コスト最小経路を選択する過程において生成される暫定経路を，別候補として経路の生成順にスタックに保管しておく．

最終的に Dijkstra のアルゴリズムによって選択されるコスト最小経路が全ての QoS 制約を満たしていれば，目的の経路として採用する．コスト最小経路が QoS 制約を満たさない場合は，スタックに保管していた経路を 1 つ取りだし，全ての QoS 制約を満たすかどうかを調べる．その結果，QoS 制約を満たしていれば目的の経路として採用し，そうでない場合は，保管していた次の経路について同様の処理を行う．

ここで，コスト最小経路を選択する過程で得られた暫定経路は，スタックに保管してあるため，自動的にコストについて昇順にソートされていることに注意する．よって，全ての QoS 制約を満たす経路が見つかった場合は，その経路が生成した QoS 制約を満たす候補の中でコスト最小の経路となる．

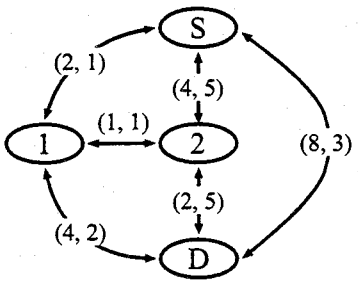
コスト最小経路を選択する過程で得られた暫定経路の中に全ての QoS 制約を満たす経路が存在しない場合は，通常の FB アルゴリズムと同様に，1 つの QoS 指標を目的関数として選択し，選択した以外の QoS 制約を無視して，選択した QoS 指標に対する最適経路を探索する．しかし，Fallback+アルゴリズムにおいては，コスト最適経路を選択する場合と同様に，各 QoS 指標に対する最小経路を選択する過程で生成される暫定経路を保管しておき，選択した最適経路が QoS 制約を満たさない場合は，保管していた経路の中で QoS 制約を満たす経路があるか探

索し，QoS 制約を満たす経路があればその経路を採用する．この操作を行うことで，目的の経路が得られる確率が高くなる．

もし，いずれの候補経路も QoS 制約を満たさなければ，以降同様にして，別の QoS 指標を 1 つずつ選択し，全ての QoS 制約を満たす経路を探索する．以降に具体例を用いて説明する．

【Fallback+の適用例】

次に，Fallback+を実際に適用した結果を示す．図 4.7 のグラフについて伝送遅延が 4 以下の制約を満たす送信ノード (S) から 受信ノード (D) までの経路の選択を行う．



	コスト	伝送遅延
S - 1 間	2	1
S - 2 間	4	5
S - D 間	8	3
1 - 2 間	1	1
1 - D 間	4	2
2 - D 間	2	5

図 4.7: 経路選択問題の例

〈 **初期経路生成** 〉送信ノード (S) から 1 ホップで到達可能なノードに至る経路が生成される．それにより，受信ノード (D) に至る 1 ホップの経路 (S→D) が生成される．またこの経路を候補として保管する (図 4.8)．

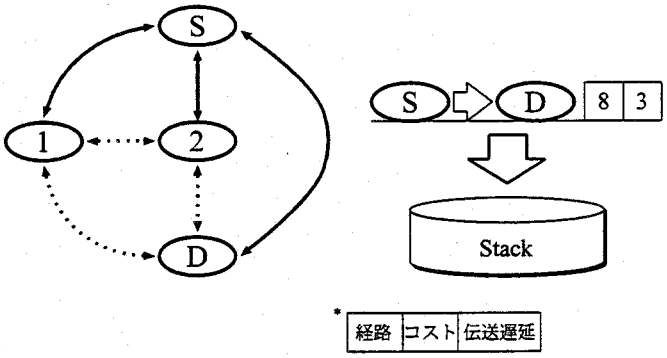


図 4.8: 初期経路生成

〈 **低コスト経路に更新** 〉送信ノード (S) からの経路が生成されているものの中で，最小コストの経路に注目し，この経路の終端ノードから 1 ホップで到達可能で，この経路を経由することで既に生成された経路よりも低コストの経路が生成可能なノードが存在する場合，そのノードへの経路が更新される．また，全てのノードは経路更新に一度だけ使用可能であることに注意する．

受信ノード (D) は，既に生成された経路の中の最小コストの経路 (S→1) の終端ノード (1) から1ホップで到達可能で，この経路を経由することで既に生成されている経路 (S→D) よりも低コストな経路 (S → 1→ D) が生成可能であるので，経路が更新される．また，更新される前の経路は，目的経路の候補としてスタックに保管する (図4.9)．この時点で，ノード1は使用済みとなる．

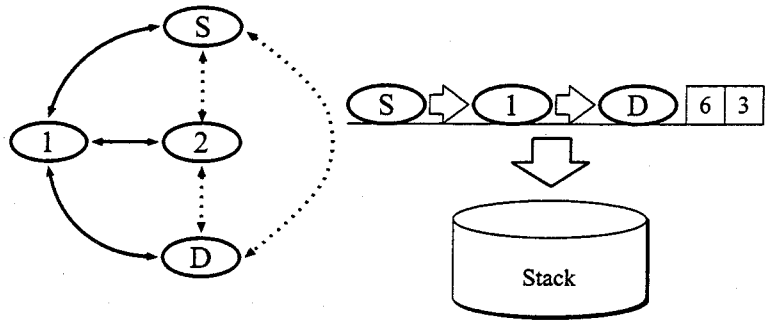


図 4.9: 低コスト経路生成

同様に，受信ノード (D) に注目すると，既に生成された経路の中で終端ノードが経路生成に未使用なものの中で最小な経路 (S→1→D) の終端ノード (2) を経由することで既に生成されている経路 (S → 1→ D) より更に低コストな経路 (S → 1 → 2 → D) を生成可能である．よって経路は更新され，更新前の経路は候補として保管される (図4.10)．

以上で，より低コストな経路を生成不可能となる．結果，3つの候補経路が見つかったことになる．

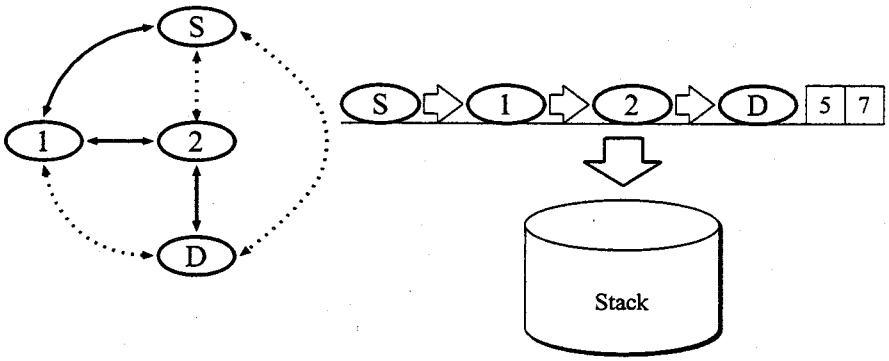


図 4.10: 経路生成終了

〈**経路検討過程**〉まず，コスト最小経路 (S→1→2→D) を経路をスタックから取り出し，QoS制約を満たしているかどうかを調べる．その結果，その経路の伝送遅延が7であり，制約条件である4を満たさないため，経路として採用しない (図4.11)．

次に，経路を保管しているスタックから次の経路 (S→1→D) を取り出し，QoS制約を満たすかどうかを調べる．この経路は伝送遅延が3であり，制約を満たしているため，目的経路として採用する (図4.12)．

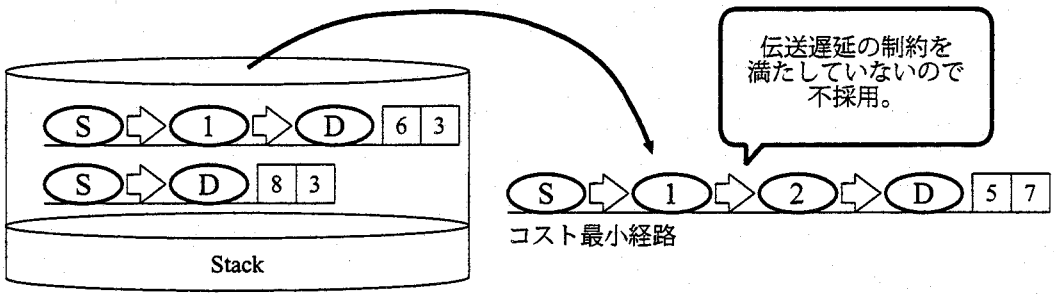


図 4.11: 経路検討中

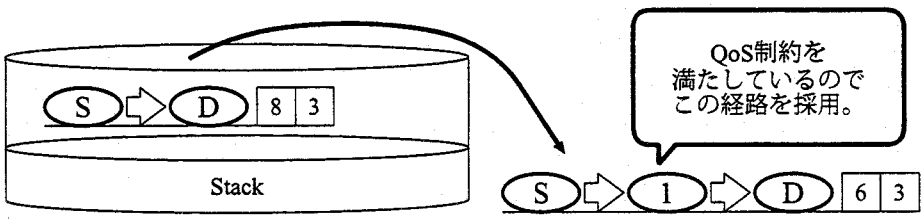


図 4.12: 経路検討終了

4.4 性能評価

本節では，提案したサーバ検索方式と経路選択アルゴリズムの性能を評価する．

4.4.1 検索速度の評価

本節では，提案したサーバ検索方式の性能をC++言語によるシミュレーションを用いて評価する．まず，性能評価のための評価条件と評価方法について述べ，その後，評価結果と得られた知見に対する考察を示す．

【評価条件と方法】

まず，総数1024（32×32）のノードを格子状に接続したネットワークを用意し，各ノード間のリンクにネットワークQoS値（伝送遅延）を付加した．単純化のため，付加するネットワークQoS指標は1つとし，その値は1~10の一様分布に従うランダムな整数値とした．

また，構築したネットワークの各ノードに，8種類のサービスコンテンツをランダムに配置した．ただし，サービスは$32i$ ($i = 1, 2, 3, \cdots, 8$) 個所のノードに配置されており，各サービスに対して，それぞれ個別のプロバイダQoS値（映像品質）を付加するものとした．単純化のため，付加するプロバイダQoS指標は1つとし，その値は1~10の一様分布に従うランダムな整数値とした．

さらに，サービス情報の階層管理に利用する木構造には2分探索木を利用し，単純化のため，ネットワークマップの保持する情報は変化しないものとした．

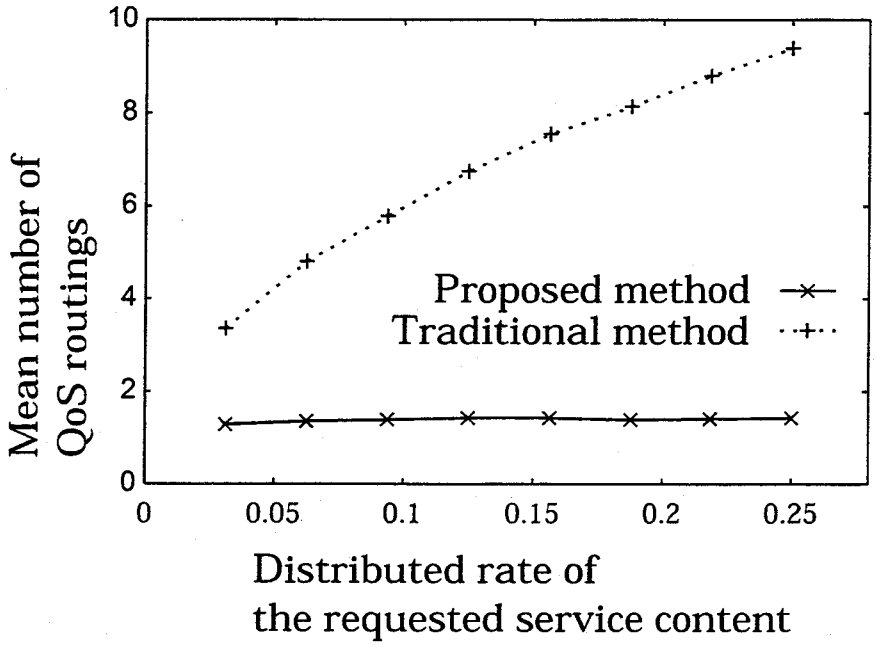


図 4.13: シミュレーション結果（QoS ルーチング実行回数）

このような条件下で，ユーザは 1~10 の一様分布に従うランダムな整数を用いて，1 つのネットワーク QoS 値と 1 つのプロバイダ QoS 値を指定し，サービスを要求する．

まず，本方式が，従来のプロバイダ QoS を満たすコンテンツサーバを探索した後にネットワーク QoS を満たす経路を生成する方式よりも高速にサービス提供可能であることを示すため，検索終了までの QoS ルーチング実行回数を評価指標とした比較を行った．なお，シミュレーションの試行回数は 50,000 である．

シミュレーション結果を図 4.13 に示す．実線は提案方式，点線は従来方式の結果を示しており，横軸はサービスコンテンツ分布率，つまり，全ノードに対して要求したサービスコンテンツを提供するプロバイダを持つノードが占める割合を示している．

これらの結果から，従来方式と比較して，経路情報を利用する本方式の有効性が確認できる．

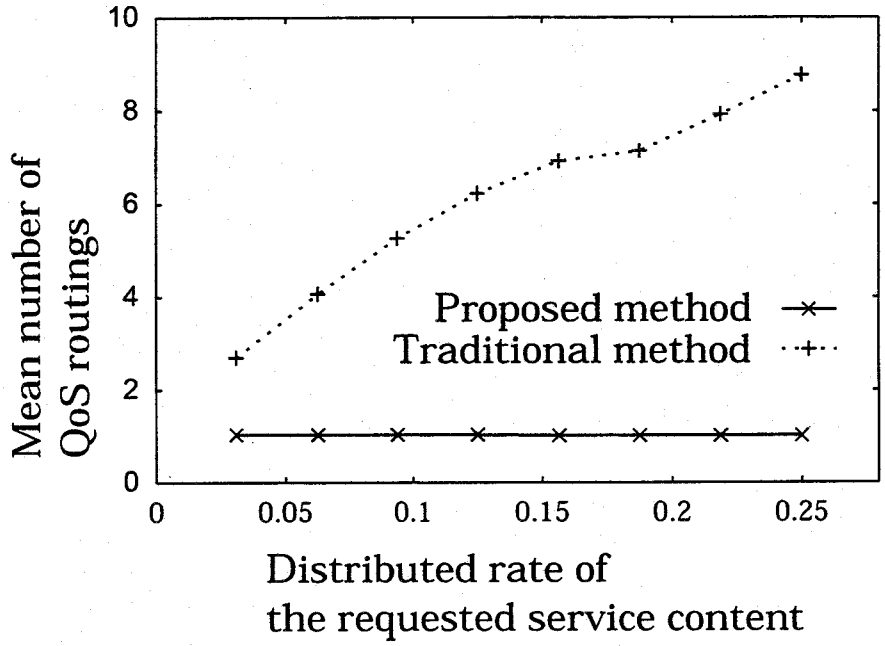
更に，この結果を詳細に分析するために，得られた結果を要求したコンテンツを提供するサーバが存在した場合とそうでない場合に分類した．なお，今回のシミュレーションでは，該当するサーバが存在した確率は 9 割程度であった．

図 4.14 は，要求したコンテンツを提供するサーバが存在した場合，図 4.15 は，要求したコンテンツを提供するサーバが存在しなかった場合の結果を示している．

これらの結果から，要求するコンテンツを提供するサーバの有無に関わらず，本方式を用いることによって，不必要な経路選択の回数が削減されていることが分かる．

次に，検索終了までに訪れたネットワークマップの数を評価指標とし，単純に探索木を辿る方式（方式 1）と，本方式で隣接ノードの親ノードに移動する代わりに，単に隣接ノードに移動する方式（方式 2）との比較を行った．なお，シミュレーションの試行回数は 50,000 である．

シミュレーション結果を図 4.16 に示す．実線は提案方式，点線は方式 1，波線は方式 2 によ



(a) The requested service content
with requested QoSs exists

図 4.14: シミュレーション結果の分析（QoS ルーチング実行回数）

る結果を示しており，横軸はサービスコンテンツ分布率を示している．

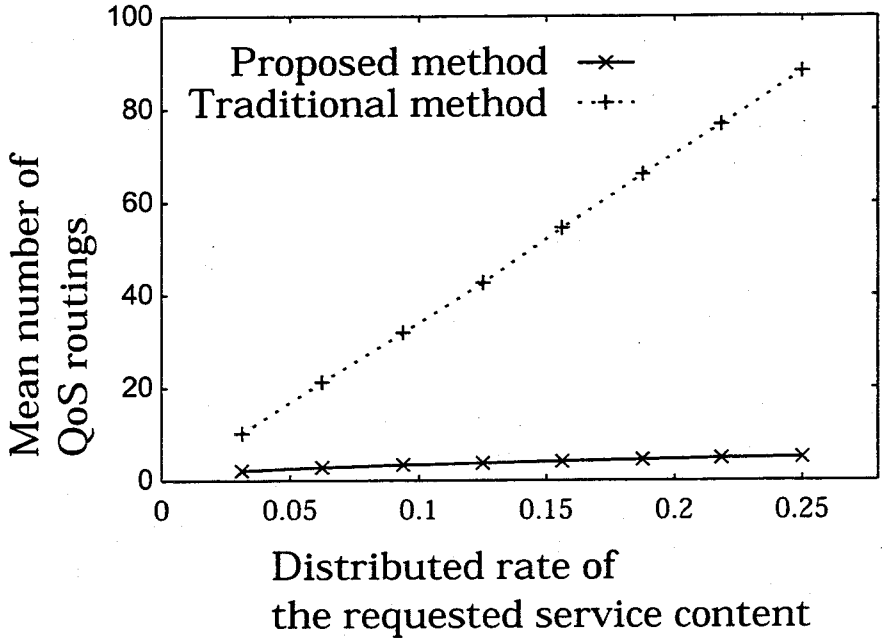
これらの結果から，方式1，方式2と比較して，本方式の検索速度が優れていることが分かる．

更に，この結果を先程と同様に2つに分類した．

図4.17は，要求したコンテンツを提供するサーバが存在した場合，図4.18は，要求したコンテンツを提供するサーバが存在しなかった場合の結果を示している．

これらの結果から，特に，要求したコンテンツを提供するサーバがネットワーク中に存在しない場合，方式1と比較して，提案方式の性能が優れていることが分かる．これは，検索範囲限定の処理の影響により，不必要な検索が削減されていることを示している．

また，要求したコンテンツを提供するサーバが存在する場合も，方式2と比較して，提案方式の性能が優れていることが分かる．これは，あるノードでの情報検索の結果として求める情報が得られなかった場合に，隣接ノードの親ノードへ移動する処理によって，効率的な検索が行われていることを示している．



(b) The requested service content with requested QoSs does not exist

図 4.15: シミュレーション結果の分析（QoS ルーチング実行回数）

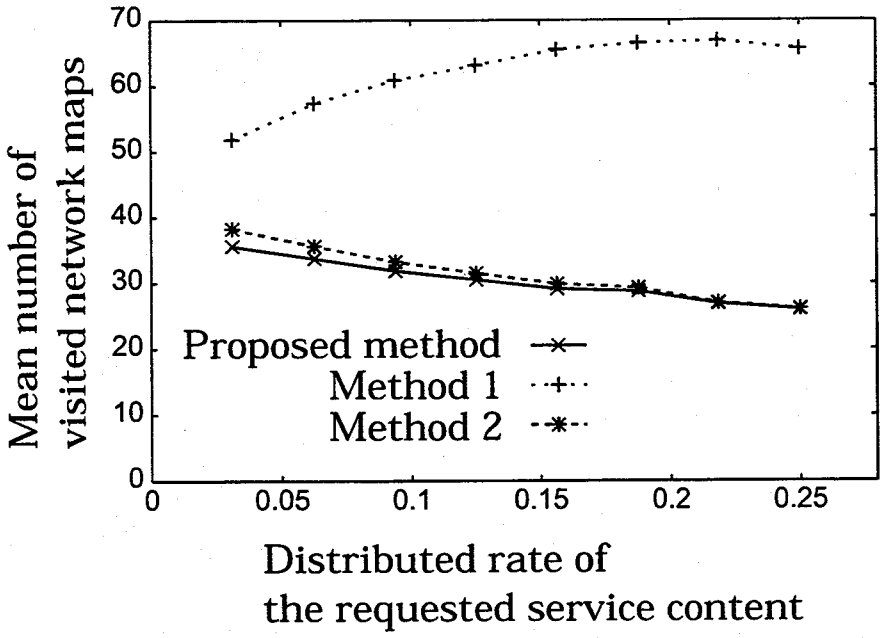
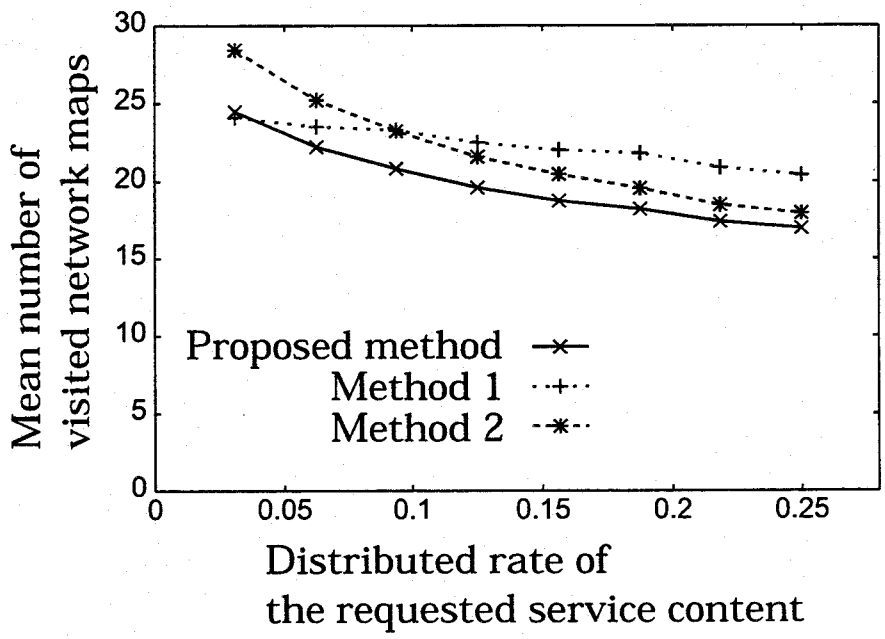


図 4.16: シミュレーション結果（ネットワークマップ訪問回数）



(a) The requested service content with requested QoSs exists

図 4.17: シミュレーション結果の分析（ネットワークマップ訪問回数）

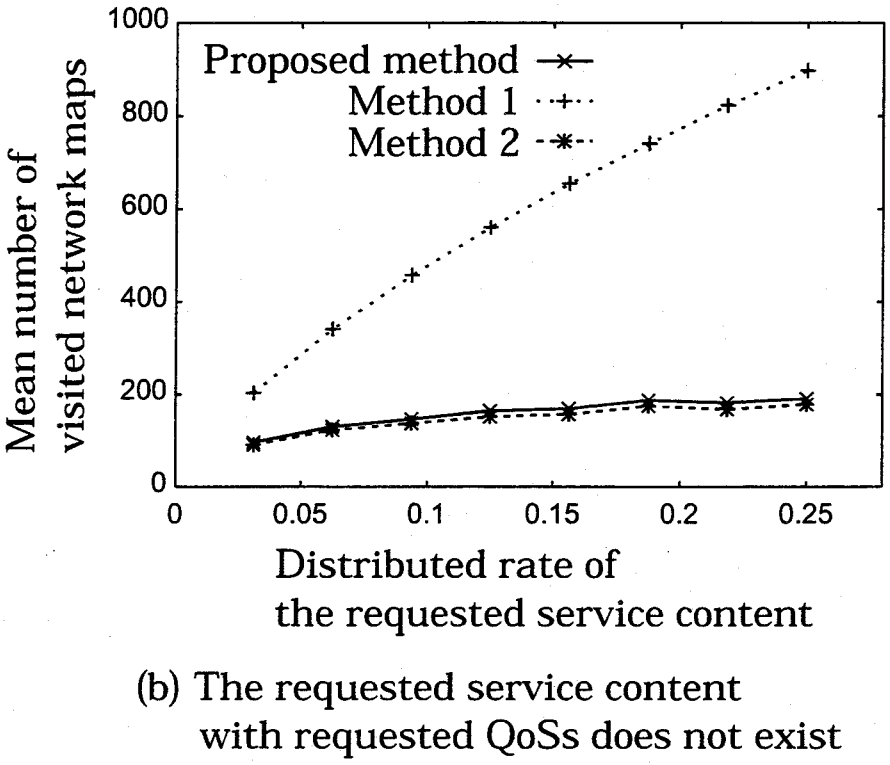
4.4.2 経路選択アルゴリズムの評価

本節では，提案した Fallback+アルゴリズムの性能を，計算量と目的経路選択率及び選択された経路の最適性から評価した結果を示す．

【最大時間計算量】

従来の FB アルゴリズムでは，ネットワーク全体のノード数を V とした時経路生成に Dijkstra のアルゴリズムを適用するため，各 QoS について $O(V^2)$ の計算量を要する．よって，全ての QoS について最適経路を生成する場合，QoS の個数を R とした時，Dijkstra のアルゴリズムを高々 R 回適用するので $O(RV^2)$ の計算量となる．

一方，Fallback+アルゴリズムでは更に，ある 1 つの QoS について Dijkstra のアルゴリズムにより生成した最適経路が 1 つでも他の制約を満たさない場合，別の QoS について Dijkstra のアルゴリズムを用いて最適経路を生成する前に，スタックにある最適経路の生成過程の途中において生成された暫定経路の中で，全ての QoS 制約を満たす経路が存在するかどうかを順次探索する手続きが存在する．全てのノード間にリンクが存在した場合，最大 $V-1$ の暫定経路が生成されるため，この手続きには $O(V^2)$ の計算量を要する．よって，各 QoS 指標について最適経路の生成後，毎回この操作を行うと最大で $O(RV^2)$ の計算量となり，結局 $O(RV^2)+O(RV^2)=O(RV^2)$



(b) The requested service content with requested QoSs does not exist

図 4.18: シミュレーション結果の分析（ネットワークマップ訪問回数）

表 4.1: 最大計算量

Algorithm	Time Complexity	Space Complexity
FB	$O(RV \log V)$	$O(RV^2)$
FB+	$O(RV^2)$	$O(RV^2)$

となる．

【最大領域計算量】

通常のFBでは，Dijkstraのアルゴリズムで使用するグラフのリンク全てについてコスト，およびQoS値の表を持つ必要があるため $O(RV^2)$ の領域が使用される．Fallback+では，更に最適経路生成する過程において生成される暫定経路（最大 V 個）を全て保管しておくため $O(V)$ の領域が必要となるが，領域計算量は $O(RV^2)$ となる．

【経路選択性効率と実行速度】

次に，提案したアルゴリズムに対する定量的な性能評価のため，以下に示す条件，方法での計算機シミュレーションを行った．

シミュレーションに使用するネットワークモデルとして，ランダムグラフ生成法 [83] を用い，1つのノードに平均4本のリンクを持ちノード数が100のグラフを10個生成した．また生成したグラフの各リンクには各々4つのQoS（コストを含む）指標を付加した．この時，4つの指標は1～10のランダムな整数値とした．

経路選択の要求である，送信ノード受信ノード及び3つのQoS制約については，通常のDijkstraのアルゴリズムをコストについて適用した場合には選択不可能で，かつ，必ず1つ以上の経路が存在するものをそれぞれのグラフにつき1000通り用意した．このとき，QoS制約は1以上50未満の一様分布に従う整数値とした．

以上の条件のシミュレーションをC言語により実装し，経路選択成功率，最適経路選択率，処理時間（CPU Time）をSun Ultra Sparc10上で計測した．ここで最適経路とは，QoS制約を満足する経路の中でコストが最小の経路を指すものとする．ただし，コストが同一の経路が存在する場合はリンク数が少ないものとする．最適経路は，事前に全探索によって全ての経路情報を得ることで導出した．

全ての試行から得られた経路選択率，処理時間の平均値と95%信頼区間を計算した結果を表4.2に示す．この結果から，経路選択成功率，最適経路選択率が共に従来のFBアルゴリズムよりも向上していることが分かる．更に，3.1.1節で示したように，定性的には処理量が増加しているにも関わらず，Fallback+はその平均処理速度も向上していることが確認できる．

これらの性能向上の要因を明らかにするため，選択された経路が，各QoS指標（コストを含む）に関する最適路である割合と，最適路を生成する過程で暫定経路としてスタックに保存されていた経路である割合とを算出した．その結果を図4.19に示す．この図において(A)はFBによる結果，(B)はFB+による結果を示している．ここで，このシミュレーションは，コストを指標とした最適経路が目的の経路とならないという条件で行ったことに注意する．

この図より，FB+では，選択された経路のうち24%の経路が従来のFBでは選択できないコストに関する最適路を生成する際に保存していた暫定経路であり，更に33%の経路が暫定経路として保存された経路であったことが分かる．

つまり，Fallback+によって最初に処理される，コストに関する最適路を生成する過程でスタックに保存された経路に目的とする経路が多く含まれているため，従来のFBと比べて処理速度が向上することが確認できる．

表 4.2: シミュレーション結果

Algorithm	FallBack	FallBack+
Success Rate(%)	92.84 ± 0.91	95.95 ± 0.66
Optimal Rate(%)	55.58 ± 3.07	61.78 ± 2.25
CPU Time (ms)	0.344 ± 0.026	0.297 ± 0.026

これらのことから，Fallback+は，暫定路を保存しそれらを目的経路の対象として制約を満たすかどうかの検討を行うことで，経路の選択率を上昇させることができることが確認できた．また，付加した操作によって従来のFBよりも多くの処理時間を必要とするように見えるが，実際には，各指標毎での最適経路が他のQoS制約を満たさないために，別の指標についてDijkstraのアルゴリズムを再び適用して経路を生成するよりも，暫定経路に対する検討を行うことで目的の経路を発見した場合の処理時間短縮効果が大きく，平均的にはより高速に目的の経路を選択可能であることが分かった．

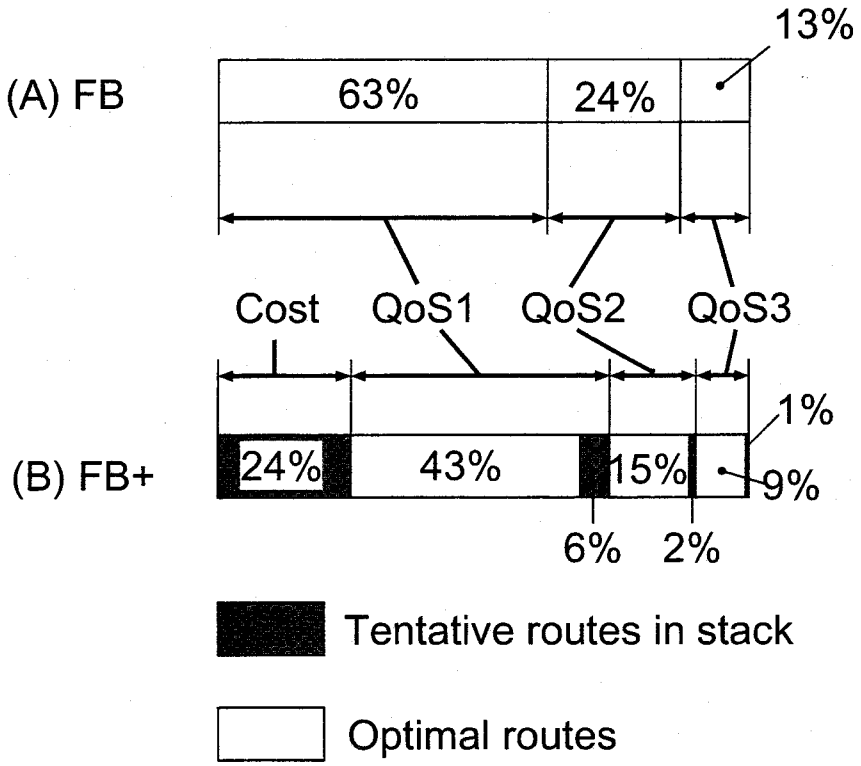


図 4.19: 経路分析結果

4.5 結言

本章では，前章までに提案したアーキテクチャ上での呼制御方式について述べた．

まず，マルチメディアサービス提供時において，サーバで提供されるQoSとそのサーバからクライアントまでの通信経路におけるQoSを同時に考慮することで，効率よく適切なサーバを検索する方式を提案した．

提案方式は，前章で提案したネットワークマップにおける木構造に沿った検索に，経路情報を基にした優先検索を加えた結果，従来に方式に比べて平均して3倍以上高速な検索を実現している．

次に，複数のQoS要求を考慮した経路選択を行うため，この問題を複数制約下の最短路問題

として定式化し，従来のFBアルゴリズムで経路選択に用いるDijkstraのアルゴリズムを拡張させたFallback+を提案した．

シミュレーションによる性能評価の結果から，提案したサーバ検索方式は経路選択成功率，最適経路選択率，処理時間の全ての性能において，Fallback+が従来のFBアルゴリズムよりも優れていることを確認し，その有効性を示した．

第5章
分散型マルチキャスト経路制御方式

5.1 緒言

近年の通信ネットワークにおいては，1対1のユニキャスト通信サービスに加えて，TV会議や音楽配信のような多人数で利用するマルチメディアアプリケーションの提供が期待されている [86,87]．このようなサービスは，ネットワークの効率的利用の観点から，マルチキャスト通信によって実現されることが望ましい．マルチキャスト通信では，同一の情報を複数のノードに送信する際に，経路が同じである限りは1つの情報として伝送し，経路が分岐するノードで情報を複製し，それぞれの受信ノードに転送するため，共通の経路上を流れるトラヒック量を削減することができる [88]．ここで，上記の通信経路は木構造をとり，マルチキャストツリーと呼ばれる．このような情報複製機能を有するノードで構成されたネットワークにおいて，マルチキャスト通信を行う際の経路制御については多くの研究がなされている [7,87,90]．

一方，映像配信などのマルチメディアサービスを提供する際には，経路全体のコストを下げるだけでなく，各受信ノードへの経路が遅延やジッタ等の複数のQoS要求を満たすことが重要となる [21,23]．これまでに，複数のQoSを考慮したユニキャストの経路選択アルゴリズムに関しては前章で示したように多数の研究事例がある．しかし，マルチキャスト通信に関しては，遅延や帯域など1つのQoS制約を満たしつつコストの最適化を図るアルゴリズムがいくつか提案されているものの，複数のQoS，つまりコストも含めて3つ以上の指標を考慮したものはない [91–96]．

そこで本研究では，2つ以上の任意の数のQoS制約を満足しつつ，経路全体のコストを削減するマルチキャスト経路制御方式について検討し，1対多及び多対多のマルチキャストツリー生成手法を提案する [35,36,38]．

5.2 1対多通信経路制御方式

5.2.1 問題の定式化

本研究で検討する複数のQoSを考慮したマルチキャストツリー生成問題は，以下のように定式化される．

まず，ネットワークトポロジーとして，接続グラフ $G = (V, E)$ を考える．ここで，V はネット

ワーク中のノードの集合，Eはリンクの集合とし，全てのリンクにはQoS指標$w_1(l), w_2(l), \cdots, w_n(l)$ $(l \in E)$とコスト$c(l)$ $(l \in E)$が付加されているものとする．ただし，nは考慮するべきQoS指標数である．

次に，マルチキャスト通信における送信ノードs $(s \in V)$と受信ノード集合D $(D \subset V - \{s\})$，及びユーザから要求されたQoS制約$r_1, r_2, \cdots, r_n$が与えられたとする．ただし，これらのQoS制約は全てのユーザに対して同一であるものと仮定する．この仮定は，NearVoDや音楽配信等のアプリケーションに適用可能である．この時，本研究で検討する問題は，sを根としてDに含まれるノードd_iを全て接続し，かつ，任意の$i = 1, 2, \cdots, |D|$と$k = 1, 2, \cdots, n$に対して，QoS制約

$$\sum_{l \in p_i} w_k(l) \leq r_k$$

を満たす木のうちで最小の$\sum_{l \in E_T} C(l)$をもつ木$T = (V_T, E_T)$を発見する問題と表せる．ただし，p_iは木E_Tでの根sから受信ノードd_iへの経路を表すものとする．ここで，$(\{s\} \cup D) \subseteq V_T \subseteq V$, $E_T \subseteq E$である．

しかし，前章でも述べたように，複数の(加法的)QoS制約を満たすユニキャスト経路選択問題はNP完全であることが知られている[22]．また，QoS制約のないマルチキャスト経路選択問題はスタイナー木問題に帰着されるが，これもまたNP完全である[89]．よって，本研究で扱う問題もまた明らかにNP完全であるため，多項式時間で決定的に解くことはできない．そこで，発見的なアルゴリズムを考える．

5.2.2 ユニキャスト経路による実現

先に述べたように，複数のQoSを考慮したユニキャスト経路選択アルゴリズムは多数提案されているが，マルチキャストに関しては，2つ以上のQoS指標（コストを含めて3つ以上の指標）を考慮したものはない．そのため，既存手法によって複数のQoS制約を満たすマルチキャスト経路を生成するためには，送信ノードから全受信ノードへのユニキャスト経路を個別に生成し，それらを結合して利用することになる．

この時，各経路を逐次的に生成すれば，2つ目以降の経路を生成する際にそれまでに決定した経路の情報を利用することで，経路全体のコストを抑えられる．本稿では，これを直列方式と呼ぶ．

例えば，図5.1のネットワークにおいて，送信ノードSから受信ノード$M1, M2, M3$への経路生成を考える．ノードa〜fは中継ノードである．ここでリンクに付与された数値は（コスト，遅延）を表すものとし，ユーザからのQoS要求として，遅延合計8以下を制約条件とする．なお，ここでは簡単のため，ただ1つのQoSを考慮しているが，任意数のQoS指標に対して同様の方法を適用可能である．また，用いるユニキャストルーチングアルゴリズムによって，要求されたQoSを満たし，かつコストが最小となる経路が必ず見つかるものと仮定している．

経路生成の対象となる受信ノードは無作為に決定されるが，この例ではM1,M2,M3の順とする．まず，M1へのユニキャスト経路生成により，経路S→b→M1が得られる．ここで，2つ目以降の受信ノードに対する経路生成でこの経路上のリンクを用いてもマルチキャスト経路全体のコストに変化はないため，リンクS→bとb→M1のコストを0とする(図5.2)．

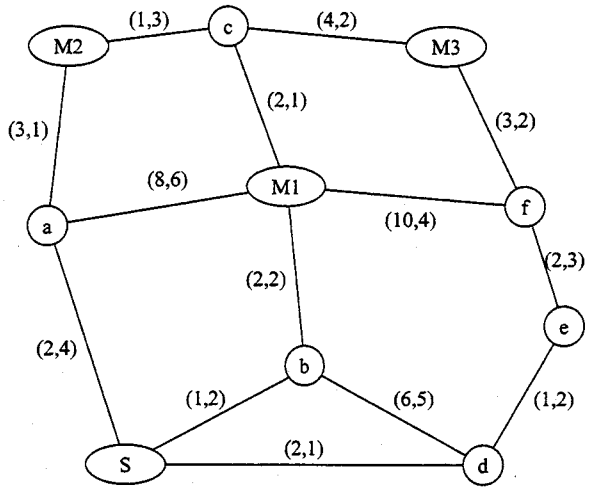


図 5.1: ネットワーク例

次に，M2への経路S→b→M1→ c→M2が得られる．更に同様にリンクM1→c，c→M2のコストを0とする (図5.3).

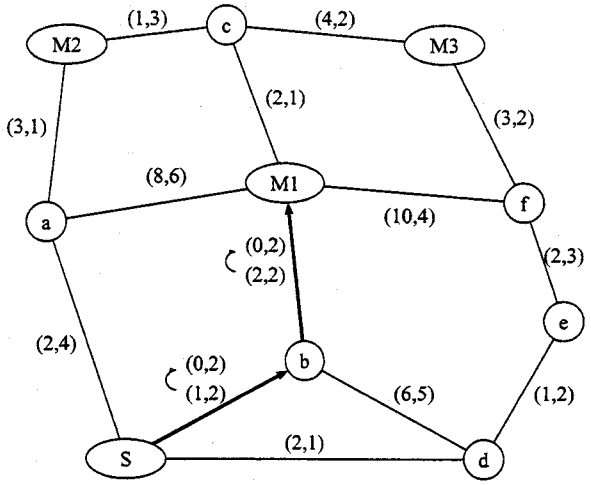


図 5.2: 直列方式の実行例 (step 1)

最後に，M3への経路が得られ，マルチキャストツリーが完成する (図5.4).

このようにして得られたマルチキャストツリーは与えられた制約条件を満たし，総コストは10となる．しかし，この方式は全ての受信ノードに対して逐次的にユニキャスト経路を生成するため，その数に比例して処理時間が増大する．従って，受信ノード数が多いときには実用的な時間でマルチキャスト経路を生成できないという問題がある．

ここで，複数のQoSを考慮したユニキャストルーチングアルゴリズムのうちで，Dijkstraのアルゴリズムを利用するアルゴリズム [20,24,26,27] では，1対1の経路を導出することで，結果的に発信ノードから他の全てのノードへの経路が同時に求まる．そこで，これらの経路をそ

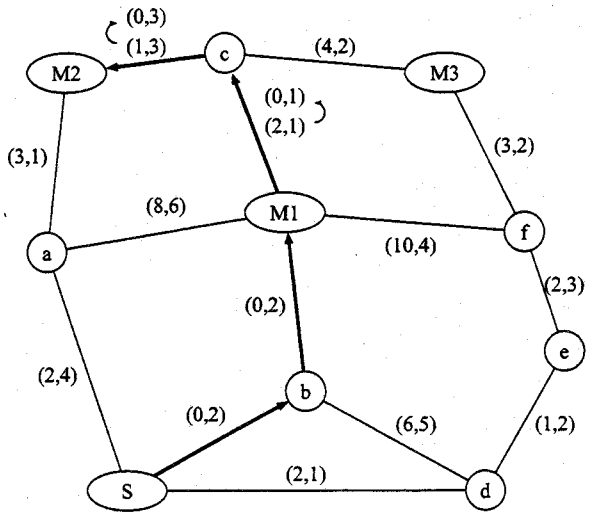


図 5.3: 直列方式の実行例 (step 2)

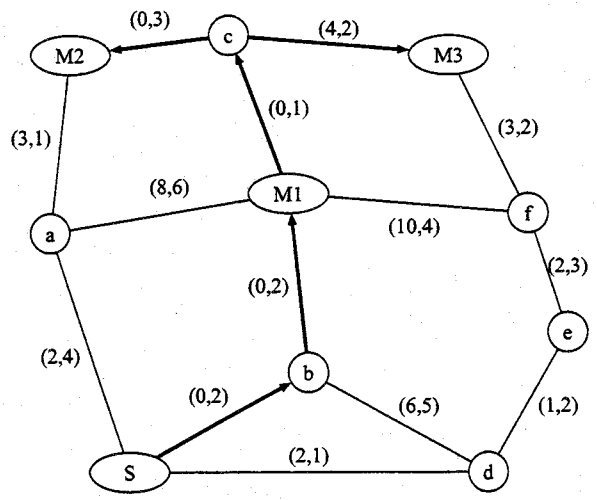


図 5.4: 直列方式の実行例 (step 3)

のまま利用すれば，処理時間の問題は回避される．しかしこの場合，直列方式のように各受信ノードへの経路の重なりを利用することができないことに注意する．この方式を本稿では並列方式と呼ぶ．

図5.1と同じ条件で，並列方式を用いて経路を生成した例を図5.5に示す．

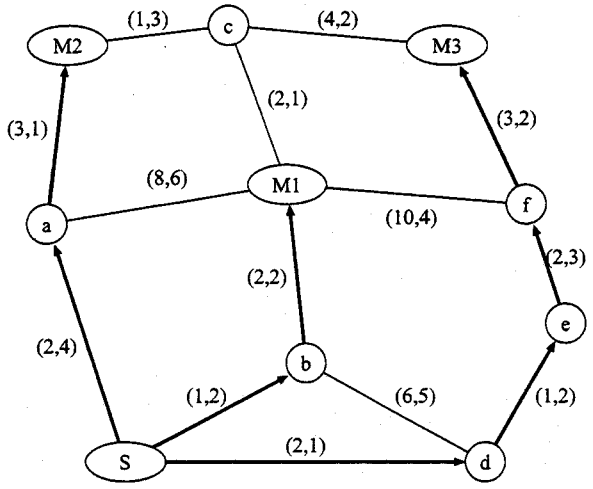


図 5.5: 並列方式の実行例

各メンバノードに対して経路を生成すると，それぞれ制約を満たす経路S→b→M1，S→a→M2，S→d→e→f→M3が得られる．並列方式ではこの1対1経路をそのまま利用してマルチキャストツリーを生成するため，総コストは16となる．

このように，並列方式では各受信ノードへの経路の重なりを利用することができないため，得られるマルチキャストツリーのコストは直列方式よりも大きくなる．

5.2.3 提案方式

本研究で提案する方式は，Dijkstraのアルゴリズムを拡張し，各受信ノードへの経路を求める際に，コスト最小の1つだけでなく，QoS要求を満たしている限りは2つ以上の経路を候補として残しておき，これらを効率的に組み合わせることでマルチキャストツリー全体のコストを削減しようとするものである．

【Dijkstraのアルゴリズムの拡張】

Dijkstraのアルゴリズムは，送信ノードから順に目的関数の値の小さい経路を辿りながら最短経路を求める．つまり，Dijkstraのアルゴリズムでは，実行する過程で暫定的な経路が次々に生成され，注目指標についてより良い経路が発見されるたびにこの暫定的な経路が廃棄される．これは，Dijkstraのアルゴリズムが単一の指標について最適な経路を導出することを目的としたアルゴリズムだからである．しかし，本研究で扱う問題に関しては，要求されたQoSを満たしていれば，必ずしもその指標に関して最適な経路でなくても良いことに注意する．

そこで，最短経路を生成する過程において得られる暫定経路を廃棄せずに，一時的にスタックに保管させる操作をDijkstraのアルゴリズムに付加する．また，目的関数の値が等しい場合にも，その経路をスタックに保管する．これによって，全てのノードに対してある指標に関する最適経路および暫定経路が得られる．この方法は，前章で提案した複数のQoS要求を考慮したユニキャストルーチングアルゴリズムFallback+ [29]でも用いられているが，この時は，1対1の経路生成を目的としているため，スタックへの保管を目的ノードでのみ行っている．本研究では，これをマルチキャストツリー生成にも適用するため，全ノードでこの暫定経路保存を行う．

例として，図5.6のネットワークで，発信ノードをSとした場合の経路生成過程を示す．リンクに付加された数字が，目的関数とするQoS指標を表すものとする．また，ネットワークの右側には前ノードテーブルが示され，注目ノードに到達するまでの目的関数の値と直前のノードが表されている．

1. まず，発信ノードSに接続されているノードa，bについて，前ノードテーブルにSが書き込まれる．ここで，目的関数を計算すると，最小で到達できるノードはbであるため，bへの経路が確定する(丸付き文字で表す)．

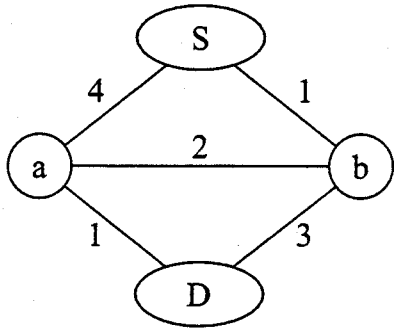
2. 次に経路の確定したノードbを経由して到達できるノードa，Dに注目すると，Dにはこれまで前ノード情報がないため，前ノードテーブルにbが書き込まれる．一方，aには既にSという前ノード情報があるが，経路S→aに要する目的関数値4に対して，経路S→b→aに要する値3の方が小さいため，前ノード情報を書き換える．ここで，Dijkstraのアルゴリズムではこれまでの前ノード情報は廃棄されるが，拡張されたアルゴリズムではこれをスタックに残す．ここで，経路S→b→Dによる目的関数値も4となり，経路S→b→aによる目的関数値3を超えるため，ノードaについても経路が確定する．

3. 残った未確定ノードDに対して，前ノードテーブル上にある経路S→b→Dと先に確定したノードaを経由する経路S→b→a→Dが比較される．ここでは，指標は等しい値となるため，これも前ノードテーブルに加えられる．

以上のようにして，Dijkstraのアルゴリズムでは得られなかった経路S→aや，どちらか一方しか残らなかった経路S→b→D，S→b→a→Dを候補経路として扱うことができる．なお，この処理を加えても時間計算量は変わらないことに注意する．

【マルチキャストツリー生成】

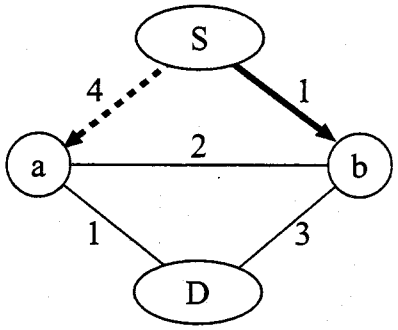
本研究で提案するマルチキャストツリー生成方式は，前項で示したように，拡張したDijkstraのアルゴリズムを利用する．まず，このアルゴリズムを用いて，コストおよび全てのQoS指標に関して最適な経路を生成し，その際得られる暫定経路を含めた全ての経路のうち，要求されたQoSを全て満たすものを候補経路とする．そして，これらの候補経路を効率的に組み合わせることによりコストを抑える．



pre-node table

	S	a	b	D
pre-node		-	-	-

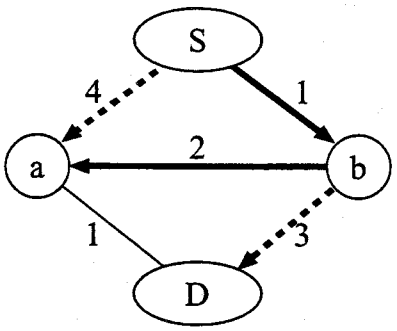
(初期状態)



pre-node table

	S	a	b	D
pre-node		S,4	S,1	-

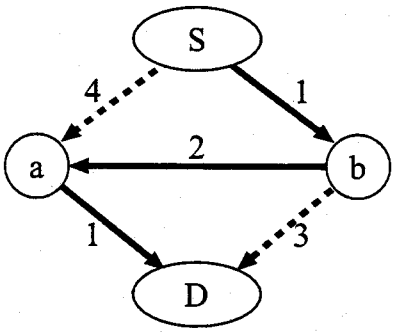
(1)



pre-node table

	S	a	b	D
pre-node		b,3	S,1	b,4
		S,4		

(2)



pre-node table

	S	a	b	D
pre-node		b,3	S,1	a,4
		S,4		b,4

(3)

図 5.6: 拡張されたDijkstraのアルゴリズムの動作例

具体的には，以下の手順でマルチキャストツリーを生成する．ここで，各経路のコストをC，経路中に含まれる受信ノードのうちで，まだマルチキャストツリーに含まれていない物の数をmとする．

1. ある受信ノードに対して，そのノードを含む経路が1つしかない場合は，その経路を採用する．このような経路が存在しない場合には，何も行わずに手順(2)へ進む．
2. 候補経路の中からC/mが最小のものを採用する．ただし，既に採用された経路に含まれている受信ノードはこの数に含めない．
3. 手順(2)で選択した経路上のリンクのコストを0として，残りの候補経路のコストを再計算する．
4. 手順(2),(3)を全ての受信ノードがマルチキャストツリーに含まれるまで続ける．

図5.7を用いて提案方式の実行例を示す．図中のSを送信ノード，M1〜M5を受信ノード，その他のノードを中継ノードとする．また，リンクに付加された数値は(コスト，遅延)とし，遅延制約を9以下とする．

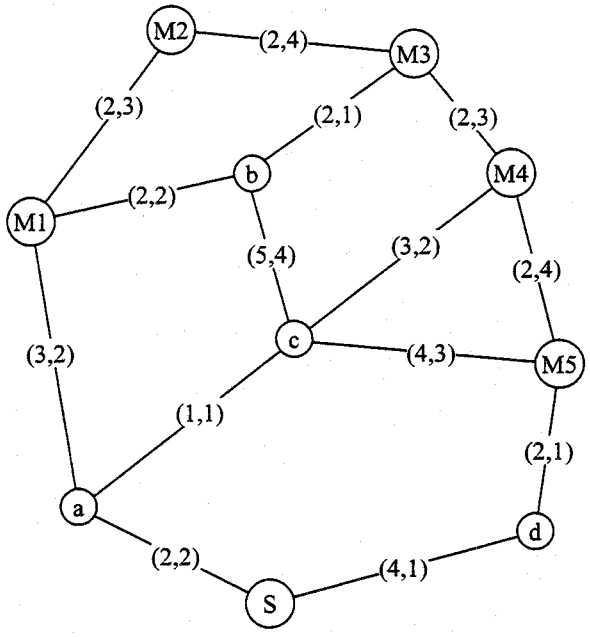


図 5.7: ネットワーク例

まず，拡張したDijkstraのアルゴリズムを用いて，目的関数とする指標をコストおよび遅延として，各受信ノードに対して経路を生成する．これによって各目的関数に対して得られる，4.1節で述べた前ノードテーブルおよびそれに対応する(コスト，遅延)の値はそれぞれ図5.8のようになる．

これより，図5.9に破線で示したような8つの遅延制約を満たす経路route1〜route8が得られる．また，これらの経路のコストは表5.1のようになる．

For cost optimal routes.

	S	a	b	c	d	M1	M2	M3	M4	M5
pre-node (cost,delay)		S (2,2)	c (8,7)	a (3,3)	S (4,1)	a (5,4)	M1 (7,7)	M4 (8,8)	c (6,5)	c (7,6)
			M1 (7,6)							d (6,2)

For delay optimal routes.

	S	a	b	c	d	M1	M2	M3	M4	M5
pre-node (cost,delay)		S (2,2)	c (8,7)	a (3,3)	S (4,1)	a (5,4)	M1 (7,7)	M4 (8,8)	M5 (8,6)	d (6,2)
			M1 (7,6)					b (9,7)	c (6,5)	

図 5.8: 前ノードテーブルと対応する (コスト，遅延) の値

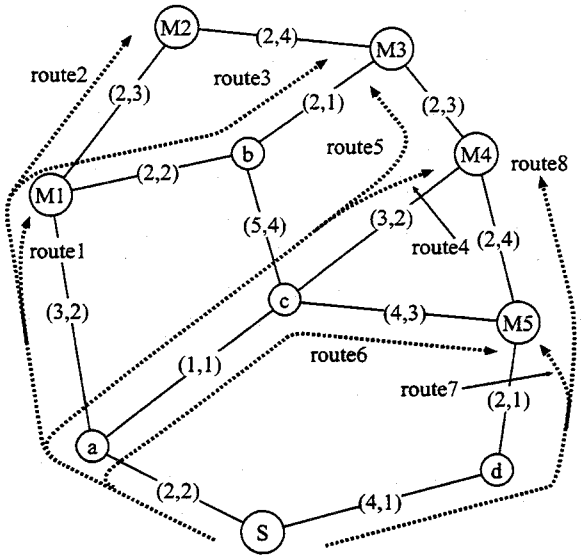


図 5.9: 提案方式で用いる候補経路

表 5.1: 候補経路のコストの初期値

route#	1	2	3	4	5	6	7	8
C	5	7	9	6	8	7	6	8
m	1	2	2	1	2	1	1	2
C/m	5	3.5	4.5	6	4	7	6	4

ここで，受信ノードM1〜M5のうち，M2を含む経路はroute2のみであるため，まずこれを採用し，この経路上のリンクのコストを0として候補経路のコストを計算し直す(図5.10，表5.2).

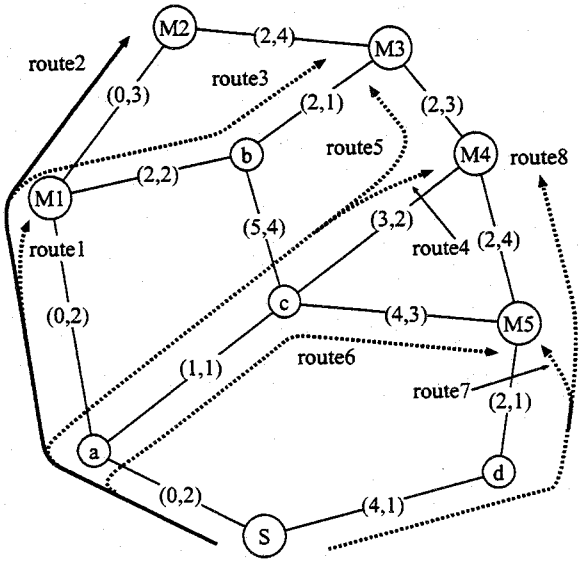


図5.10: 提案方式の実行例（step 1）

次に，表5.2より，C/mの値が最も小さいroute5を採用する．また，同様にCとmを再計算する(図5.11，表5.3).

表5.2: 候補経路のコスト（step 1）

route#	1	2	3	4	5	6	7	8
C	*	*	4	4	6	5	6	8
m	*	*	1	1	2	1	1	2
C/m	*	*	4	4	3	5	6	4

最後に，表5.3より，C/mの値が最小のroute6を採用すると，全ての受信ノードへの経路が確定し，マルチキャストツリーが完成する(図5.12).

5.3 多対多通信経路制御方式

5.3.1 既存方式

一般に多対多のマルチキャストルーチングは，前節で述べた1対多のマルチキャストツリーを利用して実現される．この時，1つのマルチキャストツリーを全ての参加ノードで共有する

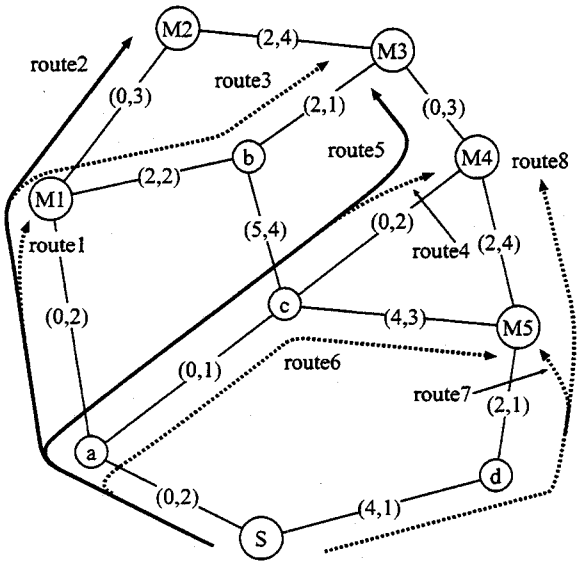


図 5.11: 提案方式の実行例（step 2）

表 5.3: 候補経路のコスト（step 2）

route#	1	2	3	4	5	6	7	8
C	*	*	*	*	*	4	6	*
m	*	*	*	*	*	1	1	*
C/m	*	*	*	*	*	4	6	*

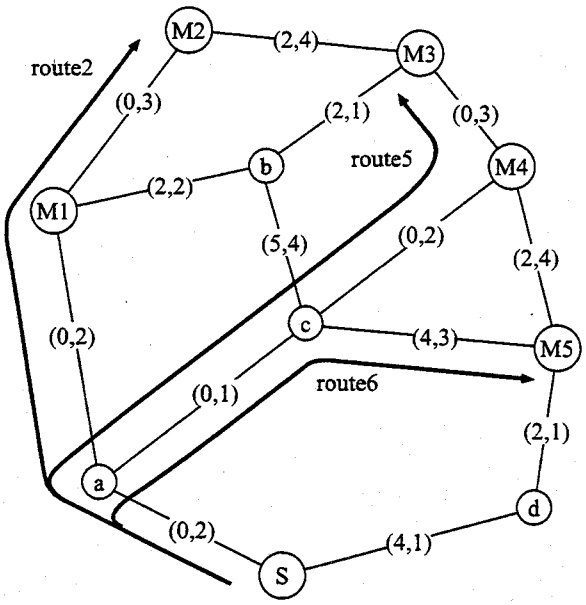


図 5.12: 提案方式の実行例（step 3）

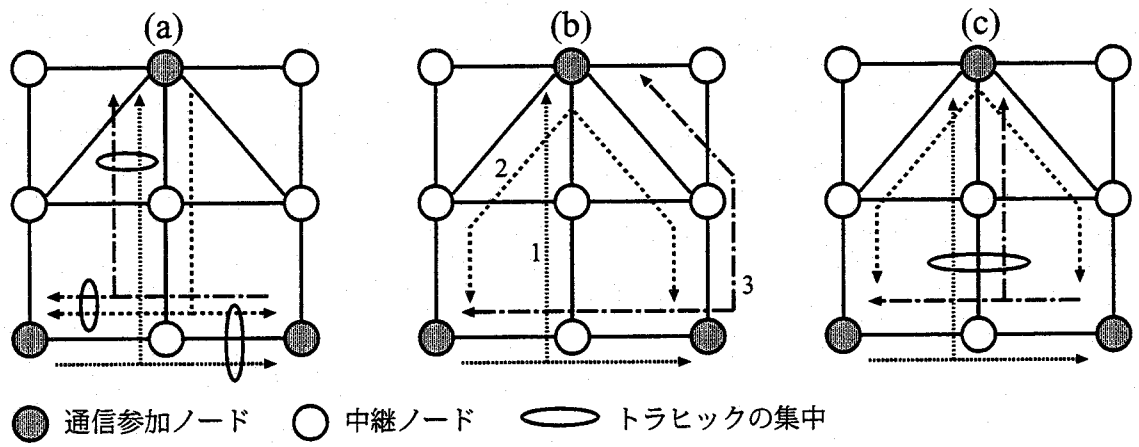


図 5.13: 多対多通信の実現法

*Shared-Tree*方式と，各ノードが独自の1対多マルチキャストツリーを生成する *Private-Tree*方式の2つの方法が考えられる [97,98].

Shared-Tree 方式では，ただ1つのツリーを構成するため，処理が単純であるが，メンバ数の増加に伴ってツリー中のリンクにトラヒックが集中するという問題がある（図 5.13(a)）.

Private-Tree 方式は，更に，各ノードに対して順番にマルチキャストツリーを求める方法と，同時に各ノードに対するマルチキャストツリー求める方法とに分類できる．前者の方法では，トラヒックの分散を考慮したツリー構成は可能であるが，参加ノード数に応じてツリーに使用されるリンク数が増加し，その上ツリー構成に多くの処理時間を要する（図 5.13(b)）．一方，後者の方法では，処理時間を削減できるが，複数のノードで類似するツリーを構成する可能性が高く，トラヒックの集中を回避することができないことが知られている（図 5.13(c)）.

そこで本研究では，メンバ数に比例しない時間でトラヒックを効率的に分散する多対多経路経路制御方式を提案する.

5.3.2 提案方式

まず，提案方式の手順について説明する（図 5.14）．この図において，参加ノードはマルチキャストに参加しているユーザを収容しているノードを指すものとする.

1. 各参加ノードが，自身をソースノードとするマルチキャストツリーを構成する．この時，前節で提案した方式を用いてツリー構成することにより，ユーザからの複数の QoS 要求を満たすと共に暫定経路を保持する (図 5.15-1)．この時，拡張された Dijkstra のアルゴリズムを用いてツリー構成を行うことで得られる暫定経路は，後のツリー再構成の際に利用する.

2. 構成したツリー情報を相互に交換して全てのリンクの使用状況を把握する (図 5.15-2)．その後，あるリンクについて，そのリンクを使用しているノード数を総参加ノード数で割った値が設定した閾値を超えた時，ツリーの再構成を行う.

3. ツリー再構成が必要と判断した場合，該当するノードは暫定経路を用いてトラヒックを分散させるようにツリーを修正する (図 5.16).

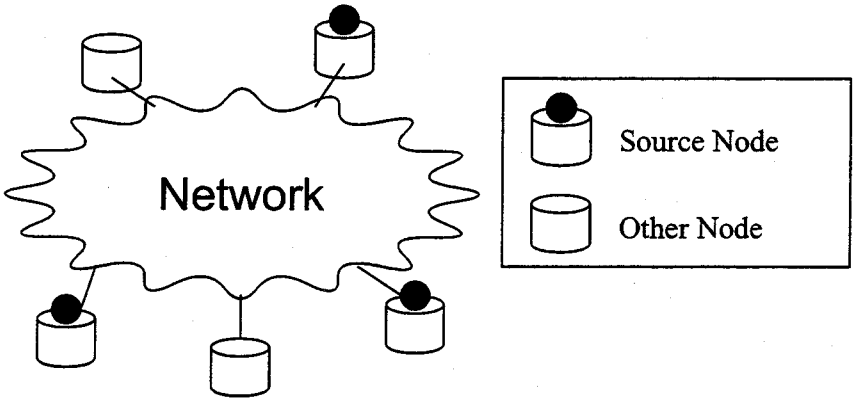


図 5.14: 手順 1

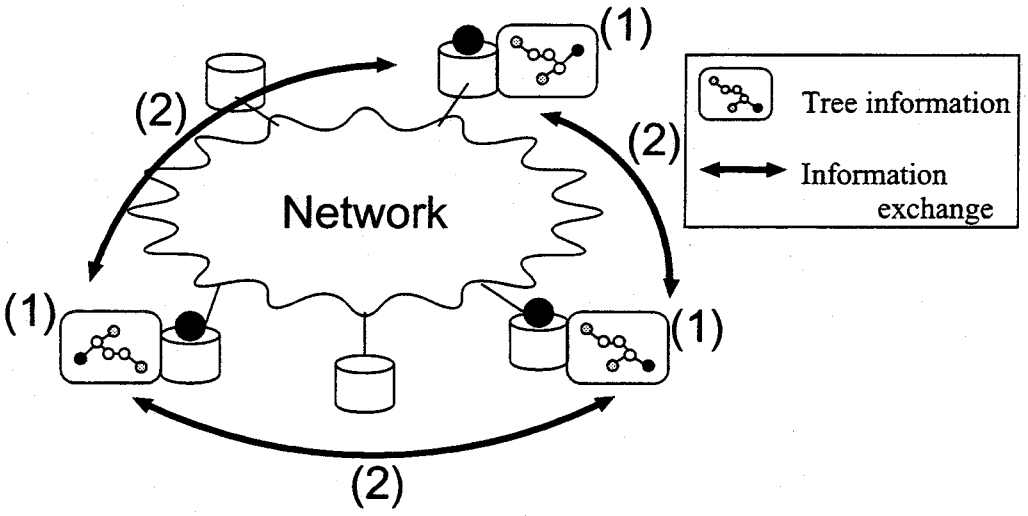


図 5.15: ツリー情報の交換

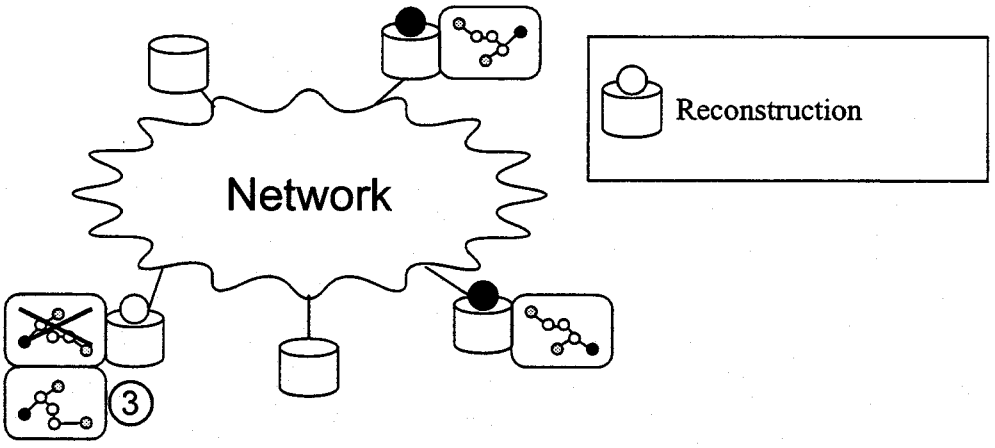


図 5.16: ツリー再構成

以下，各手順の詳細について，動作例を用いて説明する．

【マルチキャストツリー構成】

マルチキャストツリーを構成する際には，トラヒックの分散を考慮したツリーを再構成する時に有用な情報を得るために，前節で提案した方式を利用する．

提案方式では，この方式において，拡張されたDijkstraのアルゴリズムをホップ数を指標とした適用する際，同一ホップ数の経路が後から見つかった場合でも暫定経路として保持する．これにより，更に多くの暫定経路を保持できる．

図5.17を用いて，ノードSからノード1, 2,Dへのマルチキャストツリーを構成する場合について説明する．ここで表5.4は，生成された経路を各ノードからソースノードへ逆にたどる際に最初に訪れるノード情報(以下，前ノード情報)を保持するものであり，#1は先に見つかる前ノード情報を#2は後から見つかる前ノード情報を表す．

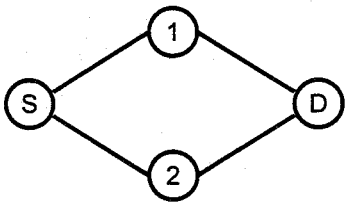


図 5.17: ツリー構成1

表 5.4: 前ノード情報:初期値

ノード	S	1	2	D
#1	X	-	-	-

まずはじめに図5.18に示す順序でソースノードSから1ホップで到達可能なノード1，2が選択される．その結果リンクテーブルは表5.5のように更新される．

表 5.5: 前ノード情報2

ノード	S	1	2	D
#1	X	S	-	-

⇓

ノード	S	1	2	D
#1	X	S	S	-

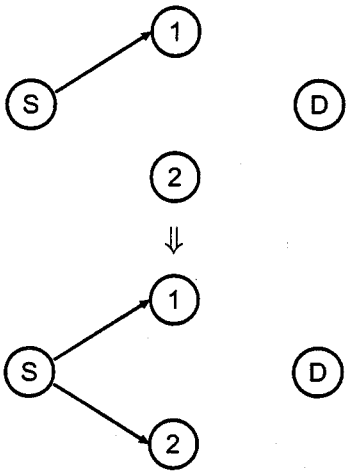


図 5.18: ツリー構成 2

次に，既に経路が生成されているノードの中で一番ホップ数の少ないノードから 1 ホップで到達可能な経路が生成される．図 5.19 に示すようにまずノード 1 からノード D までの経路が生成され，リンク使用テーブルは表 5.6 のように更新される．

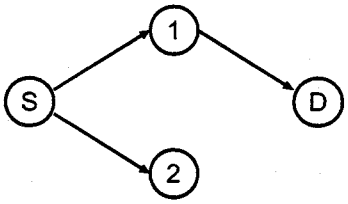


図 5.19: ツリー構成 3

表 5.6: 前ノード情報 3

ノード	S	1	2	D
# 1	X	S	S	1

通常ではソースノード S から全てのノードまでの経路が生成されているため処理が終了するが，更にノード 2 から 1 ホップで到達可能な経路があるかどうか検討し，その結果，ノード D に対して総ホップ数 2 となる経路が見つかる．ノード D へは既に総ホップ数 2 となるノード 1 からの経路が生成されているが，同一ホップであるノード 2 からの経路についても情報を保持する．そのためリンク使用テーブルは，表 5.7 のように更新される．

最終的なマルチキャストツリーは，表 5.7 の最新の前ノード情報を用いて，図 5.21 に示すように構成される．

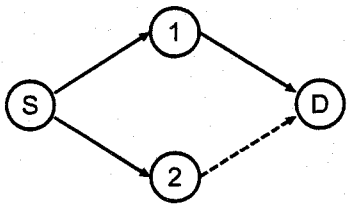


図 5.20: ツリー構成4

表 5.7: 前ノード情報4

ノード	S	1	2	D
#1	X	S	S	1
#2	-	-	-	2

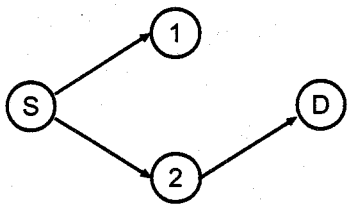


図 5.21: ツリー構成5

【再構成判定】

ツリーの再構成を行うかどうかの判定は，以下に示す式5.1と閾値thを用いて行う．

$$f(m) = \frac{m}{M} \tag{5.1}$$

式5.1において，M，m，はそれぞれマルチキャストに参加しているノード数及び，あるリンクを使用しているノード数を表す．

まず$f \leq th$となる場合は，再構成は行わない．

$f > th$の時は，該当リンクを使用するノードの中で閾値を越えた$(m - M \times th)$数に等しいノードで再構成を行う．この時，どのノードが再構成を行うかは，事前に設けた優先順位によって決定する．提案方式では，各参加ノードに固有の番号を割り付け，番号が大きいものから順に閾値を超えた数のノードで再構成を行うようにした．これよりノード間で交渉を行わずとも自律的に再構成を行える．

【再構成】

各ノードは，はじめにFallback+によりツリー構成を行った際に，保持された暫定経路を用いて再構成する．以下にソースノードを8とした場合の再構成の動作例を示す．

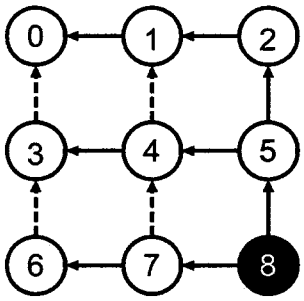


図 5.22: ツリー情報

Node	0	1	2	3	4	5	6	7	8
#1	1	2	5	4	5	8	7	8	X
#2	3	4	-	6	7	-	-	-	X

表 5.8: 前ノード情報

ノード8において，Fallback+を実行すると表5.8のような経路情報を示すテーブルが得られる．図5.22はテーブルを図示したものである．図において実線は，先に選択されたリンクを破線は後から選択されたリンクを表す．

ノード0がマルチキャストグループに参加している場合，表5.8のテーブルより，図5.23に示すような経路$8 \rightarrow 7 \rightarrow 6 \rightarrow 3 \rightarrow 0$が選択される．

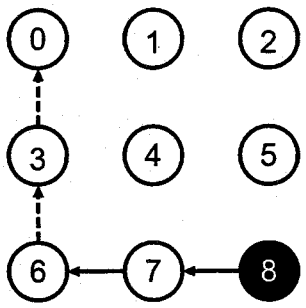


図 5.23: はじめに選択された経路

ここで，他のノードとのツリー情報を交換した結果，図5.24で示されるようにリンク6 → 5において重なりが検知され，更にツリーの再構成が必要であると判断した時，表5.8を参照しツリーの再構成すると，8から0へ向かう経路上にあるノード3に注目すると，ノード6からの経路情報以外に　ノード4, ノード3と経由する経路情報があることが分かる (図5.24の斜線に示されるリンク)．この情報を基に経路を再構成すると，重複利用リンクを避ける経路8 → 7 → 4 → 3 → 0が選択できる．

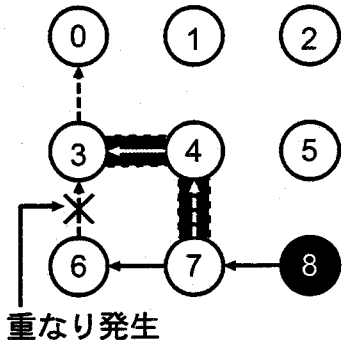


図 5.24: 重複検出

最終的に，図5.25のようなツリーが完成する．

5.4 性能評価

5.4.1 1対多通信経路制御方式に関する評価

【時間計算量】

本節では，提案方式の定性的な性能評価として最悪時間計算量を導出する．また第3章で述べた並列方式および直列方式の時間計算量も導出し，提案方式と比較する．以下では，ネットワーク中のノード数を$|V|$，メンバノード数を$|D|$，コストおよび考慮するQoS指標の総数をnとし、拡張したDijkstraのアルゴリズムによって得られる候補経路数をjとする．

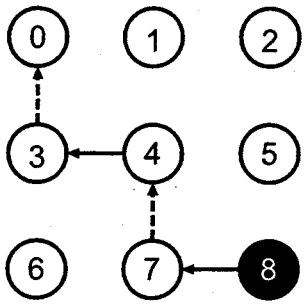


図 5.25: 再構成後の経路

まず，主要な複数のQoSを考慮した経路選択アルゴリズムの最悪時間計算量は$O(|V|^2)$もしくは$O(n|V|^2)$であることが示されている [21, 23, 26]．よって，これらのアルゴリズムを受信ノード数分逐次的に実行する直列方式の時間計算量は$O(|D||V|^2)$もしくは$O(n|D||V|^2)$となる．

次に，並列方式の計算量を導出する．並列方式では，各受信ノードに対する経路生成は1回のDijkstraのアルゴリズムによって行われるため，受信ノード数に関係なく時間計算量は$O(|V|^2)$もしくは$O(n|V|^2)$となる．

最後に提案方式の時間計算量を示す．はじめに候補経路を求める処理の時間計算量は並列方式と同じく$O(|V|^2)$もしくは$O(n|V|^2)$である．次に候補経路を組み合わせる際の時間計算量を示す．まず，C/mが最小の経路を選ぶ際の時間計算量はヒープソートを用いると$O(\log j)$である．そして残りの候補経路についてコストを計算し直すが，ひとつの経路に含まれるリンクは最大で$|V|-1$個 (全てのノード間にリンクがある場合) であるため，コスト再計算の処理の計算量は

$$O((j-1)\times(|V|-1)) = O(j|V|)$$

である．これを最大で受信ノード数分行うので，結局$O(j|D||V|)$となる．ここで，提案方式は各受信ノードに対してコストおよびQoS指標の数だけ拡張されたDijkstraのアルゴリズムを実行するので$j \propto n|D|$である．よって一方，

$$O(n|V|^2) + O(j|D||V|) = O(n|V|^2) + O(|D|^2V)$$

となる．

以上の結果より，提案方式の最悪計算量は並列方式よりは大きいものの，$|D| < |V|$であるため，直列方式よりは小さいことが分かる．

【定量的評価】

本節では，シミュレーション実験を行い，各方式を用いて得られるマルチキャストツリーのコストを比較する．また，平均実行時間についても評価する．

まず，ネットワークモデルとして，ノード数が400のランダムグラフをWaxmanの方法 [83] によって生成し，各リンクにコストおよびQoS指標を一様乱数により決定される1から6まで

の整数値として付加したものをそれぞれ100通り用意した．ここで，ノード当たりのリンク数は平均4本とした．送信ノード及び受信ノードは等確率に選ばれるものとし，受信ノード数は10〜50とした．

また，考慮するQoS指標数は2（コストを含めて3）とした．全ての受信ノードは同一のQoSを要求するものとし，その値は，受信ノード数20のとき，制約を満たす経路を得られる割合が90%程度になるような値を求めて予備実験を行い，28とした．

以上の条件で，並列方式，提案方式，直列方式それぞれによって全ての受信ノードについて制約を満たす経路を生成する処理を1つのネットワークモデルにつき100回，計10000回試行した．

このシミュレーションにおいて，各方式によって得られた経路全体のコストの平均値を並列方式による結果で正規化したものを図5.26に示す．ただし，横軸は受信ノード数であり，結果には自由度9のt分布を用いて算出した95%信頼区間を付加している．

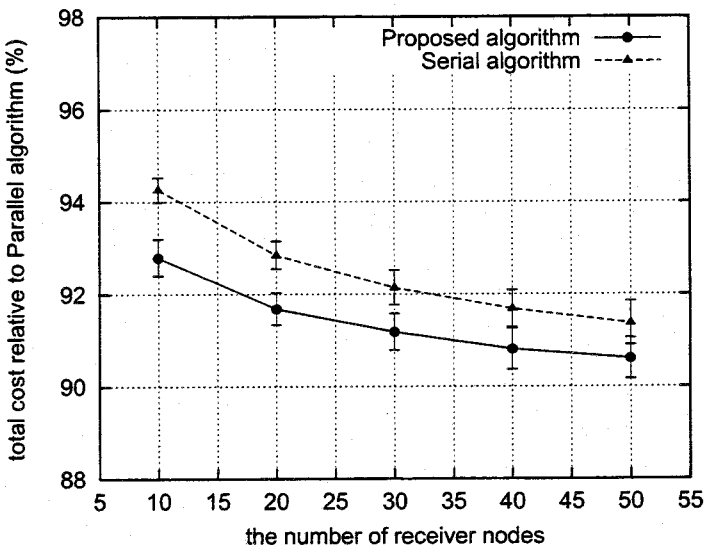


図 5.26: シミュレーション結果

この結果から，提案方式は複数のQoS要求を満たしつつコストを抑えたマルチキャストツリーを生成していることが分かる．

次に，平均実行時間（CPU時間[sec]）を表5.9に示す。なお，実行環境はSun Ultra80である．

表 5.9: 平均実行時間

受信ノード数	10	20	30	40	50
提案方式	0.1388	0.1451	0.1500	0.1553	0.1612
直列方式	1.6121	2.9179	4.1440	5.3876	6.5521
並列方式	0.1372	0.1422	0.1449	0.1475	0.1498

この結果から，提案方式は5.1節で述べた最悪時間計算量については並列方式より大きいものの，平均的には並列方式とほぼ同等の処理速度を達成していると言える．

以上のように，提案方式は複数のQoS要求を満たしつつコストを抑えたマルチキャストツリーを高速に生成可能であることが分かる．特に，受信ノード数が多くなっても処理時間がほぼ一定で，かつ，より効果的にコストを削減可能である点は，大規模なマルチキャスト通信サービスを提供するにあたって重要な特性であると言える．

【評価関数に関する考察】

4.2節で提案した方式では，候補経路からマルチキャストツリーを生成する際，受信ノード数当たりのコストC/mを評価関数に用いたが，ここで，候補経路のコストCあるいは候補経路に含まれる受信ノード数mを評価関数とすることも考えられる．そこで，提案方式において，この評価関数のみを変更した方式を，前節で行ったシミュレーションと同じ条件で評価した．この結果を，図5.27に示す。

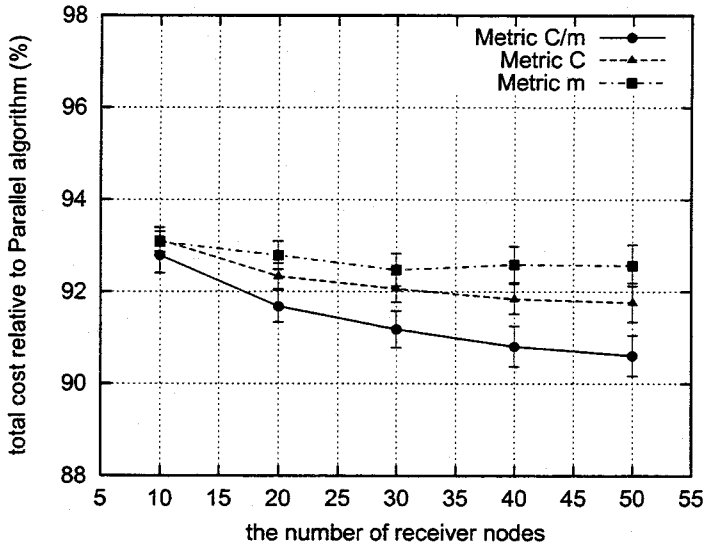


図 5.27: 評価関数の変更

この結果から，やはり評価関数としてC/mを用いた方が平均的に高品質な経路を生成することが分かる．

5.4.2 多対多通信経路制御方式に関する評価

【時間計算量】

この節では，提案したアルゴリズムの計算量を評価する．以下，V，L，Mはそれぞれ，ネットワーク上の全てのノード数，1つのノードに接続される平均リンク数，マルチキャスト通信

に参加している総ノード数を表すものとする.

ツリー構成

Fallback+を利用してツリーを構成するため $V \log V$ の計算量を要する [29].

再構成判定

各参加ノードから送られてくる最大 $V-1$ のリンクを使用するツリーの情報を基に，リンクの使用状況を表すテーブルを作成するために，$(M-1)(V-1)$ の計算量を要する．更に，そのテーブルを利用して一つのツリーで使用される最大のリンク数だけツリーの再構成が必要かどうかを判定するため $V-1$ の計算量を要する．よって再構成判定には $M(V-1)$ の計算量を要する.

再構築

全参加ノードに対してツリーの再構築を行った場合，各参加ノードに対する最大経路長が $V-1$，各ノードに対して最大 L 個の前ノード情報を保持するため $(M-1)L(V-1)$ の計算量を要する.

この結果から提案方式では，式5.2で表される計算量を要する事がわかる.

$$V \log V + M(V-1) + (M-1)L(V-1) \tag{5.2}$$

この計算量は，参加ノード数及び1ノード辺りの平均リンク数に依存するが，一般的なネットワークでは，1ノードあたりのリンク数は全ノード数に比べ十分に少なく，また参加ノード数についてもノード数よりも十分に少ないことが考えられる．このことから提案方式の最大計算量は $O(V \log V)$ となり，実用的な処理量をもつことがわかる.

【トラヒックの分散】

シミュレーション条件 提案手法の定量的性能を評価するため，以下に示す条件でシミュレーションを行った.

まず，1ノードあたり平均4本のリンクを持つ100ノードのグラフをランダムグラフ生成法 [83] によって10通り生成し，マルチキャストグループは，ランダムに選択した10ノードのグループを10000通り用意した.

また比較手法は，並列にツリーを構成し，再構成を行わない方法と，直列的にツリーを構成する方法とした .

評価指標は，あるリンクについて，そのリンクを使用しているノード数を表す「重複度」及びマルチキャストグループで使用される延べリンク使用本数を表す「リンク使用数」とした．尚，再構成条件及び再構成判定で使用する閾値は，0.2,0.3,0.4,0.5とした．また，比較手法においても，ツリー構成時にはFallback+を使用する．これによりQoSを考慮したマルチキャストルーチングを行えることを前提とし，特にツリー構成に対して制約を設けていない.

評価結果 図5.28に示すグラフは，生成されたツリーで使用されたリンク数を重複度別に表したものである．このグラフにおいて横軸は重複度を表し，縦軸は再構成前の結果によって提案方式を正規化した正規化リンク数を表す．ここで提案方式の通信回数は2回 (再構成1回) とした.

このグラフから，提案方式は7～9と重複度の高いリンクの数を高い割合で減少させていることが確認できる．また閾値別にグラフを見た場合，低い閾値を設定することによってより大

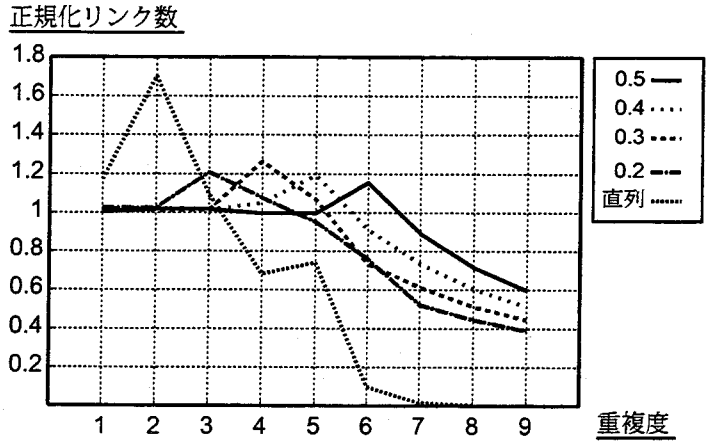


図 5.28: 重複度別リンク数 (通信回数 2)

きなトラヒックの分散が達成できることが確認できる．このことから閾値を下げて積極的に再構成を行う方が良い傾向にあることが分かる．

次に，表 5.10 は，方式別のリンク使用数の平均を表している．この表から，提案手法は再構成しない場合と同等のリンク使用数でトラヒックの分散を考慮したツリーを構成できることがわかる．一方，直列的にツリーを構成する場合は，提案方式に比べてより多くのリンク数を削減できるが，それと共にリンク使用数が増加することがわかる．

表 5.10: 方式別リンク使用数

方式	再構成前	提案方式	直列構成
リンク利用数	200.45	202.92	220.87

よって提案方式は，効果的にトラヒック分散を考慮したツリー構成を可能とすることが分かる．

5.5 結言

本章では，これまで考えられていなかった 2 つ以上の任意の数の QoS 要求を満たしつつ経路全体のコストを削減するマルチキャストツリー生成法について検討し，新たなアルゴリズムを提案した．これは，Dijkstra のアルゴリズムを拡張することで，実行時間を増大させずに要求を満たす経路をより多く保存し，それらを効率的に組み合わせるというものである．

また，提案方式の性能を時間計算量と得られるマルチキャストツリーのコストによって評価し，よりコストの低い経路を高速に生成できることを確認した．

更に，多対多マルチキャストに対応した経路制御手法を提案した．提案手法は，複数 QoS 要求を満たす経路選択を行う Fallback+を利用してツリー構成を行い，暫定経路を保持し，必要に応じて暫定経路を利用して再構成を行うものである．

シミュレーション実験による性能評価を行った結果，提案方式は，従来の再構成を行わない並列方式と比較しトラヒックの分散，ネットワーク資源の有効利用など様々な面において優れた性能であることを確認し，その有効性を示した．

今後の課題としては，通信継続中に受信ノードの離脱や新規参入が生じた際の経路生成や，ユーザ毎に要求するQoSが異なる場合への対応などが挙げられる．

第6章

結論

本研究では，マルチメディアネットワークにおける自律分散型サービス制御アーキテクチャについて多角的に検討した．

まず第1章では，情報通信ネットワークの現状について述べ，伝達系の発展に比較して，制御系の整備が遅れていることを指摘した．その上で，次世代のマルチメディアネットワークのサービス制御アーキテクチャとしては，自律性と分散性が重要であることを示し，本論文の論点を明らかにした．

第2章では，既存のサービス制御アーキテクチャについて調査し，その特徴と問題点を明らかにした．これを踏まえて，マルチメディアネットワークに対応した新たなサービス制御アーキテクチャとしてホロニックネットワークを提案し，その基本構成を示した．ホロニックネットワークは，全光の超高速伝達網とエージェント通信を基本とした柔軟な制御網で構成され，各ノードが自律分散的に動作することから，処理性能とスケーラビリティの両面を満足するアーキテクチャである．また，この制御網の処理プラットフォームとして，エージェントシステムが適していることを示し，Agent Gateway を提案した．Agent Gateway は，自律的に移動するエージェントの位置をシステムが分散管理することで，スケーラブルに位置透過性を実現し，これまでのエージェントシステムでは考えられていなかった大規模ネットワークでの利用を可能にするマルチエージェントシステムである．更に，現在の IP ネットワークからこのホロニックネットワークへの移行方法についても考察し，アクティブネットワーク技術の適用が有効であるとの見通しを得た．

第3章では，提案アーキテクチャ上でネットワーク情報を管理するデータベースであるネットワークマップの機能と情報管理方式について説明した．ネットワークマップはネットワーク上に分散配置され，局所的な情報を詳細に管理しながら，相互に協調してネットワーク全体の情報を蓄積する機能的データベースである．このとき，変動の激しい情報を縮退して交換する手法とその有効性を示した．また，経路情報だけでなく，ネットワーク上で提供されるサービスに関する情報をも統一的に管理することで，高度なサービス制御が可能となることを示し，その方法として論理的階層構造を用いた分散管理を提案した．これらの提案方式に対して，計算機シミュレーションとプロトタイプ実装による性能評価を行い，その特性と実用性を明らかにした．

第4章では，ネットワークマップを基盤とした呼制御方式について，将来のマルチメディアネットワークにおいては，複数の QoS 指標を考慮することが重要であることを示し，サービス

検索と経路制御の両面からこれに対応した方式を提案した．提案方式は，経路選択アルゴリズムとして，既存のアルゴリズムよりも高速性と最適性の面で優れているだけでなく，ネットワークマップに蓄積されたサービス情報を有効に利用して，呼制御全体での効率を高めている．

第5章では，今後のマルチメディアサービスの実現に向けて必須の技術であるマルチキャスト通信について検討した．まず，1対多のマルチキャスト通信については，複数のQoS指標を考慮した上でネットワークにかかる負荷を軽減する経路を生成することが重要であるため，前章で提案した1対1の経路選択アルゴリズムを拡張してこれを実現した．次に，多対多のマルチキャスト通信においては，この1対多の経路制御方式をエージェントが自律分散的に利用し，かつ，相互に協調することでネットワーク上の負荷集中を回避する方式を提案した．これらの方式は，共に処理時間が通信参加者数に比例しないという特徴を有しており，大規模ネットワークでの実用にも適している．

今後の課題としては，提案アーキテクチャを実現する実装技術の開発と，それを用いたより詳細な性能評価と動作検証などが挙げられる．具体的には，エージェントシステムAgent Gatewayのソフトウェア開発，提案した経路制御方式を実現するプロトコルの設計，アクティブネットワーク技術やGMPLS技術を用いた次世代ノードのハードウェア実装などが考えられる．

謝辞

本研究遂行並びに本論文作成の全過程を通じて，多大なる御指導と御鞭撻を賜わりました大阪大学大学院情報科学研究科情報ネットワーク学専攻の村上孝三教授に謹んで感謝の意を表します.

本論文をまとめるにあたり，懇切なる御教示と御助言を賜わりました大阪大学大学院情報科学研究科情報システム工学専攻の白川功教授，藤岡弘教授に厚く感謝申し上げます.

在学中には講義・学生生活を通じて，また，在職中には職務を通じて，情報システム工学全般に関して，懇切なる御教示と御助言を賜わりました大阪大学大学院情報科学研究科マルチメディア工学専攻の西尾章治郎教授，薦田憲久教授，岸野文郎教授，下條真司教授，及び，同バイオ情報工学専攻の赤澤堅造教授に感謝の意を表します.

本研究の初期の過程にあたり，直接の御指導・御討論と共に研究者としての姿勢を御教示頂きました京都大学大学院情報学研究科数理工学専攻の滝根哲哉助教授に衷心より厚く御礼申し上げます.

本研究の中期以降の過程にあたり，主にエージェントシステムの確立に御協力頂きました岡山大学総合情報処理センターの山井成良助教授に心より感謝致します.

本研究を進める際にさまざまな点でお世話になりました，大阪大学大学院情報科学研究科情報ネットワーク学専攻村上研究室の戸出英樹助教授，江木康雄技術専門官並びに事務補佐員の佃真理子氏に厚く御礼申し上げます.

本研究を進めるにあたり，研究室において御協力を頂きました，村上研究室卒業生の添田純一郎氏，宇野哲史氏，中吉功氏，高瀬正明氏，益田智員氏，永松孝之氏，松井知章氏，及び，在学生の渡邊君人氏，谷岡秀昭氏，河野圭太氏，神原正義氏，徳丸裕章氏，細川晃氏，山田直幸氏に感謝します.

参考文献

[1] M. Day, B. Cain, G. Tomlinson, P. Rzewski, "A Model for Content Delivery Internetworking (CDI)," draft-day-cdnp-model-09.txt, Nov. 2001.

[2] G. Nilsson, F. Dupuy, and M. Chapman, "An Overview of the Telecommunications Information Networking Architecture," TINA95, pp. 1–12, Feb. 1995.

[3] 石井啓之，西川博昭，小林秀承，井上友二，"TINA 型高品質マルチメディアネットワークの実現法の検討，" 信学論 B-I, Vol.J80, No.6, pp. 457–464, June 1997.

[4] H. Berndt, T. Hamada, and P. Graubmann, "TINA: Its Achievements and its Future Directions," *IEEE Surveys & Tutorials,* Vol.3, No.1, 2000.

[5] M. Kuwada, K. Watanabe, K. Kinoshita, N. Yamai, and K. Murakami, "A Design of Mobile Agent System with Network Transparency," in *Proceedings of the EUROMEDIA '99,* pp. 170–174, Munich, Germany, Apr. 1999.

[6] K. Watanabe, K. Kinoshita, N. Yamai, M. Kuwada, and K. Murakami, "Agent Gateway: A Location Transparent Agent Platform on Multi-Plane Logical Network," in *Proceedings of the 1999 International Conference on Software, Telecomuunications and Computer Networks (SoftCOM '99),* pp. 325–334, Split, Croatia, Oct. 1999.

[7] T. Finin, Y. Labrou, and Y. Peng, "Mobile Agents Can Benefit from Standards Efforts on Interagent Communication," *IEEE Communications Magazine,* No.7, pp. 50–56, July 1998.

[8] Vu Anh Pham and Ahmed Karmouch, "Mobile Software Agents: An Overview," *IEEE Communications Magazine,* No.7, pp. 26–37, July 1998.

[9] 佐藤一郎，"モバイルエージェントの動向，" 人工知能学会誌，Vol. 14, No. 4, pp. 598–605, July 1999.

[10] K. Kinoshita, T. Takine, K. Murakami, and H. Terada, "Holonic Network: A New Network Architecture for Personalized Multimedia Communications Based on Autonomous Routing," *IEICE Transactions on Communications,* Vol. E80-B, No.2, pp. 282–288, Feb. 1997.

[11] K. Murakami, S. Uno, K. Kinoshita, M. Higashida, and T. Takine, "Distributed Cooperative Networking Architecture for Multimedia Communications Based on Autonomous Routing," *Proceedings of the 1996 International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC '96)*, pp. 83–86, Seoul, Korea, July 1996.

[12] K. Kinoshita, H. Tode, and K. Murakami, "Future Personalized Multimedia Communication Network Composed of Optical CDM Switching and Cooperative Network Maps," *Proceedings of the 2001 IASTED International Conference on Advances in Communications (AIC2001)*, pp. 172–177, Rhodes, Greece, July 2001.

[13] B. Schulze, and E. R. Madeira, "Migration Transparency in Agent Systems," *Proceedings of the 4th International Symposium on Autonomous Decentralized Systems*, pp. 320–323, Mar. 1999.

[14] 桑田倫彦，木下和彦，山井成良，村上孝三，"ネットワーク透過性を考慮したエージェントシステム，" 電子情報通信学会技術研究報告, IN98-129, Nov. 1998.

[15] S. Maffeis, and D. Maffeis, "Constructing Reliable Distributed Communication Systems with CORBA," *IEEE Communicatons Magazine*, pp. 56–60, Feb. 1997.

[16] T. Luo, and K. S. Trivedi, "A Reliable CORBA-Based Network Management System," *Proceedings of the 1999 IEEE International Conference on Communications (ICC '99)*, pp. 1374–1387, Vancouver, Canada, June 1999.

[17] K. Kinoshita, T. Masuda, K. Kawano, H. Tanioka, T. Takine, and K. Murakami, "Network Map Architecture using Inter-Agent Communication Techniques for Multiple QoS Routing," *IEICE Transactions on Communications*, Vol. E85-B, No.7, pp. 1333–1341, July 2002.

[18] K. Kinoshita, H. Tanioka, N. Yamai, T. Takine, and K. Murakami, "Network Map Design for Holonic Routing," in *Proceedings of the 1997 International Seminar on Teletraffic and Network (ISTN '97)*, pp. 189–192, Xi'an, China, Nov. 1997.

[19] T. Masuda, K. Kawano, H. Tanioka, K. Kinoshita, T. Takine, and K. Murakami, "Network Map Design for QoS Routing on WAN," in *Proceedings of the 5th Asia-Pacific Conference on Communications (APCC '99)*, pp. 196–199, Beijing, China, Oct. 1999.

[20] C. Pornavalai, G. Chakraborty, and N. Shiratori, "QoS Routing Algorithm for Multimedia Communication," *Technical Report of IEICE*, IN97-30, pp.15–22, Apr. 1997.

[21] Z. Wang, and J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications," *IEEE Journal on Selected Areas in Communications*, Vol.14, No.7, pp.1228–1233, Sep. 1996.

[22] J. M. Jaffe, "Algorithms for Finding Paths with Multiple Constraints," *Networks*, Vol.14, pp.95–116, 1984.

[23] S. Chen and K. Nahrstedt, "An Overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions," *IEEE Network,* pp.64–79, Nov./Dec. 1998.

[24] W. C. Lee et al., "Routing Subject to Quality of Service Constraints in Integrated Communication Networks," *IEEE Network*, Vol.9, pp.46–55, July/Aug. 1995.

[25] S. Chen and K. Nahrstedt, "On Finding Multi-constrained Paths," *Proceedings of the IEEE ICC '98,* pp. 874–879, Georgia, USA, June 1998.

[26] T. Korkmaz and M. Krunz, "Multi-Constrained Optimal Path Selection," *Proceedings of the IEEE INFOCOM 2001,* pp. 834–843, Alaska, USA, Apr. 2001.

[27] 谷岡秀昭，木下和彦，滝根哲哉，村上孝三，"複数の QoS 要求を満たす経路選択アルゴリズム Fallback+ の提案，" 電子情報通信学会論文誌，Vol.J83-B，No.5，pp. 690–696, May 2000.

[28] H. Tanioka, K. Kinoshita, T. Takine, and K. Murakami, "Fallback+: A Routing Algorithm Subject to Multiple QoS Constraints," in *Proceedings of ITC-CSCC '98*, vol. I, pp. 319–322, Sokcho, Korea, July 1998.

[29] 谷岡秀昭，木下和彦，滝根哲哉，村上孝三，"Fallback+: 複数の QoS 要求を満たす経路選択アルゴリズム，" 電子情報通信学会技術研究報告, SSE98-28, May 1998.

[30] 谷岡秀昭，木下和彦，滝根哲哉，村上孝三，"複数 QoS を考慮した経路選択アルゴリズム，" 電子情報通信学会総合大会，B-7-155, Mar. 1999.

[31] 河野圭太，益田智員，木下和彦，村上孝三，"複数の QoS を考慮したサービス情報検索方式，" 電子情報通信学会論文誌，Vol.J84-B，No.3，pp. 443–451，Mar. 2001.

[32] K. Kawano, T. Masuda, K. Kinoshita, and K. Murakami, "An Efficient Method to Search for the Location of Network Services with Multiple QoSs Guarantee," in *Proceedings of the SoftCOM 2000,* pp. 665–674, Oct. 2000.

[33] 河野圭太，益田智員，木下和彦，村上孝三，"ネットワークマップにおける経路情報を利用したサービス情報検索，" 電子情報通信学会総合大会，B-7-25, Mar. 2000.

[34] 河野圭太，益田智員，木下和彦，村上孝三，"経路状況を考慮したマルチメディアサービス検索方式，" 電子情報通信学会技術研究報告，SSE2000-162，Oct. 2000.

[35] 木下和彦，神原正義，谷岡秀昭，村上孝三，"複数の QoS 要求を満たすマルチ キャストツリーを生成するリンク状態アルゴリズム，" 電子情報通信学会論文誌，Vol.J85-B, No.10, Oct. 2002.

[36] M. Kanbara, H. Tanioka, K. Kinoshita, and K. Murakami, "A Multicast Routing Algorithm for Multiple QoS Requirements," in *Proceedings of the 3rd International Network Conference (INC 2002)*, pp. 253–260, Plymouth, UK, July 2002.

[37] 神原正義，谷岡秀昭，木下和彦，村上孝三，"複数のQoSを考慮したマルチキャストルーチング方式，"電子情報通信学会総合大会，B-7-129，Mar. 2001.

[38] H. Tanioka, K. Kinoshita, and K. Murakami, "Multipoint-to-Multipoint Routing for Multimedia Communication Service," in *Proceedings of the ICC 2002*, C13-3, New York, USA, Apr. 28 – May 2 2002.

[39] 谷岡秀昭，木下和彦，村上孝三，"マルチメディア通信のための多対多ルーチング方式，"電子情報通信学会技術研究報告，SSE2000-276，Mar. 2001.

[40] N. Nankaku, and M. Yamashita, "Prospect of multimedia network," *IEICE Transactions on Communications*, Vol.E79-B, No.7, pp.899–903, July 1996.

[41] K. Koyanagi, T. Saito, T. Kaneda, and H. Ikeda, "Internetworking technologies for the multimedia network, " *IEICE Transactions on Communications,* Vol.E80-B, No.10, pp.1386–1392, Oct. 1997.

[42] 村田正幸，"マルチメディアネットワークにおけるサービス品質に関する動向と課題，"電子情報通信学会論文誌, Vol. J80-B, No.6, pp. 296–304, June 1997.

[43] C. Pornavalai, G. Chakraborty, and N. Shiratori, "Routing with multiple QoS requirments for supporting multimedia applications," *Telecommunication Systems*, No.9, pp.357–373, 1998.

[44] 鈴木滋彦，"高度インテリジェントネットワーク，"電子情報通信学会誌，Vol.77, No.4, pp. 410–415, Apr. 1994.

[45] 若原恭，"インテリジェントネットワークINの最新動向—サービス生成技術—，"電子情報通信学会誌, Vol.78, No.7, pp.715–720, July 1995.

[46] 長竹幸輝，坂井博，野原龍男，高見一正，"H.323をベースとした高度INとIP網の連携方式に関する検討，"電子情報通信学会技術研究報告，SSE2000-42, June 1999.

[47] 山下康治，諏訪祐一，岩田泉，"高度INにおけるJava/CORBAを用いたサービス制御方式の検討，"電子情報通信学会技術研究報告，NS2001-141, Nov. 2001.

[48] K. Murakami, and S. Kuroyanagi, "Overview of Photonic Switching Systems Using Time-Division and Wavelength-Division Multiplexing," *IEICE Transactions on Electronics*, Vol. E77-B, No. 2, pp.119–127, Feb. 1994.

[49] 古賀正文，高知尾昇，宮本裕，"波長多重光伝送システムの現状と将来，"電子情報通信学会誌， Vol. 83, No. 7, pp. 569–575, July 2000.

[50] 菊池和朗，“光ファイバ通信におけるスペクトル拡散技術，” 電子情報通信学会技術報告，OCS93-71, Mar. 1994.

[51] I. Saeki, S. Nishi, and K. Murakami, “All-optical code division multiplexing switching network based on self-routing principle,” *IEICE Transactions on Communications*, Vol. E82-B, No. 2, pp. 239–245, Feb. 1999.

[52] 松井知章，河野圭太，木下和彦，村上孝三，“ホロニックネットワークにおける端末の移動を考慮した経路切り替え方式，” 電子情報通信学会総合大会，B-6-196, Mar. 2002.

[53] K. Kawano, K. Kinoshita, and K. Murakami: “A Mobility-Based Terminal Managemant in IPv6 Networks,” *IEICE Transactions on Communications*, Vol.E85-B, No.10, Oct. 2002.

[54] K. Kawano, K. Kinoshita, and K. Murakami: “A Multilevel Hierarchical Distributed IP Mobility Management Scheme for Wide Area Networks,” *Proceedings of the 2002 International Conference on Computer Communications and Networks (ICCCN 2002)*, pp. 480–484, Miami, USA, Oct. 2002.

[55] K. Kinoshita, M. Fujisawa, T. Takine, and K. Murakami, “Performance Evaluation Method of Cellular Networks with Retrial,” *Proceedings of the 1999 IEEE International Conference on Personal Wireless Communications (ICPWC '99)*, pp. 389–393, Jaipur, India, Feb. 1999.

[56] M. Oshima, G. Kajoth, and K. Ono, Aglets Specification 1.1 Draft, http://www.trl.ibm.co.jp/aglets/spec11.html.

[57] I. Satoh, Agent Space, http://islab.is.ocha.ac.jp/agent/.

[58] General Magic Inc., “Mobile Agents White Paper,” http://www.generalmagic.com/technology/techwhitepaper.html.

[59] ObjectSpace, Voyager, http://www.objectspace.com/developers/voyager/white/VoyagerTechOview.pdf.

[60] D. Gavalas, D. Greenwood, M. Ghanbari, and M. O'Mahony, “An Infrastructure for Distibuted and Dynamic Network Management based on Mobile Agent Technology,” *Proceeding of the ICC '99*, pp. 1362–1366, Vancouver, Canada, June 1999.

[61] F. G. Chatzipapadopoulos, M. K. Perdikeas, and I. S. Venieris, “Mobile Agent and CORBA Technologies in the Broadband Intelligent Network,” *IEEE Communications Magazine*, Vol. 38, No. 6, pp. 116–124, June 2000.

[62] 渡辺君人，木下和彦，山井成良，村上孝三，“Agent Gateway: 多面論理ネットワークによる位置透過性を考慮したモバイルエージェントプラットフォーム，” 電子情報通信学会技術研究報告，SSE99-124，Dec. 1999.

[63] 渡辺君人，木下和彦，山井成良，村上孝三，“モバイルエージェント復元のための分散オブジェクトデータベース，” 電子情報通信学会総合大会，B-7-54, Mar. 2000.

[64] 渡辺君人，永松孝之，木下和彦，山井成良，村上孝三，“多面的論理ネットワークによる位置透過性を考慮したエージェントシステム Agent Gateway の提案，” 電子情報通信学会ソフトウェアエージェントとその応用ワークショップ（SAA2000），pp. 53–60, Nov. 2000.

[65] A. S. Tanenbaum, *Computer Networks Third Edition,* Prentice-Hall, 1996.

[66] 永松孝之，渡辺君人，木下和彦，山井成良，村上孝三，“エージェント間通信のための広域ルーチング方式，” 電子情報通信学会第2種研究会・ネットワーキングアーキテクチャワークショップ，pp. 77–82, Feb. 2000.

[67] T. Nagamatsu, K. Kinoshita, N. Yamai, and K. Murakami, “A Fast and Efficient Routing Method for Inter-Agent Communications,”in *Proceedings of the 2000 IASTED International Conference on Software Engineering and Applications (SEA 2000),* pp. 140–145, Las Vegas, USA, Nov. 2000.

[68] 徳丸裕章，木下和彦，山井成良，村上孝三，“エージェント間通信における迂回ルーチング方式，” 電子情報通信学会総合大会，B-7-60，Mar. 2001.

[69] H. Tokumaru, K. Kinoshita, N. Yamai, and K. Murakami, “A Fast and Efficient Dynamic Routing for Inter-Agent Communications,” in *Proceedings of the International Conferences on Info-tech & Info-net,* Conference D, pp. 73–78, Beijing, China, Oct. 29 - Nov. 1 2001.

[70] 細川晃，木下和彦，山井成良，村上孝三，“端末の移動を考慮したユーザ／エージェント間通信方式，” 電子情報通信学会総合大会，B-6-137, Mar. 2002.

[71] A. Viswanathan, et al., “Evolution of Multiprotocol Label Switching,” *IEEE Communication Magagine,* pp. 165–173, May 1998.

[72] 堀田英一，“超高速ネットワークのためのルーティング技術，” 電子情報通信学会誌, Vol. 84, No. 7, pp. 471–477, July 2001.

[73] K. Psounis, “Active Networks: Applications, Security, Safety and Architectures,” *IEEE Communication Surveys,* First Quarter, 1999.

[74] 山本 幹，“アクティブネットワークの技術動向，” 電子情報通信学会論文誌, Vol.J84-B, No.8, pp.1401–1412, Aug. 2001.

[75] 木下和彦，滝根哲哉，村上孝三，寺田浩詔，“ホロニックネットワークを指向した網制御アーキテクチャ，” 電子情報通信学会第2種研究会・ネットワーキングアーキテクチャワークショップ, 4-1-1〜4-1-6, Dec. 1996.

[76] 木下和彦，谷岡秀昭，山井成良，滝根哲哉，村上孝三，“ホロニックルーチングのためのネットワークマップ構成法，” 電子情報通信学会技術報告, SSE97-62, pp.25–30, Aug. 1997.

[77] 益田智員，河野圭太，木下和彦，滝根哲哉，村上孝三，“QoS ルーチングのためのネットワークマップ構成法，” 電子情報通信学会総合大会，B-7-151, Mar. 1999.

[78] C. Huitema, *Routing in the Internet*, Prentice-Hall, NJ, 1995.

[79] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows*, Prentice-Hall, NJ, 1993.

[80] 木下和彦，益田智員，村上孝三，“大規模ネットワークにおける QoS 保証を考慮したフロー管理方式，” 電子情報通信学会総合大会，B-7-119，Mar. 2001.

[81] 平田 富夫，アルゴリズムとデータ構造，森北出版，1990.

[82] 池永全志，川原憲治，尾家祐二，“PNNI における経路選択手法の検討と性能評価，” 電子情報通信学会技術研究報告，CQ98-85, Feb. 1999.

[83] B. M. Waxman, “Routing of Multipoint Connection,” *IEEE Journal on Selected Areas in Communications,* Vol. 6, pp. 1617–1621, 1998.

[84] 吉原 貴仁, 堀内 浩規, 杉山 敬三, 小花 貞夫, “複数 QoS に基づく近似精度可変な経路選択アルゴリズム,” 電子情報通信学会技術研究報告，SSE97-89, pp. 81–86, 1997.

[85] I. Stoica, T. S. Eugene Ng, and H. Zhang, “REUNITE: a recursive unicast approach to multicast,” *Proceedings of the INFOCOM 2000*, pp. 1644–1653, Tel Aviv, Israel, Mar. 2000.

[86] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, ”Multicasting For Multimedia Applications,” *Proceedings of the INFOCOM '92,* pp.2078–2083, New York, USA, May 1992.

[87] H. F. Salama, D. S. Reeves, and Y. Viniotis, “Evaluation of multicast routing algorithms for real-time communication on high-speed networks,” *IEEE Journal on Selected Areas in Communications,* Vol.15, No.3, pp. 332–345, Apr. 1997.

[88] H. Tode, Y. Sakai, M. Yamamoto, H. Okada, and Y. Tezuka, ”Multicast Routing Algorithm for Nodal Load Balancing,” *Proceedings of the INFOCOM '92,* pp.2086–2095, New York, USA, May 1992.

[89] P. Winter, ”Steiner Problem in Networks: A Survey,” *Networks,* Vol.17, pp.129–167, 1987.

[90] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, ”Multicast Routing for Multimedia Communication,” *IEEE/ACM Transactions on Networking,* Vol.1, pp.286–292, 1993.

[91] F. Bauer, and A. Varma, "Degree-Constrained Multicasting in Point-to-Point Networks," *Proceedings of the INFOCOM '95,* pp.369–376, Boston, USA, Apr. 1995.

[92] Q. Zhu, M. Parsa, and J. J. Garcia-Luna-Aceves, "A Source-Based Algorithm for Delay-Constrained Minimum-Cost Multicasting," *Proceedings of the INFOCOM '95,* pp.377–385, Boston, USA, Apr. 1995.

[93] H. Y. Lee, and C. H. Youn, "Scalable multicast routing algorithm for delay-variation constrained minimum-cost tree," *Proceedings of the ICC 2000,* pp. 1343–1347, New Orleans, USA, June 2000.

[94] W. M. Moh, and B. Nguyen, "An Optimal QoS-Guaranteed Multicast Routing Algorithm with Dynamic Membership Support," *Proceedings of the ICC '99,* pp. 727–732, Vancouver, Canada, June 1999.

[95] S. Chen, K. Nahstedt, and Yuval Shavitt, "A QoS-Aware Multicast Routing Protocol," *IEEE Journal on Selected Areas in Communications,* Vol. 18, No. 12, pp. 2580–2592, Dec. 2000.

[96] B. Wang, and J. C. Hou, "Multicast Routing and Its QoS Extension," *IEEE Network,* pp. 22–36, Jan./Feb. 2000

[97] 木下和彦，添田純一郎，山井成良，滝根哲哉，村上孝三，"多対多通信のための経路制御とネットワーク設計法，"電子情報通信学会論文誌，Vol.J82-B，No.10，pp. 1773–1781，Oct. 1999.

[98] K. Kinoshita, J. Soeda, N. Yamai, T. Takine, and K. Murakami, "Network Design and Control for Multipoint-to-Multipoint Communications," in *Proceedings of the 1998 IEEE International Symposium on Circuits and Systems (ISCAS '98)*, vol. 6, pp. 529–532, Monterey, CA, May 31–June 3 1998.

[99] K. Kinoshita, M. Takase, N. Yamai, and K. Murakami, "Remote Program Execution Mechanism with Network Transparency on Multi-Platform Environment," in *Proceedings of the ISTN 2000,* pp. 336–339, Hangzhou, China, Nov. 2000.

[100] 神原正義，谷岡秀昭，木下和彦，村上孝三，"マルチキャスト通信における複数のQoS要求を考慮したルーチング方式，"電子情報通信学会技術研究報告，NS2001-182，Dec. 2001.