博 士 論 文

VLANイーサネットを用いた
大規模クラスタネットワークの構築に関する研究

2009年3月

慶應義塾大学大学院理工学研究科

大塚 智宏

論文要旨

汎用の PC 群を相互接続することにより構成する PC クラスタは，高性能計算 (HPC) 分野のプラットフォームにおいて近年の主流となっている．中でも，汎用 LAN でありコスト面で圧倒的に有利なイーサネット (Ethernet) を結合網に用いたクラスタシステムが多数を占めるようになっており，その傾向は今後も続くと予想される．

一方で，イーサネットを用いたクラスタの大半は，トポロジ内にループ構造を含むことができないため，単純なツリートポロジを採用している．しかし，ツリートポロジにはトラフィックがツリーのルート付近に集中しやすいという欠点があり，特に大規模なクラスタにおいては全体のパフォーマンスに大きな影響を与えかねない．

この問題の有力な解決策として，VLAN ルーティング法と呼ばれる手法が提案されている．VLAN ルーティング法は，IEEE 802.1Q 標準のタグ VLAN 技術を応用し，イーサネットにおいてループを含むさまざまなトポロジを構築できるようにする手法である．しかし，既存の VLAN ルーティング法を大規模クラスタに適用するのは，ホスト側システムソフトウェアの VLAN への対応，および必要となる VLAN 数の増加，の 2 つの問題により現時点では困難である．

本研究は，これらの問題を解決し，イーサネットを用いた大規模クラスタシステム構築のための技術を開発することを目的とする．そのために本論文では，VLAN ルーティング法を改良した 2 つの手法を提案し，中規模クラスタに適用した際の実機評価に加え，大規模クラスタへの適用可能性や既存の他の手法との比較について議論する．

1 つ目の提案手法「スイッチタグ法」では，イーサネットフレームへの VLAN タグ挿入をホストではなくスイッチにおいて行うことで，ホスト側のシステム環境への依存をなくし，通信ライブラリ等がタグ VLAN をサポートしていない場合でも VLAN ルーティング法を利用できるようにする．また，2 つ目の手法「VLAN リネーミング法」では，フレームへの VLAN タグの付加をスイッチ内でのみ行い，タグによってフレームの出力ポートを決定する目的に VLAN の使用を限定することで，必要となる VLAN 数をスイッチのポート数以下に削減する．さらに，これらの提案に加え，VLAN ルーティング法を適用したイーサネットにおいて性能を決定する要因，特にデッドロックの問題について検証し，典型的なトポロジを構築する際の VLAN 割り当て方法やルーティングアルゴリズムの選択について議論する．

提案手法の評価として，16 スイッチ・32 ホストからなるクラスタ環境に手法を適用し，基本通信性能およびアプリケーションベンチマーク性能を測定した．その結果，提案手法は導入によるオーバヘッドがほとんどなく，提案手法を用いて構築した各トポロジはすべてのアプリケーションで理想的なフラットトポロジの 88%以上という高い性能を示すことがわかった．また，典型的なトポロジに提案手法および従来の VLAN ルーティング法を適用した際に必要となる VLAN 数を一般化し，各トポロジを採用した場合に構築可能なシステム規模を導出することにより，提案手法が従来手法に比べてより大規模なネットワークに適用可能なことを確認した．これらの評価から，本論文の提案手法によって，イーサネットを用いた大規模かつ高性能なクラスタシステムを構築できることを示した．

Abstract

PC clusters, which are built by connecting commodity PCs, are the mainstream architecture in recent high-performance computing (HPC) platforms. In particular, systems that adopt highly cost-effective, standard Ethernet for their interconnects become and certainly continue to be the majority among clusters.

On the other hand, most clusters using Ethernet adopt simple trees as their network topologies, since Ethernet cannot include loops in its topology. However, a tree topology has a disadvantage in traffic congestion at its root, which may degrade overall performance especially in large-scale clusters.

VLAN-based routing method, which makes it possible to adopt various topologies including loops in Ethernet by applying IEEE 802.1Q tagging VLAN technology, is a solution to this problem. At present, however, it is difficult to apply the existing VLAN-based routing method to large-scale clusters due to the following two problems; the capability of system software to deal with VLAN tags, and the increase in required number of VLANs.

The goal of this research is to develop a technology for building large-scale clusters using Ethernet by solving these problems. For this purpose, two methods improving the existing VLAN-based routing method are proposed and evaluated in this dissertation. Also, possibility of applying these methods to large-scale clusters as well as comparison of them with other methods is discussed.

The first method called "switch-tagged method" makes it possible to employ VLAN-based routing when communication libraries do not support VLAN tagging, by inserting VLAN tags into Ethernet frames not at hosts but at switches. The second method called "VLAN renaming method" reduces required number of VLANs to the number of ports in a switch or less, by using VLANs only inside switches for determining output ports of frames tagged at input ports. In addition to these proposals, this dissertation also discusses determining factors in the performance of Ethernet with the VLAN-based methods, especially the deadlock problem, as well as methodology of VLAN assignment and selection of routing algorithms on typical topologies.

As a result of performance evaluations using a cluster with 32 hosts and 16 switches, the proposed methods introduce almost no overhead, and topologies built by using the methods achieve over 88% performance of an ideal flat topology in all applications. Moreover, it is ascertained that the proposed methods are applicable to larger networks than the original VLAN-based routing method, through generalizing required number of VLANs to obtain the maximum system scale when typical topologies are adopted by using these methods. In conclusion, it is possible to build large-scale and high-performance clusters using Ethernet by applying the proposed methods.

目 次

表 目 次

図目次

第1章　緒論

多数の汎用 PC をネットワークで相互接続することにより構成する PC クラスタは，そのコスト対ピーク性能比の高さなどにより，高性能計算 (HPC) 分野のプラットフォームにおいて近年の主流となっている．LINPACK ベンチマーク [1] によって世界のスーパーコンピュータをランキングする TOP500 プロジェクト [2] によると，2008 年 11 月のランキングにおいて，クラスタシステムは上位 500 台中 410 台と実に 82%を占めている．特に，企業が所有するシステムではより顕著で，305 台中 301 台とほぼ全てがクラスタによって占められており，コスト対効果に敏感な企業において広く採用されていることがわかる．

このような PC クラスタのノード間を相互接続するネットワークは，大きく 2 種類に分類することができる．1 つは，InfiniBand[3][4] や Myrinet[5][6]，QsNET[7][8] に代表されるシステムエリアネットワーク (SAN) である．SAN は，クラスタ向けもしくは専用に設計されたネットワークであり，Remote Direct Memory Access (RDMA) プロトコル [9] やハードウェアによるマルチキャストのサポート，低遅延のネットワークスイッチなどにより，クラスタ上での並列分散処理に必要とされる高い転送性能を提供している．また，SAN の多くは，スイッチにおいて仮想チャネルをサポートすることにより，さまざまな並列計算機向けのネットワークトポロジを構築することを可能にしている．

一方，SAN に対し，汎用のローカルエリアネットワーク (LAN) として広く普及しているイーサネット (Ethernet)[10] を結合網に用いたクラスタシステムが近年増加しており，2008 年 11 月の TOP500 ランキングでは，クラスタシステム 410 台中イーサネットを用いたクラスタが 282 台と約 7 割を占めている．イーサネットは，管理の容易さ，高い耐故障性，SAN に比べ安価なハードウェアなどの利点を持ち，特に，ギガビット・イーサネット (Gigabit Ethernet, GbE) の普及，ツイストペアケーブルを用いる 10GBASE-T の標準化 (IEEE 802.3an-2006) などにより，リンクバンド幅においては SAN に匹敵するまでになっている．また，初期の Beowulf 型クラスタと違い，最近のイーサネットを用いた PC クラスタでは，SAN で採用されてきたゼロコピー通信やワンコピー通信をサポートするシステムソフトウェア [11][12] を利用することができ，低遅延のノード間通信が可能である．

しかしながら，SAN とは異なり，イーサネットを用いた PC クラスタの大半は単純なツリー状トポロジを採用している．これは，基本的にイーサネットがループ構造を含むトポロジを許していないためである．ツリー状ネットワークにはトラフィックがツリーのルート付近に偏りやすいという欠点があるため，リンク集約化 (IEEE 802.3ad)[13] などによってルート付近のリンクを強化するのが一般的である．しかし，クラスタが大規模になると，リンク集約化だけではツリー状ネットワークの欠点を補い切れず，また，リンク集約化のためにスイッチのポートが多数占有されるという問題も生じる．一般に，SAN 等で用いられる並列計算向けのトポロジでは，ホスト間やスイッチ間に複数の経路 (パス) を設けることにより高いバイセクションバンド幅を提供している．この点において，トポロジがツリー状に限定されるイーサネットでは，リンクあたりのバンド幅は SAN と比較しても遜色がない一方で，ホスト間やスイッチ間のパスが 1 本に限られるた

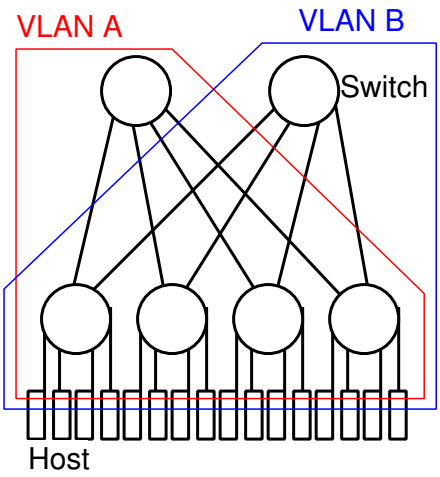


図 1.1 VLAN ルーティング法における VLAN 割り当ての例

め，SAN で実現可能なバイセクションバンド幅を提供することは困難である．これらのことから，ユーザやアプリケーションの要求に応じたトポロジやルーティングを採用している SAN や並列計算機の相互結合網に比べて，イーサネットを用いた PC クラスタはこれまで大規模化には向かないとされてきた．

この問題を解決するために，スイッチ間に複数パスを設けることでネットワークのバンド幅を向上させる方法が提案されている [14][15]．これらの方法では，ループを解消してツリー構造を維持するスパニングツリープロトコル (STP, IEEE 802.1D) を意図的に用いないようにすることで，Fat ツリーやトーラスなどのループ構造を含むトポロジの構築を可能としている．STP を用いずに大規模クラスタシステムを構築する場合，ホストの追加やスイッチの故障，操作ミス等によるブロードキャストストームの発生を回避するために，ホストの MAC アドレスの管理が 1 つの課題となる．この点において，IEEE 802.1Q 標準のタグ VLAN 技術を応用した VLAN ルーティング法 [14][16] が既存の手法の中で有力である．

VLAN ルーティング法は，IEEE 802.1Q 標準のタグ VLAN 技術を応用し，イーサネットにおいてループを含むさまざまな物理トポロジを構築できるようにする手法である．VLAN 技術は本来，同じ物理ネットワークに接続されたホストの集合を，複数の論理的なグループに分割するために用いられるが，VLAN ルーティング法ではこれをネットワークのスループット向上のために用いる．例えば，図 1.1のように，各ホストが複数の VLAN グループのメンバーになるようにしておき，各 VLAN にそれぞれ異なるリンク集合を割り当てる．ここで，各 VLAN ネットワークのトポロジはツリー構造となっているため，ブロードキャストストームは発生しない．このようにすることで，すべてのホストがどの VLAN を用いても互いに通信でき，VLAN を選択することで複数の経路を切り替えて使うことができるようになる．

しかし，既存の VLAN ルーティング法を適用して大規模な PC クラスタを構築するのは，1) ホスト側システムソフトウェアの VLAN への対応，2) VLAN 数および MAC アドレステーブルのエントリ数の制限，の 2 つの問題により現時点では困難である．工藤らは，VLAN ルーティング法を提案した論文において，VLAN ルーティング法の動作原理を示し，16 台のノードと少数のスイッチによる比較的単純なトポロジを TCP/IP ベースの通信ライブラリを用いて評価した結果を報告している [14] が，主に上に挙げた 2 つの問題により，VLAN ルーティング法を利用した数百ノード以上の大規模なクラスタはまだ構築例がなく，小規模なクラスタによる手法の有効性の評価にと

どまっているのが現状である．

また，現状では，VLAN ルーティング法を利用してクラスタを構築する際に，クラスタの利用目的に応じたトポロジを設計する方法に関する検討が不足している．一般に，SAN におけるトポロジ設計では，トポロジ形状の選択以外に，どのパスを用いてパケットを転送するかを決定するルーティングアルゴリズムの設計が性能に大きな影響を与える．イーサネットに VLAN ルーティング法を適用する場合も，ホスト間に複数パスが導入されることにより，パスの選択すなわちルーティングの方法が重要な要素となる．これに加え，VLAN ルーティング法に特有の設計要素として，構築するトポロジに VLAN をどのようにマッピングするかが大きな問題となる．VLAN のマッピングはトポロジ形状およびルーティング方法の選択と密接に関わっているため，トポロジを設計する際はこれらをまとめて検討する必要がある．

本研究は，これらの問題を解決し，イーサネットを用いた大規模 PC クラスタシステム構築のための技術を開発することを目的とする．本論文ではまず，上で述べた既存の VLAN ルーティング法の問題点を解決するために，VLAN ルーティング法を改良した 2 種類の手法を提案する．1 つ目の提案手法「スイッチタグ法」は，上記 1) の問題を解決し，ホスト上の通信ライブラリ等がフレームへの VLAN タグの挿入をサポートしていない場合にも VLAN ルーティング法を利用できるようにする．このためにスイッチタグ法では，イーサネットフレームへの VLAN タグの挿入を，ホストではなくスイッチへのフレーム入力時に行うことで，ホスト側のシステム環境への依存をなくしている．また，2 つ目の提案手法「VLAN リネーミング法」は，上記 1) に加えて 2) の問題を解決し，イーサネットを用いたクラスタの大規模化を可能とする．VLAN リネーミング法では，VLAN の使用目的をスイッチ内におけるフレームの出力ポートの決定に限定し，各スイッチへのフレーム入力時に入力ポートに応じたタグを挿入，出力時にタグを除去する．これにより，スイッチが各ホストの MAC アドレスを学習することができなくなる代わりに，スイッチ間の VLAN 設定の依存関係をなくし，必要となる VLAN 数をスイッチの使用ポート数以下に削減することができる．

さらに本論文では，現状では検討が不足している VLAN ルーティング法を用いたトポロジ設計について，提案手法を適用する場合を中心に詳細に検討する．まず，提案手法を含む VLAN ルーティング法を適用して構築したイーサネットトポロジにおいて，その性能を決定する要因を明らかにする．特に，複数パスがリンク上に重なる際の性能低下の問題と，ループ構造を含む場合のデッドロックの問題についてそれぞれ検証し，VLAN ルーティング法を適用する際のフロー制御の有効性，およびデッドロックフリールーティングの重要性について明らかにする．その上で，提案手法を用いた場合と既存の VLAN ルーティング法を用いた場合のそれぞれについて，代表的な並列計算向けトポロジの設計例を示し，各トポロジ上でのルーティングアルゴリズムの選択と VLAN 割り当ての方法，および必要 VLAN 数について検討する．

最後に，提案手法の評価として，16 スイッチ・32 ホストからなる中規模クラスタ環境に手法を適用して並列計算向けのトポロジを構築し，基本通信性能およびアプリケーションベンチマーク性能を測定する．また，各トポロジで必要となる VLAN 数をもとに，従来手法および提案手法によって構築可能なシステム規模を明らかにし，その比較を通して提案手法の大規模クラスタへの適用可能性について議論する．

本論文の以降の構成は次の通りである．まず，第 2章で既存の VLAN ルーティング法についてその概要を述べ，現状の VLAN ルーティング法の抱える問題点について明らかにする．次に，第 3章では関連研究および関連技術として，主にイーサネットを用いたクラスタを対象とした VLAN ルーティング法以外の性能改善手法を紹介する．第 4章では，本研究の提案手法であるスイッチタ

グ法および VLAN リネーミング法についての詳細を述べ，続く第 5章で，提案手法を含む VLAN ルーティング法を用いてトポロジを設計する際の検討事項について議論する．さらに，第 6章では，提案手法を中規模クラスタに適用した際の性能を評価し，大規模クラスタへの適用について検討する．最後に，第 7章で結論を述べる．

また，付録として，付録 Aに *k*-ary *n*-cube (メッシュ，トーラス) における VLAN 割り当て手法を掲載する．

第2章　VLANルーティング法

本章では，イーサネットにおいてノード間に複数の経路 (パス) を設けるための手法であり，本研究が対象とする VLAN ルーティング法について述べる．また，現状の VLAN ルーティング法の問題点についてまとめる．

2.1　VLAN ルーティング法の概要

VLAN ルーティング法 (VLAN-based Routing Method)[14][16] は，IEEE 802.1Q 標準 [17] のタグ VLAN 機能を用いることにより，レイヤ 2 イーサネットネットワーク上に複数パスを実現する手法である．VLAN ルーティング法では，ノード間あるいはスイッチ間に異なる VLAN に属する複数のパスを用意し，どの VLAN 上で通信を行うかを決定することによりパスを選択する．本手法を用いることで，イーサネット上にループ構造を含むようなトポロジを構築することが可能となる．

2.1.1　VLAN を用いた複数パスの実現

図 2.1に示すように，複数のノードが接続されたスイッチ間を複数のパスで接続し，ノード毎にパスを選択すれば，それぞれのスイッチに接続されたノード群同士の通信バンド幅は，パスが 1 つしかない場合に比べてパス数倍になる．しかし，このように複数パスを持つネットワークにはループが存在する．レイヤ 2 イーサネットでは，ネットワーク中にループが存在するとブロードキャストストームが発生し，ネットワークはダウンして使用不能となる．

このような状況を避けるため，イーサネットスイッチのほとんどはスパニングツリープロトコル (STP)[18] をサポートしている．STP では，ネットワーク中のループを検出し，ループを構成するリンクのうち 1 本を未使用とすることで，ネットワークを木構造トポロジに保つ．このため，イーサネットにおいて図 2.1のようにネットワークを構成しても，実際には複数パスを利用することはできない．

VLAN ルーティング法では，この問題を VLAN を用いることにより解決する．VLAN は，物理的なネットワーク上に仮想的な複数のネットワーク (それぞれを VLAN と呼ぶ) を構築する技術である．異なる VLAN 間ではイーサネットフレームは伝搬しないため，物理的なネットワークがループを含んでいても，それぞれの VLAN が木構造トポロジを保っている限りブロードキャストストームは発生しない．この性質を用いて，図 2.2に示すように，ノードやスイッチ間の複数パスにそれぞれ異なる VLAN を割り当てれば，いずれの VLAN を用いて通信を行うかを選択することにより，複数のパスを使い分けることができるようになる．

VLAN は通常，単一の物理ネットワーク上に複数の仮想ネットワーク (VLAN) を構成し，それぞれの VLAN を独立したネットワークセグメントとして利用するために用いられる．これに対して，VLAN ルーティング法では複数パスを構成するために VLAN を用いており，基本的にいずれ

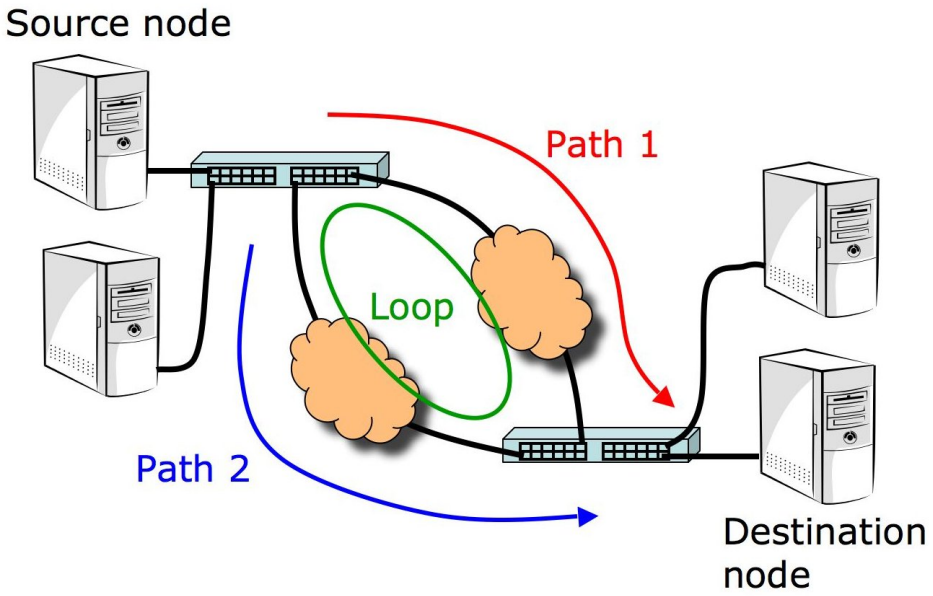


図 2.1　スイッチ間に複数パスを持つネットワーク

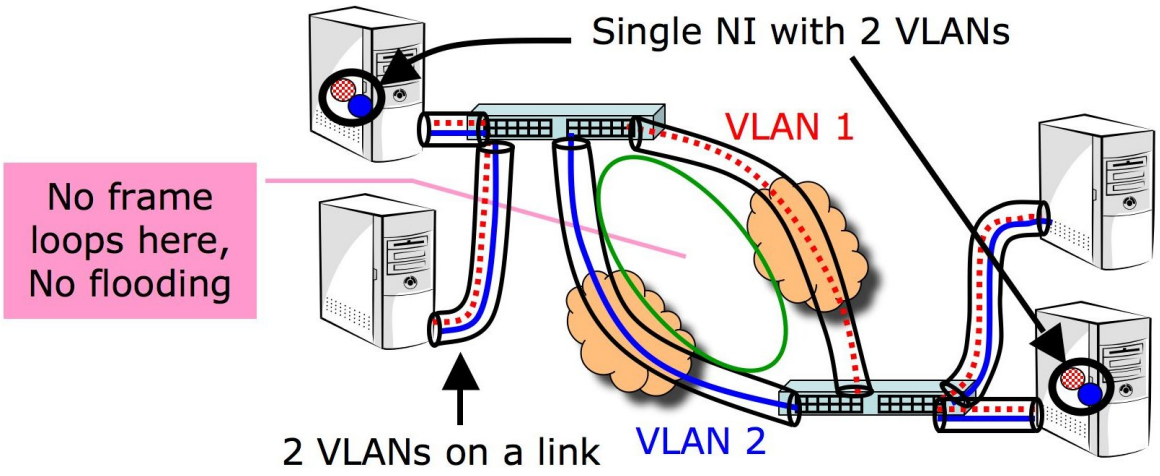


図 2.2　VLAN ルーティング法による複数パスの実現

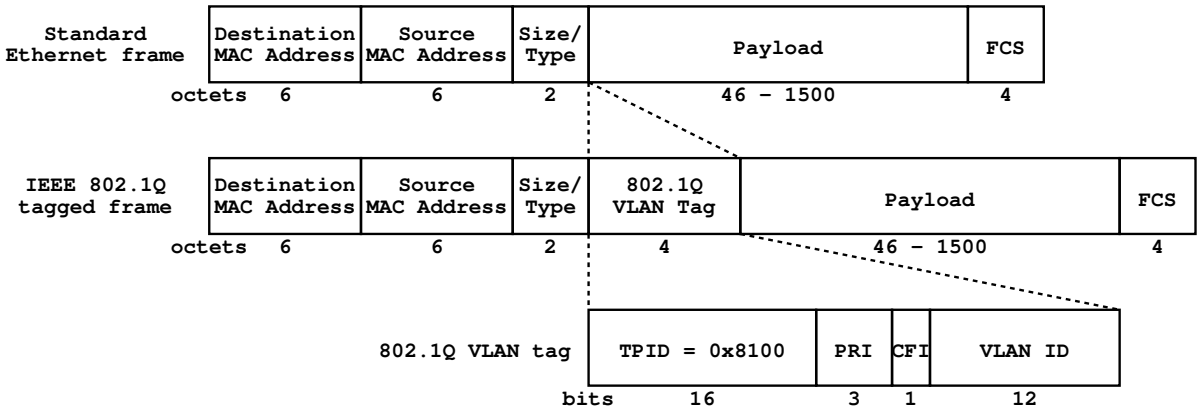


図 2.3　IEEE 802.3 イーサネットフレームのフォーマット

のノードも他のすべてのノードとレイヤ 2 で通信を行うことができる．

2.1.2　VLAN 対応スイッチの動作

VLAN をサポートするために，イーサネットフレームには VLAN タグと呼ばれるフィールドが付加される．IEEE 802.1Q 標準による VLAN (タグ VLAN) の規定では，イーサネットフレームにはタグ付きのものとタグなしのものがあり，タグなしのフレームは通常のイーサネットフレームである (図 2.3上)．一方，タグ付きのフレームには 4 オクテットの VLAN タグが挿入され，そのうち 12 ビットを使用する VLAN 番号 (VLAN ID) フィールドによって，フレームが属している VLAN が決定される (図 2.3下)．同一リンクに単一の VLAN しか割り当てられない場合にはタグなしフレームを使うことができるが，複数の VLAN を割り当てる場合には，識別のためにタグ付きフレームを用いる必要がある．

IEEE 802.1Q タグ VLAN をサポートするイーサネットスイッチでは，各ポートに対する VLAN の設定項目として以下の 2 種類がある．

- デフォルトの VLAN ID (ポート VLAN ID，PVID)
- 各 VLAN のメンバ情報

まず，イーサネットフレームがスイッチのポートに入力された際，そのフレームがスイッチ内でどの VLAN に属するかは以下のように決定される．

- タグなしフレームの場合，その入力ポートに設定された PVID を VLAN ID とするタグが挿入され，PVID で示される VLAN に属すると見なされる．
- タグ付きフレームの場合，タグはそのまま維持され，タグ内の VLAN ID フィールドで示される VLAN に属すると見なされる．

スイッチは，登録されている各 VLAN について，どのポートが VLAN のメンバとなっているかを示す情報 (メンバセット) を保持している．ここで，VLAN のメンバ情報には以下の 3 種類がある．

- “タグ付き” (tagged) メンバ

- “タグなし” (untagged) メンバ
- 非メンバ (VLAN に所属しない)

スイッチ内部では，入力された各フレームは属する VLAN の (tagged または untagged の) メンバとなっているポートにのみ転送され，非メンバであるポートには決してフレームは転送されない．

一方，フレームがスイッチのポートから出力される際，出力フレームがタグ付きフレームとなるかタグなしフレームとなるかは以下のように決定される．

- 出力ポートが VLAN の tagged メンバである場合，出力フレームはタグ付きフレームとなる (VLAN タグはそのまま維持される)．
- ポートが VLAN の untagged メンバである場合，出力フレームはタグなしフレームとなる (タグは除去される)．

VLAN 設定を全く行っていないスイッチでは，以下の状態となっている．

- すべてのポートの PVID が 1 である．
- すべてのポートが VLAN #1 の untagged メンバである．

この状態のスイッチに対してタグなしフレームを送受信する場合，各フレームはスイッチ内部では仮想的に VLAN ID を 1 とするタグが挿入された状態で転送される．実際のスイッチ内部での VLAN の扱いは実装に依存するが，subsec:tag-overhead 節で述べる実験結果の通り，VLAN を用いた場合と用いない場合とで帯域や遅延に違いはほとんど生じない．

なお，通常，ポートの PVID として登録される VLAN のメンバセットにはそのポートが含まれる．すなわち，各ポートは，PVID で示される VLAN の tagged または untagged のメンバとなっている必要がある(注 1)．

2.1.3 ノードの設定

VLAN ルーティング法を利用する場合，スイッチやリンクだけでなく各ノードも VLAN に所属させる必要がある．一般に，VLAN をサポートするオペレーティングシステムでは，VLAN が割り当てられたリンクに接続されている物理ネットワークインタフェース上に，各 VLAN に対応する仮想インタフェースを作成し，仮想インタフェースを用いてデータを送受信することで VLAN に参加する．

例えば，Linux オペレーティングシステムでは，物理インタフェース `eth0` に対し，VLAN ID 2 および 3 に対応する仮想インタフェース `eth0.2`，`eth0.3` を作成することができる．これらの仮想インタフェースはアプリケーションからは物理インタフェースと同様に扱うことが可能であり，仮想インタフェース経由で送信されるイーサネットフレームはすべて対応する VLAN ID でタグ付けされた上で物理インタフェースから送信される．逆に，物理インタフェース上で受信されたタグ付きフレームは，タグ内の VLAN ID に対応する仮想インタフェース経由で受信される．仮

(注 1)PVID の VLAN を非メンバとする設定も可能ではあるが，その場合，入力フレームの VLAN ID をチェックするイングレスフィルタを無効とする必要がある．入力フレームが属する VLAN のメンバセットに入力ポートが含まれていない (非メンバである) 場合，イングレスフィルタによってフレームは破棄される．タグなしフレームは PVID の VLAN に属すると見なされるため，イングレスフィルタを無効にしない限り入力時に破棄される．

想インタフェースの MAC アドレスは物理インタフェースの MAC アドレスと同じであるが，それぞれの仮想インタフェースには異なる IP アドレスを割り当てることができる．

各ノードは，そのノードがデータを送受信する可能性がある VLAN すべてについて仮想インタフェースを持つ必要がある．その上で，送信側ノードは，どの仮想インタフェースから送信するかを選択することで用いる VLAN を決定する．

PM/Ethernet[19][12] 等の軽量通信ライブラリで用いられるようなレイヤ 2 でデータを送受信する方式では，送信する際に用いる仮想インタフェースにより VLAN を選択し，宛先ノードは MAC アドレスによって指定すればよい．ただし，現在の PM/Ethernet の実装では，宛先によって用いるインタフェースを選択する機能がないため，拡張が必要となる．

一方，IP を用いた通信では，仮想インタフェースにそれぞれ異なる IP アドレスを持たせておくことで，IP アドレスのみで用いる VLAN と宛先ノードを指定することができる．VLAN ID ごとにネットワークセグメントを分離しておけば，ルーティングテーブルの設定も VLAN 数分のエントリだけで済む．

2.1.4 VLAN ルーティングの振舞い

各 VLAN 内では，イーサネットフレームは通常のイーサネットの仕組みに従って転送される．イーサネットでは，フレーム中の宛先 MAC アドレスフィールド (図 2.3) によってフレームの宛先が指定される．ネットワーク内のパス上にスイッチ (ブリッジデバイス) が存在する場合，スイッチは自身が持つ MAC アドレステーブルにより，受け取ったフレームをどのポートに出力すればよいかを知る．

通常，MAC アドレステーブルへのエントリの登録は，学習によって自動的に行われる．まず，フレームを受信した際，スイッチはその送信元 MAC アドレスを参照し，フレームの VLAN ID および入力されたポート番号とともに MAC アドレステーブルに登録する．次に，宛先 MAC アドレスを参照し，テーブルを引いてそのアドレスのエントリがあるかどうかを調べる．エントリが見つかった場合，登録されているポートからフレームが出力される．一方，エントリが見つからなかった場合，フレームは所属する VLAN のメンバとなっているすべてのポートから出力される (これをフラッディングと呼ぶ(注 2))．この機能により，宛先 MAC アドレスを持つデバイスがその VLAN 上にあれば，最終的にフレームはそのデバイスへ確実に到達する．この宛先となっていた MAC アドレスのエントリは，宛先ホストからの返信フレームを受信した際に登録されるため，以後はフラッディングを伴わずにフレームの交換が実現されるようになる．

Myrinet[5][6] などの SAN では，送信元が通信パスを指定するソースルーティングが用いられる場合が多い．ソースルーティングでは，送信側が任意のパスを選択することが可能である．これに対して VLAN ルーティング法では，VLAN によって送信フレームが通過可能な部分ネットワーク (VLAN) が決定され，その VLAN の中ではスイッチの学習機能によってルーティングが行われる．従って，あるパスでデータを送信したい場合，そのパス全体を含む VLAN を作成しておけばよい．ただし，次節で述べる通り利用できる VLAN の数は限られるため，全体として少ない VLAN 数で効率よくトラフィックを分散できるようなトポロジおよび VLAN 構成が望ましい．

SAN 同様，VLAN ルーティング法においても同一の物理的なネットワークトポロジ上で複数のルーティングが考えうる．上記の制約に従って，構築するクラスタシステムに適したトポロジとルーティング方法を選択する必要がある．

(注 2) ブロードキャストストームは，マルチキャストフレームの送信時以外にもこのフラッディングが原因で発生する．

2.2 現状における VLAN ルーティング法の問題点

VLAN ルーティング法は，イーサネットにおいてホスト間に複数パスを導入可能であるなど，ループを含むさまざまなトポロジを構築できることから，イーサネットを用いた大規模クラスタの実現手段として有力である．しかし，現状では，VLAN ルーティング法を適用して実際に大規模な PC クラスタを構築するのは，以下の 2 つの問題により困難である．

1. ノード間通信に用いられる通信ライブラリの VLAN への対応
2. VLAN 数および MAC アドレステーブルのエントリ数の制限

まず，1. の問題は，ホスト側のシステムソフトウェアが VLAN に対応していない場合があることによる．工藤らは，VLAN ルーティング法を提案した論文において，VLAN ルーティング法の動作原理を示し，少数のスイッチによる比較的単純なトポロジを TCP/IP ベースの通信ライブラリを用いて評価した結果を報告している [14]．この評価において，経路を選択するための VLAN 番号 (VLAN ID) は，ID に関連付けられた仮想インタフェースの IP アドレスによって示される．

一方，現在のイーサネットを用いた高性能 PC クラスタは，TCP/IP をバイパスする軽量な通信ライブラリを利用している場合が多い．しかし，残念ながら，このようなシステムソフトウェアは通常イーサネットフレームの VLAN タグ付けをサポートしていないため，VLAN ルーティング法を利用するにはライブラリに手を加える必要があるという問題点がある．TCP/IP であれば VLAN に対応している場合がほとんどであるが，TCP はクラスタ上の並列アプリケーションおいてノード間通信に用いるには遅延が大きく，パフォーマンス上問題となる場合が多い．

また，IP を使用する場合，VLAN ごとの仮想インタフェースにそれぞれ IP アドレスを割り振る必要があるが，MPI 等の IP 上の通信ライブラリの実装では，ホストの指定に IP アドレス (またはそれに対応するホスト名) を用いることが多く，同一のホストに複数の IP アドレス (ホスト名) がある状態を扱うのが難しいという大きな問題がある．

次に 2. の問題について，既存の VLAN ルーティング法を適用するために必要となる VLAN 数は，ネットワークの規模に応じて増加するため，VLAN 数が大規模クラスタを構築する場合の制限要因となりうる．図 2.3に示した通り，IEEE 802.1Q の規定では VLAN タグ内の VLAN ID フィールドは 12 ビットであり，0 と 4,095 は特殊な用途に予約されているため，識別できる VLAN 数は 4,094 ($2^{12} - 2$) 個である．しかしながら，コストパフォーマンスの高い安価な商用スイッチでは，数十〜数百個程度の VLAN しかサポートしていないものも多い．

さらに，VLAN 数が増加するにつれ，その管理も複雑になる．特に，スイッチ内で静的または (学習により) 動的に登録されるホストの MAC アドレスは，同じアドレスであっても VLAN ごとに別々に登録される．そのため，VLAN 数が増加した場合，MAC アドレステーブルのエントリが不足するという状態になりかねない．

これらの理由により，VLAN ルーティング法を利用した大規模なクラスタはまだ構築例がなく，小規模なクラスタによる手法の有効性の評価にとどまっているのが現状である．本研究では，これらの問題を解決し，イーサネットを用いて大規模 PC クラスタのインタコネクトを構築するための手法を確立することを目的とする．

第3章　関連研究・関連技術

本章では，関連研究および関連技術として，主にイーサネットを用いたクラスタを対象としたVLAN ルーティング法以外の性能改善手法について紹介し，VLAN ルーティング法との比較について述べる．

3.1　リンク集約化

リンク集約化 (link aggregation) は，主にイーサネットにおいて図 3.1に示すように 2 台のネットワーク機器間を複数の物理リンクで接続し，論理的に 1 本のリンクとして扱えるようにする技術であり，複数リンクへの負荷分散による帯域幅の増加の他，伝送路の冗長性を確保するために用いられる．ポートトランキング (port trunking)，NIC ボンディング (bonding)，NIC チーミング (teaming) などの別名があるが，すべて基本的には同じものを指す用語である．

以前はネットワーク機器のベンダによりさまざまな実装およびプロトコルが存在し，互換性の問題が大きかったが，2000 年に IEEE 802.3ad[13] として標準化されて以降は互換性の問題は解消されつつある．IEEE 802.3ad リンク集約化標準では，LACP (Link Aggregation Control Protocol) と呼ばれるプロトコルが規定されており，LACP によって機器間の調停を行い，集約化されたリンクを設定することが想定されている．

複数リンクへのトラフィックの分散方法としては，送信側および受信側の MAC アドレスや IP アドレス，ポート番号，(スイッチの) 入力ポートなどのハッシュを用いるのが一般的である．ラウンドロビンが用いられる場合もあるが，フレームの順序入れ替えが発生する場合があるため一般的ではない．そのため，トラフィックのパターンによる性能のばらつきが大きく，必ずしもリンク本数分の帯域幅を得られるわけではない．

リンク集約化は，イーサネットのスイッチ間やノード-スイッチ間リンクの帯域を強化する目的で使うことが可能であり，ツリー状ネットワークの欠点を補う手段として用いられることも多い．しかしながら，クラスタが大規模になると，特にツリーのルート付近に大量のトラフィックが集中するため，リンク集約化だけではツリー状ネットワークの欠点を補い切れない．また，リンク集約化のためにスイッチのポートを多数占有するため，次数が低く直径の大きなトポロジしか構築できないという問題も生じる．

なお，リンク集約化と VLAN ルーティング法は相反する技術ではなく，集約化したリンクを VLAN トポロジを構成する 1 本のリンクとして設定することで，相互補完的に用いることが可能である．

3.2　レイヤ 3 ルーティングを用いる方法

レイヤ 3 でのルーティングを用いれば，VLAN ルーティング法を用いたレイヤ 2 イーサネットネットワークと同様に，ループを含むトポロジを構築することができる．Cisco Systems 社 [20] や

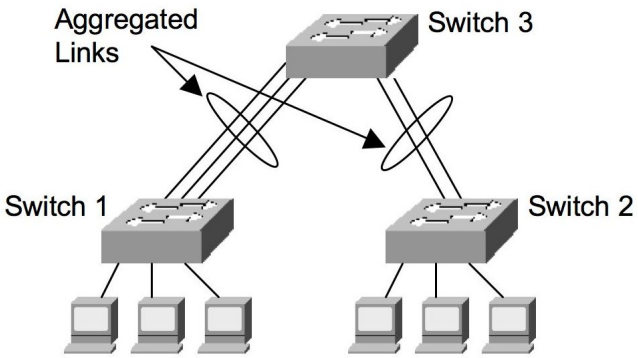


図 3.1　スイッチ間のリンク集約化

Force10 Networks 社 [21]，Fulcrum Microsystems 社 [22] などは，レイヤ 3 スイッチを用いた Fat ツリーおよび Clos 網 [23] 状のネットワークトポロジをクラスタ用に提案している．これらのレイヤ 3 ネットワークでは，データ送信時のパスは送信ノードと受信ノードの組み合わせにより決定され，実行時に送信側ノードがパスを明示的に選択することを想定していない．ただし，VLAN ルーティング法と同様に，仮想ネットワークインタフェースにより単一の物理ネットワークインタフェースに複数の IP アドレスを持たせることにより，明示的にパスを選択することも可能である．

レイヤ 3 ルーティングを用いることにより，VLAN ルーティング法と同等の並列計算向けトポロジを構築できると考えられる．しかし，レイヤ 3 ルーティングには以下のような問題点がある．

1. 一般に，レイヤ 3 ルーティングをサポートするスイッチは，レイヤ 2 スイッチングのみをサポートするスイッチと比べ高価である．
2. レイヤ 3 ルーティングは，レイヤ 2 スイッチングに比べてオーバヘッドが大きい場合が多い．
3. レイヤ 3 ルーティングの設定によって Fat ツリーやハイパークロスバ網のような複雑なトポロジを構築する場合，スイッチの設定等が煩雑になると予想される．

このうち 3. については，レイヤ 3 ルーティングを用いる場合，スイッチの各ポートに対しそのポートにルーティングされるノードの IP アドレスを登録しなければならない．すなわち，各ノードのネットワークインタフェースに (複数の) IP アドレスを割り当てた上で，各アドレス宛のトラフィックが通過する可能性のある全スイッチの全出力ポートにそのアドレスを登録する必要がある．これに対し，VLAN ルーティング法を用いた場合は，スイッチに対する VLAN の設定さえ行っておけば，2.1.4節で述べたレイヤ 2 スイッチのアドレス学習機能により，MAC アドレステーブルは最初にフレーム交換を行った際に自動的に学習される．このため，個々の IP アドレス (MAC アドレス) ごとのルーティングを設定する必要はなく，レイヤ 3 ルーティングを用いる場合と比べて設定が簡単である．

3.3　レイヤ 2 イーサネット上のルーティングに関する研究

レイヤ 2 イーサネットにおいて，スパニングツリープロトコル (STP) を用いずにデッドロックフリーを保証する高性能なルーティングアルゴリズムについての研究も行われている．

Pellegrini らの研究 [24] では，Up*/Down*ルーティング (5.4.4節参照) を改良した Tree-Based Turn-Prohibition (TBTP) と呼ばれるアルゴリズムを提案している．TBTP では，スパニングツリーをもとに禁止ターンを決定していくことで，Up*/Down*ルーティングよりも少ない禁止ターン数で循環除去を行い，デッドロックフリーを保証することができる．

Mejia らの研究 [25][26] では，メッシュ・トーラス網向けに Segment-based Routing (SR) と呼ばれるルーティングアルゴリズムを提案している．SR では，トポロジを subnet および subnet をさらに分割した segment と呼ばれる部分に分割し，各 segment 内にただ 1 つの禁止ターンを設けることによって循環を除去し，トポロジ全体のデッドロックフリーを保証することができる．なお，SR はメッシュ・トーラス向けとされているが，実際には任意のトポロジに適用可能であり，メッシュ・トーラス内の一部のリンクやスイッチが欠落 (故障を想定) しているようなトポロジにおいて高い性能を発揮する．

Reinemo らの研究 [27][28] では，VLAN タグ (図 2.3) 内の優先度フィールドを仮想チャネルの番号として用いることで，LASH ルーティング [29] 等の仮想チャネルの使用を前提としたルーティングアルゴリズムをイーサネットに適用することを提案している．さらに，リンクレベルフロー制御 (5.1節参照) で用いられる PAUSE フレームのフォーマットに優先度のフィールドを加えることにより，デッドロックフリーなロスレスネットワークをイーサネットで実現できるとしている．

これらの手法を用いることで，STP によるトポロジの制限を回避し，VLAN ルーティング法と同様にレイヤ 2 イーサネットにおいてループを含むトポロジを構築可能になると考えられる．しかし，評価についてはすべてシミュレーションによって行われており，どのようにイーサネット上に実現するかについては明らかでない部分が多い．

3.4 Viking

Sharma らによって開発された Viking[15] は，VLAN ルーティング法と同様，イーサネットにおいて VLAN を用いてホスト間に複数のパスを設定する手法をベースとしたシステムである．Viking は VLAN ルーティング法とほぼ同時期 (2004 年) に提案されたが，VLAN ルーティング法がクラスタの内部ネットワークを対象とし，並列処理向けのトポロジを構築することを目的としているのに対して，Viking はメトロポリタンエリアネットワーク (MAN) 上の広域イーサネットを主な対象としており，トポロジについては特に限定をせず，不規則なトポロジ上での複数パスの利用を目的とする点で異なる．

Viking では，Viking Node Controller (VNC) と呼ばれるクライアントと Viking Manager (VM) と呼ばれるサーバが各ホスト上で起動される．そして，VNC と VM が協調してトラフィックをモニタリング・解析し，それをもとに動的なパス (VLAN) 選択を行うことによって，トラフィックの最適な分散や耐障害性を実現する．また，現在 Lin らによって開発されている後継システムである Viking2[30] では，ノードの追加・削除などによってネットワーク構成が変更されたり，トラフィックパターンが大幅に変化した場合に対応するため，VLAN 構成を動的に変更できるようになるとされている．

3.5 PACS-CS

PACS-CS[31][32] は，筑波大学計算科学研究センターに設置されているクラスタ型スーパーコンピュータであり，2006 年 6 月の TOP500 ランキングにおいて 34 位にランクインしたシステム

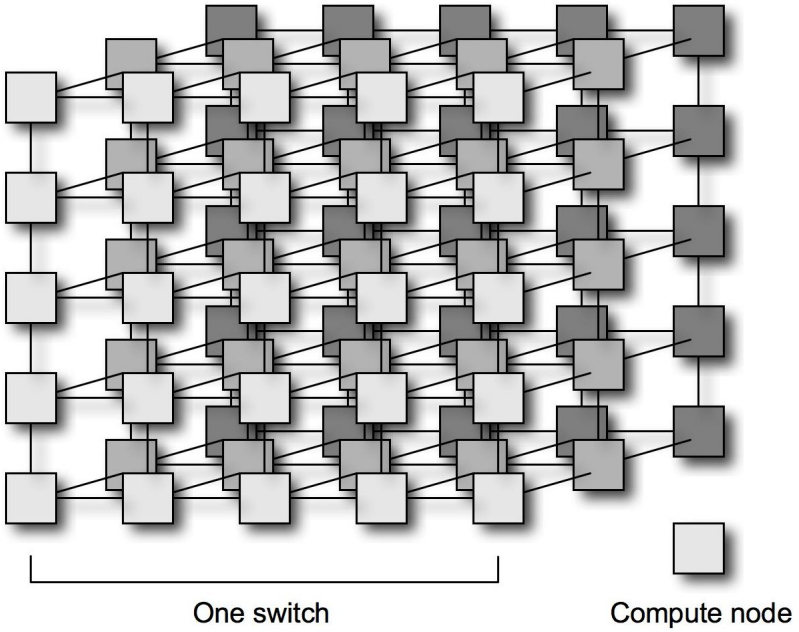


図 3.2　PACS-CS における 3 次元ハイパークロスバ網 ($5 \times 5 \times 3$)

である．

PACS-CS では，汎用プロセッサを搭載した計算ノード間を接続するネットワークとして，図 3.2に示すようなイーサネットによる 3 次元ハイパークロスバ網を構築し，ゼロコピー通信などを実現する専用通信ライブラリ PM/Ethernet-HXB[33][34] を新たに開発している．図 3.2において，各計算ノードは 3 つの次元それぞれに対応するネットワークインタフェースを持ち，各次元に連なる 1 台のスイッチ (図では 1 本の線として表現) に接続されている．なお，VLAN ルーティング法によるハイパークロスバ網 [14] では節点において各次元方向のスイッチを直接接続するが，PM/Ethernet-HXB では，ホスト上の PM ドライバが複数 NIC 間でのイーサネットフレームの中継 (ルーティング) を担当する点が異なる．

VLAN ルーティング法と比較した場合，PM/Ethernet-HXB は PACS-CS で実現されている 3 次元ハイパークロスバ網での利用を前提としており，ハイパークロスバ網以外のトポロジでは使用できない．これに対し，VLAN ルーティング法はハイパークロスバ網を含むさまざまなトポロジを構築可能な手法であり，汎用性の面で優れていると言える．

3.6 Data Center Ethernet

Data Center Ethernet (DCE) は，Converged Enhanced Ethernet (CEE) または Data Center Bridging (DCB)[35] とも呼ばれ，IBM や Cisco Systems 等が提唱し，現在 IEEE 等で 10 ギガビット・イーサネットの拡張仕様として標準化作業が進められている次世代イーサネットのアーキテクチャである．既存のイーサネットアーキテクチャでは実現できない高信頼・高性能な通信を提供し，Fibre Channel over Ethernet (FCoE)[36] や InfiniBand over Ethernet (IBoE) などのカプセル化プロトコルを使用することによって，主にサーバ・ストレージ分野において LAN および SAN (Fibre Channel, InfiniBand 等) のインタフェースを統合するのが DCE の目的である．Cisco Systems の Nexus シ

リーズや Woven Systems[37] の Ethernet Fabric など，すでに DCE の一部機能のサポートを謳う製品が 2008 年頃より登場している．

DCE は，主に以下の各機能からなる．

輻輳通知 (Congestion Notification, CN) [38]　Fibre Channel 等，輻輳制御機構を持たないプロトコル向けに End-to-End の輻輳制御を提供するもので，トラフィックの発信元で帯域制限を行う．TCP 等の輻輳制御機構を実装しているプロトコルにおいても，より適切な輻輳制御を行えるようになることが期待される．

拡張伝送選択 (Enhanced Transmission Selection, ETS) [39]　優先度グルーピングとも呼ばれ，各優先度のトラフィック内に複数のトラフィッククラスを作成し，特定のクラスに帯域幅を割り当てる等の処理の差別化を実現するためのフレームワークである．

優先度ベースフロー制御 (Priority-based Flow Control, PFC) [40]　優先度ごとに独立して制御されるリンクレベルフロー制御 (5.1節参照) を提供する．この機構により，最終的にはイーサネットにおいてパケットロスのない (ロスレス) 通信を提供するとされている．

ブリッジデータ交換 (Data Center Bridging Exchange Protocol, DCBX)　CN や PFC，トラフィッククラス等のパラメータを近隣デバイスやスイッチと交換し，ネットワーク全体の整合性を保つためのプロトコルであり，LLDP (Link Layer Discovery Protocol)[41] の拡張として標準化される予定である．

マルチパスルーティング　VLAN ルーティング法と同様，レイヤ 2 イーサネット上で複数パスを用いた最短パスルーティングを行うためのフレームワークである．

このうちマルチパスルーティングについては，現在 IEEE と IETF でそれぞれ以下の仕様の策定が同時に進められている．

Shortest Path Bridging (SPB) [42]　互いに MAC アドレステーブルを共有可能な複数のトポロジを用いて最短パスでの L2 フォワーディングを行うもので，IEEE 802.1aq として標準化作業が行われている．スパニングツリープロトコル (STP) を用いる方法や，リンクステート型プロトコルである IS-IS を用いて動的ルーティングを行う方法などが提案されている．仕様策定段階のため変更される可能性もあるが，STP を用いる方法はトポロジの識別に VLAN ID を用いることから VLAN ルーティング法と類似した手法である．

Transparent Interconnection of Lots of Links (TRILL) [43]　ルータ (レイヤ 3) とブリッジ (レイヤ 2) の両方の特徴を持つ RBridge という実体を定義し，手動での設定なしに RBridge による動的ルーティングが実現されるもので，IETF へ提案され現在議論されれている．ルーティングプロトコルとしては IS-IS が用いられる予定である．

現在の DCE は主にサーバおよびストレージをターゲットとしているが，PFC によるロスレス通信や，マルチパスルーティングにより複数の最短パスを提供可能，など点からクラスタのノード間通信用インターコネクトとしても有望である．ただし，クラスタ用インターコネクトとしてはすでに InfiniBand および (通常の) イーサネットが十分な採用実績を誇っており，高性能な InfiniBand，低コストなイーサネット，という棲み分けがなされている状況であるため，DCE がこれらのシェアを奪うには，InfiniBand に比べてかなりの低価格で提供されるようになる必要があると考えられる．

3.7 まとめ

本章では，クラスタ内ネットワーク向けを中心としたイーサネットの性能改善手法についてまとめた．いずれの手法も，イーサネットにおいてホストやスイッチ間に複数のパスを設けることでトラフィックを分散させ，ネットワークのバンド幅向上を図っている点では共通しており，イーサネットの性能向上の鍵がツリー状トポロジの制約の打破にあることが示されている．

その中でも，Data Center Ethernet (DCE) は，イーサネットのアーキテクチャに SAN で用いられる技術を導入することよって SAN との統合を進めようとしており，近い将来クラスタのノード間通信用ネットワークとしても使用されるようになる可能性が高い．現在はまだ標準化作業が行われている段階であるが，すでに各ベンダから製品が出荷され始めており，今後の動向が注目される．

第4章　提案手法

本章では，現状の VLAN ルーティング法の持つ問題点を解決するための方法として，VLAN ルーティング法を改良した 2 種類の手法を提案する．

1 つ目の提案手法「スイッチタグ法」では，イーサネットフレームへの VLAN タグ挿入をホストではなくスイッチにおいて行うことで，ホスト側のシステム環境への依存をなくし，通信ライブラリ等がタグ VLAN をサポートしていない場合にも VLAN ルーティング法を利用できるようにする．また，2 つ目の手法「VLAN リネーミング法」では，スイッチタグ法をさらに改良し，VLAN の使用を各スイッチ内でフレームの出力ポートを決定する目的に限定することにより，必要となる VLAN 数を大幅に削減する．

4.1　スイッチタグ法

VLAN ルーティング法は，第 2章で述べた通り，IEEE 802.1Q で標準化されているタグ VLAN 技術を利用することで，イーサネットにおいてループを含むさまざまなトポロジを構築可能にする手法である．工藤らは，VLAN ルーティング法を提案した論文 [14] において，VLAN ルーティング法の動作原理を示した上で，最大で 8 台のスイッチ(注 1)による比較的単純なトポロジを TCP/IP ベースの MPI 通信ライブラリ [44] を用いて評価した結果を報告している．この評価において，経路 (パス) を選択するための VLAN 番号 (VLAN ID) は，ID に関連付けられた仮想インタフェースの IP アドレスによって示される．

一方，最近のイーサネットを用いた PC クラスタは，TCP/IP をバイパスする軽量な通信ライブラリを利用している場合が多い．しかし，残念ながら，このようなシステムソフトウェアは通常 VLAN をサポートしていないため，VLAN ルーティング法を利用するにはライブラリに手を加える必要があるという問題点がある．

本節では，スイッチへイーサネットフレームが入力される際に VLAN タグを付加することで，VLAN をサポートしていない通信ライブラリからでも VLAN ルーティング法を利用できるようにする手法「スイッチタグ法」[45][46] を提案する．本手法は，IEEE 802.1Q 標準で定義されたタグ VLAN の機能のみを用いて実現するため，一般的な VLAN 対応のスイッチ以外に特殊な機器やソフトウェアは一切必要なく，すべて既存のものを利用できる．

4.1.1　スイッチタグ法によるルーティングの動作

スイッチタグ法では，ある 1 つのホストからフレームを送る場合のパスは，宛先ホストによらずすべて単一の VLAN に属するように設定する．2.1.2節で説明した VLAN 対応スイッチの動作に従い，ホストと直接接続されたスイッチにおいて，ホストからの入力フレームに VLAN タグを

(注 1)物理的には 3 台のスイッチを VLAN により分割し，論理的に 8 台として用いている．

付加し，ホストへ出力するフレームから VLAN タグを除去する．このために，ホストと接続された各ポートに対し，以下の 2 種類の設定を行う．

- ポートの PVID として，接続されたホストがフレームを送信する際に使用する VLAN の ID を設定する．すなわち，このホストが送信するフレームが通過するパスは，宛先によらずすべてこの VLAN に含まれる．
- 各リモートホストから送られてくるフレームの VLAN タグを除去するため，ポートをネットワーク全体で使われる全 VLAN の untagged メンバとして登録しておく．

スイッチタグ法の適用例を図 4.1に示す．図の (a)〜(c) において，VLAN A はスイッチ s1, s2, s3, s4, s5 によって，VLAN B はスイッチ s1, s2, s3, s4, s6 によって構成されており，(c) においてスイッチ s1, s2 のホスト側ポートには PVID A が，スイッチ s3, s4 のホスト側ポートには PVID B が割り当てられている．(c) において，例えばホスト 1 から送出されたフレームは，スイッチ s1 の入力ポートにおいて VLAN A のタグを付与され，すべての宛先について VLAN A 内でルーティングされる．そして，宛先ホストと接続された末端スイッチ s2, s3, s4 の出力ポートにおいて VLAN タグは除去される．一方，ホスト 3 から送出されたすべてのフレームは同様に VLAN B 内でルーティングされる．

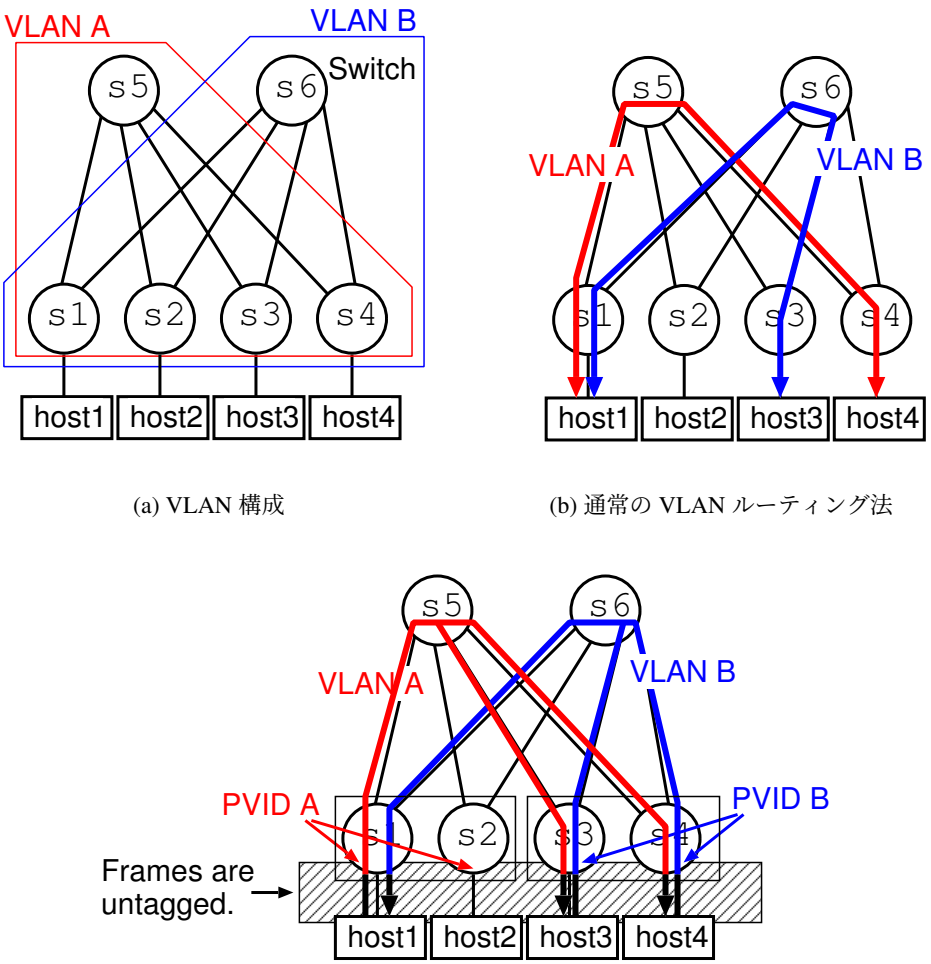
このようにすることで，ホスト側で VLAN がサポートされていなくても，さまざまなトポロジにおいて全ホストが相互に通信できるようになる．同じパス集合によるパケット転送は，図 4.1(b) のように従来の VLAN ルーティング法でも実現可能であるが，VLAN タグの処理の仕方が異なる．

フレームが転送される際の動作の詳細は，以下の通りである．

1. まず，送信側ホストは，通常の (タグなし) フレームを，IP アドレスや MAC アドレスで宛先ホストを指定することによって送出する．
2. スイッチのポートに入力される際に，フレームはそのポートに設定された PVID によってタグ付けされ，以後はタグによって示される VLAN に属すると見なされる．これにより，ホストにおいて常に単一の VLAN タグを付加してフレームを送出するのと同じ効果が得られる．
3. スイッチ間の転送においては，フレームは通常の VLAN ルーティング法と同様に，レイヤ 2 イーサネットの動作に従って転送される．
4. 最後に，受信側ホストに接続されたスイッチの出力ポート (VLAN の untagged メンバに設定されている) から出力される際に，フレームのタグは除去される．
5. 受信側ホストはタグなしフレームを受け取るため，システムソフトウェアでは通常通りの処理を行うことができる．

4.1.2 VLAN 割り当てアルゴリズム

並列計算機の結合網や SAN における固定ルーティングアルゴリズムは，以下の 3 種類に分類することができる [23]．ここで，N はノード集合，P はパス集合，C はチャネル集合をそれぞれ表し，パスとはチャネルの順序付きリストである．イーサネットの場合，チャネルはスイッチ間やホスト-スイッチ間の「リンク」に相当し，スイッチから見た場合は接続されたリンクとポートが 1 対 1 に対応するため,「ポート」と置き換えてもほぼ同義となる．また，イーサネットではフレー



(a) VLAN 構成

(b) 通常の VLAN ルーティング法

(c) スイッチタグ法

図 4.1　Fat ツリーにおけるスイッチタグ法の例

ムの送信元や宛先がスイッチとなることはないため，ノード集合は「ホスト集合」に対応すると考えてよい．なお，$A \times B \mapsto C$ という表記は，集合 A の要素と B の要素の組み合わせによって C の要素がただ 1 つ定まることを示す．

$N \times N \mapsto P$ 型　All-at-once とも呼ばれ，パケットがネットワークに注入された際に，与えられた送信元ノードと宛先ノードの組からただ 1 つのパスが決定される．

$N \times N \mapsto C$ 型　各スイッチにおいて，パケットの送信元ノードと宛先ノードを参照し，出力すべきチャネル (ポート) を決定する．

$C \times N \mapsto C$ 型　各スイッチにおいて，パケットの入力チャネル (ポート) と宛先ノードを参照し，出力すべきチャネル (ポート) を決定する．

スイッチタグ法では，このうち $N \times N \mapsto P$ 型の固定ルーティングアルゴリズムを実装することができる．ただし，あるホストからのパスはすべて単一のツリー (VLAN) に含まれていなければならない，という制約がある．

任意の物理トポロジにおいて，トポロジに接続されているホスト数を n，スイッチ数を m とした場合，以下の手順に従って VLAN を割り当てることで，スイッチタグ法によるルーティングを実装することができる．

1. あるホストについて，そのホストから送信されるフレームが通過するパス集合を内包する木 (最大で $n+m-1$ 本のリンクから構成される) を作成する．これをすべてのホストについて行い，n 個の木を作成する．
2. n 個の木のうち，同一のものを 1 つにまとめる．これを同一の木がなくなるまで繰り返す．
3. それぞれの木にユニークな VLAN ID を割り当てる．
4. ホストと接続された各スイッチポートについて，1. でそのホストに対して作成した木の VLAN ID をポートの PVID として登録する．
5. それぞれの木の各スイッチ間リンクについて，リンクの両端に接続されたポートを，その木に割り当てたの VLAN の tagged メンバとして登録する．
6. すべての VLAN について，ホストと接続された各スイッチポートを VLAN の untagged メンバとして登録する．

以上の手順からも明らかなように，スイッチタグ法を適用する際に必要となる VLAN 数は，最大でホスト数 (n) 個である．

4.2 VLAN リネーミング法

既存の VLAN ルーティング法を適用するために必要となる VLAN 数は，ネットワークの規模に応じて増加するため，VLAN 数が大規模クラスタを構築する場合の制限要因となりうる．2.2節で述べた通り，VLAN タグによって識別できる VLAN 数は限られており，安価なスイッチではさらに少数の VLAN しか扱えない場合が多い．また，ホストの MAC アドレスは VLAN ごとに別々に登録されるため，VLAN 数の増加によって，システムに接続可能なホスト数が制限される．

本節では，この問題を解決する手法として，スイッチタグ法をさらに改良し，VLAN の使用を各スイッチ内でフレームの出力ポートを決定する目的に限定することにより必要となる VLAN 数を大幅に削減することのできる「VLAN リネーミング法」[47] を提案する．

4.2.1 VLAN リネーミング法によるルーティングの動作

VLAN リネーミング法では，2.1.2節で説明した IEEE 802.1Q 標準のタグ VLAN 対応スイッチの機能を以下のように利用する．

- 各スイッチのすべてのポートにおいて，出力されるイーサネットフレームの VLAN タグを除去する．
- スイッチの入力ポートには常に VLAN タグなしフレームのみが到着し，それらは PVID に基づいてタグ付けされ，出力ポートに転送される．

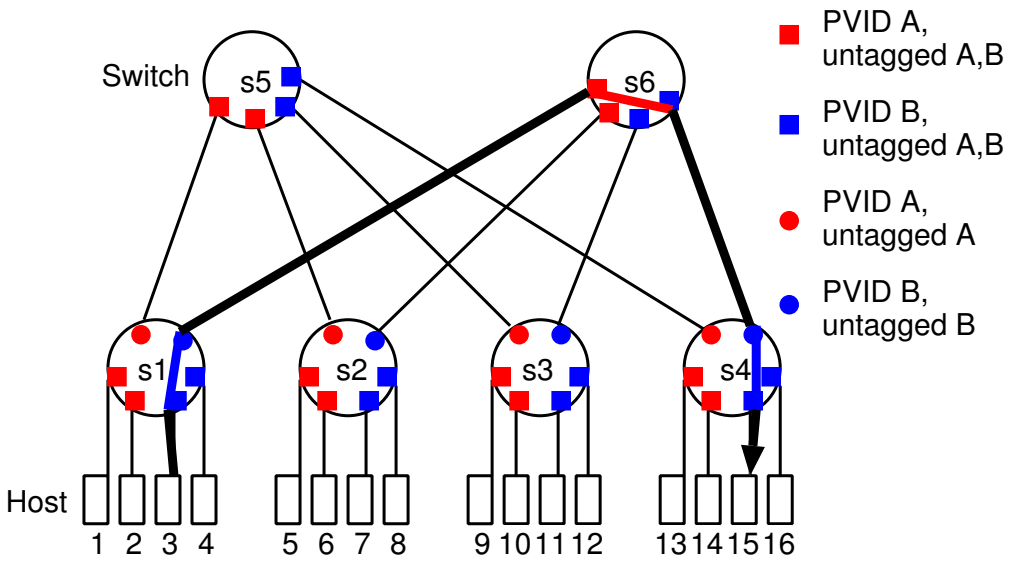


図 4.2　Fat ツリーにおける VLAN リネーミング法の例

VLAN 対応スイッチをこのように利用することにより，各スイッチ間およびホスト-スイッチ間でやり取りされるフレームはすべてタグなしフレームとなるため，各スイッチの設定および動作を VLAN とは関係なく独立にできる．また，VLAN リネーミング法では，スイッチタグ法と同様，ホストの MAC アドレスを学習ではなく各 VLAN 内で静的にスイッチに登録する．

VLAN リネーミング法の Fat ツリートポロジへの適用例を図 4.2に示す．この例では，各スイッチの PVID A のポートに接続されたホスト (1, 2, 5, 6, 9, 10, 13, 14) から送信されたフレームはスイッチ s5 を経由して転送され，PVID B のポートに接続されたホスト (3, 4, 7, 8, 11, 12, 15, 16) から送信されたフレームはスイッチ s6 を経由して転送されるように設定している．図において，凡例はそのポートに設定された PVID と VLAN メンバ情報を示す．例えば「PVID A, untagged A, B」は，PVID が A であり，VLAN A および B の untagged メンバとなっているポートを表している．

スイッチ s1 に着目すると，例えばホスト 1 と接続しているポート (PVID A) から入力されたフレームは，VLAN A でタグ付けされた上で，目的地に応じてスイッチ s5 およびホスト 2, 3, 4 に接続されたポート (いずれも VLAN A の untagged メンバ) のいずれかへと転送される．一方，スイッチ s6 へのポートは VLAN A のメンバではないため，ホスト 1 から入力されたフレームが転送されることはない．

例えば，ホスト 3 からホスト 11 へフレームを転送する場合 (図 4.2の矢印のパス)，スイッチ s1 では以下のように処理が行われる．

1. ホスト 3 から送信された (タグなし) フレームが受信される．この際，PVID の設定によりフレームは VLAN ID B でタグ付けされ，VLAN B に属すると見なされる．

2. フレームの宛先 MAC アドレス (ホスト 11 の MAC アドレス) を，VLAN B の MAC アドレステーブル内で検索し，出力すべきポートがスイッチ s6 へのポートであることを得る (テーブルの設定はあらかじめ行っておく)．

3. スイッチ s6 へのポートからフレームを出力する．このとき，ポートが VLAN B の untagged メンバであるため，出力フレームの VLAN タグを除去する．

スイッチ s6, s3 でも同様の処理が行われ，最終的にフレームはホスト 11 へ転送される．この際，スイッチ s6 内ではフレームは VLAN A で，スイッチ s3 内では VLAN B でそれぞれタグ付けされてルーティングされる．つまり，VLAN リネーミング法では，フレームは各スイッチ内において異なる VLAN ID に乗せ換えられながら転送される．

4.2.2 VLAN リネーミングアルゴリズム

VLAN リネーミング法では，4.1.2節で述べた 3 種類の固定ルーティングアルゴリズムのうち，$C \times N \mapsto C$ 型を実装することができる．

任意の物理トポロジにおいて，各スイッチに対して以下の手順に従って VLAN の設定を行うことで，VLAN リネーミング法によるルーティングを実装することができる．

1. 各ポート i に対して PVID v_i を割り当て，VLAN v_i の untagged メンバとして設定する (図 4.3(a)).
2. 各ポート i について，i から入力されたフレームが転送されうる出力ポートを，VLAN v_i の untagged メンバとして設定する (図 4.3(b)).
3. まったく同一の出力ポート集合 (同一の untagged メンバセット) で構成される VLAN ID 群を，VLAN v_j にまとめる (図 4.3(c)).

定理 4.1 VLAN リネーミング法においてデッドロックフリー (固定) ルーティングアルゴリズムを用いた場合，ブロードキャストストームは生じない．

証明 デッドロックフリー (固定) ルーティングアルゴリズムではチャネル (リンク) 間の循環依存が存在しない．すなわち，フラッディング等によりブロードキャストが発生した場合でも，フレームが一度通過したポートに再度到達することはない．よって，ブロードキャストストームは発生しない． ∎

また，VLAN に対応したイーサネットスイッチにおいて，2.1.4節で述べた MAC アドレスのアドレステーブルへの登録は，VLAN ごとに行われる．VLAN リネーミング法では，この性質を利用して VLAN 毎にホストの MAC アドレスをスイッチに静的に登録することにより，様々なルーティングアルゴリズムを実装することができる．VLAN 技術を用いずに，同じように静的にホストの MAC アドレスを登録することでルーティングを行う方法も検討されている [48] が，VLAN を用いた場合に比べて使用可能なルーティングアルゴリズムが限定される．

例えば，任意のトポロジ上で使用可能なデッドロックフリー固定ルーティングアルゴリズム (5.2節参照) として知られる Up*/Down*ルーティング [49] では，図 4.4において S2 から D へは点線の経路により 1 ホップで転送されるが，S1 から D へは実線の経路により 3 ホップで転送される．ここで，VLAN を用いない場合，S2 において同一宛先のフレームの出力ポートを入力ポート毎に設定することができないため，Up*/Down*ルーティングは実装することができない．一方，VLAN を用いた場合，入力ポート毎に異なる VLAN (PVID) を割り当てることで，様々なルーティングアルゴリズムを実装することができる．

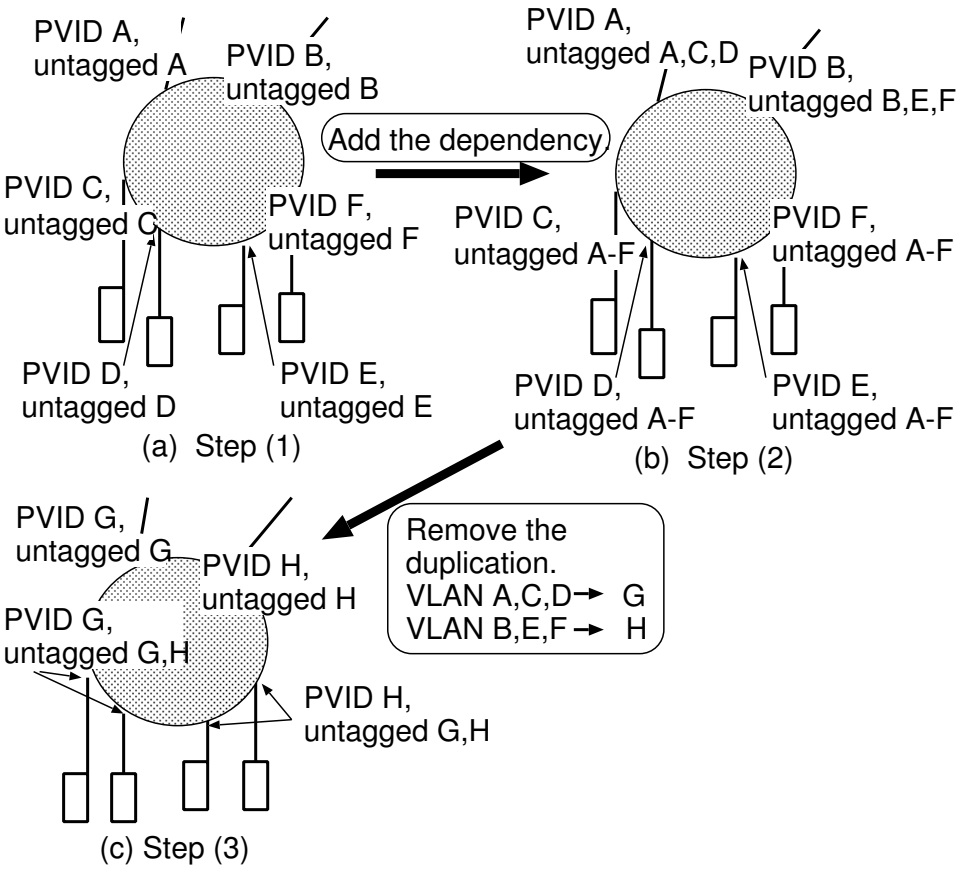


図 4.3　VLAN リネーミング法における VLAN ID の割当て

定理 4.2 VLAN リネーミング法では，トポロジの次数(注 2)以下の VLAN 数で $C \times N \mapsto C$ 型固定ルーティングアルゴリズムを実装することができる．ここで，C はチャネル集合，N はノード集合を表す．

証明　VLAN リネーミング法では，入力ポートに割り当てられた PVID と目的地の MAC アドレスにより出力ポートが決定される．よって，VLAN リネーミング法において $C \times N \mapsto C$ 型固定ルーティングアルゴリズムを実装することができる．また，スイッチの各入力ポートにはただ 1 つの PVID が割り当てられ，各ポートにおいてメンバ設定を行うべき VLAN ID は，いずれかの入力ポートに PVID として設定された VLAN ID に限られる．よって，各スイッチで使用する VLAN ID 数は，最大でも入力ポート数以下である．VLAN の設定は各スイッチで独立であるため，VLAN リネーミング法はトポロジの次数以下の VLAN 数で実装することができる． ∎

4.3　スイッチの設定例

本節では，スイッチタグ法および VLAN リネーミング法を適用する際の実際のスイッチ設定について述べる．

スイッチタグ法，VLAN リネーミング法ともに，必要となるスイッチの設定項目は以下の通り

(注 2) トポロジ内の各スイッチに接続されているリンク数の最大値.

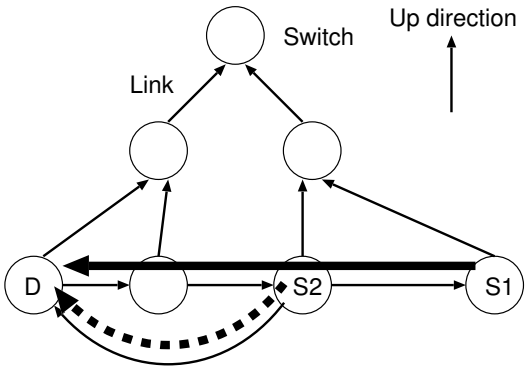


図 4.4　Up*/Down*ルーティングの例

である．

- スパニングツリープロトコル (STP) を無効にする．
- 使用する VLAN ID を登録する．
- 各ポートについて，所属する VLAN のメンバ設定を行い，PVID を設定する．
- 各 VLAN について，各ホストの MAC アドレスを登録する．
- 性能向上のためにリンクレベルフロー制御を有効にする．(オプション)

以下，スイッチタグ法，VLAN リネーミング法のそれぞれについて，実際のスイッチ設定の例を挙げながら説明する．なお，具体的なコマンドはベンダおよびスイッチの種類毎に異なるが，設定内容はほぼ同様となる．

4.3.1　スイッチタグ法の場合

図 4.1に示した Fat ツリートポロジにおいて，スイッチとして Dell PowerConnect 5324[50] を使用する場合のスイッチ s1 の設定ファイルの記述例を図 4.5に示す．ここで，図 4.1において，スイッチ s1 のポート 1, 5, 6 に，ホスト 1 およびスイッチ s5, s6 がそれぞれ接続されていると仮定し，VLAN ID には A = 101, B = 102 を割り当てる．なお，図 4.5の各行の左端の数字は説明のための行番号であり，実際には記述されない．

例として，図 4.1のホスト 1 からホスト 3 へフレームを送信する場合，スイッチ s1 では以下のように処理が行われる．

1. ホスト 1 から送信された (タグなし) フレームがポート 1 で受信される．この際，図 4.5の 14 行目の PVID の設定により，フレームは VLAN 101 (VLAN A) に属すると見なされる．
2. フレームの宛先 MAC アドレス (ホスト 3 の MAC アドレス) を，VLAN 101 の MAC アドレステーブル (27〜35 行目で設定) 内で検索し，出力すべきポートがポート 5 (スイッチ s5 へのポート) であることを得る (33 行目).
3. ポート 5 の VLAN メンバ設定 (16〜20 行目) において，VLAN 101 がメンバとして登録されているかどうかを調べ，tagged メンバであることを得る (19 行目).

```
 1:  // スパニングツリープロトコルを無効にする
 2:  no spanning-tree
 3:
 4:  // VLAN ID の登録
 5:  vlan database
 6:   vlan 101,102 // VLAN ID 101,102 を登録
 7:  exit
 8:
 9:  // 各ポートの VLAN の設定
10:  interface ethernet g1 // ポート 1（ホスト 1 へのポート）の設定
11:   switchport mode general // ポートの VLAN モードの設定（802.1Q フルサポートモード）
12:   // VLAN 101,102 の untagged メンバとして登録
13:   switchport general allowed vlan add 101,102 untagged
14:   switchport general pvid 101 // PVID を 101 に設定
15:  exit
16:  interface ethernet g5 // ポート 5（スイッチ s5 へのポート）の設定
17:   switchport mode general
18:   // VLAN 101 の tagged メンバとして登録
19:   switchport general allowed vlan add 101 tagged
20:  exit
21:  interface ethernet g6 // ポート 6（スイッチ s6 へのポート）の設定
22:   switchport mode general
23:   switchport general allowed vlan add 102 tagged
24:  exit
25:
26:  // 各 VLAN 毎の MAC アドレスの登録
27:  interface vlan 101 // VLAN 101 に関する設定
28:   // ホスト 1 の MAC アドレスを登録（ホスト 1 はポート 1 に接続されている）
29:   bridge address MAC_Host1 ethernet g1 delete-on-reset
30:   // ホスト 2〜4 の MAC アドレスを登録
31:   //（VLAN 101 では，ホスト 2〜4 はポート 5（スイッチ s5）の先にある）
32:   bridge address MAC_Host2 ethernet g5 delete-on-reset
33:   bridge address MAC_Host3 ethernet g5 delete-on-reset
34:   bridge address MAC_Host4 ethernet g5 delete-on-reset
35:  exit
36:  interface vlan 102 // VLAN 102 に関する設定
37:   // ホスト 1 の MAC アドレスを登録
38:   bridge address MAC_Host1 ethernet g1 delete-on-reset
39:   // ホスト 2〜4 の MAC アドレスを登録
40:   //（VLAN 102 では，ホスト 2〜4 はポート 6（スイッチ s6）の先にある）
41:   bridge address MAC_Host2 ethernet g6 delete-on-reset
42:   bridge address MAC_Host3 ethernet g6 delete-on-reset
43:   bridge address MAC_Host4 ethernet g6 delete-on-reset
44:  exit
45:
46:  // フローコントロールを全ポートで有効にする（オプション）
47:  interface range ethernet all
48:   flowcontrol on
49:  exit
```

図 4.5　Fat ツリー (2,4,2) におけるスイッチタグ法のスイッチ設定例

4. フレームをポート 5 から出力する．このとき，tagged メンバであるため出力フレームは VLAN 101 でタグ付けされたままである．

逆に，図 4.1のホスト 3 からホスト 1 へフレームを送信する場合，スイッチ s1 では以下のように処理が行われる．

1. ホスト 3 から送信され，スイッチ s3 において VLAN 102 (VLAN B) でタグ付けされたフレームがポート 5 (スイッチ s5 からのポート) で受信される．

2. フレームの宛先 MAC アドレス (ホスト 1 の MAC アドレス) を，VLAN 102 の MAC アドレステーブル (36〜44 行目で設定) 内で検索し，出力すべきポートがポート 1 であることを得る (38 行目)．

3. ポート 1 の VLAN メンバ設定 (10〜15 行目) において，VLAN 102 がメンバとして登録されているかどうかを調べ，untagged メンバであることを得る (13 行目)．

4. フレームをポート 1 から出力する．このとき，untagged メンバであるため出力フレームの VLAN タグを除去する．

同様の設定をスイッチ s2〜s6 に対しても行うことで，図 4.1の Fat ツリーにおけるスイッチタグ法の設定が完了する．この際，スイッチ s2〜s4 においては，それぞれホスト 2, 3, 4 をポート 1 に接続し，スイッチ s5, s6 をポート 5, 6 にそれぞれ接続する．これにより，スイッチ s2〜s4 の設定は，PVID の値と MAC アドレスの登録部分を除いてスイッチ s1 と全く同じとなる．

4.3.2 VLAN リネーミング法の場合

図 4.2に示した Fat ツリートポロジにおいて，スイッチとして Dell PowerConnect 5324 を使用した場合のスイッチ s1 の設定ファイルの記述例を図 4.6に示す．ここで，図 4.2において，スイッチ s1 のポート 1〜6 に，ホスト 1, 2, 3, 4 およびスイッチ s5, s6 がそれぞれ接続されていると仮定し，VLAN ID には A = 101, B = 102 を割り当てる．なお，図 4.6の各行の左端の数字は説明のための行番号であり，実際には記述されない．

例として，図 4.2のホスト 3 からホスト 11 へフレームを送信する場合 (矢印のパス)，スイッチ s1 では以下のように処理が行われる．

1. ホスト 3 から送信された (タグなし) フレームがポート 3 で受信される．この際，図 4.6の 19 行目の PVID の設定により，フレームは VLAN 102 (VLAN B) に属すると見なされる．

2. フレームの宛先 MAC アドレス (ホスト 11 の MAC アドレス) を，VLAN 102 の MAC アドレステーブル (46〜54 行目で設定) 内で検索し，出力すべきポートがポート 6 (スイッチ s6 へのポート) であることを得る (51〜53 行目．ただし当該行は省略してある)．

3. ポート 6 の VLAN メンバ設定 (26〜30 行目) において，VLAN 102 がメンバとして登録されているかどうかを調べ，untagged メンバであることを得る (28 行目)．

4. フレームをポート 6 から出力する．このとき，untagged メンバであるため出力フレームの VLAN タグを除去する．

```
 1:  // スパニングツリープロトコルを無効にする
 2:  no spanning-tree
 3:
 4:  // VLAN ID の登録
 5:  vlan database
 6:   vlan 101,102 // VLAN ID 101,102 を登録
 7:  exit
 8:
 9:  // 各ポートの VLAN の設定
10:  interface range ethernet g(1,2) // ポート 1,2（ホスト 1,2 へのポート）を同時に設定
11:   switchport mode general // ポートの VLAN モードの設定（802.1Q フルサポートモード）
12:   // VLAN 101,102 の untagged メンバとして登録
13:   switchport general allowed vlan add 101,102 untagged
14:   switchport general pvid 101 // PVID を 101 に設定
15:  exit
16:  interface range ethernet g(3,4) // ポート 3,4（ホスト 3,4 へのポート）を同時に設定
17:   switchport mode general
18:   switchport general allowed vlan add 101,102 untagged
19:   switchport general pvid 102
20:  exit
21:  interface ethernet g5 // ポート 5（スイッチ s5 へのポート）の設定
22:   switchport mode general
23:   switchport general allowed vlan add 101 untagged
24:   switchport general pvid 101
25:  exit
26:  interface ethernet g6 // ポート 6（スイッチ s6 へのポート）の設定
27:   switchport mode general
28:   switchport general allowed vlan add 102 untagged
29:   switchport general pvid 102
30:  exit
31:
32:  // 各 VLAN 毎の MAC アドレスの登録
33:  interface vlan 101 // VLAN 101 に関する設定
34:   // ホスト 1〜4 の MAC アドレスを登録
35:   //（ホスト 1〜4 はポート 1〜4 にそれぞれ接続されている）
36:   bridge address MAC_Host1 ethernet g1 delete-on-reset
37:   bridge address MAC_Host2 ethernet g2 delete-on-reset
38:   bridge address MAC_Host3 ethernet g3 delete-on-reset
39:   bridge address MAC_Host4 ethernet g4 delete-on-reset
40:   // ホスト 5〜16 の MAC アドレスを登録
41:   //（VLAN 101 では，ホスト 5〜16 はポート 5（スイッチ s5）の先にある）
42:   bridge address MAC_Host5 ethernet g5 delete-on-reset
43:   bridge address MAC_Host6 ethernet g5 delete-on-reset
44:   ...  // 以下同様
45:  exit
46:  interface vlan 102 // VLAN 102 に関する設定
47:   // ホスト 1〜4 の MAC アドレスを登録
48:   ...（VLAN 101 と同様なので省略）
49:   // ホスト 5〜16 の MAC アドレスを登録
50:   //（VLAN 102 では，ホスト 5〜16 はポート 6（スイッチ s6）の先にある）
51:   bridge address MAC_Host5 ethernet g6 delete-on-reset
52:   bridge address MAC_Host6 ethernet g6 delete-on-reset
53:   ...  // 以下同様
54:  exit
55:
56:  // フローコントロールを全ポートで有効にする（オプション）
57:  interface range ethernet all
58:   flowcontrol on
59:  exit
```

図 4.6　Fat ツリー (2,4,2) における VLAN リネーミング法のスイッチ設定例

同様の設定をスイッチ s2〜s6 に対しても行うことで，図 4.2の Fat ツリーにおける VLAN リネーミングの設定が完了する．この際，スイッチ s2〜s4 においては，ホスト 5〜8, 9〜12, 13〜16 をポート 1〜4 にそれぞれ接続し，スイッチ s5, s6 をポート 5, 6 にそれぞれ接続する．また，スイッチ s5, s6 においては，スイッチ s1〜s4 をポート 1〜4 にそれぞれ接続する．これにより，スイッチ s2〜s6 の設定は，MAC アドレスの登録部分を除いてスイッチ s1 と全く同じとなる．

4.3.3 設定ファイルの記述とアップロード

以上に挙げたようなスイッチ設定ファイルの記述は，現段階ではクラスタ設計者が手動で行う必要がある．このため，スイッチ数やホスト数が増加した場合にはその分工数がかかる．規則的なトポロジであれば，スクリプトファイル等を記述することにより半自動化することは可能だが，あらゆるトポロジに対応するのは困難である．

また，トポロジの構築のために，記述した設定ファイルを各スイッチにアップロードする際は，SNMP 等の管理用プロトコルを用いるか，構築する結合網とは別のネットワーク (別系統のイーサネットや，シリアルインタフェース等) を経由して各スイッチにアップロードする必要がある．ただし，以上の手順は，SAN を用いたクラスタや，多くの並列分散システムにおいても同様であり，多くの工数を必要とすることに変わりはない．

4.4 適用範囲および他の手法との比較

本節では，提案手法および従来の VLAN ルーティング法を用いてトポロジを構築する際の適用可能範囲，すなわち適用可能なイーサネットスイッチ，構築可能なシステム規模，実装可能なルーティングアルゴリズムのそれぞれについて検討する．また，VLAN ルーティング法を実現する他の手法を紹介し，提案手法との比較について議論する．

4.4.1 適用可能なイーサネットスイッチ

現在，イーサネットスイッチの価格は，1000BASE-T に限定したとしても数千円〜百万円前後まで多岐にわたる．しかし，極めて安価なスイッチは VLAN にすら対応していないものが多く，ごく一部の VLAN 機能のみを提供している安価なイーサネットスイッチでは VLAN ルーティング法を利用できない場合がある．

IEEE 802.1Q 準拠と仕様に書かれたスイッチの場合でも，2.1.2節で述べた VLAN タグ操作のすべてを実現していない場合があるため，注意が必要である．例えば，Cisco Systems の Catalyst 3550 シリーズでは，あるポートにおいて，PVID (Catalyst 3550 では native port と呼ばれる) として設定された VLAN 以外の VLAN では，メンバー設定において untagged メンバとして設定することができない．つまり，このようなスイッチでは，出力ポートにおいて PVID 以外の VLAN に属するフレームからタグを除去することができず，スイッチタグ法，VLAN リネーミング法ともに実装することは不可能である．

提案手法を適用可能かどうか判断するもう 1 つの基準は，スイッチの静的な MAC アドレス登録のサポート状況である．イーサネットスイッチは通常，2.1.4節で述べた手順で MAC アドレスを学習する．しかし，提案手法を含む VLAN ルーティング法では複数の VLAN を利用するため，ホスト A から B へのパスと B から A へのパスが異なる VLAN を使っている場合が当然考えられ

表 4.1　各手法の適用に必要なスイッチの機能

	VLAN	SW-TAG	RENAME
VLAN ID にもとづく転送ポートの決定	○	○	○
PVID によるフレーム入力時のタグ付け		○	○
untagged 設定による任意の VLAN ID のタグ除去		○	○
MAC アドレステーブルエントリの静的登録		△	○

る．MAC アドレスの学習は VLAN ごとに独立して行われるため，それぞれのパスの中間スイッチ群では，たとえ 2 つのパスが全く同じスイッチの集合から構成されていても，宛先ホスト側の MAC アドレスの学習が不可能となる．

この問題は，従来の (一般の) VLAN ルーティング法の場合にも当てはまるが，スイッチタグ法の場合，各パスの往路と復路に同じ VLAN を割り当てようとすると，各ホストからのパスがすべて 1 つの VLAN に属していなければならないという制限のために，全体として 1 つの VLAN しか使用できない．また，VLAN リネーミング法の場合は，フレームは各スイッチにおいて異なる VLAN ID に乗せ換えられながら転送されるため，そもそも MAC アドレスの学習は全く不可能である．

幸い，最近の商用イーサネットスイッチでは，Dell PowerConnect 5324[50] のような比較的安価なものであっても，MAC アドレステーブルの静的な設定ができるようになっているものが多い．このため，スイッチタグ法および VLAN リネーミング法では，4.3節の設定例でも示したように，各スイッチにおいて静的に MAC アドレステーブルエントリを設定することを前提としている．ただし，例えば前述の Catalyst 3550 のように，MAC アドレステーブルのエントリ数自体は 12,000 個であるが，静的に登録可能な MAC アドレス数が 128 個に制限されているようなスイッチも存在する．この場合，スイッチタグ法や VLAN リネーミング法の利用は可能であるが，構築可能なシステムの規模が制限される．これについては 4.4.2節で議論する．

なお，スイッチタグ法に限り，以下の手続きを踏むことで強制的に学習を行わせて MAC アドレスを登録することも可能であるが，VLAN リネーミング法では適用不可能である．

1. 各ホストで，ネットワーク内で使われる全 VLAN に対応する仮想インタフェース (I/F) を作成する．さらに，それぞれ別々のセグメントに属するように各仮想 I/F に IP アドレスを割り振る．

2. 各ホストから，各 VLAN セグメント内でイーサネットのブロードキャストフレームを送信する．これにより，各スイッチにおいて，VLAN ごとの MAC アドレステーブルに送信ホストの MAC アドレスが登録される．

以上をまとめると，表4.1のようになる．なお，“VLAN” は従来の VLAN ルーティング法，“SW-TAG” はスイッチタグ法，“RENAME” は VLAN リネーミング法をそれぞれ表し，“○” はその機能が必要であることを示す．

4.4.2　構築可能なシステム規模

2 つの提案手法，および従来の VLAN ルーティング法を適用することで構築可能な可能なシステムの規模は，使用するイーサネットスイッチの仕様により定まる．その際に参照されるパラメー

タは以下の 2 種類である．

1. 使用 (登録) 可能な VLAN 数
2. アドレステーブルに登録可能な MAC アドレス数

2.2節で述べた通り，スイッチ内で使用できる VLAN 数には制限があるため，同種のスイッチのみを使用すると仮定し，ネットワーク全体で使用する VLAN 数を V，個々のスイッチに登録できる VLAN 数を $V_{\max}$ とすると，

$$V \leq V_{\max} \tag{4.1}$$

を満たす場合のみそのネットワークを構築可能である．なお，VLAN リネーミング法の場合，各スイッチの VLAN 設定は全く独立であるため，V は正確には「各スイッチで使用する VLAN 数の最大値」を意味する．

一方，スイッチ内で静的または (学習により) 動的に登録されるホストの MAC アドレスは，同じアドレスであっても VLAN ごとに別々に登録される．このため，登録する MAC アドレス数，つまりシステムに接続するホスト数 H は，使用する VLAN 数 V，スイッチに登録できる MAC アドレス数 M として以下の式を満たす必要がある．

$$H \leq \frac{M}{V} \tag{4.2}$$

ここで，M は，スイッチタグ法および VLAN リネーミング法の場合には「静的に」登録可能な MAC アドレス数を表す．前節で述べた通り，スイッチによっては静的に登録可能な MAC アドレス数が MAC アドレステーブルの総エントリ数に比べて制限されていたり，静的な登録をサポートしていない場合がある．ただし，スイッチタグ法のみ，前節で述べた方法により学習による登録を行うことも可能である．なお，主要なトポロジに対し各手法を適用する際に必要となる VLAN 数 V については，5.4節で議論する．

4.4.3 実装可能なルーティングアルゴリズム

スイッチタグ法および VLAN リネーミング法では，その性質上，従来の VLAN ルーティング法に比べてとりうるパス集合にそれぞれ異なる制限がある．従来の (一般の) VLAN ルーティング法およびスイッチタグ法では，4.1.2節で挙げた 3 種類の固定ルーティングアルゴリズムのうち $N \times N \mapsto P$ 型を実装することができるが，スイッチタグ法には「あるホストからのパスはすべて 1 つの VLAN に属していなければならない」という制限がある．例えば，従来の VLAN ルーティング法による図 4.1(b) に示されるパス集合は，スイッチタグ法では実現できない．ホスト 1 からのパスが，VLAN A と B の 2 つの VLAN を使っているからである．

この制限により，不規則なトポロジにおいては，スイッチタグ法を用いて効率よくトラフィックを各パスに分散させるのは難しい．しかし，並列計算機で用いられているような規則的なトポロジであれば，適切にパスを分散させることのできる VLAN 集合を割り当てるのは比較的容易である．これは 5.4節で議論する．

また，VLAN リネーミング法で実装できるルーティングアルゴリズムは，4.2.2節で述べた通り $C \times N \mapsto C$ 型である．すなわち，一般の VLAN ルーティング法 ($N \times N \mapsto P$ 型) ではすべての送信元-宛先間に任意のパスを設定できるのに対し，VLAN リネーミング法では，各スイッチにおいて VLAN ID (PVID) と宛先をもとに次のルーティング先を決定する方式であるため，任意の $N \times N \mapsto P$ 型ルーティングを実装することはできない．

ただし，これも 5.4節で議論するように，一般に用いられる規則的なトポロジでは，ルーティングアルゴリズムに関しても，$C \times N \mapsto C$ 型で実装可能な規則性を持った手法を採用するのが一般的である．同じアルゴリズムを VLAN ルーティング法と VLAN リネーミング法でそれぞれ実装する場合は，使用 VLAN 数を少なく抑えることが可能な分 VLAN リネーミング法の方が優れていると言える．なお，VLAN リネーミング法は不規則トポロジの扱いにおいても一般の VLAN ルーティング法より優れており，Up*/Down*ルーティング等，不規則トポロジ向けのルーティングアルゴリズムの実装も比較的容易である (5.4.4節参照).

4.4.4 他の手法との比較

三浦らが開発している VFREC-Net[51][52][53][54] では，送信に用いる VLAN を選択してフレームへのタグ付けを行うための Linux 用デバイスドライバを実装し，TCP/IP を用いた VLAN ルーティング法の利用を実現している．この手法では，フレームヘッダに格納された送信元および宛先の MAC アドレスをもとに，ドライバが自動的に使用する VLAN ID を決定してタグ付けを行うため，上位レイヤのソフトウェア環境に手を加えることなく VLAN ルーティング法を実現できる．この点で柔軟性が高く，本研究の提案手法とともに，VLAN ルーティング法を利用した PC クラスタの構築方法として有力であると言える．また，どちらも VLAN ID 制御のオーバヘッドが小さい点で優れている．

ただし，本研究の 2 つの提案手法と VFREC-Net では，次の点でその適用範囲が異なる．

- 本研究の提案手法は，オペレーティングシステムやそのバージョンに依存しない．
- VLAN リネーミング法では，必要となる VLAN 数をスイッチのポート数以下に削減することができ，より大規模なシステムを構築可能である．
- VFREC-Net によって実装可能なルーティングアルゴリズムは，本研究の提案手法によって実装可能なルーティングアルゴリズムよりも対象範囲が広い．

両者とも，変化が激しい HPC 分野を対象としているため，柔軟性は重要である．VFREC-Net も，Linux 用のデバイスドライバであるため高い柔軟性を持っているが，本手法はさらに，例えば最新バージョンのカーネルや，オペレーティングシステムとして Windows を用いるような PC クラスタにも，ホスト側の動作検証を行うことなくそのまま適用することができる．この点で，提案手法は利用する際の柔軟性が高く，管理が簡単であると言える．また，VLAN リネーミング法を用いた場合には，他の手法に比べ必要となる VLAN 数を少なく抑えることが可能であり，4.4.2節の式 4.2から，より多数のホストを接続した大規模なシステムを構築することができることがわかる．

一方，VFREC-Net で実装可能なルーティングアルゴリズムは，従来の (一般の) VLAN ルーティング法と同様 $N \times N \mapsto P$ 型であり，4.4.3節で述べた通りとりうるパス集合に制限がある提案手法に比べて，より柔軟なパス選択が可能である．さらに，提案手法，特に VLAN リネーミング法では，VFREC-Net のようにスイッチにおける MAC アドレスの学習は利用できず，各スイッチにおいて静的に MAC アドレスを登録しなければならない．ただし，5.4節で述べる通り，提案手法を用いても，規則性を持つ多くのトポロジにおいて典型的な最短パスルーティングを実装することは可能である．

表 4.2　各手法の比較

	VLAN	SW-TAG	RENAME
ホストでの VLAN サポート	必要	不要	不要
使用可能な VLAN スイッチ	制限なし	制限あり	制限大
必要 VLAN 数の決定要因	トポロジ規模 & ルーティング	トポロジ規模 & ルーティング	スイッチのポート数 & ルーティング
最大ノード数の決定要因	必要 VLAN 数 & MAC エントリ数	必要 VLAN 数 & MAC エントリ数	必要 VLAN 数 & 静的登録可能な MAC エントリ数
実装可能なルーティング	$N \times N \mapsto P$	$N \times N \mapsto P$ (制限あり)	$C \times N \mapsto C$

4.4.5 まとめ

表 4.2に，各手法の特徴をまとめる．“VLAN” は従来の VLAN ルーティング法，“SW-TAG” はスイッチタグ法，“RENAME” は VLAN リネーミング法をそれぞれ表す．

VLAN リネーミング法は，ホストでの VLAN サポートを必要とせず，使用 VLAN 数がトポロジの規模に依存しない点で，大規模化に最も適している．一方で，スイッチに対する要求が厳しく，任意の VLAN ID のタグを除去する機能に加え，ホストの MAC アドレスを静的に登録する必要があるため，静的登録をサポートしないスイッチには適用できず，静的登録可能な MAC アドレス数が制限されるスイッチではその分最大ノード数が減少する．

これに対し，従来の VLAN ルーティング法は，実装可能なルーティングアルゴリズムの範囲が最も広く，必要なスイッチの機能もタグ付きフレームの送受信のみである．しかし，ホスト側ソフトウェアによる VLAN タグのサポートを必要とする上，使用 VLAN 数がトポロジの規模に応じて増加するため，それに従って最大ノード数が減少する．また，使用 VLAN 数は使用するルーティングの種類にも左右され，一般にパスを分散させるほど使用 VLAN 数は増加する．

また，スイッチタグ法は，使用 VLAN 数および最大ノード数に関しては従来の VLAN ルーティング法と同じ特徴を持つが，VLAN リネーミング法と同様ホストでの VLAN サポートを必要としない上，VLAN リネーミング法では不可能な MAC アドレスの学習も可能である．ただし，ホストと接続されるスイッチには任意の VLAN ID を除去する機能が必要であり，ルーティングアルゴリズムに関して，あるホストからのパスがすべて 1 つの VLAN に属していなければならないという制限がある．

第5章　VLANルーティング法を用いたクラスタ向けトポロジの設計

本章では，VLAN ルーティング法，および第 4章で提案した VLAN ルーティング法を改良する 2 種類の手法をイーサネットに適用してトポロジを構築する際の検討事項について述べる．

まず，イーサネットにおけるフロー制御の有効性と，VLAN によるループを含むトポロジを構築した際に発生するデッドロックの問題について検証した結果を示し，VLAN ルーティング法を適用する際のデッドロックフリールーティングの重要性について明らかにする．その上で，VLAN を利用して構築可能な並列計算向けトポロジの例を示し，各トポロジ上でのルーティングアルゴリズムの選択，VLAN 割り当ての方法について検討する．

5.1　パスの衝突とフロー制御の有効性

本節では，イーサネットにおいて複数のパスがリンク上に重なる場合の性能低下の問題について検証し，イーサネットが提供するフロー制御機構が転送性能の改善に有効であることを示す．

5.1.1　イーサネットにおけるフロー制御

一般に，並列計算機の結合網やシステムエリアネットワーク (SAN) におけるフロー制御は，ハードウェアや NIC 上のファームウェアで行う場合が多い．これに対しイーサネットでは，通常は TCP 等の上位 (トランスポート層) プロトコルによってフロー制御を行う．しかし，TCP のフロー制御は End-to-End で行われるため，トポロジが大規模化すると，パスのホップ数が増加して処理オーバヘッドが増大する問題がある．また，End-to-End のフロー制御では，パス上にあるスイッチのバッファ不足を検出できないため，複数のパスがリンク上に重なる場合には効率が低下する．

一方，IEEE 802.3x 標準において，イーサネットでのフロー制御が規定されている [55]．このフロー制御では，PAUSE フレームと呼ばれる特殊なイーサネットフレームを用いて，単純な Stop/Go 方式のフロー制御をリンクレベルで行うものである．PAUSE フレームは，接続先デバイスがフレームの送信をどれくらいの時間待てばよいかを 512 ビット時間単位で指定するフィールドを持つ．

しかし，IEEE 802.3x フロー制御はイーサネットスイッチにおける初期設定では無効になっていることが多く，現状ではあまり用いられていない．これは主に，QoS (Quality of Service) 機能と同時に使用すると QoS がうまく動作しない，という互換性の問題が原因と考えられるが，クラスタ環境ではあらゆるホスト対が相互に通信するため，特定のアプリケーションを想定しない限り QoS を設定して利用することは考えにくい．そのため，IEEE 802.3x フロー制御を使用することによって，複数パスが重なった際の性能を改善できる可能性があると考えられる．そこで，実際のクラスタ環境において，その有効性を評価する実験を行った．

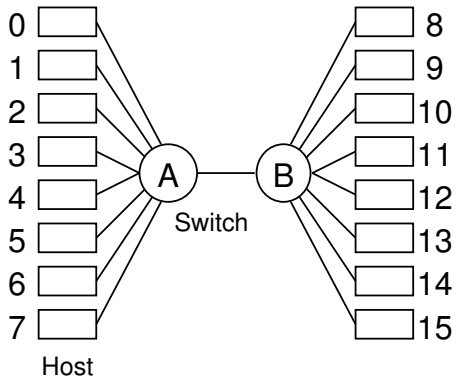


図 5.1　単純な 2 スイッチトポロジ

表 5.1　クラスタ 1 の各ノードの仕様

プロセッサ	Intel Pentium4 2.4C (2.4GHz)
メモリ	PC3200 DDR SDRAM 512MB
チップセット	Intel D865GLC
NIC	Intel 82547EI (オンボード，CSA 接続)
NIC ドライバ	Intel e1000 5.2.39
OS	Red Hat Linux 9 (kernel 2.4.21)

5.1.2　PAUSE フロー制御の評価実験

図 5.1に示す単純な 2 スイッチトポロジにおいて，IEEE 802.3x 標準の PAUSE フレームを用いたリンクレベルのフロー制御を用いる場合と用いない場合の転送バンド幅およびパケット消失率を測定した．用いたクラスタ (クラスタ 1 とする) の各ホストの仕様は表 5.1に示す通りで，イーサネットスイッチには，ノンブロッキングレイヤ 2 スイッチである Dell PowerConnect 5224 (1000BASE-T × 24 ポート) を用いた．また，測定プログラムには Iperf 2.0.2[56] の UDP 転送機能を用いた．UDP であるため，上位プロトコルによる End-to-End のフロー制御は行われない．UDP データグラムの送信レートは 1,000Mbps，バッファサイズは 128KB とした．

測定結果を表 5.2に示す．表において,「2-SW-SW-2」は 2 台のスイッチを介して 2 組の送信元-宛先対が接続されている場合，例えば図 5.1のホスト 0, 1 からホスト 8, 9 へそれぞれデータ転送を行う場合を表している．スイッチ間は 1 本のリンクで接続されているため，各送信元-宛先対間で転送されるフレームは，スイッチ間のリンク (図 5.1の中央のリンク) においてパスが重なる．なお，ここでのバンド幅は，全通信対におけるバンド幅の総和である．

表 5.2の結果より，フロー制御を用いた場合と用いない場合とで転送バンド幅はほとんど変わらないが，フロー制御なしの場合，転送パスが重なったとき (2-SW-SW-2, 4-SW-SW-4, 8-SW-SW-8) に大量のパケット消失が発生していることがわかる．一方で，フロー制御を用いた場合は，これらのパケット消失を完全に防止できている．一方で，1-SW-1, 2-SW-2, 4-SW-4, 1-SW-SW-1 の 4 つのパターンについては，パケットの消失を完全に 0 にはできていない．これは，NIC からスイッチに対するフロー制御が適切に働いていないことが原因として考えられる．

次に，同じく図 5.1のトポロジにおいて，上位層による End-to-End のフロー制御が存在する場合の転送バンド幅を測定した．使用したクラスタ (クラスタ 2 とする) の各ホストの仕様は表 5.3に示す

表 5.2　2 ホスト間の UDP 転送における転送バンド幅とパケット消失率

	フロー制御なし		フロー制御あり	
	バンド幅 [Mbps]	消失率 [%]	バンド幅 [Mbps]	消失率 [%]
1-SW-1	820	0.0249	821	0.0331
2-SW-2	1642	0.0334	1640	0.0921
4-SW-4	3283	0.0632	3275	0.0470
1-SW-SW-1	821	0.0208	821	0.0159
2-SW-SW-2	958	41.6	957	0
4-SW-SW-4	957	70.8	960	0
8-SW-SW-8	958	85.2	959	0

表 5.3　クラスタ 2 の各ノードの仕様

プロセッサ	Intel Xeon 2.8GHz × 2 (SMP)
メモリ	PC2-3200 DDR2 SDRAM 1GB
チップセット	Intel E7520
PCI	64bit/133MHz PCI-X
NIC	Intel 82545 (Intel PRO/1000 MT Server Adapter)
NIC ドライバ	Intel e1000 6.2.15
OS	Fedora Core 1 (kernel 2.4.21)

通りで，イーサネットスイッチには Dell PowerConnect 5324 (1000BASE-T × 24 ポート，ノンブロッキング) を用いた．クラスタには，オープンソースのクラスタシステムソフトウェア SCore[57][58] バージョン 5.8.2 が搭載されている．

測定プログラムには Intel MPI Benchmarks (IMB)[59] 2.3 の Multi-PingPing テストを用い，MPI レベルの転送バンド幅を測定した．MPI ライブラリには SCore 5.8.2 に付属の MPICH-SCore を用いている．MPICH-SCore は，MPICH-1.2.5[60][61] をベースにした MPI ライブラリで，下位の軽量通信ライブラリ PM[11][12][19] 上に構築されている．イーサネットの場合は PM/Ethernet[12][19] が使われ．TCP と同様 End-to-End のフロー制御が行われる．

測定結果を図 5.2に示す．グラフは，MPI の転送サイズを変化させた際の，8 ホスト間 (表 5.2における 8-SW-SW-8 の場合) の双方向バンド幅の平均値をプロットしたものである．ここで，“FC All” は IEEE 802.3x PAUSE フロー制御を用いた場合，“FC None” は用いない場合をそれぞれ表している．

結果より，PAUSE フロー制御を用いなかった場合は，転送サイズが 2KB を越えたところからバンド幅が低下しているが，PAUSE フロー制御を用いた場合はバンド幅の低下は見られない．このことから，PAUSE フレームによるリンクレベルのフロー制御は，上位プロトコルによる End-to-End のフロー制御が行われている場合にも有効であることがわかる．

以上の実験結果より，リンクレベルのフロー制御を用いることによって，フレームの転送パスが重なる場合のパケット消失を抑制し，性能低下を回避することができることがわかった．よって，IEEE 802.3x 標準による PAUSE フロー制御は，クラスタ環境において非常に有効であると言える．

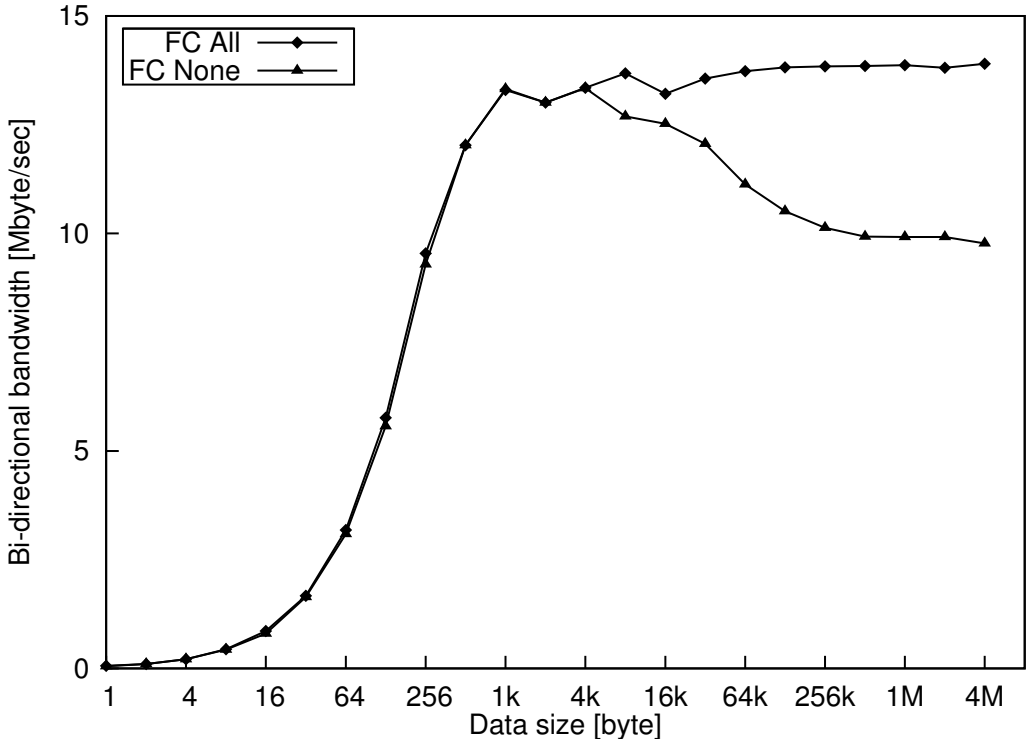


図 5.2 IMB Multi-PingPing テストにおける転送バンド幅

5.2 デッドロックの問題

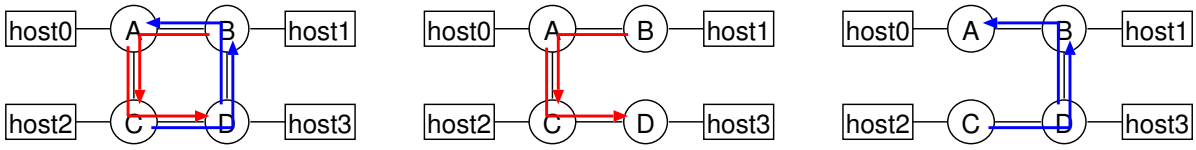
本節では，VLAN ルーティング法等で VLAN を導入し，ループを含むトポロジを構築することによって発生するフレーム間のデッドロックの問題について説明および検証を行う．そして，トポロジおよびその上でのルーティングアルゴリズムを検討する際に，デッドロックフリールーティングを選択するべきであることを示す．

5.2.1 イーサネットにおけるデッドロックの発生と回避

並列計算機の相互結合網や SAN では，メッシュ網などのループ構造を含むトポロジが広く採用されている．VLAN ルーティング法や，第 4章で提案した VLAN ルーティング法の改良手法を用いることで，イーサネットにおいてもループを含むトポロジを構築可能となるが，同時に，通常のイーサネットでは考慮する必要のなかったデッドロックが問題となってくる．デッドロックが発生することで，ネットワークのスループットが劇的に低下する場合がある．

例えば，図 5.3において，以下のフレーム転送が同時に発生したとする．一般的なイーサネットスイッチは，VLAN ID に対応した仮想チャネルなどは持たないため，この場合，4 つの転送パス間にデッドロックを発生させる要因となる (物理) チャネル循環依存が存在する．

- ホスト 0 から VLAN A を用いてホスト 3 へ
- ホスト 1 から VLAN A を用いてホスト 2 へ
- ホスト 2 から VLAN B を用いてホスト 1 へ



(a) 2×2 メッシュ　(b) VLAN A　(c) VLAN B

図 5.3　デッドロックを引き起こすフレーム転送

- ホスト 3 から VLAN B を用いてホスト 0 へ

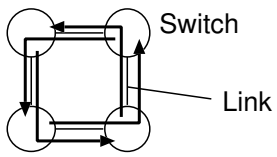
一般に，イーサネットのフレーム転送においてスイッチやネットワークインタフェースのバッファが一杯である場合，フレームは破棄される．この際，通常であれば，TCP 等の上位層の End-to-End 再送制御プロトコルが破棄フレームの再送処理を行うため，ネットワークのスループットは低下するものの，デッドロックとなるような転送であっても実際には問題は発生しない．ところが，SAN 等で行われているのと同様に，IEEE 802.3x のリンクレベルフロー制御を用いた場合，5.1.2節で検証したようにフレームはほとんど破棄されない．リンクレベルフロー制御は転送パスが重なった際のスループット低下を抑制する効果があるが，循環依存のあるパスでフレームを転送した場合，イーサネットにおいてもデッドロックが発生する．

フレーム間のデッドロックの問題を回避するためには，一般にデッドロックフリーの固定ルーティングアルゴリズムが有効である．このようなアルゴリズムでは，デッドロックフリーを保証するために，チャネル依存グラフ (Channel Dependency Graph, CDG) においてチャネル間の循環依存をすべて除去する，という操作を行う．イーサネットのように仮想チャネルを持たないネットワーク上ですべての循環依存を除去するデッドロックフリールーティングアルゴリズムは，並列計算機や SAN 向けに数多く提案されている [62]．例えば，*k*-ary *n*-cube (メッシュ，トーラス) における次元順ルーティング (Dimension-order Routing)[23][63] では，図 5.3において，ホスト 0 から 3，および 2 から 1 への転送で VLAN B，1 から 2，および 3 から 0 への転送で VLAN A を用いることによりデッドロックを回避する．

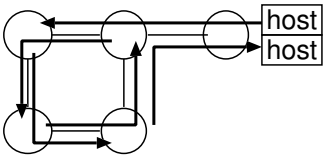
5.2.2　デッドロックの評価実験

イーサネットにおけるデッドロックの影響を検証するため，複数のフレーム転送パス間のチャネル循環性の有無が転送バンド幅に与える影響を測定した．なお，ここでは，デッドロックの問題が VLAN ルーティング法の実現方法によらない一般的な問題であることを示すために，従来のホストで VLAN ID を付与する VLAN ルーティング法とスイッチタグ法の両方を対象として実験した．

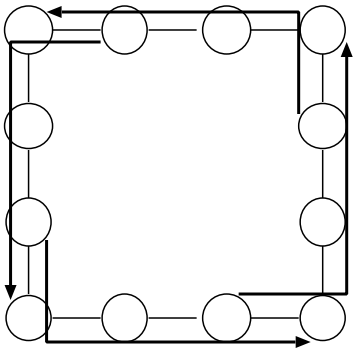
図 5.4に示す計 6 種類のフレーム転送パターンにおいて，各転送パス間の TCP および UDP の転送バンド幅およびパケット消失率を測定した．図 5.4において，(a), (c), (e) の各転送パターンではデッドロックを引き起こすパス間の循環構造が含まれているが，それぞれに対応する (b), (d), (f) の各パターンでは循環は形成されていない．1 つのチャネル (リンク) を使用するパスはどのパターンでも最大 2 個であり，パスのホップ数も対応するパターン対間（(a) と (b)，(c) と (d)，(e) と (f)）で等しいため，循環性の有無以外に各パターン対間の条件に違いはない．測定プログラムには Tperf 1.4[64] を用い，図 5.4(b) に示したように，送信プロセスと受信プロセスは同じスイッ



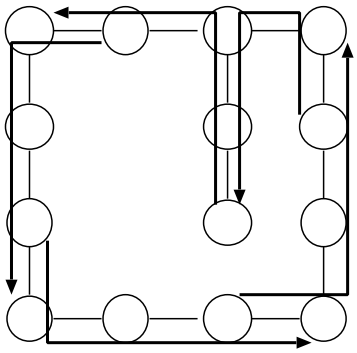
(a) Cyclic, 4sw



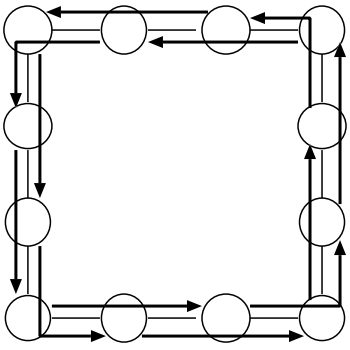
(b) Acyclic, 5sw



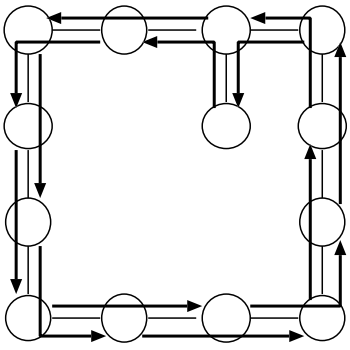
(c) Cyclic, indirect 12sw



(d) Acyclic, indirect 14sw



(e) Cyclic, 12sw



(f) Acyclic, 13sw

図 5.4　循環および非循環のフレーム転送パターン

表 5.4　各転送パターンにおけるバンド幅とフレーム消失率 (VLAN ルーティング法)

	FC None		FC All		FC Host		FC SW	
(a)/UDP	364	61.8%	0.324	28.3%	477	0.0589%	0.551	99.9%
(b)/UDP	394	58.7%	317	0.275%	506	7.11%	230	75.9%
(c)/UDP	366	61.6%	470	0.0389%	477	0.163%	0.527	99.9%
(d)/UDP	438	54.1%	475	0.0607%	507	6.73%	248	74.0%
(e)/UDP	356	62.7%	2.14	9.17%	478	0.342%	1.20	99.9%
(f)/UDP	361	62.1%	205	0.0554%	494	4.18%	106	88.8%
(a)/TCP	465	-	27.3	-	457	-	75.7	-
(b)/TCP	487	-	324	-	442	-	346	-
(c)/TCP	466	-	469	-	466	-	119	-
(d)/TCP	487	-	469	-	466	-	428	-
(e)/TCP	465	-	56.7	-	464	-	129	-
(f)/TCP	469	-	177	-	430	-	237	-

チに接続された異なるホストで起動した．評価環境は，5.1.2節の実験で用いたクラスタ 2 である．

ホストで VLAN ID を付与する従来の VLAN ルーティング法を用いた場合と，スイッチタグ法を用いた場合の測定結果を表 5.4, 表 5.5にそれぞれ示す．各表において，バンド幅 (単位 Mbps) の値は各パスごとの測定結果の平均値であり，%付きの数値はフレームの消失率である．また，リンクレベル PAUSE フロー制御の影響を調べるため，4 種類のフロー制御パターンについてそれぞれ測定した．“FC None” はフロー制御を使用しない場合，“FC All” は全てのリンクでフロー制御を使用した場合であり，さらに，“FC Host” および “FC SW” はそれぞれ，ホスト-スイッチ間のリンクにのみフロー制御を用いた場合，スイッチ間リンクにのみフロー制御を用いた場合である．

表 5.4および表 5.5から，循環を形成する転送パターン ((a), (c), (e)) では，ほとんどの場合において循環のないパターン ((b), (d), (f)) に比べてバンド幅が低下していることがわかる．特に，スイッチ間のリンクレベルフロー制御を使用している場合 (“FC All” および “FC SW”) に，バンド幅が著しく低くなっており，多くがゼロに近い値となった．これは，従来の VLAN ルーティング法，スイッチタグ法のいずれを用いた場合についても当てはまることから，VLAN ルーティング法を適用した場合の一般的な現象であると言える．

なお，これらのパターンの多くで，測定開始後すぐにネットワークの状態が不安定になり，プログラムが終了しなかった．プログラムの強制終了後もネットワークは回復せず，スイッチをリセットしない限りノード間の通信が全くできない状態が続いた．これは，フロー制御によりパケットの破棄が抑制され，実際に循環依存によるデッドロックが発生して回復不可能になったためと考えられる．

この現象について解析するため，GtrcNET-1[65][66] を用いて図 5.4(a) のパターンと同様の転送実験を行った．GtrcNET-1 は，搭載する FPGA により機能をプログラム可能なネットワーク試験用装置であり，ネットワークの遅延やバンド幅の計測，トラフィックのモニタリング，流量制御などをギガビット・イーサネットのワイヤレートで行うことができる．スイッチ間で転送されているフレームを GtrcNET-1 でキャプチャして解析したところ，フロー制御設定を “FC All” または “FC SW” にしたとき，循環を形成する各リンクにおいてフロー制御用の PAUSE フレームが大量に転送されており，通常のフレームがほとんど転送されていない状態であることが判明した．ここで，

表 5.5　各転送パターンにおけるバンド幅とフレーム消失率 (スイッチタグ法)

	FC None		FC All		FC Host		FC SW	
(a)/UDP	345	63.9%	209	23.2%	477	0.0666%	0.256	100%
(b)/UDP	403	57.8%	318	0.0209%	506	6.93%	230	75.9%
(c)/UDP	339	64.5%	465	0.0859%	477	0.110%	0.300	100%
(d)/UDP	344	64.0%	477	0.0228%	478	0.0419%	247	74.1%
(e)/UDP	349	63.5%	1.92	10.7%	479	1.19%	19.2	97.9%
(f)/UDP	378	60.5%	167	0%	487	3.00%	109	88.6%
(a)/TCP	445	-	0.772	-	455	-	90.4	-
(b)/TCP	465	-	318	-	441	-	345	-
(c)/TCP	462	-	469	-	466	-	401	-
(d)/TCP	443	-	464	-	470	-	428	-
(e)/TCP	466	-	47.8	-	447	-	154	-
(f)/TCP	472	-	158	-	417	-	227	-

PAUSE フレームは VLAN とは無関係に転送されるため，これは PAUSE フレーム転送の循環依存によってデッドロックが発生し，他のフレームが全く転送できない状態であると考えられる．

並列計算機の結合網や SAN では，スイッチング技術としてカットスルーもしくはワームホールルーティングを用いている．一方，イーサネットスイッチの大半はストアアンドフォワード方式を用いているため，SAN などに比べてデッドロックは発生しにくいと考えられる．しかし，この結果からは，VLAN によりループを含むトポロジを導入し，リンクレベルのフロー制御を用いた場合には，イーサネットでもデッドロックが現実問題として十分発生し得ることがわかる．

さらに，表 5.4および表 5.5の結果から，スイッチ間のフロー制御を使用しない場合 (“FC None” および “FC Host”) においても，非循環の転送パターンでは循環を含むパターンに比べて 2 割程度バンド幅が向上しており，循環を含むパターンでは，UDP による転送において大量のパケット消失が発生している．

これらの結果から，VLAN ルーティング法等で VLAN を用いてループ構造を含むトポロジを構築する場合，デッドロックフリーを満たすパス集合を選択することが，リンクバンド幅を効率的に使うために非常に有効であることがわかる．特に，IEEE 802.3x のリンクレベル PAUSE フロー制御を使用する場合，デッドロックフリールーティングを用いることが PAUSE フレーム間の循環依存を防ぐために必須となる．

5.3　トポロジの性能決定要因

本節では，次の 5.4節への準備として，イーサネットに VLAN ルーティング法を適用した際のトポロジおよびルーティングの性能を決定する要因についてまとめる．

5.3.1　パスのホップ数

並列計算機の結合網やシステムエリアネットワーク (SAN) では，スイッチング技術としてカットスルー方式もしくはワームホールルーティング [23] を用いている場合が多いが，イーサネッ

トスイッチの大半はストアアンドフォワード方式を採用している．ストアアンドフォワード方式は，カットスルー方式に比べて遅延が大きいため，一般にイーサネットでは SAN よりもスイッチング遅延がかなり大きくなる傾向にある．例えば，典型的な SAN のスイッチである RHiNET-2 スイッチ [67][68] は，スイッチングに最小 270nsec しか必要としないのに対し [69]，低コストな 1000BASE-T のイーサネットスイッチの遅延はその約 10 倍 (2μsec から 4μsec 程度) である．

SAN や並列計算機の結合網では，転送パスの平均ホップ数がルーティングアルゴリズムの性能に大きな影響を与えることが知られている [23]．イーサネットの場合，ストアアンドフォワード方式でフレームを転送するため，各スイッチでの遅延がすべて加算され，パスのホップ数は SAN の場合に比べてより重要な性能決定要因になると考えられる．そのため，トポロジおよびルーティングの設計においては平均ホップ数が小さくなるようにし，特にルーティングの設計においてはなるべく最短パスを用いるようにすることが望ましい．

パスのホップ数が転送バンド幅やレイテンシに与える影響については，6.2.2節で評価する．

5.3.2 パスの多重度

MPI[70] 等のメッセージパッシングモデルによる並列プログラミングでは，マルチキャストに代表される集団通信がしばしば用いられる．このため，QsNET[7][8] 等，SAN の中にはハードウェアによるマルチキャストをサポートしているものもあり，ブロードキャストやマルチキャスト操作において転送されるフレーム数を削減することができる [62].

一方，イーサネットでもマルチキャストはサポートされているが，使用する宛先アドレスやフレームのフィルタリングの管理をアプリケーション側で行う必要があり，SAN に比べて設定が繁雑である．そのため，現在一般に用いられている MPI 等の通信ライブラリでは使用されておらず，1 対 1 通信を繰り返し行うことでマルチキャストが実現されている．MPI では，MPI_Alltoall や，MPI_Reduce，MPI_Barrier 等のさまざまな集団通信操作が用いられ，これらの性質がトポロジやルーティングアルゴリズムを設計する上でも重要な要素となる．中でも，一度に大量の通信が発生する MPI_Alltoall (全対全通信) では，すべてのプロセスが自分以外のすべてのプロセスとデータを交換するため，1 対 1 通信をもとにした実装では最適化が困難である．よって，トポロジおよびルーティングを設計する際は，全対全通信において 1 つのチャネル (リンク) に重なるパス数 (多重度) の最大値がなるべく小さくなるようにパスを分散させることが望ましい．

また，複数のパスがリンク上に重なる状況においては，5.1.2節で評価した通り，IEEE 802.3x 標準のリンクレベルフロー制御を有効にすることで実効バンド幅の低下を防ぐことが可能である．

5.3.3 デッドロックフリー性

5.2.2節で検証した通り，VLAN ルーティング法によってループ構造を含むトポロジを構築する場合，デッドロックフリーを満たすパス集合をとるようにルーティングアルゴリズムを設計するのが望ましい．特に，IEEE 802.3x リンクレベルフロー制御を使用する場合は，デッドロックフリールーティングを用いることが必須である．

5.4 主要なトポロジにおける VLAN 割り当て手法

本節では，前節までの検証結果および検討を踏まえ，並列計算機や SAN で採用されている典型的な並列計算向けトポロジを VLAN を利用して構築する例を示す．従来の VLAN ルーティング法と，提案したスイッチタグ法，VLAN リネーミング法のそれぞれについて，各トポロジ上でのルーティングアルゴリズムの選択，VLAN 割り当ての方法について検討し，適用例を示してそれぞれ必要となる VLAN 数を示す．トポロジとしては，Fat ツリー，Myrinet-Clos 網，k-ary n-cube (メッシュおよびトーラス)，不規則トポロジを取り上げる．各トポロジにおいて必要となる VLAN 数を明示することにより，4.4.2節で示した最大システム規模がトポロジごとに算出できるようになる．

なお，4.4.3節で述べた通り，従来の VLAN ルーティング法，スイッチタグ法，および VLAN リネーミング法ではそれぞれ実装可能な固定ルーティングアルゴリズムの種類が異なっており，(従来の) VLAN ルーティング法とスイッチタグ法では $N \times N \mapsto P$ 型(注 1)，VLAN リネーミング法では $C \times N \mapsto C$ 型である．このため，以下の各節における VLAN 割り当て例では，説明のため，3 つの手法それぞれで同じルーティングアルゴリズムを実装するものとし，VLAN リネーミング法における割り当て例から順に説明する．

5.4.1 Fat ツリー

Fat ツリー (図 5.5) は，木構造ネットワークを多重化したトポロジであり，各レイヤにおけるスイッチの上位リンク数 u と下位リンク数 d，およびレイヤ数 r をパラメータとする拡張性に優れたトポロジである．ここで，$u \geq d$ であれば，フルバイセクションバンド幅を持つネットワークとなる．例として，SAN である QsNET[7][8] では，Fat ツリー (4,4,r) を基本トポロジとして採用している．Fat ツリーにおける最短ルーティングは，5.4.4節で述べる Up*/Down*ルーティングの条件を満たしているため，デッドロックフリーとなることが保証される．

図 5.6に示すように，任意のレイヤ数 r の Fat ツリー $(2, 4, r)$ に対し，4.2.1節の例に示した VLAN 設定 (図 4.2) を階層的に各スイッチに適用することで，VLAN リネーミング法によるルーティングを実装することができる．この場合，r の値にかかわらず VLAN A および B の 2 個の VLAN しか必要としないが，パスは各レイヤにおいてリンク間に等しく分散される．一般の Fat ツリー (u, d, r) の場合では，u 個の VLAN が必要となる．

一方，従来の VLAN ルーティング法では，例えば Fat ツリー (2,4,2) 上で VLAN リネーミング法と同じパス集合を構築する場合，図 5.7に示す 4 つのトポロジそれぞれに VLAN を割り当てる必要がある．一般の Fat ツリー (u, d, r) の場合では，u^r 個の VLAN が必要となる．これらは，スイッチタグ法の場合も全く同様であり，1 つのホストからのパスが複数の VLAN にまたがることができない点のみが異なる．

Fat ツリーおよび次節で述べる Myrinet-Clos 網においては，あらゆる最短ルーティングでデッドロックが発生しない．このため，どのパス (VLAN) を用いてフレームを転送するかについては，比較的選択の自由度が高い．特に Fat ツリーにおいては，VLAN トポロジとして図 5.7のように Fat ツリーを構成する木構造トポロジを用いる限り，あらゆる VLAN トポロジにおいてすべてのホスト間が最短ルーティングとなる．このため，パスの分散のみを考えて VLAN を選択すればよい．第 6章で述べるトポロジごとの性能評価では，送信側のホスト ID を VLAN 数で割った余りによって使用する VLAN ID を選択する手法を用いている．

(注 1)ただし，スイッチタグ法では，1 つのホストからのすべてのパスが単一の VLAN に含まれる場合に限られる．

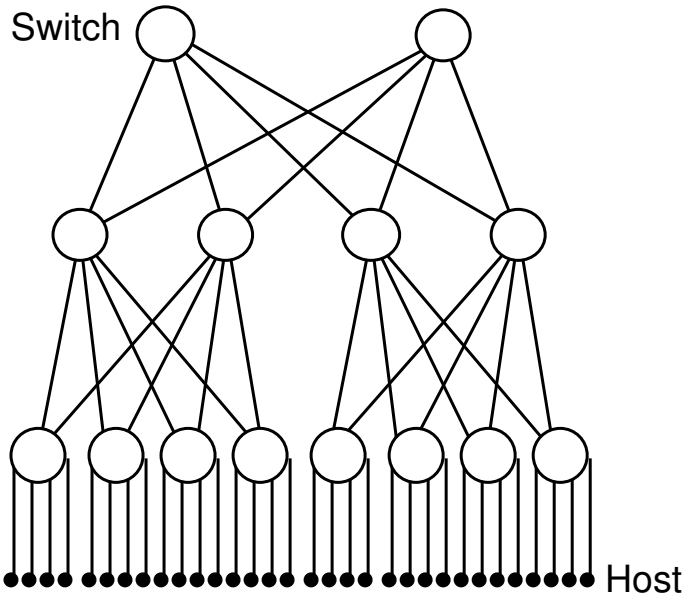


図 5.5　Fat ツリー (2,4,2)

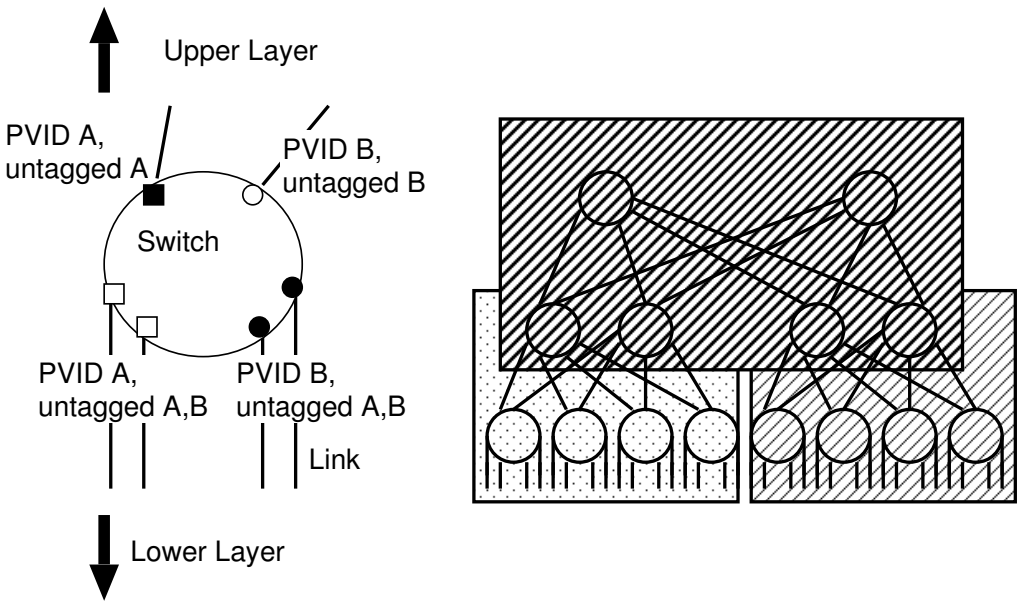


図 5.6　Fat ツリー (2,4,2) への VLAN リネーミング法の適用

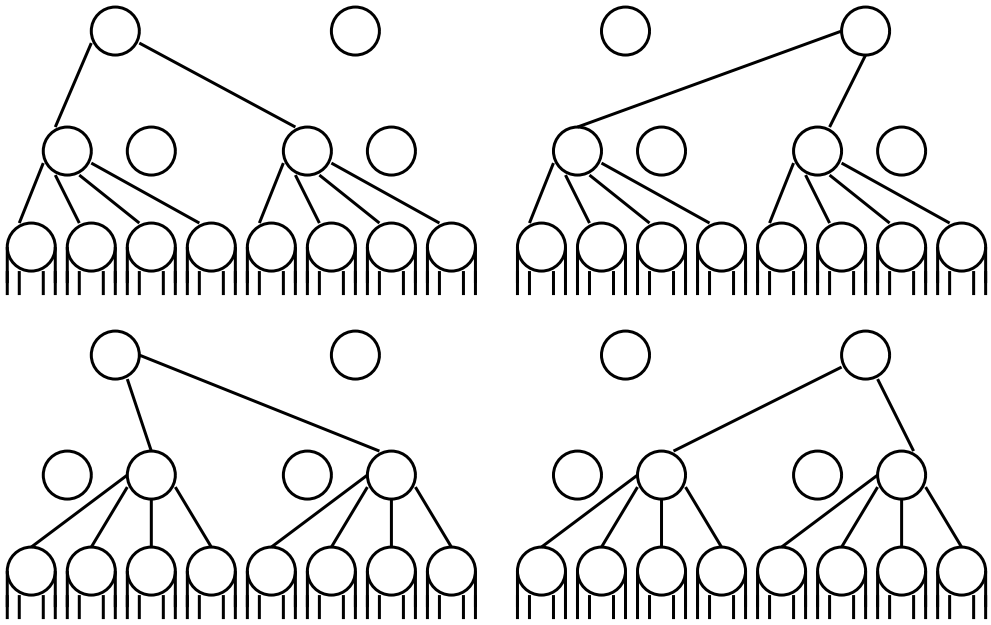


図 5.7　Fat ツリー (2,4,2) への VLAN ルーティング法の適用

5.4.2　Myrinet-Clos 網

Myrinet-Clos 網 (図 5.8) は，Fat ツリーと同様，木構造ネットワークを多重化したトポロジであり，SAN である Myrinet[5][6] のトポロジとして Myricom 社が推奨している，やはりフルバイセクションバンド幅を持つネットワークである．Fat ツリーとの違いは上位スイッチ側にもホストが接続されるという点であり，より密結合となる．ただし，上位スイッチにホストを接続することによって Fat ツリーの持つ拡張性が失われるため，より大規模なネットワークを構築する際にはレイヤ数を増やす必要があるが，この場合のトポロジは Fat ツリーとほぼ同じとなる．Myrinet-Clos 網における最短ルーティングは，Fat ツリーと同様 5.4.4節で述べる Up*/Down*ルーティングの条件を満たしているため，デッドロックフリーとなることが保証される．

図 5.9は，図 5.8の Myrinet-Clos (4×4) に VLAN リネーミング法を適用した場合のスイッチ 0 の VLAN 設定を示したものである．ホストから入力されたフレームは VLAN A でタグ付けされ，任意のポートに転送されて最短パスでのルーティングが行われる．一方，上位スイッチ側から入力されたフレームは，直上にあるスイッチ 4 から入力された場合はやはり VLAN A でタグ付けされるが，その他の上位スイッチ 5, 6, 7 から入力された場合はそれぞれ異なる VLAN B, C, D でタグ付けされ，ホストに接続されたポートにのみ転送される．同様の設定を他のスイッチに対しても行うことにより，デッドロックフリーを保証しつつ最短ルーティングを行うことができる．この場合，VLAN A〜D の 4 個の VLAN を必要とするが，パスは各リンクに等しく分散される．なお，Myrinet-Clos 網には，レイヤ数が増えた場合に Fat ツリー (u, d, r) のようなトポロジの記述法が存在しないが，便宜上，図 5.8のトポロジを上位リンク数 (= 下位リンク数) $u = 4$，レイヤ数 $r = 1$ として Myrinet-Clos (4,1) と表記することにすると，一般の Myrinet-Clos (u, r) の場合では，u 個の VLAN が必要となる．

一方，従来の VLAN ルーティング法では，例えば Myrinet-Clos (4×4) 上で VLAN リネーミング法と同じパス集合を構築する場合，図 5.10に示す 4 つのトポロジそれぞれに VLAN を割り当てる必要がある．一般の Myrinet-Clos (u, r) の場合では，u^r 個の VLAN が必要となる．これらは，ス

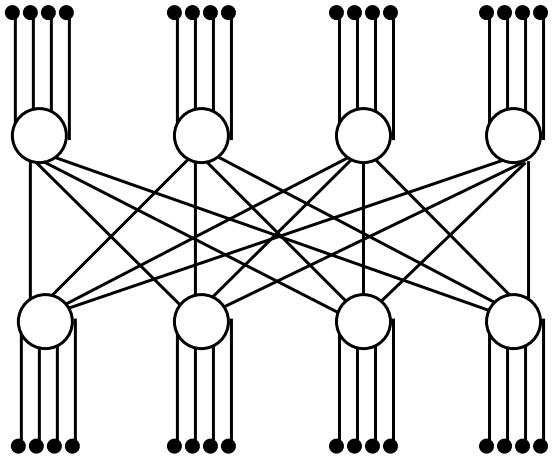


図 5.8　Myrinet-Clos (4×4)

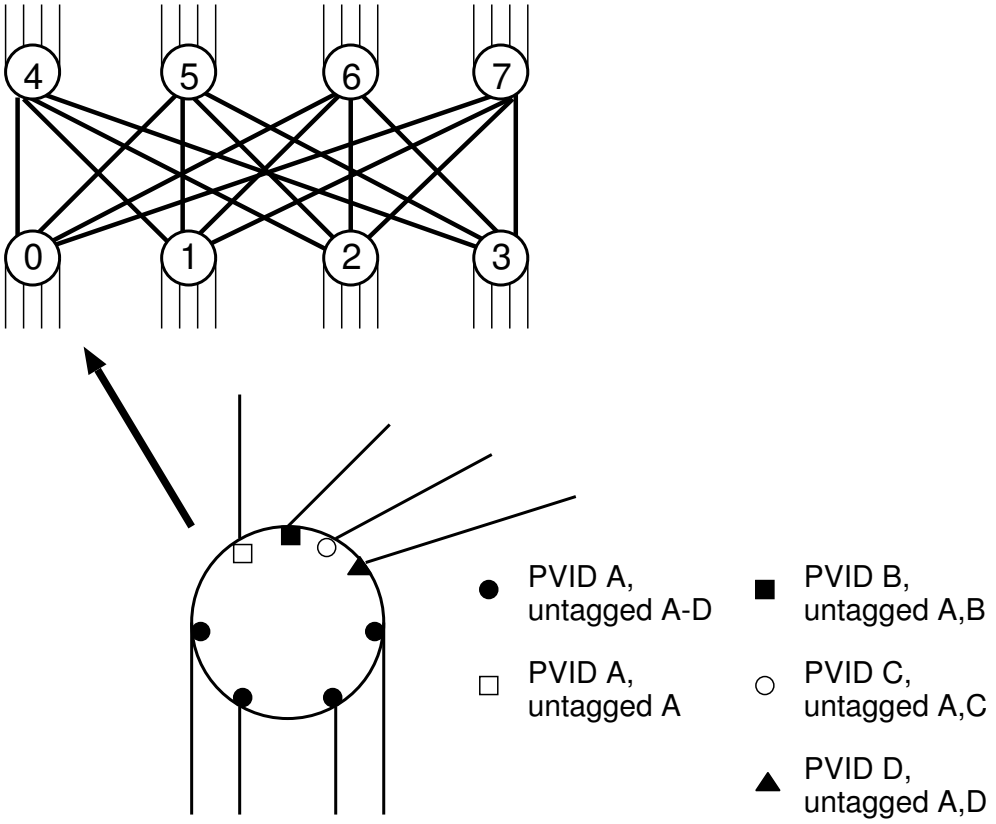


図 5.9　Myrinet-Clos (4×4) への VLAN リネーミング法の適用

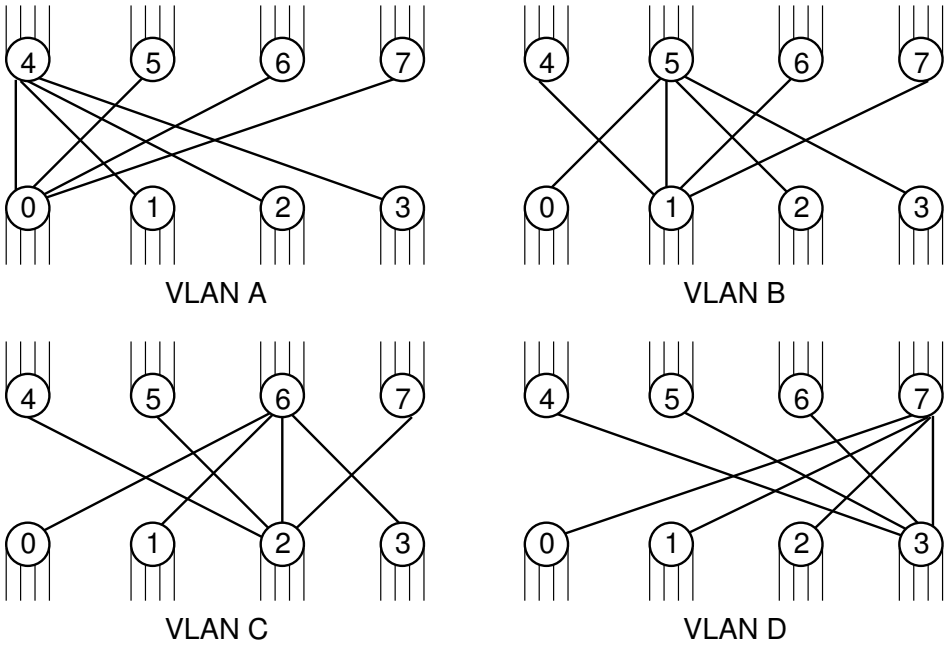


図 5.10　Myrinet-Clos (4×4) への VLAN ルーティング法の適用

イッチタグ法の場合も同様であり，1 つのホストからのパスが複数の VLAN にまたがることができない点のみが異なる．

Myrinet-Clos 網では，Fat ツリーと同様あらゆる最短ルーティングでデッドロックが発生しない．しかし，Fat ツリーと違い Myrinet-Clos 網では，各 VLAN トポロジにおいてすべてのホスト間が最短ルーティングとなるわけではない．例えば，上で述べた VLAN ルーティング法およびスイッチタグ法を Myrinet-Clos (4×4) に対する VLAN の割り当て方法 (図 5.10) では，最短ルーティングとなる VLAN ID の選択は一意に定まる．すなわち，図 5.10において，スイッチ 0, 4 に接続されたホストは VLAN A を用いてフレームを送信すればよい．同様に，スイッチ 1, 5, スイッチ 2, 6, スイッチ 3, 7 に接続されたホストはそれぞれ VLAN B, C, D を用いる．

5.4.3 *k*-ary *n*-cube

k-ary *n*-cube は，図 5.11や図 5.12に示すような格子状ネットワークの一般形である．図 5.12の上下および左右の切れているリンクは実際にはそれぞれつながっており，wrap-around リンクと呼ばれる．*k*-ary *n*-cube のうち，この wrap-around リンクを含むものがトーラス (torus，図 5.12)，含まないものがメッシュ(mesh，図 5.11) である．Fat ツリーや Myrinet-Clos 網と比較すると，フルバイセクションバンド幅こそ持たないものの，トポロジの直径や平均パスホップ数，およびトポロジ内のスイッチ数およびリンク数を少なく抑えることが可能である．このため，主要な通信が 1 対 1 通信で，集団通信が多く発生しないような場合に特に適している．例として，IBM のスーパーコンピュータ Blue Gene/L[71][72] では，ノード間の 1 対 1 通信専用ネットワークに 3 次元トーラス (*k*-ary 3-cube トーラス) が用いられている．なお，クラスタネットワークにおいて *k*-ary *n*-cube を構築する場合は，格子点である各スイッチにそれぞれ何台かずつのホストを接続する．

Fat ツリーや Myrinet-Clos 網と違い，*k*-ary *n*-cube における最短ルーティングは，一般にはデッドロックフリーとはならない．このため，*k*-ary *n*-cube において，デッドロックを回避しつつバラ

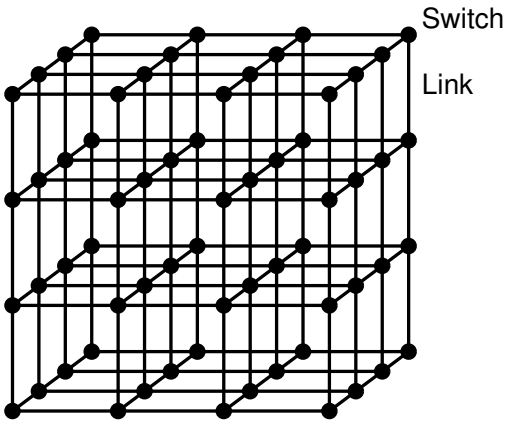


図 5.11　4-ary 3-cube メッシュ (4×4×4 3 次元メッシュ)

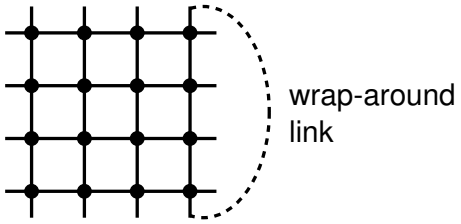


図 5.12　4-ary 2-cube トーラス (4×4 2 次元トーラス)

ンスよく分散されたパス集合を構築するための手法として，次元順ルーティング (Dimension-order Routing)[23][63] が用いられる．次元順ルーティングは，各フレームを X 方向，Y 方向，Z 方向と次元順にそれぞれ必要ホップ数転送することにより，最短パスを取ることができるデッドロックフリー固定ルーティングアルゴリズムである．以下では，*k*-ary *n*-cube メッシュおよびトーラストポロジに次元順ルーティングを実装する場合の VLAN 設定方法を示す．

図 5.13は，3 次元メッシュ(*k*-ary 3-cube メッシュ) に VLAN リネーミング法を適用した場合の各スイッチにおける VLAN 設定を示したものである．図に示す通り，Z 次元入力ポート (PVID C) に入力されたフレームは，Z 次元出力ポートもしくはホストに転送される．同様に，Y 次元入力ポート (PVID B) に入力されたフレームは，Y, Z 次元出力ポートもしくはホストに転送される．また，X 次元入力ポートまたはホストと接続されたポート (PVID A) に入力されたフレームは，X, Y, Z 次元出力ポートもしくはホストに転送される．このように，*k*-ary *n*-cube メッシュトポロジにおける次元順ルーティングでは，VLAN は各次元に対して 1 つ，つまり *n* 個必要となる．

一方，従来の VLAN ルーティング法およびスイッチタグ法において，次元順ルーティングを実装するためには，以下のように VLAN 割当てを行う [73]．図 5.13の 3 次元メッシュにおいて，矢印で示されるスイッチ (S とする) に接続されたホストは，図 5.14(a) に示すツリー状トポロジを用いることにより，どの宛先ホストに対しても次元順ルーティングに従う最短パスでフレームを送信することができる．このトポロジは，スイッチ S を通る X 次元方向のリンク (集合) L_X (図 5.14(b))，L_X と交差する Y 次元方向のリンク集合 L_Y (図 5.14(c))，L_Y と交差する Z 次元方向のリンク集合 L_Z (図 5.14(d))(注 2)で構成される．このようなトポロジはスイッチの Y 座標と Z 座標の組 (y, z) に対し 1 つ定まり，各トポロジに VLAN を割り当てることにより，全ホスト間で次元順ルーティン

(注 2)L_Z は Z 次元方向の全リンクで構成される．

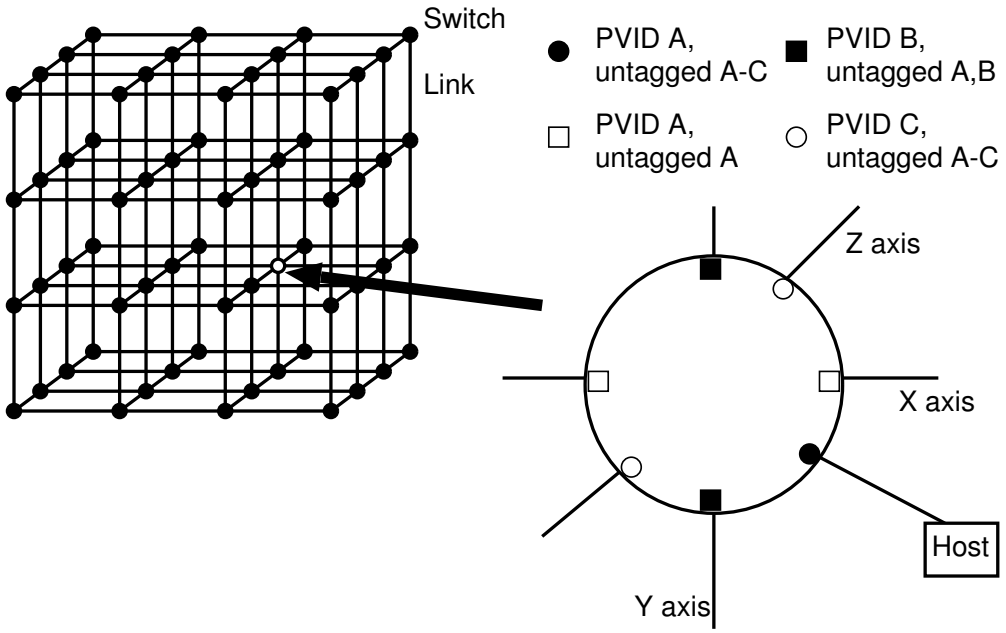


図 5.13　3 次元メッシュへの VLAN リネーミング法の適用

グに従うパス集合を構築することができる．よって，図 5.13の 3 次元メッシュ(4-ary 3-cube メッシュ) では $4^2 = 16$ 個の VLAN が必要となる．一般に k-ary n-cube メッシュの場合では，k^{n-1} 個の VLAN が必要となる．なお，本手法の詳細については，付録 Aで述べる．

次に，一般に並列計算機の相互結合網において次元順ルーティングをトーラストポロジに適用する場合，デッドロックフリーを保証するために仮想チャネルが必要となる．ここで，イーサネットには仮想チャネルがないため，VLAN リネーミング法を適用する場合，スイッチ間に複数リンク (それぞれ CA リンク，CH リンクと呼ぶ) を設けておき，それぞれのリンクに異なる VLAN ID を割り当てる．

図 5.15の 1 次元トーラス (k-ary 1-cube トーラス，リングトポロジ) において，wrap-around リンクを通過する前のフレームは CA リンクを，wrap-around リンクを一度通過した (今後通過しない) フレームは CH リンクを利用して転送する．この制御によりチャネル循環依存は除去され，デッドロックフリーが保証される [62]．このように，k-ary n-cube トーラストポロジにおける次元順ルーティングでは，VLAN は各次元に対して 3 つ (図 5.15の B, C, D) 必要であるが，メッシュの場合と違いホストに接続されたポートに別の PVID を割り当てなければならないため (図 5.15の A)，必要な VLAN 数は $3n + 1$ 個となる．

一方，従来の VLAN ルーティング法を例えば 4×4 2 次元トーラス (4-ary 2-cube トーラス) に適用する場合，図 5.16の (a)〜(h) に示す 8 つのトポロジにそれぞれ VLAN を割り当てる [73]．VLAN リネーミング法の場合 (図 5.15) と同様，循環依存を除去するためにスイッチ間に複数リンクを設ける必要があるが，図では省略している．

図 5.16において，例えば矢印で示すスイッチ S に接続されたホストは，宛先に応じて VLAN B か F のいずれかを選択することにより，すべての宛先ホストに対して次元順ルーティングに従う最短パスでフレームを送信することができる．一般に k-ary n-cube トーラスの場合では，$2k^{n-1}$ 個の VLAN が必要となる．本手法の詳細については，付録 Aで述べる．

なお，スイッチタグ法の場合，あるホストからのフレーム送信に用いる VLAN は 1 つに固定さ

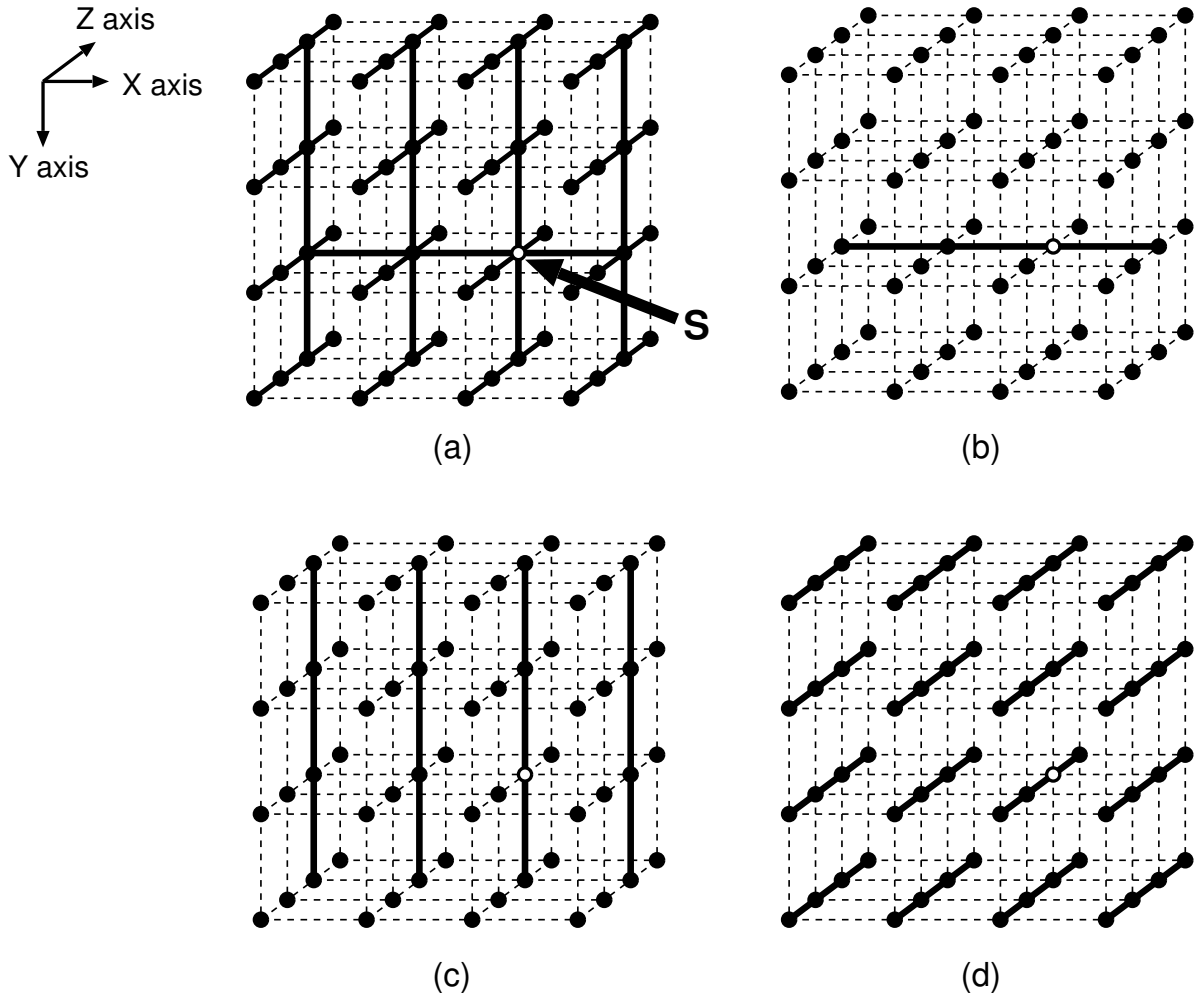


図 5.14　3 次元メッシュへの VLAN ルーティング法の適用

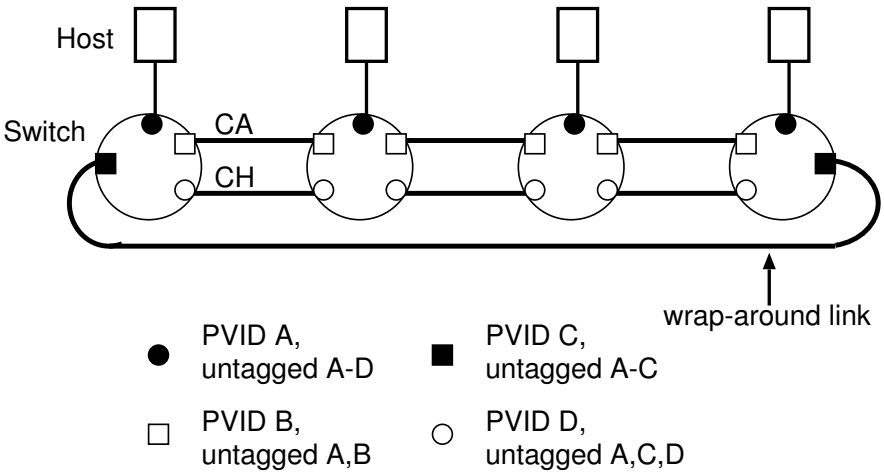


図 5.15　1 次元トーラスへの VLAN リネーミング法の適用

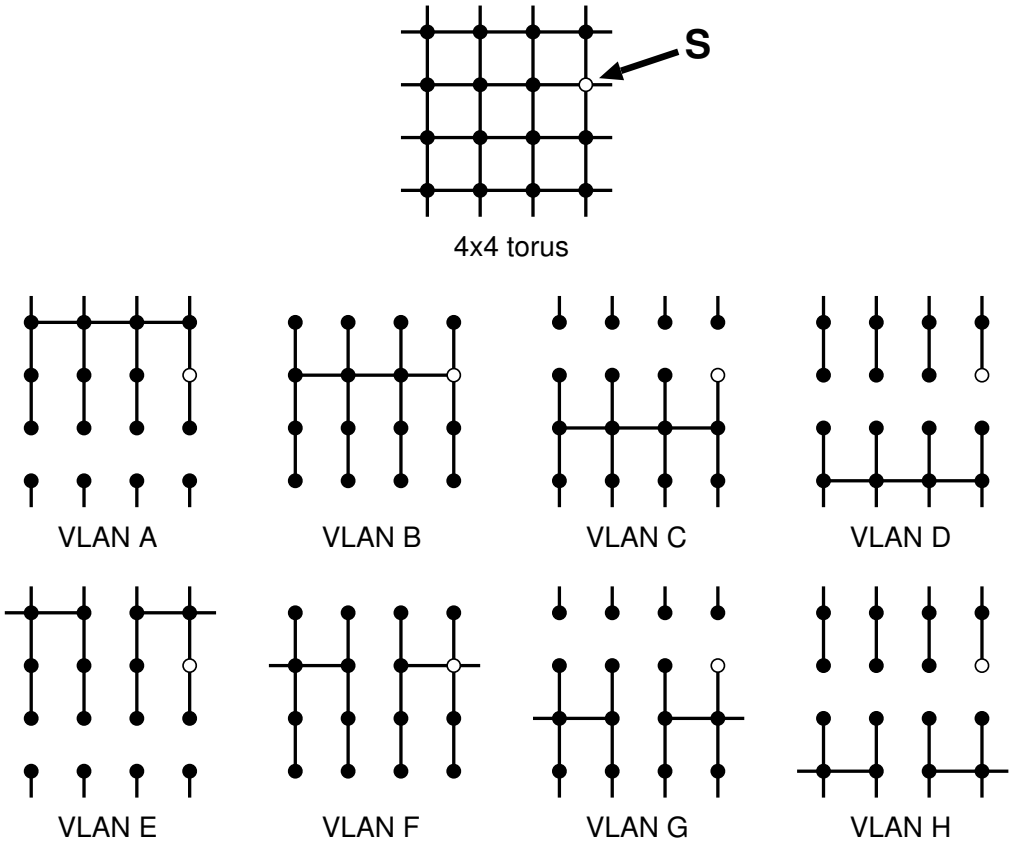


図 5.16　2 次元トーラスへの VLAN ルーティング法の適用

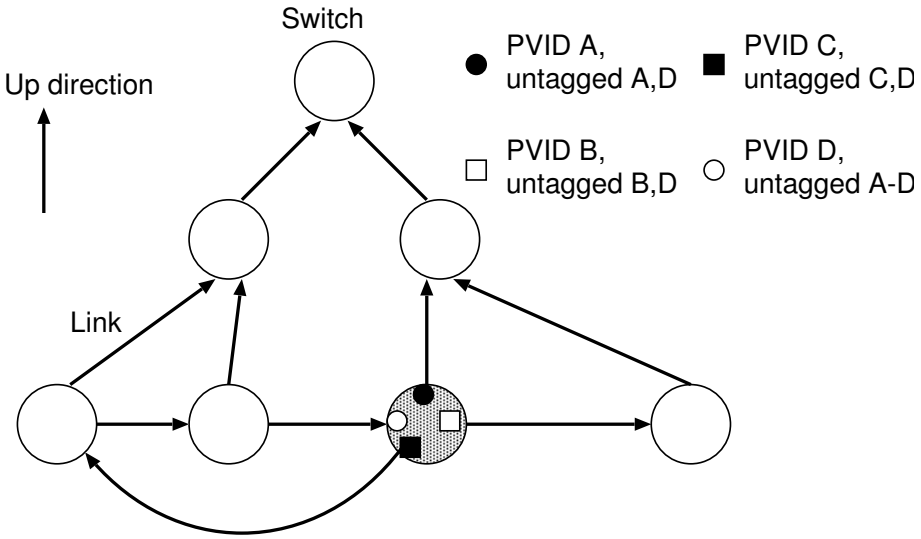


図 5.17　Up*/Down*ルーティングにおける VLAN リネーミング法

れるため，宛先に応じて使用する VLAN を選択する上記の方法は適用できず，次元順ルーティングを実装するにはさらに多くの VLAN が必要となる．

5.4.4　不規則トポロジ

任意の不規則なトポロジ上でも使用可能なルーティングアルゴリズムとして，Up*/Down*ルーティング [49] がある．Up*/Down*ルーティングは，スパニングツリーに基づいた有向グラフを用いることで，任意のトポロジに適用することができる典型的なデッドロックフリー固定ルーティングアルゴリズムである．各リンクは単方向チャネル 2 本で構成されており，図 5.17のように，ツリーのルート方向へ向かうチャネルに Up 方向，リーフ方向へ向かうチャネルに Down 方向を割り当てる．すべてのフレームは，0 回以上 Up 方向に転送された後に，0 回以上 Down 方向に転送されることで宛先ホストまで到達する．Down 方向から Up 方向へのターンを行うことができないため，チャネル間の循環依存が除去され，デッドロックフリーが保証される．

不規則トポロジに VLAN リネーミング法を適用して Up*/Down*ルーティングを実装する場合，Down 方向から Up 方向へのフレームの転送を防ぐために各スイッチにおいて必要となる VLAN 数は，Down リンク数 +1 となる．

一方，従来の VLAN ルーティング法やスイッチタグ法の場合は，不規則なトポロジにおいて Up*/Down*ルーティングを実現するための VLAN 割当てを一般化するのは困難である．

第6章　評価および検討

本章では，イーサネットを用いたクラスタに第4章で提案したVLANルーティング法の改良手法を適用し，その有効性を検証した結果について述べる．

まず，コストパフォーマンスに優れる一般的なVLAN対応イーサネットスイッチにおいて，フレーム入出力時のVLANタグの処理がスループットおよびレイテンシに与える影響が小さいことを示す．さらに，これまでイーサネットではほとんど採用例のないFatツリーやトーラス等の並列計算機向けトポロジを実際に中規模クラスタ上に構築し，並列ベンチマーク等を用いてクラスタシステムとしての性能評価を行った結果を示す．最後に，各トポロジにおける必要VLAN数の比較等を通して，大規模クラスタへの適用可能性について検討する．

6.1　評価環境

評価に用いた環境は，国立情報学研究所 (NII) に設置されているノード32台，スイッチ16台からなるPCクラスタ (図6.1) であり，各ノードの仕様は5.1.2節で用いたクラスタ2 (表5.3) と同様である．クラスタには，クラスタシステムソフトウェアとしてSCore[57][58]バージョン5.8.2が搭載されている．SCoreはオープンソースのPCクラスタ向けシステムソフトウェアで，低レベル軽量通信ライブラリPM[11][12][19]やMPICH-1.2.5[60][61]をベースにしたMPIライブラリMPICH-SCoreを提供する．

提案手法の評価の方法としては，一般的にはシミュレーションによる評価も考えられるが，以下の理由でイーサネットを用いたクラスタのシステムレベルの評価にはシミュレーションは適さない．

- イーサネットスイッチは非常に多くの機能を実装しており複雑である．また，スイッチのバッファ不足によるフレームの廃棄も発生するため，イーサネットの正確な挙動をシミュレーションにより再現するのは不可能に近い．
- シミュレーションの場合，アプリケーションベンチマークの評価をとるにはホストの挙動も含めてシミュレーションを実行しなければならない．特にクラスタが大規模化しホストの数が多くなるとこれには膨大な時間がかかり，実用的でない．
- シミュレーションによるトラフィックパターン等の評価では，極端な性能差が出ることも多く，必ずしも実際のアプリケーション実行時の性能差を表していない．

このため，本評価では上記の中規模クラスタ環境を用いて基本性能およびアプリケーションベンチマーク性能を測定し，大規模クラスタへの適用についてはその実現可能性を検討するにとどめた．

図 6.1　評価に用いた NII 設置のクラスタ

6.2　基本性能評価

本節ではまず，イーサネットスイッチにおけるフレーム入出力時の VLAN タグ処理のオーバヘッドを測定し，コストパフォーマンスに優れる一般的な VLAN 対応スイッチにおいて，VLAN タグの処理がスループットおよびレイテンシに与える影響が小さいことを示す．次に，イーサネットを用いた並列計算向けトポロジの基礎評価として，2 ホスト間の MPI レベルのレイテンシおよびバンド幅を測定する．

6.2.1　VLAN タグ処理のオーバヘッド

VLAN 対応スイッチにおけるフレーム入力時の VLAN タグ付け，および出力時の VLAN タグ除去のオーバヘッドを測定した結果を表 6.1に示す．表の数値は，2 ホスト間の ping (ICMP echo request/reply メッセージ) を用いて，Dell PowerConnect 5324 におけるフレーム通過時間を GtrcNET-1 (5.2.2節参照) で各 300 回測定した際の最小値 (Min)・平均値 (Ave)・最大値 (Max) をそれぞれとったものである．ICMP echo request/reply メッセージのサイズはヘッダを含めて 64byte とした．よって，イーサネットフレームのデータサイズは IP データグラムのヘッダ 20byte を含めて 84byte である．

表 6.1において，U-U は VLAN を一切用いない場合，T-T はホストにおいて VLAN タグ付きフレームを送受信した場合 (従来の VLAN ルーティング法に相当)，RENAME はスイッチ内でのみ PVID に基づくルーティングを行う場合 (VLAN リネーミング法に相当) をそれぞれ示す．なお，ス

表 6.1 PowerConnect 5324 におけるフレーム通過遅延 (μsec)

	Min	Ave	Max
U-U	2.47	2.74	2.79
T-T	2.47	2.76	2.79
RENAME	2.47	2.75	2.79

表 6.2 PowerConnect 5324 を介した TCP/UDP 転送のバンド幅 (Mbps)

U-U (TCP)	941.1
T-T (TCP)	936.9
RENAME(TCP)	941.1
U-U (UDP)	957.0
T-T (UDP)	954.4
RENAME(UDP)	957.0

イッチタグ法では，あるパスにおいてホストと接続されていないスイッチを通過する場合は T-T に相当し，ホストと接続されたポートでのみ RENAME に相当する処理が行われる．結果より，VLAN タグ処理による遅延はほとんどないことがわかる．なお，スイッチタグ法，VLAN リネーミング法とも，ホストでの VLAN 処理は一切行わないため，ホストにおいて提案手法を導入することによるオーバヘッドは一切ない．

次に，Tperf 1.5[64] を用いた TCP/UDP 転送のバンド幅の測定結果を表 6.2に示す．結果より，ホストが VLAN タグ付きフレームを送受信する場合 (表の T-T) のみ，数 Mbps のバンド幅低下が見られる．これは，タグ付きフレームでは，フレーム全体に占めるペイロードの割合が VLAN タグ (4byte) の分だけ少なくなるためである．

一方で，RENAME 処理によるバンド幅の低下はないことがわかる．ここで，VLAN リネーミング法では，フレームはリンク上においては VLAN タグを含まない．つまり，VLAN リネーミング法では，VLAN の処理をスイッチ内でのみ行うことにより，従来の VLAN ルーティング法と違って VLAN の導入によるバンド幅低下を生じない．なお，スイッチタグ法の場合は，スイッチ間リンクにおいてタグ付きフレームが転送されるため，T-T の場合と同様にバンド幅は数 Mbps 低下する．

これらの結果から，提案手法であるスイッチタグ法，VLAN リネーミング法と，従来の VLAN ルーティング法の間に，VLAN タグ処理によるオーバヘッドの差はほとんどないと見なすことができる．このため，以降の評価では，スイッチタグ法を適用してトポロジを構築した場合に絞って性能測定を行った．

6.2.2 2 ホスト間の通信性能

イーサネットを用いた並列計算向けトポロジの基礎評価として，Intel MPI Benchmarks (IMB)[59] 2.3 を用いて，VLAN を用いない場合と，スイッチタグ法を用いてスイッチでタグ付けを行った場合の MPI レベルの遅延とバンド幅を測定した．MPI ライブラリには，SCore 5.8.2 に付属の MPICH-SCore を用いた．MPICH-SCore は，下位の軽量通信ライブラリ PM 上に構築されており，イーサネットの場合は PM/Ethernet[12][19] が使われる．

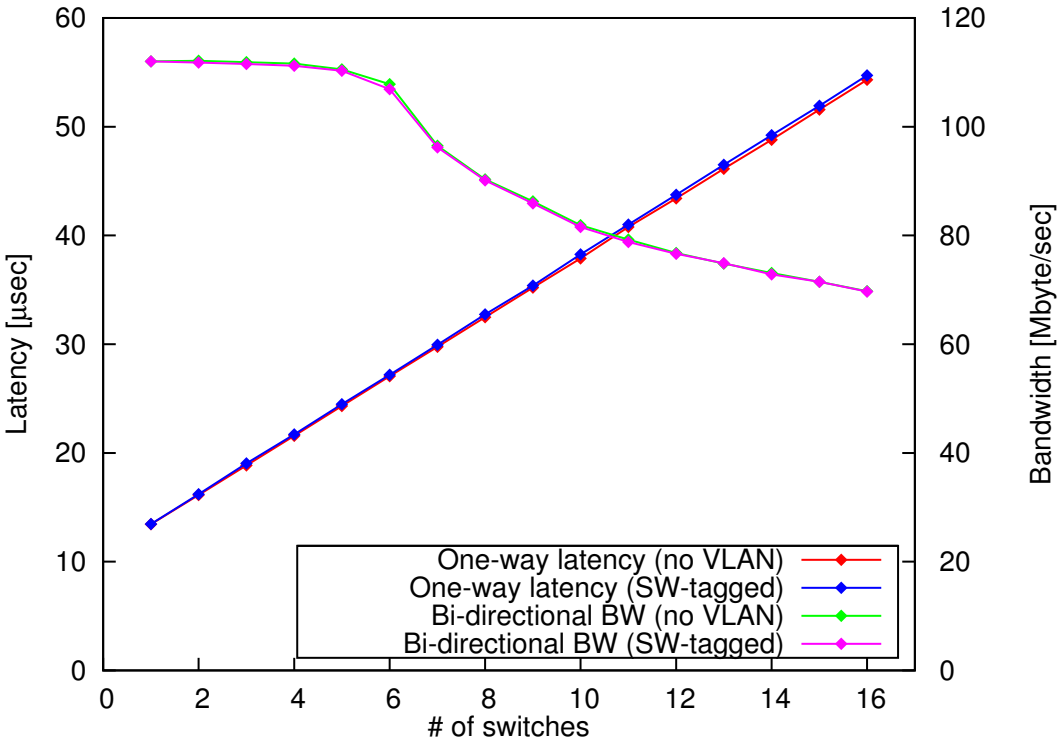


図 6.2　MPI 片道遅延と双方向バンド幅

図 6.2は，2 ホスト間の経由スイッチ数を変化させた際の，IMB PingPong テストによる MPI の片道遅延と，IMB PingPing テストによる双方向バンド幅の測定結果である．図より，片道遅延は経由スイッチ数の増加に伴いほぼ線形に増加している．また，双方向バンド幅も経由スイッチ数が 6 を越えたあたりから大きく低下しており，遅延だけでなくバンド幅もパスのホップ数による影響を大きく受けることがわかる．このようにバンド幅が低下する理由としては，PM/Ethernet ライブラリによる End-to-End の再送制御プロトコルによる影響が考えられる．宛先ホストからの Ack パケットを受信するまでの時間が長い場合，Ack を待たずに一度に送信できるパケット数を使い切って待ち状態となっている時間が発生する．経由スイッチ数が多くなるほど Ack を受信するまでの時間が長くなるため，経由スイッチ数が 6 を越えるとこの状態が発生するようになるものと考えられる．このような再送制御は TCP などでも行われるが，TCP には動的に送信 window のサイズを最適化する機能があるため，あるホップ数を越えたところから急に性能が低下するようなことはない．

一方で，図 6.2で VLAN を用いない場合とスイッチでタグ付けを行った場合の性能には差が見られないことから，スイッチでの VLAN タグ処理は性能にほとんど影響しないと言える．これは，6.2.1節で述べた結果とも一致する．残念ながら，PM/Ethernet は現時点では VLAN 処理に対応していないため，MPICH-SCore を用いた本評価では，従来の VLAN ルーティング法を用いた場合との比較は不可能である．しかし，5.2節において表 5.4および表 5.5に示した TCP/UDP のバンド幅測定結果より，スイッチタグ法と，従来のホストでタグ付けを行う VLAN ルーティング法との間にほとんど性能差はないと言える．

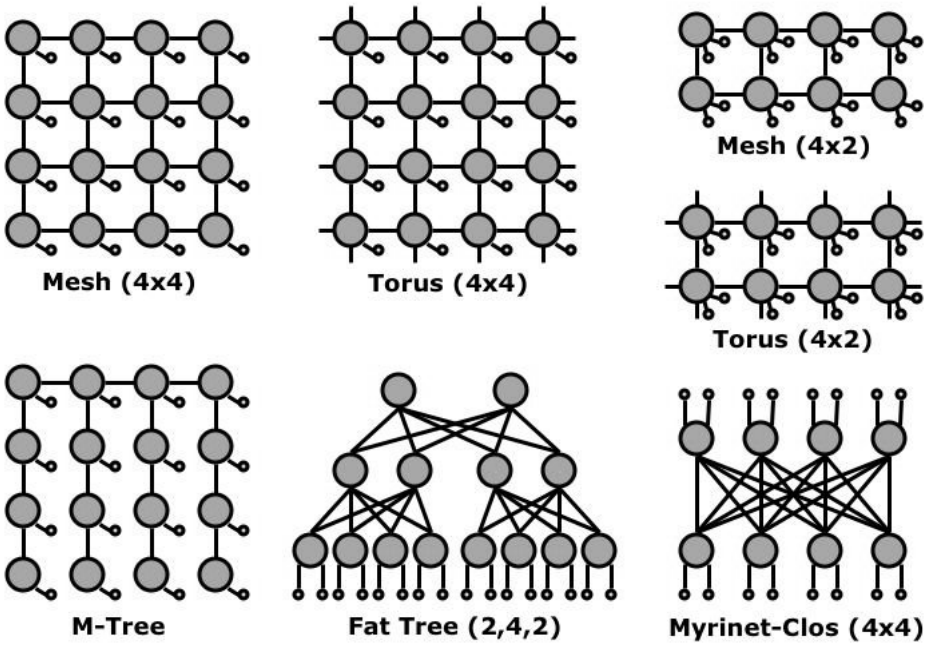


図 6.3　構築したトポロジ

6.3　クラスタシステムレベルの性能評価

本節では，提案手法を用いて並列計算向けの主要なトポロジを評価環境上に構築し，トラフィックパターンおよび並列ベンチマークを用いてクラスタシステムとしての性能評価を行った結果を示す．

6.3.1　構築したトポロジ

評価対象として，図 6.3に示すそれぞれ 16 ホストで構成されるトポロジをスイッチタグ法を適用して構築した．各トポロジにおける性能決定要因 (5.3節参照)，必要となる VLAN 数などをまとめたものを表 6.3に示す．表において，“#sw”，“#link”，“#VL” はそれぞれトポロジを構成するスイッチ数，リンク数 (ホスト-スイッチ間含む)，VLAN 数である．VLAN の割り当ては，5.4節で述べた方法に従って行った．また，“AH” はパスの経由スイッチ数の平均値，“MH” はその最大値であり，“CP” 値は，各トポロジにおいて 1 つのチャネルに重なるパス数の最大値を表す．例えば，各スイッチに 1 ホストのみを接続した 4 × 4 2 次元メッシュ上で次元順ルーティング (DOR) に従うパス集合を構築した場合，図 6.4に示すように，1 つのチャネルに最大で 16 のパスが重なる．

トポロジのうち “M-Tree” は，VLAN を用いない場合との比較のために導入した単純なツリー状トポロジである．使用するリンクの本数などの違いはあるが，表 6.3に示したように，これらのトポロジを 8 スイッチで構成されるトポロジと 14 または 16 スイッチで構成されるトポロジとに分類することで，それぞれのグループ内でおおよそ公平に比較できると考えられる．

表 6.3　評価に用いたトポロジの諸元

Topology	#sw	#link	#VL	AH	MH	CP
Mesh (4×2)	8	26	4	2.75	5	24
Torus (4×2)	8	28	4	2.50	3	32
Myrinet-Clos (4×4)	8	32	4	2.25	3	12
M-Tree	16	31	(1)	4.81	10	64
Mesh (4×4)	16	40	4	3.50	7	16
Torus (4×4)	16	48	4	3.00	5	12
Fat Tree (2,4,2)	14	40	4	3.75	5	16

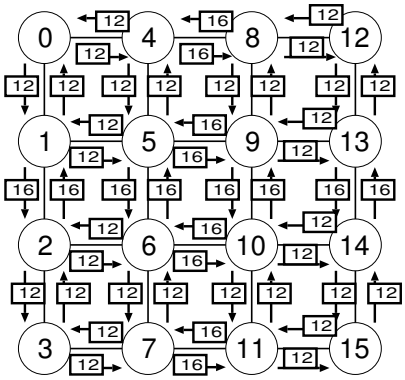


図 6.4　4×4 メッシュ上の各チャネルに重なるパス数

6.3.2　トラフィックパターンにおける通信性能

まず，相互結合網の評価で用いられる典型的な通信トラフィックパターンとして Bit-Reversal と Matrix Transpose[62] の 2 種類を使用し，ネットワーク全体のスループットを測定した．測定には Tperf 1.4 の UDP 転送を使用し，すべてのトポロジでホスト数は 16 とした．すなわち，Mesh (4×2) および Torus (4 × 2) ではスイッチあたり 2 台のホストを使用している．各ホストで送信プロセスと受信プロセスをそれぞれ起動し，UDP データグラムのサイズは最大の 1470byte とした．

図 6.5に，各トポロジの測定結果におけるすべての通信対のバンド幅の平均値を示す．比較のために，16 台のホスト全てを 1 つのスイッチに接続したフラットなトポロジ (Flat) でも測定を行った．なお，このようなフラットトポロジはあくまで比較対象としての理想的なものであり，多数のホストを接続する大規模クラスタでは実現不可能である．

図より，フラットトポロジと比較するとほとんどのトポロジで平均バンド幅は低下しているが，提案手法によるトポロジ (Mesh, Torus, Fat Tree, Myrinet-Clos) は，単純なツリー状トポロジ (M-Tree) に比べて高い平均バンド幅を達成している．特に，Torus (4 × 4) と Myrinet-Clos (4×4) がどちらのトラフィックパターンの場合にも高い性能を示した．

これらの平均バンド幅の測定結果は，パスの多重度により算出される値とほぼ一致する．例えば，4 × 4 2 次元メッシュにおける Bit-Reversal トラフィックでは，図 6.6に示すように，1 つのパスは最大で他の 2 つのパスとチャネルを共有する．すなわち，スイッチ 3 からスイッチ 12 へのパスは，スイッチ 11 からスイッチ 15 へのチャネルを他の 2 つのパスと共有している．データグラムサイズが 1470byte の場合，UDP の理論転送性能は約 958Mbps であるため，1 つのパスあたり

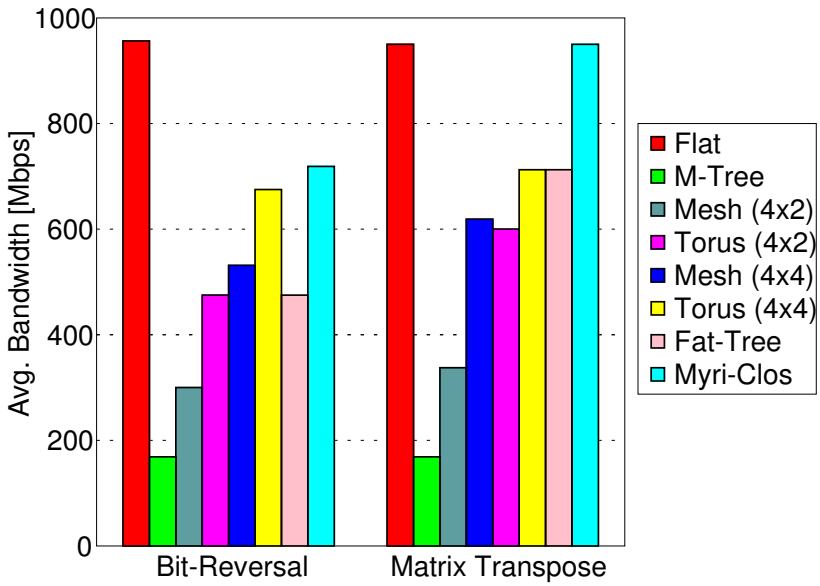


図 6.5　トラフィックパターンにおけるバンド幅

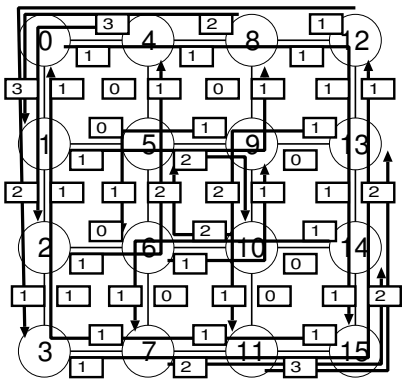


図 6.6　Bit-Reversal トラフィックにおける 4×4 メッシュ上の各チャネルに重なるパス数

のバンド幅は 319Mbps が上限となる．4×4 2 次元メッシュ上の Bit-Reversal トラフィックにおける全パスの平均では 521Mbps となり，測定値の 533.6Mbps と概ね一致している．このことは，他のトポロジについても同様に成立する．

なお，本測定では，パスの多重度が直接バンド幅に与える影響を明らかにするために UDP を用いて測定しているため，パケットのホップ数はほとんど性能に影響していない．一方，TCP や，PM のようなクラスタ内通信向けのプロトコルでは，Ack 等を用いてパケットの到着保証を提供する機構を備えているのが一般的である．6.2.2節で述べたように，PM では，ホップ数が増加することによってレイテンシおよびバンド幅が悪化する (図 6.2) ため，実際のアプリケーションを実行した場合には，ホップ数が性能に影響を与えると考えられる．

6.3.3　NAS 並列ベンチマーク性能

次に，NAS 並列ベンチマーク 3.2[74][75] を用いて，提案手法を適用して構築した各トポロジにおいてアプリケーション実行性能を測定した．各ベンチマークの問題サイズはクラス B，実行プロ

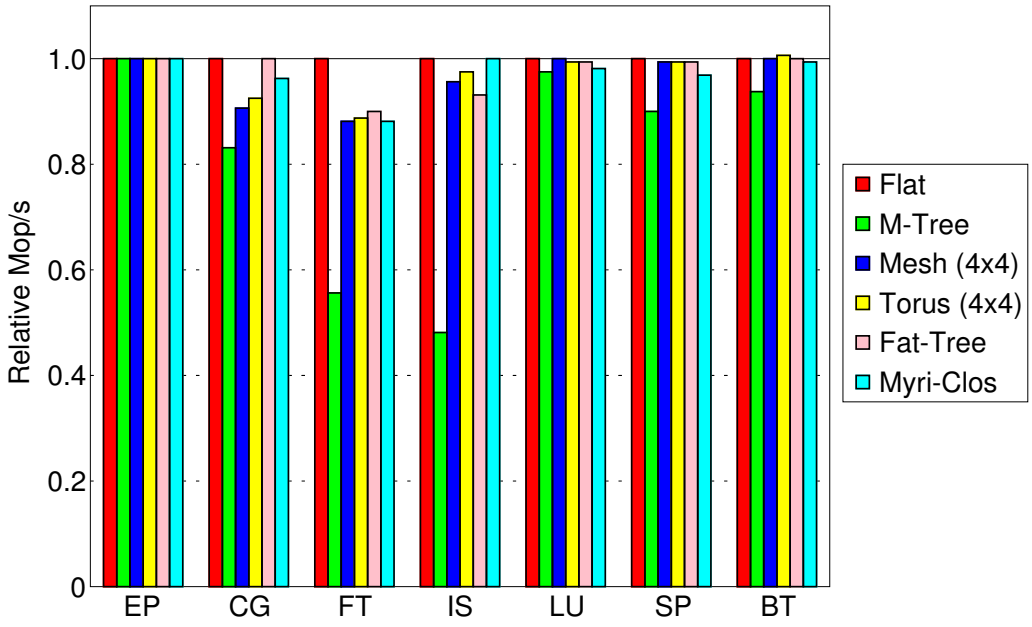


図 6.7　NAS 並列ベンチマーク性能

セス数はすべて 16 とし，コンパイルは gcc/g77 3.3.2 を用いてオプションを-O3 として行った．図 6.7に，各トポロジにおけるベンチマーク性能の測定結果 (Mop/s) を，比較のために用いたフラットトポロジ (Flat) の場合を 1 として正規化した相対性能を示す．

結果より，提案手法を用いて構築した各トポロジは，ほとんどのベンチマークにおいて理想的なフラットトポロジの 9 割以上の性能を達成している．前節でも述べたように，全ホストを 1 つのスイッチに接続するフラットトポロジはあくまで理想的であり，本評価環境のように比較的小規模なクラスタでは構築可能だが，数百〜数千のホストを接続する大規模クラスタではほぼ実現不可能である．一方，提案手法では，8~24 ポート程度の比較的小規模かつ低コストなイーサネットスイッチを用いて，5.4節で示したような，並列計算機や SAN で用いられてきたさまざまなトポロジを構築することが可能である．この点で，本手法は大規模なクラスタを構築する場合にも有効な手法であると言える．

一方で，VLAN を用いない単純なツリー状トポロジである M-Tree は，EP を除くすべてのベンチマークで提案手法を用いたトポロジに比べて性能が低くなっており，特に FT と IS では著しく性能が悪い．FT および IS は，全対全通信である MPI_Alltoall が頻繁に実行されるために高いバイセクションバンド幅を要求するベンチマークであることが知られており，VLAN ルーティング法によるホスト間の複数パスへのトラフィックの分散が非常に有効であることがわかる．

6.4　大規模化に関する検討

本節では，前節までの中規模クラスタを用いた性能評価結果を踏まえ，提案手法の大規模クラスタへの適用可能性について検討する．

まず，6.3.3節で行った NAS 並列ベンチマークの評価結果より，小〜中規模のクラスタ環境において，提案手法を適用して構築したトポロジは理想的なフラットトポロジに迫る高い性能を発揮することがわかった．これを大規模クラスタに適用する際に問題となるのは，主に以下の 2 つと

考えられる．

1. パスのホップ数 (経由スイッチ数) の増加によるホスト間のフレーム転送遅延の増大

2. 各スイッチにおける MAC アドレステーブルエントリ数の制限

まず，1. について，6.2.1節で述べた通り，提案手法を適用する際の処理オーバヘッド，すなわちスイッチへのフレーム入出力時に VLAN タグ付けおよび除去操作を行うことによる遅延やスループットへの影響はほとんどないことがわかっている．つまり，1. の問題は，各スイッチでの遅延がホップごとに純粋に加算されることに起因しており，提案手法を含む VLAN ルーティング法に限らず，イーサネットを用いたクラスタ全般に共通する問題である．

6.2.2節での 2 ホスト間の MPI 通信性能の測定結果 (図 6.2) より，低レベル通信ライブラリとして PM/Ethernet を用いた場合には，パスの経由スイッチ数が 6 以下であればバンド幅はそれほど低下しない．ここで，例えば本章の評価で用いた Dell PowerConnect 5324 のように，比較的安価に入手可能な 24 ポートのイーサネットスイッチを用いてトポロジを構築することを考えた場合，Fat ツリー (12,12,2) であれば，すべてのパスの経由スイッチ数が 5 以下で，288 ホストまで接続可能なフルバイセクションバンド幅を持つネットワークを構築することができる．

フルバイセクションバンド幅を維持しつつ，Fat ツリーでこれ以上のホスト数を接続しようとした場合，24 ポートスイッチでは Fat ツリー (u, d, r) の階層数 r を増やす以外ないため，パスによっては経由スイッチ数が 6 より大きくなり，図 6.2の結果からはホスト間のバンド幅は低下するものと予想される．ただし，PM/Ethernet では，スイッチのバッファ不足によってフレームが破棄された場合に，パケットの再送を行う際のタイムアウト時間はパラメータで調整可能であり，これを大きくすることによってバンド幅の低下はある程度防ぐことが可能である．

また，本章の評価で用いた低コストなギガビット・イーサネット (1000BASE-T) のスイッチでは遅延が 2μsec から 4μsec 程度の遅延のものが多いが，最近の 10 ギガビット・イーサネット (10GBASE-CX4) のスイッチには，Fulcrum Microsystems 社の FocalPoint シリーズのように 200nsec から 300nsec 程度の超低遅延を実現しているものも少なくない．今後，これら 10 ギガビット・イーサネットスイッチのポート単価が下がっていけば，SAN と比べた際のイーサネットのレイテンシはあまり大きな問題にならなくなると考えられる．

次に，2. の問題について，これは VLAN ルーティング法の実現方法および構築するトポロジによって異なってくるため，5.4節で VLAN 割り当て方法を示したトポロジそれぞれについて検討を行う．表 6.4に，一般化した各トポロジにおいて，ホストで VLAN タグ付けを行う従来の VLAN ルーティング法 (“VLAN”) と VLAN リネーミング法 (“RENAME”) のそれぞれで必要となる VLAN 数をまとめる．表において，D はトポロジ中で 1 つのスイッチが持つ Down 方向のチャネル数の最大値を表す．なお，スイッチタグ法において必要となる VLAN 数は，トーラスを除いて (5.4.3節参照) 従来の VLAN ルーティング法の場合と等しい．

表 6.4より，ホストにおいてフレームの VLAN タグ付けを行う従来の方法に比べ，VLAN リネーミング法では少数の VLAN でトポロジを構築できることがわかる．ここで，4.4.2節の式 (4.2) に示した通り，トポロジに接続できるホスト数 H は，必要となる VLAN 数 V に反比例する．例えば，24 ポートのスイッチを用いて前述の 12 進 Fat ツリー（Fat ツリー $(12, 12, r)$）を構築する場合，従来の VLAN ルーティング法では，レイヤ数 $r = 2$ で $V = 12^2 = 144$ 個，$r = 3$ で $V = 12^3 = 1,728$ 個となる．スイッチの MAC アドレステーブルのエントリ数を $M = 12,000$ と仮定すると，最大でもそれぞれ $H = 12,000/144 = 83$ ホスト，$H = 12,000/1,728 = 6$ ホストしか接続できない．一方，VLAN

表 6.4　必要となる VLAN 数の比較

	VLAN	RENAME
Fat ツリー (u, d, r)	u^r	u
Myrinet-Clos (u, r)	u^r	u
k-ary n-cube メッシュ	k^{n-1}	n
k-ary n-cube トーラス	$2k^{n-1}$	$3n + 1$
不規則 (UD ルーティング)	-	$D + 1$

リネーミング法では，レイヤ数 r によらず $V = 12$ 個の VLAN で済むため，$H = 12,000/12 = 1,000$ ホストまで接続することが可能である．なお，低コストなスイッチは数十〜数百個の VLAN しか登録できないものが多いため，従来の VLAN ルーティング法で $r = 3$ のときの $V = 1,728$ 個という VLAN 数は，そもそも登録不可能である可能性が高い．

ただし，VLAN リネーミング法はどのスイッチにも適用できるわけではないことには注意が必要である．4.4.1節で述べた通り，イーサネットスイッチの中には，静的に登録可能な MAC アドレス数 M が MAC アドレステーブルの総エントリ数に比べて制限されていたり，静的な登録をサポートしていないものも存在する．このようなスイッチでは VLAN リネーミング法を利用することができない．

以上より，適用可能なイーサネットスイッチに制限はあるものの，提案手法である VLAN リネーミング法を用いることにより，従来の VLAN ルーティング法に比べ多数のスイッチを含む大規模なネットワークを構築可能であると言える．今後，提案手法を含む VLAN ルーティング法や，3.6節で紹介した Data Center Ethernet (DCE) 等の関連技術の普及により，HPC クラスタのインターコネクトとしてのイーサネットの需要がさらに高まれば，現在よりも多くの VLAN や MAC アドレスを登録可能なスイッチが低コストで入手できるようになるものと予想される．これにより，現在は高々1,000〜2,000 ホスト程度のシステムが限界であるが，今後はさらに大規模なシステムを低い導入コストで構築できるようになると期待される．

第7章　結 論

7.1　本研究のまとめ

本研究では，イーサネットを用いた大規模 PC クラスタシステム構築のための技術の開発を行った．環境の制限により，残念ながら実際に大規模クラスタを構築することはできなかったが，中規模クラスタにおける性能評価を踏まえた検討の結果，大規模クラスタへの適用も十分可能であり，低コストかつ高性能なクラスタシステムの実現が期待されることがわかった．

現在，コストパフォーマンスに優れるイーサネットをインターコネクトに用いたクラスタシステムが高性能計算 (HPC) 分野のプラットフォームにおいて多数を占めるようになっている．しかし，システムエリアネットワーク (SAN) と違ってイーサネットはループ構造を含むことができないため，イーサネットを用いたクラスタの大半は現状では単純なツリー状トポロジを採用している．これに対する有力な解決策として，IEEE 標準の VLAN 技術を応用する VLAN ルーティング法が提案されていたが，ホスト側システムソフトウェアの VLAN への対応と，VLAN 数および MAC アドレステーブルエントリ数の制限の 2 つの問題により実用化には至っていなかった．

本論文ではまず，イーサネットを用いた大規模クラスタ構築のための技術として，これらの問題を解決する 2 つの手法を提案した．1 つ目の提案手法「スイッチタグ法」では，イーサネットフレームへの VLAN タグ挿入をホストではなくスイッチへのフレーム入力時に行う．これにより，ホスト側のシステム環境への依存をなくし，通信ライブラリやドライバ等がフレームへの VLAN タグ挿入をサポートしていない場合にも VLAN ルーティング法を利用できるようになる．また，2 つ目の提案手法「VLAN リネーミング法」では，スイッチタグ法をさらに改良し，VLAN の使用を各スイッチ内でフレームの出力ポートを決定する目的に限定することにより，必要となる VLAN 数を大幅に削減することができる．本論文では，これらの提案手法について，その概要およびアルゴリズムを示すとともに，実際に提案手法を用いてクラスタを構築する際のイーサネットスイッチの設定例を示し，その適用可能範囲についても明らかにした．

さらに本論文では，これらの提案手法を含めた VLAN ルーティング法をイーサネットに適用してループを含むトポロジを構築した際の性能を決定する要因，および主要な並列計算向けトポロジの構築方法についても検討を行った．まず，イーサネットにおけるフロー制御とデッドロックの問題について検証を行い，提案手法を効率的に適用するためには，リンクレベルのフロー制御およびデッドロックフリールーティングを用いることが重要であることを明らかにした．また，提案手法を用いて実際にイーサネット上に並列計算向けのトポロジを構築する際の VLAN 割り当ての方法，およびルーティングアルゴリズムの選択についても検討を行い，主要なトポロジにおける VLAN の設定例を示した．

提案手法の性能評価として，16 スイッチ・32 ホストからなるクラスタ環境に実際に適用し，基本通信性能およびアプリケーションベンチマーク性能を測定した．その結果，提案手法は導入によるオーバヘッドがほとんどなく，提案手法を用いて構築した各トポロジはすべてのアプリケーションで理想的なフラットトポロジの 88%以上という高い性能を示した．また，典型的なトポロ

ジに提案手法および従来の VLAN ルーティング法を適用した際に必要となる VLAN 数を一般化し，各トポロジを採用した場合に構築可能なシステム規模を導出することにより，提案手法が従来の VLAN ルーティング法に比べてより大規模なネットワークに適用可能なことを確認した．提案手法は，多くの商用 L2 イーサネットスイッチにおいてサポートされている VLAN 機能を制御することにより実現できる点，集約化したリンク群に対して VLAN ID を割り当てることによりリンク集約化技術と併用することができる点において，高い実用性と汎用性を持つ技術であると言える．また，これらはクラスタ内部のイーサネットに閉じた手法ではなく，柔軟な経路を設定するためにグリッドや LAN 技術の一部として利用することも可能である．

最後に，これらの評価結果を踏まえ，提案手法の大規模クラスタへの適用可能性について検討した結果，従来の VLAN ルーティング法では高々数十ホストしか接続できないようなトポロジにおいても，提案手法である VLAN リネーミング法によって，現状でも 1,000 ホスト程度のフルバイセクションバンド幅を持つ大規模ネットワークを構築可能であり，300 ホスト程度までは大きなパフォーマンスの低下を伴うことなく運用できる見込みであることが明らかになった．今後，提案手法を含む VLAN ルーティング法や Data Center Ethernet 等の関連技術の普及により，HPC クラスタのインターコネクトとしてのイーサネットの需要がさらに高まれば，最新の 10 ギガビット・イーサネットスイッチのような超低遅延のスイッチや，現在よりも多くの VLAN や MAC アドレスを登録可能なスイッチが低コストで入手可能となり，提案手法を用いたさらに大規模かつ高性能なシステムを構築できるようになると期待される．

7.2 おわりに

本研究では，イーサネットを用いた大規模クラスタ構築のための技術を開発し，実際に大規模化が可能であることを示した．この点において本研究の第一の目的は達成されたと言えるが，本論文においてこれまで十分に言及していない，さらなる検討事項も存在する．以下に現時点で筆者が考えつく限り列挙しておく．

まず，トポロジの構築時に限った問題ではないが，SMP やマルチコアへの対応が挙げられる．2009 年現在，コアを複数搭載するプロセッサは珍しいものではなくなり，特にサーバ機であれば大抵のシステムがマルチコアの CPU を採用している．また，ノード内に複数 CPU を搭載するシステムも一般的になっており，クラスタを構築する際に 1 ノードに 1 つのプロセッサコアしかない，という状況はほぼ考えられない．複数コアを搭載するシステムをトポロジに接続する場合，スイッチとの間のリンクが 1 本では明らかに不足するため，複数ポートを持つ NIC や複数の NIC を搭載し，複数リンクによってトポロジに接続する必要がある．その際の接続方法としては，リンク集約化，VLAN による複数パスでの接続，あるいは別々のスイッチへの接続などが考えられ，検討すべき課題と言える．スイッチ間についてもリンクが 1 本では不足するため，リンク集約化や VLAN を用いた負荷分散がますます重要になると予想される．

また，イーサネット上のマルチキャストの利用も課題として挙げられる．5.3.2で述べたように，イーサネットにおいてもマルチキャストはサポートされており，MAC アドレス空間内の予約領域を用いて，レイヤ 2 イーサネット上の任意の宛先ホスト集合へのマルチキャストを行うことができる．しかし，使用する宛先アドレスやマルチキャストフレームのフィルタリングの管理は基本的にホスト側アプリケーションに任されており，SAN に比べて設定が繁雑である．そのため，現在一般に用いられている MPI 等の通信ライブラリでは使用されておらず，クラスタ内通信においてイーサネットのマルチキャストは利用できないのが現状である．マルチキャストは 1 回の送信

ですべての必要な宛先にフレームを送ることができるため，並列アプリケーションでも利用価値は高く，実現が期待される．

さらに，イーサネットにおける耐故障性の実現や，省電力化についても今後ますます必要とされる技術であると考えられ，これらについては筆者も参加している研究グループにおいてすでに検討，提案している [76]．いずれも VLAN を用いて複数パスを導入する手法をベースとしており，本研究の延長線上にある．また，本研究に直接関係する問題としては，4.3.3で述べたトポロジ全体にわたるスイッチ群の管理方法，すなわち，設定ファイルの記述やそのアップロード，運用時の監視の方法なども課題として挙げることができる．

第 1章で述べたように，イーサネットを結合網に用いたクラスタは最新の TOP500 ランキングにおいて 500 台中 282 台と過半数を占めており，コストパフォーマンスに優れる高性能計算システムとしての地位を確立していると言える．しかしながら，ランクインしている各システムのイーサネットがどのようなトポロジ構成となっているかは明らかになっていない場合がほとんどであり，トポロジについてシステム間での比較検討はできないのが現状である．今後，本研究や他の優れた研究の成果を通して，イーサネットのトポロジやルーティングについての関心が高まり，よりオープンな議論ができるようになることを期待する．

なお，本研究の成果はすでに，同志社大学理工学部に設置されている SuperNova クラスタ (Opteron 1.8GHz × 2 × 256 ノード，計 512 コア)[77] で利用されており，8 台の Dell PowerConnect 6248 (1000BASE-T × 48 ポート) をスイッチ間を 2 本の集約化リンクとする完全結合で接続し，225 ノード (450 コア) で LINPACK ベンチマークを実行した結果において，1.081TFLOPS の性能を記録した [78][79]．これは，336 ポートを持つ Force10 社の E1200 スイッチを使用した際の 256 ノード (512 コア) での値 1.169TFLOPS を実効性能割合で上回るものであり，提案手法の有効性が改めて確認されたと言える．また，同学部において現在構築中の Misc クラスタ (Quad-Core Opteron 2.3GHz × 2 × 66 ノード，計 528 コア) でも提案手法が利用され，現在評価が行われている．本研究が，今後のクラスタ向けインターコネクトの研究の発展に寄与できれば，筆者にとって何よりの喜びである．

謝辞

本研究は，多くの方々のご助力をいただいて初めて完了させることができたものであり，ここに謹んで感謝の意を表します．

本研究の機会を与えて下さり，9 年間もの長きにわたり絶えずご指導下さった，慶應義塾大学理工学部 天野 英晴 教授に深く感謝いたします．RHiNET のアプリケーション開発に始まり，クラスタの構築や SCore の移植，DIMMnet や ExpEther プロジェクトへの参加など，多くの貴重な経験をさせていただきました．その一方で，なかなか論文を書こうとしなかったり，仕事の都合であまり研究室に顔を出せない時期もあり，いろいろとご心配をおかけしたことと思いますが，常に変わらぬ手厚いご指導をいただきました．9 年間，本当にありがとうございました．

本論文の執筆に際し，お忙しい中副査をお引き受け下さり，審査の過程において多くの有益なご助言をいただきました，慶應義塾大学理工学部 寺岡 文男 教授，河野 健二 准教授，西 宏章 准教授に心より感謝いたします．非常にタイトなスケジュールにもかかわらず，丁寧に査読をしていただいたことで，論文の質が格段に向上したと思います．本当にありがとうございました．

国立情報学研究所 鯉渕 道紘 助教には，共同研究者として，また研究室の先輩として，常に様々なご助言をいただき，数多くの有益な議論を行わせていただきました．心より感謝いたします．先輩の研究に対する真摯な姿勢から筆者は多くのことを学ばせていただきました．本当にありがとうございました．

VLAN ルーティング法の提案者でもある産業技術総合研究所 工藤 知宏 博士には，共同研究者として，実際的な観点からの示唆に富んだ多くのご助言をいただきました．心より感謝いたします．本研究の初期の段階でつくばの研究所にお邪魔した際には，実験用のクラスタ環境を提供いただいただけでなく，車での移動から宿泊施設の手配に至るまで大変お世話になりました．また，その際にさまざまな局面でお世話になり，有益な議論を交わさせていただいた，産業技術総合研究所 児玉 祐悦 博士，松田 元彦 博士，清水 敏行 氏，岡崎 史裕 氏をはじめとする皆様にも心より感謝いたします．ありがとうございました．

同志社大学生命医科学部 廣安 知之 教授，同大学理工学部 中尾 昌広 氏，同 渡辺 崇文 氏には，本研究の成果をクラスタ環境において利用いただき，また多くの有益なご助言をいただきました．心より感謝いたします．SuperNova クラスタにおける LINPACK ベンチマーク性能の測定結果があったからこそ，本研究の有効性が大きな説得力を持つことができたと思っております．本当にありがとうございました．

同期としてともに博士課程に在籍した，成蹊大学理工学部 長名 保範 助教，東芝 セミコンダクター社 渡邊 幸之介 博士には，公私にわたり非常にお世話になりました．心より感謝いたします．両氏のおかげで筆者の研究室生活は大変充実したものとなりました．また，今回の学位審査に際し，両氏には多くの励ましの言葉をいただきました．おかげさまで，遅ればせながらどうにか学位を取得できそうです．本当にありがとうございました．

東芝 セミコンダクター社 田邉 靖貴 博士，株式会社日立製作所 堤 聡 博士，日本電気株式会社 辻 聡 博士，株式会社東芝 長谷川 揚平 博士の各氏には，研究室生活の長い時間をともに過ごさせていただき，公私ともに大変お世話になりました．心より感謝いたします．頼りない先輩ですが，今後ともよろしくお願いいたします．本当にありがとうございました．

筆者が助教および学生アシスタントとして 8 年間にわたり奉職させていただいた，慶應義塾インフォメーションテクノロジーセンターの関係者の方々，特に，小林 啓樹 事務長，金子 康樹 事務長，落合 啓一 事務長，廣野 哲郎 事務長，徳永 澄香 氏，澤木 敏郎 氏，降旗 ゆかり 氏，および 金森 勇壮 氏の各氏には日頃より大変お世話になりました．深く感謝いたします．本論文の執筆に際し，何度もやむを得ず欠勤することを快く受け入れて下さったばかりか，多くの励ましの言葉をいただきました．今後ともお世話になることと存じますが，どうぞよろしくお願いいたします．本当にありがとうございました．

慶應義塾大学理工学部 天野研究室 松谷 宏紀 博士，同 西川 由理 氏には，学位審査の過程でさまざまな面でご協力いただき，大変お世話になりました．また，同 吉見 真聡 氏，同 王 代涵 氏には，同時期に学位審査を受ける仲間として多くの刺激を与えていただき，また情報を提供いただきました．ここに深く感謝いたします．ありがとうございました．

本研究に対し惜しみないご協力を頂きました，天野研究室 PDARCH グループの現役生の三氏，今田 啓介 氏，酒井 洋介 氏，矢部 裕也 氏に心より感謝いたします．これからも是非 PDARCH グループを盛り上げていっていただければと切に願っております．本当にありがとうございました．

筆者が本研究に先立って参加させていただいていた，RHiNET プロジェクトのすべての関係者の方々に心より感謝いたします．RHiNET での経験がなければ，本研究を進めることはできなかったと思います．本当にありがとうございました．

また，日頃より暖かいご支援をいただきました天野研究室のすべての卒業生，現役生の皆様に心より感謝いたします．9 年間，公私にわたり本当にお世話になり，ありがとうございました．

最後になりましたが，9 年にもわたる不規則な研究生活を暖かく見守ってくれた大切な家族と，精神面で大きな支えとなり，常に励ましてくれた 大北 博美 女史に本論文を捧げたいと思います．本当に，本当にありがとうございました．

2009 年 2 月
矢上キャンパス 26-107A にて
大塚 智宏

参考文献

[1] HPL - A Portable Implementation of the High-Performance Linpack Benchmark for Distributed Memory Computers. http://www.netlib.org/benchmark/hpl/.

[2] TOP500 Supercomputing Sites. http://www.top500.org/.

[3] InfiniBand Trade Association. http://www.infinibandta.org/.

[4] InfiniBand Trade Association. InfiniBand architecture. Specification Volume 1, Release 1.0.a. *available from the InfiniBand Trade Association,* http://www.infinibandta.org/, June 2001.

[5] Myricom. http://www.myri.com/.

[6] N. J. Boden, D. Cohen, R. E. Felderman, A. E. Kulawik, C. L. Seitz, J. N. Seizovic, and W. Su. Myrinet: A Gigabit-per-Second Local Area Network. *IEEE Micro*, Vol. 15, No. 1, pp. 29–36, 1995.

[7] Quadrics. http://www.quadrics.com/.

[8] F. Petrini, W. C. Feng, A. Hoisie, S. Coll, and E. Frachtenberg. The Quadrics Network (QsNet): High-Performance Clustering Technology. In *Proc. of Hot Interconnets 9*, pp. 125–130, August 2001.

[9] RDMA Consortium. http://www.rdmaconsortium.org/.

[10] IEEE 802.3 Ethernet Working Group. http://www.ieee802.org/3/.

[11] Toshiyuki Takahashi, Shinji Sumimoto, Atsushi Hori, Hiroshi Harada, and Yutaka Ishikawa. PM2: High Performance Communication Middleware for Heterogeneous Network Environment. In *Proc. of SC2000 High Performance Networking and Computing Conference*, pp. 52–53, November 2000.

[12] 住元真司, 堀敦史, 手塚宏史, 高橋俊行, 原田浩, 石川裕. 高速通信機構 PM2 の設計と評価. 情報処理学会論文誌ハイパフォーマンスコンピューティングシステム, Vol. 41, No. SIG 05(HPS 1), pp. 80–90, August 2000.

[13] IEEE 802.3ad Link Aggregation Task Force. http://www.ieee802.org/3/ad/.

[14] 工藤知宏, 松田元彦, 手塚宏史, 児玉祐悦, 建部修見, 関口智嗣. VLAN を用いた複数パスを持つクラスタ向き L2 Ethernet ネットワーク. 情報処理学会論文誌コンピューティングシステム, Vol. 45, No. SIG 6(ACS 6), pp. 35–43, May 2004.

[15] Srikant Sharma, Kartik Gopalan, Susanta Nanda, and Tzi-cker Chiueh. Viking: A Multi-Spanning-Tree Ethernet Architecture for Metropolitan Area and Cluster Networks. In *Proc. of 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom 2004)*, pp. 2283–2294, March 2004.

[16] T. Kudoh, H. Tezuka, M. Matsuda, Y. Kodama, O. Tatebe, and S. Sekiguchi. VLAN-based Routing: Multi-path L2 Ethernet Network for HPC Clusters. In *Proc. of 2004 IEEE International Conference on Cluster Computing (Cluster 2004)*, September 2004.

[17] IEEE 802.1Q - Virtual LANs. `http://www.ieee802.org/1/pages/802.1Q.html`.

[18] IEEE 802.1D - MAC bridges. `http://www.ieee802.org/1/pages/802.1D-2003.html`.

[19] 住元真司, 堀敦史, 手塚宏史, 原田浩, 高橋俊行, 石川裕. 既存 OS の枠組みを用いたクラスタシステム向け高速通信機構の提案. 情報処理学会論文誌, Vol. 41, No. 6, pp. 1688–1696, June 2000.

[20] Cisco Systems, Inc. `http://www.cisco.com/`.

[21] Force10 Networks. `http://www.force10networks.com/`.

[22] Fulcrum Microsystems. `http://www.fulcrummicro.com/`.

[23] W. D. Dally and B. Towles. *Principles and Practices of Interconnection Networks*. Morgan Kaufmann, 2003.

[24] F. D. Pellegrini, D. Starobinski, M. G. Karpovsky, and L. B. Levitin. Scalable Cycle-Breaking Algorithms for Gigabit Ethernet Backbones. In *Proc. of 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom 2004)*, pp. 2175–2184, March 2004.

[25] Andres Mejia, Jose Flich, Jose Duato, Sven-Arne Reinemo, and Tor Skeie. Segment-Based Routing: An Efficient Fault-Tolerant Routing Algorithm for Meshes and Tori. In *Proc. of 20th International Parallel and Distributed Processing Symposium (IPDPS 2006)*, April 2006.

[26] Andres Mejia, Jose Flich, Jose Duato, Sven-Arne Reinemo, and Tor Skeie. Boosting Ethernet Performance by Segment-Based Routing. In *Proc. of the 15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing (PDP'07)*, pp. 55–62, February 2007.

[27] Sven-Arne Reinemo and Tor Skeie. Ethernet as a Lossless Deadlock Free System Area Network. In *Proc. of Third International Symposium on Parallel and Distributed Processing and Applications (ISPA'05)*, pp. 901–914, November 2005.

[28] Sven-Arne Reinemo and Tor Skeie. Effective Shortest Path Routing for Gigabit Ethernet. In *Proc. of the IEEE International Conference on Communications 2007 (ICC-2007)*, pp. 6419–6424, June 2007.

[29] Tor Skeie, Olav Lysne, and Ingebjorg Theiss. Layered Shortest Path (LASH) Routing in Irregular System Area Networks. In *Proc. of 16th International Parallel and Distributed Processing Symposium (IPDPS 2002)*, pp. 162–169, April 2002.

[30] Shibiao Lin, Srikant Sharma, and Tzi-cker Chiueh. Autonomic Resource Management for Multiple-Spanning-Tree Metro-Ethernet Networks. In *Proc. of the 6th International Symposium on Network Computing and Applications (NCA07)*, pp. 239–248, July 2007.

[31] PACS-CS Project 2005-2007. http://www.ccs.tsukuba.ac.jp/PACS-CS/.

[32] Taisuke Boku, Mitsuhisa Sato, Akira Ukawa, Daisuke Takahashi, Shinji Sumimoto, Kouichi Kumon, Takashi Moriyama, and Masaaki Shimizu. PACS-CS: A Large-Scale Bandwidth-Aware PC Cluster for Scientific Computations. In *Proc. of the 6th IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2006)*, pp. 233–240, May 2006.

[33] Shinji Sumimoto, Kazuichi Ooe, Kouichi Kumon, Taisuke Boku, Mitsuhisa Sato, and Akira Ukawa. A Scalable Communication Layer for Multi-Dimensional Hyper Crossbar Network Using Multiple Gigabit Ethernet. In *Proc. of 20th ACM International Conference on Supercomputing (ICS 2006)*, pp. 107–115, June 2006.

[34] 住元真司, 大江和一, 久門耕一, 朴泰祐, 佐藤三久, 宇川彰. 複数 Gigabit Ethernet を用いた PACS-CS のための高性能通信機構の設計と評価. 情報処理学会論文誌コンピューティングシステム, Vol. 47, No. SIG 12(ACS 15), pp. 25–34, September 2006.

[35] IEEE 802.1 Data Center Bridging Task Group. http://www.ieee802.org/1/pages/dcbridges.html.

[36] Fibre Channel over Ethernet. http://www.t11.org/fcoe.

[37] Woven Systems. http://www.wovensystems.com/.

[38] IEEE 802.1Qau - Congestion Notification. http://www.ieee802.org/1/pages/802.1au.html.

[39] IEEE 802.1Qaz - Enhanced Transmission Selection. http://www.ieee802.org/1/pages/802.1az.html.

[40] IEEE 802.1Qbb - Priority-based Flow Control. http://www.ieee802.org/1/pages/802.1bb.html.

[41] IEEE 802.1AB - Station and Media Access Control Connectivity Discovery. http://www.ieee802.org/1/pages/802.1ab.html.

[42] IEEE 802.1aq - Shortest Path Bridging. http://www.ieee802.org/1/pages/802.1aq.html.

[43] Transparent Interconnection of Lots of Links (trill). http://www.ietf.org/html.charters/trill-charter.html.

[44] LAM/MPI Parallel Computing. http://www.lam-mpi.org/.

[45] 大塚智宏, 鯉渕道紘, 工藤知宏, 天野英晴. スイッチでタグ付けを行う VLAN ルーティング法. 情報処理学会論文誌コンピューティングシステム, Vol. 47, No. SIG 12(ACS 15), pp. 46–58, September 2006.

[46] Tomohiro Otsuka, Michihiro Koibuchi, Tomohiro Kudoh, and Hideharu Amano. A Switch-tagged VLAN Routing Methodology for PC Clusters with Ethernet. In *Proc. of the 2006 International Conference on Parallel Processing (ICPP-06)*, pp. 479–486, August 2006.

[47] 大塚智宏, 鯉渕道紘, 工藤知宏, 天野英晴. VLAN イーサネットを用いた PC クラスタ向け大規模ネットワーク構築法. 情報処理学会論文誌コンピューティングシステム, Vol. 1, No. 3, pp. 96–107, December 2008.

[48] 森川誠一. グリッドコンピューティングに要求される通信技術. In *NetWorld+Interop 2003 Tokyo Conference Notes*, pp. 75–87, June 2003.

[49] Michael D. Schroeder, Andrew D. Birrell, Michael Burrows, Hal Murray, Roger M. Needham, and Thomas L. Rodeheffer. Autonet: a High-speed, Self-configuring Local Area Network Using Point-to-point Links. *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 8, pp. 1318–1335, October 1991.

[50] Dell PowerConnect 5324 Product Details. `http://www.dell.com/content/products/productdetails.aspx/pwcnt_5324`.

[51] Shin'ichi Miura, Taisuke Boku, Takayuki Okamoto, and Toshihiro Hanawa. A Dynamic Routing Control System for High-Performance PC Cluster with Multi-path Ethernet Connection. In *Proc. of 22nd International Parallel and Distributed Processing Symposium (IPDPS 2008)*, April 2008.

[52] 三浦信一, 岡本高幸, 朴泰祐, 塙敏博. マルチパスネットワークを持つ PC クラスタにおける動的経路制御システム. 情報処理学会論文誌コンピューティングシステム, Vol. 48, No. SIG 18(ACS 20), pp. 56–68, December 2007.

[53] 三浦信一, 岡本高幸, 朴泰祐, 佐藤三久, 高橋大介. VFREC-Net: ドライバ制御による tagged-VLAN を用いた PC クラスタ向けマルチパスネットワーク. 情報処理学会論文誌コンピューティングシステム, Vol. 47, No. SIG 12(ACS 15), pp. 35–45, September 2006.

[54] Shin'ichi Miura, Takayuki Okamoto, Taisuke Boku, Mitsuhisa Sato, and Daisuke Takahashi. Low-cost High-bandwidth Tree Network for PC Clusters based on Tagged-VLAN Technology. In *Proc. of the 8th International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN 2005)*, pp. 84–93, December 2005.

[55] Rich Seifert. *The Switch Book: The Complete Guide to LAN Switching Technology*. Wiley, 2000.

[56] Iperf - The TCP/UDP Bandwidth Measurement Tool. `http://dast.nlanr.net/Projects/Iperf/`.

[57] Yutaka Ishikawa, Hiroshi Tezuka, Atsushi Hori, Shinji Sumimoto, Toshiyuki Takahashi, Francis O'Carroll, and Hiroshi Harada. RWC PC Cluster II and SCore Cluster System Software – High Performance Linux Cluster. In *Proc. of the 5th Annual Linux Expo*, pp. 55–62, May 1999.

[58] PC Cluster Consortium. `http://www.pccluster.org/`.

[59] Intel Cluster Tools. `http://www.intel.com/cd/software/products/asmo-na/eng/cluster/`.

[60] MPICH2: High-performance and Widely Portable MPI. http://www.mcs.anl.gov/research/projects/mpich2/.

[61] MPICH Home Page. http://www-unix.mcs.anl.gov/mpi/mpich1/.

[62] J. Duato, S. Yalamanchili, and L. Ni. *Interconnection Networks: an engineering approach*. Morgan Kaufmann, 2002.

[63] W. J. Dally and C. L. Seitz. Deadlock-Free Message Routing in Multiprocessor Interconnection Networks. *IEEE Transactions on Computers*, Vol. 36, No. 5, pp. 547–553, May 1987.

[64] Tperf. http://www.am.ics.keio.ac.jp/~terry/tperf/.

[65] GtrcNET: Programmable Network Testbed. http://projects.gtrc.aist.go.jp/gnet/.

[66] Y. Kodama, T. Kudoh, R. Takano, H. Sato, O. Tatebe, and S. Sekiguchi. GNET-1: Gigabit Ethernet Network Testbed. In *Proc. of 2004 IEEE International Conference on Cluster Computing (Cluster 2004)*, September 2004.

[67] Shinji Nishimura, Tomohiro Kudoh, Hiroaki Nishi, Junji Yamamoto, Katsuyoshi Harasawa, Nobuhiro Matsudaira, Shigeto Akutsu, and Hideharu Amano. 64-Gbit/s highly reliable network switch (RHiNET-2/SW) using parallel optical interconnection. *IEEE journal of Lightwave Tehnology (Special issue on Optical Networks)*, Vol. 18, No. 12, pp. 1620–1627, December 2000.

[68] 西宏章, 多昌廣治, 西村信治, 山本淳二, 工藤知宏, 天野英晴. LASN 用 8Gbps/port 8x8 One-chip スイッチ: RHiNET-2/SW. 並列処理シンポジウム JSPP2000 論文集, pp. 173–180, May 2000.

[69] Michihiro Koibuchi, Konosuke Watanabe, Tomohiro Otsuka, and Hideharu Amano. Performance Evaluation of Deterministic Routings, Multicasts, and Topologies on RHiNET-2 Cluster. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 16, No. 8, pp. 747–759, August 2005.

[70] Message Passing Interface Forum. http://www.mpi-forum.org/.

[71] Blue Gene Project. http://www.research.ibm.com/bluegene/index.html.

[72] Kei Davis, Adolfy Hoisie, Greg Johnson, Darren J. Kerbyson, Mike Lang, Scott Pakin, and Fabrizio Petrini. A Performance and Scalability Analysis of the BlueGene/L Architecture. In *Proc. of SC2004 High Performance Computing, Networking and Storage Conference (SC2004)*, pp. 1–9, November 2004.

[73] Tomohiro Otsuka, Michihiro Koibuchi, Akiya Jouraku, and Hideharu Amano. VLAN-based Minimal Paths in PC Cluster with Ethernet on Mesh and Torus. In *Proc. of the 2005 International Conference on Parallel Processing (ICPP-05)*, pp. 567–576, June 2005.

[74] D. Bailey, T. Harris, W. Saphir, R. Wijngaart, A. Woo, and M. Yarrow. The NAS Parallel Benchmarks 2.0. In *NAS Technical Report NAS-95-020*, December 1995.

[75] W. Saphir, R. Wijngaart, A. Woo, and M. Yarrow. New Implementations and Results for the NAS Parallel Benchmarks 2. In *8th SIAM Conference on Parallel Processing for Scientific Computing*, March 1997.

[76] Michihiro Koibuchi, Tomohiro Otsuka, Hiroki Matsutani, and Hideharu Amano. An On/Off Link Activation Method for Low-Power Ethernet in PC Clusters. In *Proc. of 23rd International Parallel and Distributed Processing Symposium (IPDPS 2009)*, May 2009.

[77] Cluster Group -Intelligent Systems Design Laboratory-. `http://mikilab.doshisha.ac.jp/dia/research/cluster/`.

[78] 渡辺崇文, 中尾昌広, 廣安知之, 鯉渕道紘, 大塚智宏. VLAN イーサネットを用いた大規模 PC クラスタの検討. 情報処理学会研究報告 2008-ARC-179, pp. 169–174, August 2008.

[79] Takafumi Watanabe, Masahiro Nakao, Tomoyuki Hiroyasu, Tomohiro Otsuka, and Michihiro Koibuchi. Impact of Topology and Link Aggregation on a PC Cluster with Ethernet. In *Proc. of 2008 IEEE International Conference on Cluster Computing (Cluster 2008)*, pp. 280–285, September 2008.

論文目録

本研究に関する論文

公刊論文

1. 大塚 智宏, 鯉渕 道紘, 工藤 知宏, 天野 英晴. VLAN イーサネットを用いた PC クラスタ向け大規模ネットワーク構築法. 情報処理学会論文誌コンピューティングシステム, Vol. 1, No. 3, pp. 96–107, December 2008.
2. 大塚 智宏, 鯉渕 道紘, 工藤 知宏, 天野 英晴. スイッチでタグ付けを行う VLAN ルーティング法. 情報処理学会論文誌コンピューティングシステム, Vol. 47, No. SIG 12(ACS 15), pp. 46–58, September 2006.

国際会議

1. Michihiro Koibuchi, Tomohiro Otsuka, Hiroki Matsutani, and Hideharu Amano. An On/Off Link Activation Method for Low-Power Ethernet in PC Clusters. In *Proc. of the 23rd International Parallel and Distributed Processing Symposium (IPDPS 2009)*, May 2009. (採録決定済)
2. Takafumi Watanabe, Masahiro Nakao, Tomoyuki Hiroyasu, Tomohiro Otsuka, and Michihiro Koibuchi. Impact of Topology and Link Aggregation on a PC Cluster with Ethernet. In *Proc. of 2008 IEEE International Conference on Cluster Computing (Cluster 2008)*, pp. 280–285, September 2008.
3. Tomohiro Otsuka, Michihiro Koibuchi, Tomohiro Kudoh, and Hideharu Amano. A Switch-tagged VLAN Routing Methodology for PC Clusters with Ethernet. In *Proc. of the 2006 International Conference on Parallel Processing (ICPP-06)*, pp. 479–486, August 2006.
4. Tomohiro Otsuka, Michihiro Koibuchi, Akiya Jouraku, Tomohiro Kudoh, and Hideharu Amano. VLAN-based Minimal Paths in PC Cluster with Ethernet on Mesh and Torus. In *Proc. of the 2005 International Conference on Parallel Processing (ICPP-05)*, pp. 567–576, June 2005.

研究会ほか

1. 鯉渕 道紘, 大塚 智宏, 松谷 宏紀, 天野 英晴. マルチパスイーサネットにおける省電力 On/Off リンクアクティベーション法. 情報処理学会研究報告 2009-ARC-174/2009-HPC-119, February 2009. (掲載予定)
2. 渡辺 崇文, 中尾 昌広, 廣安 知之, 鯉渕 道紘, 大塚 智宏. VLAN イーサネットを用いた大規模 PC クラスタの検討. 情報処理学会研究報告 2008-ARC-179, pp. 169–174, August 2008.

3. 鯉渕 道紘, 大塚 智宏, 工藤 知宏, 天野 英晴. イーサネットを用いた大規模クラスタネットワーク構築法. 第 6 回情報科学技術フォーラム (FIT2007) 論文集, pp. 71–74, September 2007.

4. 大塚 智宏, 鯉渕 道紘, 工藤 知宏, 天野 英晴. スイッチでタグ付けを行う VLAN ルーティング法の提案と評価. 情報処理学会研究報告 2006-ARC-167/2006-HPC-105, pp. 91–96, February 2006.

5. 大塚 智宏, 鯉渕 道紘, 上樂 明也, 工藤 知宏, 天野 英晴. VLAN を用いたマルチパス Ethernet における経路構築法. 情報処理学会研究報告 2005-ARC-164, pp. 121–126, August 2005.

その他の論文

公刊論文

1. 渡邊 幸之介, 大塚 智宏, 天野 英晴. 並列分散処理環境 RHiNET-2 システムの実装と評価. 電子情報通信学会論文誌, Vol. J90-D, No. 9, pp. 2465–2482, September 2007.

2. Konosuke Watanabe, Tomohiro Otsuka, Junichiro Tsuchiya, Hiroaki Nishi, Junji Yamamoto, Noboru Tanabe, Tomohiro Kudoh, and Hideharu Amano. Martini: A Network Interface Controller Chip for High Performance Computing with Distributed PCs. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 18, No. 9, pp. 1282–1295, June 2007.

3. Michihiro Koibuchi, Konosuke Watanabe, Tomohiro Otsuka, and Hideharu Amano. Performance Evaluation of Deterministic Routings, Multicasts, and Topologies on RHiNET-2 Cluster. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 16, No. 8, pp. 747–759, August 2005.

4. 鯉渕 道紘, 大塚 智宏, 渡邊 幸之介, 天野 英晴. RHiNET-2 クラスタにおけるユニキャストをもとにしたマルチキャストアルゴリズムの評価. 電子情報通信学会論文誌, Vol. J88-DI, No. 4, pp. 791–799, April 2005.

5. 鯉渕 道紘, 渡邊 幸之介, 大塚 智宏, 上樂 明也, 天野 英晴. RHiNET-2 クラスタを用いたデッドロックフリー固定ルーティングの実機評価. 情報処理学会論文誌コンピューティングシステム, Vol. 45, No. SIG 11(ACS 7), pp. 432–444, October 2004.

6. 鯉渕 道紘, 渡邊 幸之介, 大塚 智宏, 天野 英晴. RHiNET-2 クラスタを用いたシステムエリアネットワーク向けトポロジの実機評価. 情報処理学会論文誌コンピューティングシステム, Vol. 45, No. SIG 11(ACS 7), pp. 420–431, October 2004.

7. 渡邊 幸之介, 大塚 智宏, 天野 英晴. ネットワークインタフェース用コントローラチップ Martini における乗っ取り機構の実装と評価. 情報処理学会論文誌コンピューティングシステム, Vol. 45, No. SIG 11(ACS 7), pp. 393–407, October 2004.

国際会議

1. Akira Kitamura, Yoshihiro Hamada, Yasuo Miyabe, Tetsu Izawa, Tomotaka Miyashiro, Konosuke Watanabe, Tomohiro Otsuka, Noboru Tanabe, Hironori Nakajo, and Hideharu Amano. Evaluation of Network Interface Controller on DIMMnet-2 Prototype Board. In *Proc. of The 6th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT2005)*, pp. 778–780, December 2005.

2. Akira Kitamura, Konosuke Watanabe, Tomohiro Otsuka, and Hideharu Amano. The evaluation of dynamic load balancing algorithm on RHiNET-2. In *Proc. of the 15th IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2003)*, pp 262–267, November 2003.

3. Konosuke Watanabe, Tomohiro Otsuka, Jun-ichiro Tsuchiya, Hiroshi Harada, Junji Yamamoto, Hiroaki Nishi, Tomohiro Kudoh, and Hideharu Amano. Performance Evaluation of RHiNET-2/NI: A Network Interface for Distributed Parallel Computing Systems. In *Proc. of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2003)*, pp. 318–325, May 2003.

4. Konosuke Watanabe, Hideharu Amano, Junji Yamamoto, Jun-ichiro Tsuchiya, Tomohiro Otsuka, and Tomohiro Kudoh. Taking over mechanism: a Cooperation Methodology of Hardware and Software in Network Controller. In *Proc. of the 11th Workshop on Synthesis and System Integration of Mixed Information Technologies (SASIMI 2003)*, pp. 386–393, April 2003.

5. Tomohiro Otsuka, Konosuke Watanabe, Jun-ichiro Tsuchiya, Hiroshi Harada, Junji Yamamoto, Hiroaki Nishi, Tomohiro Kudoh, and Hideharu Amano. Performance Evaluation of a Prototype of RHiNET-2: A Network-based Distributed Parallel Computing System. In *Proc. of the 21st IASTED International Multi-Conference on Applied Informatics (AI 2003)*, pp. 738–743, February 2003.

研究会ほか

1. 今田 啓介, 酒井 洋介, 大塚 智宏, 鈴木 順, 樋口 淳一, 飛鷹 洋一, 天野 英晴. ExpEther における RDMA 通信のためのソフトウェア環境の構築. 情報処理学会研究報告 2008-ARC-179, pp. 163–168, August 2008.

2. 内山 幸憲, 今田 啓介, 辻 聡, 大塚 智宏, 鈴木 順, 樋口 淳一, 飛鷹 洋一, 天野 英晴. ExpEther における RDMA 通信機構の実装. 情報処理学会研究報告 2008-ARC-177/2008-HPC-114, pp. 175–180, March 2008.

3. 北村 聡, 伊豆 直之, 伊沢 徹, 宮代 具隆, 宮部 保雄, 渡邊 幸之介, 大塚 智宏, 濱田 芳博, 田邊 昇, 中條 拓伯, 天野 英晴. FPGA を用いたメモリスロット装着型ネットワークインタフェースの設計. FPGA/PLD Design Conference ユーザ・プレゼンテーション, pp. 13–20, January 2005.

4. 北村 聡, 伊豆 直之, 田邊 昇, 濱田 芳博, 中條 拓伯, 渡邊 幸之介, 大塚 智宏, 天野 英晴. DIMMnet-2 ネットワークインタフェースボードの試作. 情報処理学会研究報告 2004-ARC-159, pp. 151–156, September 2004.

5. 鯉渕 道紘, 渡邊 幸之介, 大塚 智宏, 天野 英晴. RHiNET-2 クラスタを用いたシステムエリアネットワーク向けトポロジの実機評価. 先進的計算基盤システムシンポジウム SACSIS2004 論文集, pp. 381–388, May 2004.

6. 大塚 智宏, 渡邊 幸之介, 北村 聡, 鯉渕 道紘, 山本 淳二, 西 宏章, 工藤 知宏, 天野 英晴. RHiNET プロジェクトの最終報告. 情報処理学会研究報告 2004-ARC-158, pp. 31–36, May 2004.

7. 鯉渕 道紘, 大塚 智宏, 渡邊 幸之介, 天野 英晴. RHiNET-2 クラスタにおけるユニキャストを基にしたマルチキャストアルゴリズムの評価. 情報処理学会研究報告 2004-EVA-8, pp. 25–30, March 2004.

8. 大門 優, 松尾 亜紀子, 大塚 智宏, 渡邊 幸之介, 天野 英晴. 反応を伴った圧縮性流体計算による RHiNET-2 の評価. 電子情報通信学会技術研究報告 CPSY2003-22, pp. 19–24, August 2003.

9. 大塚 智宏, 渡邊 幸之介, 北村 聡, 原田 浩, 山本 淳二, 西 宏章, 工藤 知宏, 天野 英晴. 分散並列処理用ネットワーク RHiNET-2 の性能評価. 先進的計算基盤システムシンポジウム SACSIS2003 論文集, pp. 45–52, May 2003.

10. 北村 聡, 天野 英晴, 渡邊 幸之介, 大塚 智宏. PC クラスタ用ネットワーク RHiNET-2 上における動的負荷分散アルゴリズムの評価. 情報処理学会研究報告 2003-ARC-152, pp. 73–78, March 2003.

11. 大塚 智宏, 渡邊 幸之介, 土屋 潤一郎, 原田 浩, 山本 淳二, 西 宏章, 工藤 知宏, 天野 英晴. RHiNET ネットワークインタフェースの性能評価. 電子情報通信学会技術研究報告 CPSY2002-44, pp. 23–28, August 2002.

12. 大塚 智宏, 横山 知典, 土屋 潤一郎, 宮脇 達朗, 清水 敏行, 山本 淳二, 西 宏章, 工藤 知宏, 天野 英晴. RHiNET ネットワークインタフェースプロトタイプの性能評価. 情報処理学会研究報告 2001-ARC-143, pp. 13–18, May 2001.

付 録A　*k*-ary *n*-cubeにおける VLAN割り当て手法

本付録では，*k*-ary *n*-cube メッシュおよびトーラストポロジにおいて，最短かつ効率的なトラフィックの分散を実現するパス集合を構築するための VLAN 割り当て手法について述べる．

A.1　メッシュにおける VLAN 割り当て手法

本節では，メッシュ上で最短パス集合を構築するための 2 種類の VLAN 割り当て手法を提案する．1 つ目の手法は，*k*-ary *n*-cube 上でデッドロックフリーを保証しつつパスの分散を実現する手法として知られる次元順ルーティングに従う最短パスを保証する．2 つ目の手法では，トラフィックの分散若干の偏りが生じるが，必要となる VLAN 数を 1 つ目の手法の約半分に削減しつつ最短パスを保証することができる．

A.1.1　準備

図 A.1に，4×4 2 次元メッシュ(4-ary 2-cube) とそれへの VLAN 割り当て例を示す．図において，各頂点および辺はそれぞれイーサネットスイッチとリンクを表す．各スイッチには何台かのホストを接続することができるが，ここでは省略している．

2 次元メッシュは一般に *k*-ary 2-cube として定義され，各頂点 (スイッチ) は以下のように 2 次元座標にマッピングされる．

$$\begin{array}{cccc} (0,0) & (1,0) & \cdots & (k-1,0) \\ (0,1) & (1,1) & \cdots & (k-1,1) \\ \vdots & \vdots & \ddots & \vdots \\ (0,k-1) & (1,k-1) & \cdots & (k-1,k-1) \end{array}$$

定義 A.1 (2 次元メッシュ) 各頂点 (スイッチ) に，$0 \le x, y < k$ として 2 次元座標 (x, y) を割り当てる．頂点 (x, y) を頂点 $(x-1, y)$, $(x+1, y)$, $(x, y-1)$, $(x, y+1)$ とそれぞれ $x-1 \ge 0$, $x+1 < k$, $y-1 \ge 0$, $y+1 < k$ の場合に接続することにより，$k \times k$ 2 次元メッシュが構成される． ∎

図 A.1の各 VLAN トポロジ A〜C は，物理ネットワークのスパニングツリー (全域木) となっており，k^2 個のスイッチと k^2-1 本のリンクから構成される．図に示すように，メッシュ上の VLAN トポロジにはさまざまなものが考えられるが，VLAN C のような不規則なトポロジを他のトポロジとの組み合わせで用いるのは困難であるため，提案手法では，VLAN A や B のような単純かつ規則的なトポロジを組み合わせて最短パス集合を構築する．ここで，提案手法で用いる VLAN トポロジを識別するため，以下の記法を導入する．

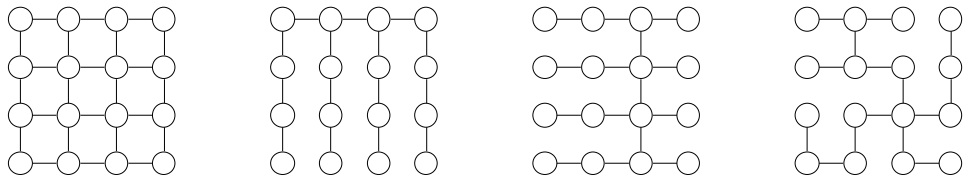


図 A.1　4×4 2 次元メッシュと VLAN トポロジの例

定義 A.2 (2 次元メッシュ上の線形接続) 2 次元メッシュにおける垂直接続 $l(x_0, -)$ とは，頂点集合 $\{(x_0, y) \mid 0 \le y < k\}$ 内の各頂点 (x_0, y) を頂点 $(x_0, y-1)$, $(x_0, y+1)$ とそれぞれ $y-1 \ge 0$, $y+1 < k$ の場合に接続してできる y 軸に平行な直線トポロジである．

同様に，2 次元メッシュにおける水平接続 $l(-, y_0)$ とは，頂点集合 $\{(x, y_0) \mid 0 \le x < k\}$ 内の各頂点 (x, y_0) を頂点 $(x-1, y_0)$, $(x+1, y_0)$ とそれぞれ $x-1 \ge 0$, $x+1 < k$ の場合に接続してできる x 軸に平行な直線トポロジである． ∎

VLAN トポロジは，1 本の垂直接続または水平接続と，もう一方の次元の k 本の線形接続すべてを用いることで構成できる．例えば，図 A.1の VLAN A は，接続 $l(0, -)$, $l(1, -)$, $l(2, -)$, $l(3, -)$, $l(-, 0)$ で構成され，VLAN B は $l(-, 0)$, $l(-, 1)$, $l(-, 2)$, $l(-, 3)$, $l(2, -)$ で構成される．このような VLAN トポロジを記述するために，以下の記法を導入する．

定義 A.3 (2 次元メッシュ上の VLAN トポロジ) 2 次元メッシュ上の VLAN トポロジ $V(-, y_0)$ および $V(x_0, -)$ とは，全頂点 (スイッチ) と，それぞれ以下の $k+1$ 本の線形接続から構成されるスパニングツリートポロジである．

$$V(-, y_0): \quad \{l(x, -) \mid 0 \le x < k\} \cup \{l(-, y_0)\}$$
$$V(x_0, -): \quad \{l(-, y) \mid 0 \le y < k\} \cup \{l(x_0, -)\}$$

∎

この定義により，図 A.1の VLAN A, B はそれぞれ $V(-, 0)$, $V(2, -)$ と表される．

A.1.2　2 次元メッシュ上の DOR VLAN 集合

本節では，2 次元メッシュ上で次元順ルーティング (Dimension-order Routing, DOR)[63] に従う最短パス集合を構築するための VLAN 割り当て手法を提案する．次元順ルーティングは，k-ary n-cube 上でデッドロックフリーを保証しつつ効率的なパスの分散を実現するルーティングアルゴリズムとして知られている．2 次元メッシュ (k-ary 2-cube) 上の次元順ルーティングでは，パケットはまず x 方向に必要ホップ数転送された後，y 方向に転送される．

定義 A.4 (2 次元メッシュ上の DOR VLAN 集合) $k \times k$ 2 次元メッシュ上の DOR VLAN 集合は，以下の k 個の VLAN で構成される．

$$\{V(-, y) \mid 0 \le y < k\}$$

∎

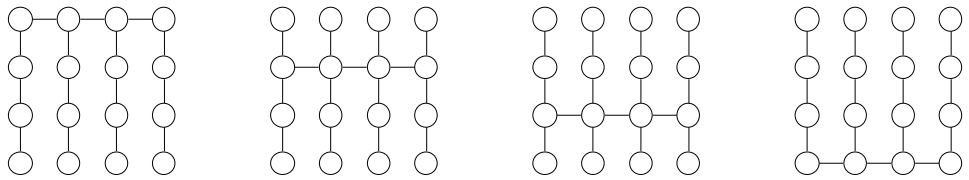


図 A.2　4×4 2 次元メッシュ上の DOR VLAN 集合

ここで，スイッチ (x_S, y_S) からのパスを VLAN $V(-, y_S)$ に割り当てることにより，次元順ルーティングに従う最短パス集合を構築することができる．図 A.2は DOR VLAN 集合の例であり，4×4 2 次元メッシュにおいて，$V(-,0)$, $V(-,1)$, $V(-,2)$, $V(-,3)$ の 4 つの VLAN が DOR VLAN 集合を構成することを示している．

定理 A.1 2 次元メッシュにおいて，DOR VLAN 集合は次元順ルーティングに従う最短パス集合を提供する．

証明　VLAN $V(-, y_S)$ は水平接続 $l(-, y_S)$ と k 本の垂直接続から構成される．よって，$V(-, y_S)$ 上でスイッチ (x_S, y_S) をソースとするすべてのパスは次元順ルーティングに従う最短パスである．DOR VLAN 集合は k 個の VLAN $\{V(-, y) \mid 0 \le y < k\}$ から構成されるため，各スイッチ (x, y) をソースとするパスはそのうちの 1 つ $V(-, y)$ 上で必ず次元順ルーティングに従う最短パスとなる．∎

A.1.3　2 次元メッシュ上の PDOR VLAN 集合

本節では，2 次元メッシュにおいて DOR VLAN 集合よりも少ない VLAN 数で次元順ルーティングに近い最短パス集合を提供する手法を提案する．

定義 A.5 (2 次元メッシュ上の PDOR VLAN 集合) $k \times k$ 2 次元メッシュ上の PDOR VLAN 集合は，以下の $k/2 + 1$ 個の VLAN で構成される．

$$\left\{ V(-, 2i+1) \;\middle|\; 0 \le i < \frac{k-1}{2} \right\} \cup \{V(x_0, -)\}$$

∎

図 A.3は PDOR VLAN 集合の例であり．4×4 2 次元メッシュにおいて，$V(-,1)$, $V(-,3)$, $V(1,-)$ の 3 つの VLAN が PDOR VLAN 集合を構成することを示している．なお，x_0 の値は任意であり，$V(x_0, -)$ 内の垂直接続に属するリンクがパスに含まれることはない．

ここで，スイッチ $(x, 2i+1)$ からのパスは，VLAN $V(-, 2i+1)$ に割り当てることによりすべて次元順ルーティングに従う最短パスとなる．しかし，スイッチ $(x, 2i)$ からのパスは，DOR VLAN 集合と違い VLAN $V(-, 2i)$ が存在しないため次元順ルーティングに従うパスとはならない．そのため，すべてのソースからのパスが最短となるように，スイッチ (x_S, y_S), (x_D, y_D) をそれぞれソースおよびデスティネーションとするパスは，以下の手続きに従って使用する VLAN を決定する．

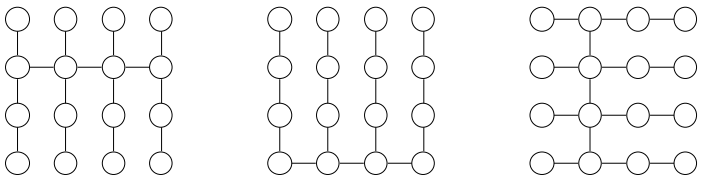


図 A.3　4×4 2 次元メッシュ上の PDOR VLAN 集合

if y_S **mod** 2 = 1 **then** use $V(-, y_S)$;
else if $y_D < y_S$ **then** use $V(-, y_S - 1)$;
else if $y_D > y_S$ **then** use $V(-, y_S + 1)$;
else $\{y_D = y_S\}$ use $V(x_0, -)$;

例えば，4×4 2 次元メッシュにおいて $(0,0)$ をソース，$(3,2)$ をデスティネーションとするパスは，図 A.3に示した VLAN $V(-,1)$ を用いて以下の通りとなる．

$$(0,0) \to (0,1) \to (1,1) \to (2,1) \to (3,1) \to (3,2)$$

定理 A.2 2 次元メッシュにおいて，PDOR VLAN 集合は最短パス集合を提供する．

証明　VLAN $V(x_0,-)$ はすべての水平接続を含んでいるため，スイッチ $(x_S, 2i)$ から $(x, 2i)$ $(0 \le x < k)$ へのパスは $V(x_0,-)$ を用いることで最短パスとなる．他のデスティネーションへは，$y_D > y_S$ の場合は $(x_S, 2i+1)$ 経由で $V(-, 2i+1)$ を用い，$y_D < y_S$ の場合は $(x_S, 2i-1)$ 経由で $V(-, 2i-1)$ を用いることにより，$(x_S, 2i)$ からのパスはすべて最短となる．

一方，定理 A.1より，$(x_S, 2i+1)$ からのパスは $V(-, 2i+1)$ を用いることで最短となる．よって，2 次元メッシュ上の PDOR VLAN 集合は最短パス集合を提供する． ∎

この手法では，$(x_S, 2i)$ をソースとするパスは次元順ルーティングに沿ったパスにはならないが，違いは最初の y 方向への転送のみであり，トラフィックは各パスに十分に分散される．

A.1.4　*n* 次元メッシュへの一般化

本節では，DOR VLAN 集合および PDOR VLAN 集合を n 次元メッシュ(k-ary n-cube) へと拡張し一般化する．まず，定義 A.1を拡張し，n 次元メッシュ上の各頂点 (スイッチ) に n 次元座標 $(x_0, x_1, \ldots, x_{n-1})$ (ただし，$0 \le x_0, x_1, \ldots, x_{n-1} < k$) を割り当てる．次に，定義 A.2を拡張し，$l(x_0, x_1, \ldots, x_{i-1}, -, x_{i+1}, \ldots, x_{n-1})$ を $\{(x_0, x_1, \ldots, x_i, \ldots, x_{n-1}) \mid 0 \le x_i < k\}$ の k 個の頂点を含む i 次元方向に平行な線形接続として定義する．

定義 A.6 (n 次元メッシュ上の VLAN トポロジ) n 次元メッシュ上の VLAN トポロジ

$$V\left(x_0, x_1, \ldots, x_{i_0-1}, -, x_{i_0+1}, \ldots, x_{n-1} \mid (i_0, i_1, \ldots, i_{n-1})\right) \quad \left(0 \le i_j < n,\ i_j \ne i_k\ (j \ne k)\right)$$

は，全頂点と，以下の $k^n + k^{n-1} + \ldots + k + 1$ 個の線形接続 (i_0 次元方向に平行な 1 本の接続，i_1 次元方向に平行な k 本の接続，...) から構成される．

$$
\begin{aligned}
& \{l(x_0, x_1, \ldots, x_{i_0-1}, -, x_{i_0+1}, \ldots, x_{n-1})\} \\
\cup\ & \{l(x_0, x_1, \ldots, x_{i_1-1}, -, x_{i_1+1}, \ldots, x_{n-1}) \mid 0 \le x_{i_0} < k\} \\
\cup\ & \{l(x_0, x_1, \ldots, x_{i_2-1}, -, x_{i_2+1}, \ldots, x_{n-1}) \mid 0 \le x_{i_0}, x_{i_1} < k\} \\
& \qquad\qquad \vdots \\
\cup\ & \{l(x_0, x_1, \ldots, x_{i_{n-1}-1}, -, x_{i_{n-1}+1}, \ldots, x_{n-1}) \mid 0 \le x_{i_0}, x_{i_1}, \ldots, x_{i_{n-2}} < k\}
\end{aligned}
$$

∎

定義 A.7 (n 次元メッシュ上の DOR VLAN 集合) k^n n 次元メッシュ上の DOR VLAN 集合は，以下の k^{n-1} 個の VLAN で構成される．

$$
\left\{ V(-, x_1, x_2, \ldots, x_{n-1} \mid A) \;\middle|\; 0 \le x_i < k,\ 1 \le i < n \right\} \quad (A = (0, 1, \ldots, n-1))
$$

∎

ここで，スイッチ $(x_{0_S}, x_{1_S}, \ldots, x_{(n-1)_S})$ からのパスを VLAN $V(-, x_{1_S}, x_{2_S}, \ldots, x_{(n-1)_S} \mid A)$ に割り当てることにより，次元順ルーティングに従う最短パス集合を構築することができる．

定義 A.8 (n 次元メッシュ上の PDOR VLAN 集合) k^n n 次元メッシュ上の PDOR VLAN 集合は，以下の $k^{n-1}/2 + 1$ 個の VLAN で構成される．

$$
\begin{aligned}
& \left\{ V(-, x_1, x_2, \ldots, x_{n-1} \mid A) \;\middle|\; \sum x_i \equiv 1 \bmod 2,\ 0 \le x_i < k,\ 1 \le i < n \right\} \\
\cup\ & \{V(x_0, x_1, \ldots, x_{n-2}, - \mid B)\} \qquad (A = (0, 1, \ldots, n-1),\ B = (n-1, n-2, \ldots, 0))
\end{aligned}
$$

∎

$(x_{0_S}, x_{1_S}, \ldots, x_{(n-1)_S})$ から $(x_{0_D}, x_{1_D}, \ldots, x_{(n-1)_D})$ へのパスは，以下の手続きに従って使用する VLAN を選択することで最短パスとなる．

if $\sum_{i \neq 0} x_{i_S}$ **mod** 2 = 1 **then**
  use $V(-, x_{1_S}, x_{2_S}, \ldots, x_{(n-1)_S} \mid A)$;
else begin
  selected := *false*;
  for i := 1 **to** $n - 1$ **do**
    if $x_{i_D} > x_{i_S}$ **then begin**
      use $V(-, x_{1_S}, x_{2_S}, \ldots, x_{(i-1)_S}, x_{i_S} + 1, x_{(i+1)_S}, \ldots, x_{(n-1)_S} \mid A)$;
      selected := *true*;
      break;
    end else if $x_{i_D} < x_{i_S}$ **then begin**
      use $V(-, x_{1_S}, x_{2_S}, \ldots, x_{(i-1)_S}, x_{i_S} - 1, x_{(i+1)_S}, \ldots, x_{(n-1)_S} \mid A)$;
      selected := *true*;
      break;

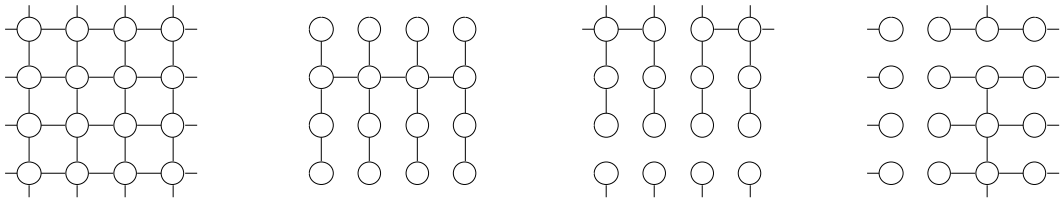


図 A.4　4×4 2 次元トーラスと VLAN トポロジの例

end;
end;
if *selected* ≠ *true* **then** use $V(x_0, x_1, \ldots, x_{n-2}, - \mid B)$;
end;

ここで，VLAN $V(x_0, x_1, \ldots, x_{n-2}, - \mid B)$ が選択されるのは，以下の条件を満たす場合である．

$$\sum x_{i_S} \equiv 0 \bmod 2,\ x_{i_S} = x_{i_D}\ (1 \le i < n)$$

A.2　トーラスにおける VLAN 割り当て手法

本節では，トーラス上で最短パス集合を構築するための 2 種類の VLAN 割り当て手法を提案する．提案手法はそれぞれ，メッシュにおける割り当て手法と同様，次元順ルーティングに従う最短パスを保証する手法と，必要となる VLAN 数をその約半分に削減する手法である．

A.2.1　準備

図 A.4に，4×4 2 次元トーラス (4-ary 2-cube) とそれへの VLAN 割り当て例を示す．メッシュと違い，トーラスには頂点 $(x, k-1)$ と $(x, 0)$，$(k-1, y)$ と $(0, y)$ をそれぞれ接続する wrap-around リンクが存在し，図では切れているが，実際にはそれぞれつながっている．

2 次元トーラスは一般に *k*-ary 2-cube として定義され，各頂点 (スイッチ) は 2 次元メッシュの場合と同様に 2 次元座標にマッピングされる．

定義 A.9 (2 次元トーラス) 各頂点 (スイッチ) に，$0 \le x, y < k$ として 2 次元座標 (x, y) を割り当てる．頂点 (x, y) を 4 つの頂点 $((x \pm 1 + k) \bmod k, y)$，$(x, (y \pm 1 + k) \bmod k)$ とそれぞれ接続することにより，$k \times k$ 2 次元トーラスが構成される． ∎

定義 A.10 (2 次元トーラス上の線形接続) 2 次元トーラスにおける垂直接続 $l(x_0, - : y_0)$ とは，頂点集合 $\{(x_0, y) \mid 0 \le y < k\}$ 内の各頂点 (x_0, y) を 2 頂点 $(x_0, (y \pm 1 + k) \bmod k)$ と接続 (ただし，$y_r = (y_0 + k/2) \bmod k$ として (x_0, y_r)，$(x_0, (y_r + 1) \bmod k)$ 間のリンクを除く) してできる y 軸に平行な直線トポロジである．

同様に，2 次元トーラスにおける水平接続 $l(- : x_0, y_0)$ とは，頂点集合 $\{(x, y_0) \mid 0 \le x < k\}$ 内の各頂点 (x, y_0) を 2 頂点 $((x \pm 1 + k) \bmod k, y_0)$ と接続 (ただし，$x_r = (x_0 + k/2) \bmod k$ として (x_r, y_0)，$((x_r + 1) \bmod k, y_0)$ 間のリンクを除く) してできる x 軸に平行な直線トポロジである． ∎

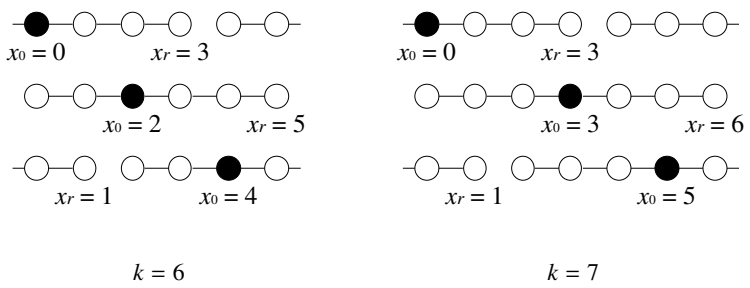


図 A.5　2 次元トーラス上の水平接続の例

ここで，図 A.5に示すように，トーラス上の線形接続 $l(x_0, -: y_0)$ および $l(-: x_0, y_0)$ において，頂点 (x_0, y_0) (図で黒く塗り潰した頂点) は直線トポロジの「中心」に位置している．

定義 A.10によれば，図 A.4の VLAN A は接続 $l(0, -: 1)$, $l(1, -: 1)$, $l(2, -: 1)$, $l(3, -: 1)$, $l(-: 1, 1)$ で構成される．同様に，VLAN B は接続 $l(0, -: 0)$, $l(1, -: 0)$, $l(2, -: 0)$, $l(3, -: 0)$, $l(-: 3, 0)$，VLAN C は接続 $l(-: 2, 0)$, $l(-: 2, 1)$, $l(-: 2, 2)$, $l(-: 2, 3)$, $l(2, -: 2)$ でそれぞれ構成される．

定義 A.11 (2 次元トーラス上の VLAN トポロジ) 2 次元トーラス上の VLAN トポロジ $V(-: x_0, y_0)$ および $V(x_0, -: y_0)$ とは，全頂点 (スイッチ) と，それぞれ以下の $k + 1$ 本の線形接続から構成されるスパニングツリートポロジである．

$$V(-: x_0, y_0): \quad \{l(x, -: y_0) \mid 0 \le x < k\} \cup \{l(-: x_0, y_0)\}$$
$$V(x_0, -: y_0): \quad \{l(-: x_0, y) \mid 0 \le y < k\} \cup \{l(x_0, -: y_0)\}$$

∎

この定義により，図 A.4の VLAN A〜C はそれぞれ $V(-: 1, 1)$, $V(-: 3, 0)$, $V(2, -: 2)$ と表される．

定義 A.12 (トーラス上の 2 頂点間の距離) 2 次元トーラスにおいて，2 頂点 (x_S, y_S), (x_D, y_D) 間の x 座標の正の距離 $d^+(x_S, x_D)$，負の距離 $d^-(x_S, x_D)$ とは，それぞれ以下の値である．

$$d^+(x_S, x_D) = (x_D - x_S + k) \bmod k$$
$$d^-(x_S, x_D) = (x_S - x_D + k) \bmod k$$

また，$d^+(x_S, x_D)$ と $d^-(x_S, x_D)$ の小さい方の値を $d(x_S, x_D)$ と表し，単に x 座標の距離と呼ぶ．

$$d(x_S, x_D) = \min\left(d^+(x_S, x_D), d^-(x_S, x_D)\right)$$

同様に，(x_S, y_S), (x_D, y_D) 間の y 座標の距離とは，以下の値である．

$$d^+(y_S, y_D) = (y_D - y_S + k) \bmod k$$
$$d^-(y_S, y_D) = (y_S - y_D + k) \bmod k$$
$$d(y_S, y_D) = \min\left(d^+(y_S, y_D), d^-(y_S, y_D)\right)$$

∎

例えば，4 × 4 2 次元トーラスにおいて，スイッチ (0, 3) から (3, 1) への距離は以下の通りである．

$$d^+(x_S, x_D) = 3, \quad d^-(x_S, x_D) = 1, \quad d(x_S, x_D) = 1$$
$$d^+(y_S, y_D) = 2, \quad d^-(y_S, y_D) = 2, \quad d(y_S, y_D) = 2$$

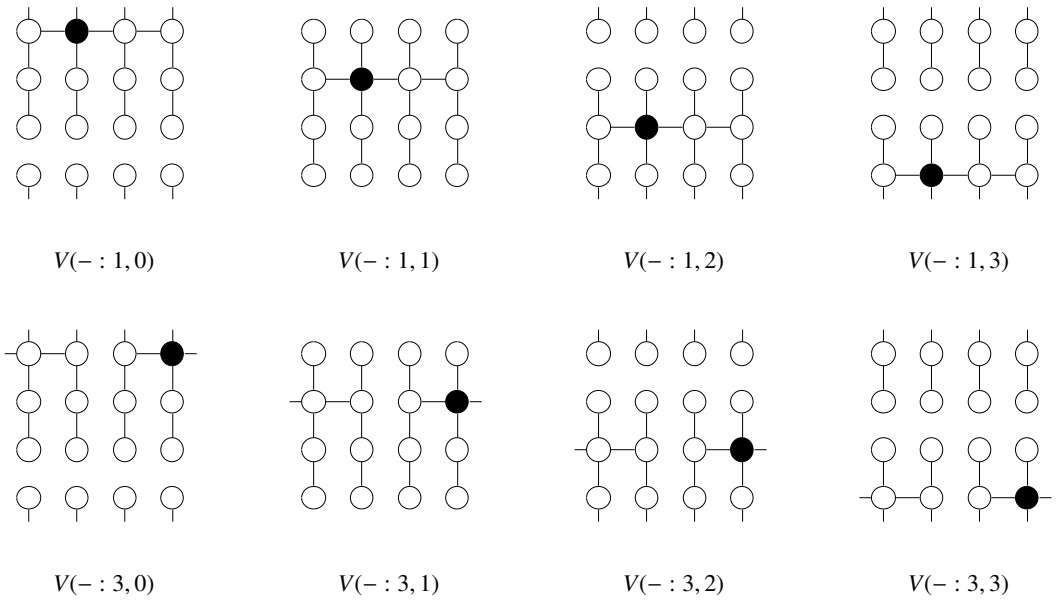


図 A.6　4×4 2 次元トーラス上の DOR VLAN 集合

A.2.2　2 次元トーラス上の DOR VLAN 集合

本節では，2 次元トーラス上で次元順ルーティング (DOR) に従う最短パス集合を構築するための VLAN 割り当て手法を提案する．

定義 A.13 (2 次元トーラス上の DOR VLAN 集合) $k \times k$ 2 次元トーラス上の DOR VLAN 集合は，以下の $2k$ 個の VLAN で構成される．

$$\{V(-:x,y) \mid x = a,b,\ 0 \le y < k\} \quad \left(a = \frac{k-1}{2},\ b = k-1\right)$$

∎

この VLAN 割り当ては，メッシュ上の DOR VLAN 集合と似ているが，wrap-around リンクが存在するために 2 倍の VLAN 数を使用する．図 A.6はトーラス上の DOR VLAN 集合の例であり，4 × 4 2 次元トーラスにおいて，$V(-:1,0)$，$V(-:1,1)$，$V(-:1,2)$，$V(-:1,3)$，$V(-:3,0)$，$V(-:3,1)$，$V(-:3,2)$，$V(-:3,3)$ の 8 つの VLAN が DOR VLAN 集合を構成することを示している．

ここで，スイッチ (x_S, y_S) から (x_D, y_D) へのパスは，以下の手続きに従って $V(-:a,y_S)$，$V(-:b,y_S)$ のいずれかの VLAN に割り当てることにより，すべて次元順ルーティングに従う最短パスとなる．

```
function select_ab(s, d, N : integer) : integer;
begin
  if d+(s,d) ≤ d−(s,d) then begin
    if s < k/2 then select_ab := a;
    else select_ab := b;
  end else {d+(s,d) > d−(s,d)} begin
    if s < k/2 then select_ab := b;
```

```
        else select_ab := a;
    end;
end;
use V(−: select_ab(x_S, x_D, k), y_S);
```

例えば，4×4 2 次元トーラスにおいて $(0,0)$ をソース，$(3,2)$ をデスティネーションとするパスは，図 A.6に示した VLAN $V(-:3,0)$ を用いて以下の通りとなる．

$$(0,0) \to (3,0) \to (3,1) \to (3,2)$$

補題 A.1 トーラスにおいて，ある 1 つの次元に沿ったすべての最短パスは，それぞれ $a = (k-1)/2$ と $b = k-1$ を中心とする 2 つの線形接続のどちらかに含まれる．

証明 x 次元に沿ったパスを仮定し，2 つの線形接続を $l(-:a,y_0)$, $l(-:b,y_0)$ とする．$l(-:a,y_0)$ は $(k-1,y_0)$, $(0,y_0)$ 間の wrap-around リンクのみを欠いており，x 座標の距離 $d(x_S,x_D)$ の最大値は $k/2$ であるため，$l(-:a,y_0)$ に含まれない最短パスの集合は以下の頂点集合にわたる．

$$\{(x,y_0) \mid x = x_1, x_1+1, \ldots, k-1, 0, 1, \ldots, x_2\} \quad \left(x_1 = k - \frac{k}{2},\ x_2 = \left(k - 1 + \frac{k}{2}\right) \bmod k\right)$$

ここで，$x_1 > x_2$ である．一方，$l(-:b,y_0)$ の中央の x 座標 $x_r = (b + k/2) \bmod k = x_2$ であるため，$l(-:b,y_0)$ は上記の頂点集合にわたるパスをすべて含む．よって，$\{(x,y_0) \mid 0 \le x < k\}$ 内の任意の頂点をソースおよびデスティネーションとする最短パスは，2 つの線形接続 $l(-:a,y_0)$, $l(-:b,y_0)$ のどちらかに必ず含まれる． ∎

定理 A.3 2 次元トーラスにおいて，DOR VLAN 集合は次元順ルーティングに従う最短パス集合を提供する．

証明 VLAN $V(-:a,y_S)$, $V(-:b,y_S)$ はそれぞれ水平接続 $l(-:a,y_S)$, $l(-:b,y_S)$ と k 本の垂直接続 $\{l(x,y_S) \mid 0 \le x < k\}$ から構成される．補題 A.1より，$\{(x,y_S) \mid 0 \le x < k\}$ 内の 2 頂点間の最短パスは，この 2 つの水平接続のいずれかに含まれる．また，k 本の各垂直接続 $l(x,y_S)$ において，(x,y_S) は接続の中心に位置するため，(x,y_S) から (x,y) $(0 \le y < k)$ への最短パスはすべてそれぞれの垂直接続に含まれる．DOR VLAN 集合は $2k$ 個の VLAN $\{V(-:x,y) \mid x = a,b,\ 0 \le y < k\}$ で構成されるため，(x_S,y_S) からのパスは，$V(-:a,y_S)$, $V(-:b,y_S)$ のうち適切な VLAN を選択することですべて次元順ルーティングに従う最短パスとなる． ∎

A.2.3 2 次元トーラス上の PDOR VLAN 集合

本節では，2 次元トーラスにおいて DOR VLAN 集合よりも少ない VLAN 数で次元順ルーティングに近い最短パス集合を提供する手法を提案する．

定義 A.14 (2 次元トーラス上の PDOR VLAN 集合) $k \times k$ 2 次元トーラス上の PDOR VLAN 集合は，以下の $2((k+1)/2) + 2$ 個の VLAN で構成される．

$$\left\{V(-:x,2i) \;\middle|\; x = a,b,\ 0 \le i < \frac{k}{2}\right\} \cup \{V(a,-:y_0),\ V(b,-:y_1)\} \quad \left(a = \frac{k-1}{2},\ b = k-1\right)$$

∎

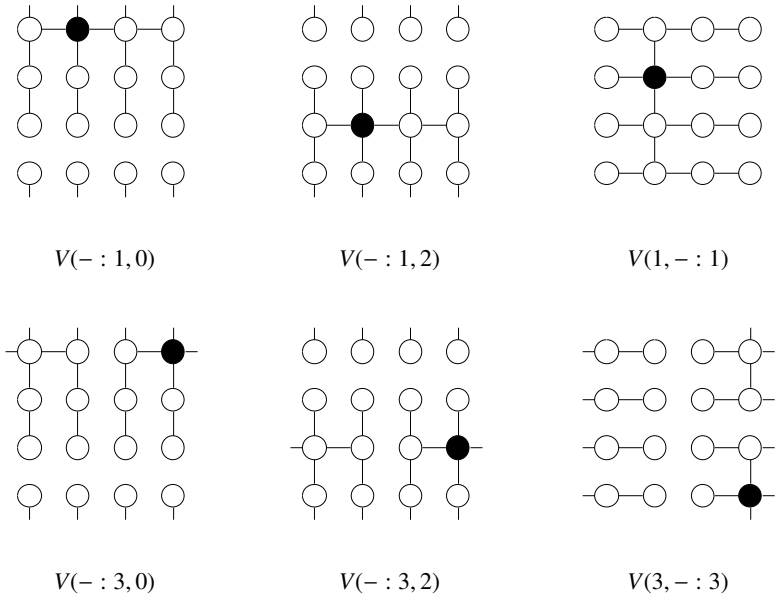


図 A.7　4×4 2 次元トーラス上の PDOR VLAN 集合

図 A.7はトーラス上の PDOR VLAN 集合の例であり，4 × 4 2 次元トーラスにおいて，$V(-:1,0)$, $V(-:1,2)$, $V(-:3,0)$, $V(-:3,2)$, $V(1,-:1)$, $V(3,-:3)$ の 6 つの VLAN が DOR VLAN 集合を構成することを示している．なお，y_0, y_1 の値は任意であり，$V(a,-:y_0)$, $V(b,-:y_1)$ 内の垂直接続に属するリンクがパスに含まれることはない．

ここで，スイッチ $(x_S, 2i)$ からのパスは，DOR VLAN 集合の場合と同じ方法により $V(-:a,2i)$, $V(-:b,2i)$ のいずれかの VLAN に割り当てることで，次元順ルーティングに従う最短パスとなる．一方，スイッチ $(x_S, 2i+1)$ からのパスは，$V(-:a,2i)$, $V(-:b,2i)$, $V(-:a,(2i+2) \bmod k)$, $V(-:b,(2i+2) \bmod k)$, $V(a,-:y_0)$, $V(b,-:y_1)$ の 6 つの VLAN のいずれかに割り当てることにより最短パスとなる．スイッチ (x_S, y_S) をソース，(x_D, y_D) をデスティネーションとしたとき，2 次元トーラス上の PDOR VLAN 集合における VLAN 選択手続きは以下のようになる (関数 *select_ab* は A.2.2節で示した).

$ab := select\_ab(x_S, x_D, k)$;
if y_S **mod** 2 = 0 **then** use $V(-:ab, y_S)$;
else begin
　if $y_D = y_S$ **then** use $V(ab,-:y_{ab})$;
　else if $d^+(y_S, y_D) \le d^-(y_S, y_D)$ **then** use $V(-:ab, (y_S+1) \bmod k)$;
　else $\{d^+(y_S, y_D) > d^-(y_S, y_D)\}$ use $V(-:ab, y_S - 1)$;
end;

なお，$ab = a$ のとき $y_{ab} = y_0$，$ab = b$ のとき $y_{ab} = y_1$ である．例えば，4×4 2 次元トーラスにおいて (0,3) をソース，(1,1) をデスティネーションとするパスは，図 A.7に示した VLAN $V(-:1,0)$ を用いて以下の通りとなる．

$$(0,3) \to (0,0) \to (1,0) \to (1,1)$$

定理 A.4　2 次元トーラスにおいて，PDOR VLAN 集合は最短パス集合を提供する．

証明 $V(a,-:y_0)$, $V(b,-:y_1)$ はそれぞれ $\{l(-:a,y) \mid 0 \le y < k\}$ と $\{l(-:b,y) \mid 0 \le y < k\}$ の k 本の水平接続を含むため，補題 A.1より，$(x_S, 2i+1)$ から $(x, 2i+1)$ $(0 \le x < k)$ へのパスは $V(a,-:y_0)$, $V(b,-:y_1)$ のいずれかに割り当てることで最短パスとなる．$(x_S, 2i+1)$ から他のデスティネーションへのパスは，$d^+(y_S,y_D) \le d^-(y_S,y_D)$ の場合は $V(-:a,(2i+2) \bmod k))$ または $V(-:b,(2i+2) \bmod k))$ を用いて $(x_S,(2i+2) \bmod k)$ を経由することにより，また $d^+(y_S,y_D) > d^-(y_S,y_D)$ の場合は $V(-:a,2i)$ または $V(-:b,2i)$ を用いて $(x_S,2i)$ を経由することにより最短パスとなる．

一方，定理 A.3より，$(x_S,2i)$ からのパスは $V(-:a,2i)$, $V(-:b,2i)$ のいずれかを用いることにより最短パスとなる．よって，2 次元トーラス上の PDOR VLAN 集合は最短パス集合を提供する．∎

この手法では，$(x_S, 2i+1)$ をソースとするパスは次元順ルーティングに沿ったパスにはならないが，メッシュにおける PDOR VLAN 集合と同様，違いは最初の y 方向への転送のみであり，トラフィックは各パスに十分に分散される．

A.2.4 n 次元トーラスへの一般化

本節では，DOR VLAN 集合および PDOR VLAN 集合を n 次元トーラス (k-ary n-cube) へと拡張し一般化する．まず，定義 A.9を拡張し，n 次元トーラス上の各頂点 (スイッチ) に n 次元座標 $(x_0, x_1, \ldots, x_{n-1})$ (ただし，$0 \le x_0, x_1, \ldots, x_{n-1} < k$) を割り当てる．次に，定義 A.10を拡張し，$l(x_0, x_1, \ldots, x_{i-1}, -:x_{i_0}, x_{i+1}, \ldots, x_{n-1})$ を $\{(x_0, x_1, \ldots, x_i, \ldots, x_{n-1}) \mid 0 \le x_i < k\}$ の k 個の頂点を含み $(x_0, x_1, \ldots, x_{i-1}, x_{i_0}, x_{i+1}, \ldots, x_{n-1})$ を中心とする i 次元方向に平行な線形接続として定義する．

定義 A.15 (n 次元トーラス上の VLAN トポロジ) n 次元トーラス上の VLAN トポロジ

$$V\left(x_0, x_1, \ldots, x_{i_0-1}, -:x_{i_0}, x_{i_0+1}, \ldots, x_{n-1} \mid (i_0, i_1, \ldots, i_{n-1})\right) \quad \left(0 \le i_j < n,\ i_j \ne i_k\ (j \ne k)\right)$$

は，全頂点と，以下の $k^n + k^{n-1} + \ldots + k + 1$ 個の線形接続 (i_0 次元方向に平行な 1 本の接続，i_1 次元方向に平行な k 本の接続，…) から構成される．

$$\begin{aligned}
& \{l(x_0, x_1, \ldots, x_{i_0-1}, -:x_{i_0}, x_{i_0+1}, \ldots, x_{n-1})\} \\
\cup\ & \{l(x_0, x_1, \ldots, x_{i_1-1}, -:x_{i_1}, x_{i_1+1}, \ldots, x_{n-1}) \mid 0 \le x_{i_0} < k\} \\
\cup\ & \{l(x_0, x_1, \ldots, x_{i_2-1}, -:x_{i_2}, x_{i_2+1}, \ldots, x_{n-1}) \mid 0 \le x_{i_0}, x_{i_1} < k\} \\
& \qquad\qquad \vdots \\
\cup\ & \{l(x_0, x_1, \ldots, x_{i_{n-1}-1}, -:x_{i_{n-1}}, x_{i_{n-1}+1}, \ldots, x_{n-1}) \mid 0 \le x_{i_0}, x_{i_1}, \ldots, x_{i_{n-2}} < k\}
\end{aligned}$$

∎

定義 A.16 (n 次元トーラス上の DOR VLAN 集合) k^n n 次元トーラス上の DOR VLAN 集合は，以下の $2k^{n-1}$ 個の VLAN で構成される．

$$\left\{V(-:x_0, x_1, x_2, \ldots, x_{n-1} \mid A) \;\middle|\; x_0 = a, b,\ 0 \le x_i < k,\ 1 \le i < k\right\}$$
$$\left(A = (0, 1, \ldots, n-1),\ a = \frac{k-1}{2},\ b = k-1\right)$$

∎

ここで，スイッチ $(x_{0_S}, x_{1_S}, \ldots, x_{(n-1)_S})$ から $(x_{0_D}, x_{1_D}, \ldots, x_{(n-1)_D})$ へのパスを以下の VLAN に割り当てることにより，次元順ルーティングに従う最短パス集合を構築することができる (関数 *select_ab* は A.2.2節で示した).

$$V(-: select\_ab(x_{0_S}, x_{0_D}, k), x_{1_S}, x_{2_S}, \ldots, x_{(n-1)_S} \mid A)$$

k が奇数の場合，トーラスにおける PDOR VLAN 集合は複雑となり，かつ必要な VLAN 数は wrap-around リンクの存在により $k+1$ の場合とほとんど変わらないため，以下では k を偶数と仮定した場合の PDOR VLAN 集合についてのみ記述する.

定義 A.17 (n 次元トーラス上の PDOR VLAN 集合) k^n n 次元トーラス上の PDOR VLAN 集合は，以下の $k^{n-1}+2$ 個の VLAN で構成される.

$$\begin{aligned} & \left\{ V(-: x_0, x_1, x_2, \ldots, x_{n-1} \mid A) \;\middle|\; x_0 = a, b,\ \sum x_i \equiv 0 \bmod 2,\ 0 \le x_i < k,\ 1 \le i < k \right\} \\ \cup\ & \left\{ V(x_0, x_1, \ldots, x_{n-2}, -: x_{n-1} \mid B) \;\middle|\; x_0 = a, b \right\} \\ & \left(A = (0, 1, \ldots, n-1),\ B = (n-1, n-2, \ldots, 0),\ a = \frac{k-1}{2},\ b = k-1 \right) \end{aligned}$$

∎

$(x_{0_S}, x_{1_S}, \ldots, x_{(n-1)_S})$ から $(x_{0_D}, x_{1_D}, \ldots, x_{(n-1)_D})$ へのパスは，以下の手続きに従って使用する VLAN を選択することで最短パスとなる.

$ab := select\_ab(x_{0_S}, x_{0_D}, k)$;
if $\sum_{i \neq 0} x_{i_S}$ **mod** $2 = 0$ **then**
  use $V(-: ab, x_{1_S}, x_{2_S}, \ldots, x_{(n-1)_S} \mid A)$;
else begin
  $selected := false$;
  for $i := 1$ **to** $n-1$ **do**
    if $x_{i_S} = x_{i_D}$ **then continue**;
    else if $d^+(x_{i_S}, x_{i_D}) \le d^-(x_{i_S}, x_{i_D})$ **then begin**
      use $V(-: ab, x_{1_S}, x_{2_S}, \ldots, x_{(i-1)_S}, (x_{i_S}+1) \bmod k, x_{(i+1)_S}, \ldots, x_{(n-1)_S} \mid A)$;
      $selected := true$;
      break;
    end else $\{d^+(x_{i_S}, x_{i_D}) > d^-(x_{i_S}, x_{i_D})\}$ **begin**
      use $V(-: ab, x_{1_S}, x_{2_S}, \ldots, x_{(i-1)_S}, (x_{i_S}-1+k) \bmod k, x_{(i+1)_S}, \ldots, x_{(n-1)_S} \mid A)$;
      $selected := true$;
      break;
    end;
  end;
  if $selected \neq true$ **then** use $V(ab, x_1, x_2, \ldots, x_{n-2}, -: x_{n-1} \mid B)$;
end;

ここで，VLAN $V(ab, x_1, x_2, \ldots, x_{n-2}, -: x_{n-1} \mid B)$ が選択されるのは，以下の条件を満たす場合である.

$$\sum x_{i_S} \equiv 1 \bmod 2,\ x_{i_S} = x_{i_D}\ (1 \le i < n)$$