システムエリアネットワークにおける ルーティングに関する研究

平成14年度

鯉渕　道紘

論文要旨

PC クラスタの PC 間やストレージシステムなどの高速 I/O システム間を接続するシステムエリアネットワーク (SAN, サーバエリアネットワーク) は，最近の高性能並列分散システムの性能向上の鍵となっている．SAN は，専用の高速スイッチ群と大容量の point-to-point リンクを用いて構成されるが，ローカルエリアネットワーク (LAN) と異なり，高速なダイレクトメモリ通信を行うためにバーチャルカットスルー方式 (VCT 方式) もしくはワームホール方式 (WH 方式) によりパケットを転送する．そのため，SAN では効率的なデッドロックフリールーティングが必要となる．しかし，SAN は並列計算機の結合網と異なり，不規則なトポロジをとることが多いため，経路保証とデッドロックフリーの両立が難しい．そのため，SAN における既存のルーティングの多くはトポロジ上へスパニングツリーのマッピングを行ない，ツリー構造が持つ連結性および非循環性の特性を利用している．しかし，この方法では単純にツリー構造を利用したことにより，(1) 非最短経路が発生する，(2) トラフィックに偏りが生じやすい，という問題を持つ．

本論文では，これらの問題を解決し，ルーティング技術における様々な側面から，SAN のバンド幅を向上させる技術の開発を目的とする．

まず，不規則なトポロジの SAN において (デッドロック除去のための) パケット転送制限を分散させる left-up first turn (L-turn) ルーティングと right-down last turn (R-turn) ルーティングを提案する．L-turn ルーティングと R-turn ルーティングは動的な経路選択ができる適応型アルゴリズムであり，物理チャネルを時分割で共有する仮想チャネル，バッファの追加なしにあらゆるトポロジの SAN に適用することができる．L-turn ルーティングと R-turn ルーティングは，SAN を従来の 1 次元ではなく，2 次元座標 (垂直方向と水平方向) を持つ有向グラフにマッピングする．そして，L-turn ルーティングと R-turn ルーティングは 4 つの論理方向を使ってパケットの転送制限を分散させる．確率モデルシミュレーションの結果，仮想チャネルを持たない SAN において L-turn ルーティングは最大 80%のスループット向上が確認された．

次に，適応型ルーティングにおける出力選択機構 (output selection function: OSF) について検討を行う．動的に複数経路の中から使用する経路を選択することができる適応型ルーティングはルーティング機構と選択機構により構成される．ルーティング機構では適応型アルゴリズムが出力チャネルの候補の集合を定める．一方，選択機構では OSF がその中から実際にパケットの出力チャネルを決定する．そのため，適応型ルーティングのスループットを向上させるためには適応型アルゴリズムと同様に OSF についても基礎的な技術を確立させる必要がある．そこで，仮想チャネル間，物理チャネル間のトラフィックの分散を考慮する load-dependent 選択機構 (LDSF), LRU 選択機構および minimal multiplexed and least recently used (MMLRU) 選択機構を提案する．この 3 つの OSF では各スイッチが各自の物理チャネルと仮想チャネルの利用状況に基づいてトラフィックの混雑状況を判断する．確率モデルシミュレーションの結果，提案した 3 つの OSF は安定した性能を示し，有効であることが確認さ

れた．

最後に，不規則なトポロジのSANにおいて仮想チャネルを用いてデッドロックの除去とスループット向上の両方を達成するdescending layers (DL) ルーティングを提案する．DLルーティングは静的に経路を定める固定ルーティングであり，仮想チャネルを用いてネットワークを同一トポロジのサブネットワークの層に分割するアイデアを基にしている．DLルーティングはサブネットワーク間の切り換えにより非最短経路の割合を減らし，かつ，経路を分散させることができる点が特徴である．確率モデルシミュレーションの結果，DLルーティングは仮想チャネルを用いたSANにおいてパケットの平均ホップ数を減らし，スループットを最大266%向上させることが確認された．

Abstract

System area network (SAN or server area network), which connects personal computers (PCs) of PC clusters, high-performance storage systems and I/O systems, is one of the crucial components of modern high-performance parallel and distributed systems. SAN is a switch-based network using point-to-point links, and unlike local area network (LAN), virtual cut-through (VCT) or wormhole routing (WH) is used for low latency direct-communication in switching technique. In such networks, a high-performance deadlock-free routing is required. However, unlike interconnection networks used in parallel machines, SAN accepts irregular topologies, and it introduces difficulty on guarantee of connectivity and deadlock-free packet transfer. In traditional systems, spanning tree based routings which use the connectivity and acyclicity of spanning tree are used as practical solutions. However, they have common problems: (1) they must accept non-minimal routes, and (2) they tend to generate unbalanced traffic.

This thesis addresses these problems from various aspects of routing techniques, and aims to develop efficient routing techniques for SAN.

First, "left-up first turn (L-turn) routings" and "right-down last turn (R-turn) routings" are proposed for avoiding traffic unbalance. The L-turn routings and the R-turn routings are adaptive algorithms, which can select a route of packet dynamically, and they use a special directed graph. This graph introduces two dimensions and four directions instead of traditional one dimension and two directions. The L-turn routings and the R-turn routings try to set routing restrictions considering traffic balance on this graph. The L-turn routings and the R-turn routings have the advantage of requiring no additional virtual channels, which use a physical channel by time-sharing, and they can be applied to any topologies. Results of simulations show that the L-turn routings achieve up to 80% improvement on throughput under using no virtual channels.

Secondly, I focus on output selection functions (OSFs). An adaptive routing, which can select a route of packet dynamically, consists of routing function and selection function. In routing function, an adaptive algorithm provides a set of suitable (deadlock-free) outgoing channels. Then, in selection function, one of outgoing provided channels is selected by the OSF. Thus, the fundamental technique of the OSF as well as that of the adaptive algorithm is needed to improve the performance of adaptive routing. Here, I propose "load-dependent selection function (LDSF)", "LRU selection function", and "minimal multiplexed and least recently used (MMLRU) selection function", which dynamically consider traffic balance. In the three proposed ones, each switch locally grasps the congestion information by the utilization ratio of its own physical and virtual channels. Results of simulations show

that the three proposed OSFs are advantageous and achieve stable performance.

Finally, "descending layers (DL) routing" is proposed for using virtual channels to guarantee deadlock-free and improve the throughput. The DL routing is a deterministic routing, which statically determines a route, and it is based on dividing the network into the layers of similar sub-networks. Through switching sub-networks, the DL routing reduces the path hops, and can consider traffic balance. Results of simulations show that the DL routing decreases the routing hops and achieves up to 266% improvement on throughput under using virtual channels.

目 次

図 目 次

表目次

第1章　緒 論

1.1　システムエリアネットワーク

近年，パーソナルコンピュータ (PC) およびワークステーション (WS) の性能向上と低価格化が著しいものとなっている．そのため，数十から数千台の PC/WS を高速なシステムエリアネットワーク (SAN，サーバエリアネットワーク) で接続したクラスタシステムを用いて高性能コンピューティングを行う研究が盛んに行われている (Myrinet[N.J95], ServerNet[Hor96], QsNET[PFH01], RHiNET [TSJ+99][西 宏 00][STH+00][NKN+01], DIMMnet[NJH+00a][NJH+00b])．PC クラスタは同規模の並列計算機よりもコストの点で有利であり [Sea99][Sea01][HP02][T. 95]，将来に渡って高性能計算の主力マシンの一角を占めると予想される．

PC クラスタでは，PC 間を結ぶ SAN が構成，性能に影響を与える．SAN は大規模計算で発生するダイレクトメモリ通信を高速に行うために，従来の大規模並列計算機で用いられてきた相互結合網 (T3D[Oed93], T3E[ST96], Cavallino[JF96], Spider[M.G97], Jump-1[YAA+01]) と同様に高バンド幅，低レイテンシであることが求められる．そのため，SAN はバス接続などの複数の形態，速度のネットワークを混在させて構成することは少なく，専用の高速スイッチ群と大容量の point-to-point リンクを用いて構成される．したがって，パケットは複数のスイッチを経由して目的地に到達することになり，ルーティングがスループット向上及び低レイテンシを実現する鍵となる．

SAN では各スイッチが Store-and-Forward 方式 (SF 方式) ではなく，バーチャルカットスルー方式 (VCT 方式)[KK79] もしくはワームホール方式 (WH 方式)[DS87] を用いてパケット転送を行う．そのため，SAN ではデッドロックに対する対処が重要となるが，通常，デッドロックフリールーティングを用いてこれを解決する．他の解決策としてデッドロックが発生した場合，パケットの廃棄，再送処理によりパケット転送を保証するデッドロックリカバリー方式 [KT95b] [KT95a][KTJ96][JZA94] も提案されてはいるが，ソフトウェアのオーバヘッドが大きくなるため，現実的ではない．

WH 方式はパケットレイテンシを著しく削減する一方で，物理チャネル[1]の利用率を下げてしまう問題がある．

これは WH 方式ではパケットが一度ブロックされると，複数のスイッチのバッファを占有しながら停止してしまうためである．この解決策として，仮想チャネル [Dal92] が提案されている．仮想チャネルを用いることで複数のパケットが物理チャネルを共有することができ，その結果ネットワーク資源を効率良く使用することができる．しかし，仮想チャネルはバッファ量の増加，ハンドシェーク線の増加による実装の複雑さから，必ずしもすべての SAN で用いられるわけではない．

[1]SAN においてリンクは 1 本の双方向物理チャネルにより構成されている．

デッドロックフリールーティングを設計する場合，仮想チャネルの有無のみならず，スイッチの動的なチャネル切り換え —物理，仮想の両方を含む— の可否も性能に影響を与える．動的なチャネル切り換えができる適応型ルーティングではある経路が混雑した場合，別の経路を使ってパケットを転送することができる．つまり，適応型ルーティングは物理チャネルの利用率を向上させることにより，SAN のスループットを高めることができる．適応型ルーティングは，出力チャネル —物理，仮想の両方を含む— の候補の集合を求めるルーティング機構である適応型アルゴリズムと，その中から実際にパケットの出力チャネルを決定する出力選択機構 (output selection function: OSF) により構成される．

一方，固定ルーティングは，出発地から目的地まで静的に定まった経路でパケットを送る方法である．固定ルーティングでは，各スイッチが混雑状況に応じて動的に経路や仮想チャネルを変更する機能を省くことができる．そのため，固定ルーティングはスイッチの高速動作により SAN のスループットを高めることができる．固定ルーティングはこの他に，MPI ライブラリ [Mea96] の一部で必要となるパケット配達の FIFO 性の保証ができる利点も持つ．

1.2　本研究で解決すべき問題と従来の諸研究

本論文では，SAN のバンド幅の使用率 (スループット) を向上させるために，次の3つのデッドロックフリールーティング技術を開発し，その効果を評価する．

(a) 不規則なトポロジの SAN における適応型アルゴリズム

(b) 適応型ルーティングにおける OSF

(c) 不規則なトポロジの SAN における仮想チャネルを用いた固定ルーティング

1.2.1　不規則なトポロジの SAN における適応型アルゴリズム

PC クラスタではシステムの拡張性および各部の故障時の可用性が重視される．そのため，SAN は多くの場合，不規則なネットワークトポロジをサポートする．また，現在，専用のクラスタではなく，高速なネットワークを用いて机上に配置された PC や WS を接続し，専用のクラスタシステムと同様の性能を実現することを可能にするシステムも提案されており [TSJ+99][西 宏 00][STH+00][NKN+01]，この場合は物理的な配置の制約から不規則なネットワークトポロジをサポートしなければならない．

しかし，現在，多くの不規則なトポロジの SAN では Up*/Down* ルーティング [Mae91][2] を使わざるを得ない．Up*/Down* ルーティングはトポロジ上へスパニングツリー[3]のマッピングを行ない，ツリー構造が持つ連結性および非循環性の特性を利用して経路保証とデッドロックフリーを実現する．しかし，Up*/Down* ルーティングは従来の並列計算機で用いられているメッシュなどの固定トポロジのルーティングに比べて，スループットにおいて不利な点が多い [SD99]．これは Up*/Down* ルーティングが単純にツリー構造を利用

[2] パケットが up 方向に必要ホップ数移動した後，down 方向に移動するため，名称に*という表現を用いている．

[3] グラフ (ネットワーク) 内のすべての頂点 (スイッチ) を含むツリー

することにより，(1) 非最短経路が発生し，かつ，(2) トラフィックに偏りが生じやすい，という問題を持つことに起因する．

現在，この問題の解決策として (1) 最短経路とデッドロックフリーを保証するために各ホストPCに大量のバッファを用意する方法 [JPMJ02][JMPJ02], および，(2) 仮想チャネルを導入し，最短経路の割合を増加させる方法 [SD00][FJ00][SLT02][JPJ+02] について検討されている．しかし，これらの議論はハードウェアを付加することを前提としているため，限定的な用途に限られる．

そこで，この問題点を改善するために，本論文ではL-turn ルーティングと R-turn ルーティングを提案する [MAAH01a][AMAH02]．L-turn ルーティングと R-turn ルーティングは既存の1次元 (垂直方向) の有向グラフを拡張して2次元 (垂直方向と水平方向) の有向グラフである H/V グラフを用いる．H/V グラフでは各物理チャネルに割当てる論理方向が2つから4つに増加したことにより，スイッチにおけるパケットの入力方向と出力方向の組み合わせが従来の 2 つから 6 倍の 12 個に細分化される (入出力方向が一直線上になる場合を除く)．L-turn ルーティングと R-turn ルーティングはこの細分化された方向を管理することによりパケットの転送制限[4]を各スイッチに分散させることができる．L-turn ルーティングと R-turn ルーティングは (1) 仮想チャネルやホストPCのバッファなどのハードウェアの追加なしに実装でき，かつ，(2) あらゆるトポロジの SAN に適用できる，という2点から現実的な方法である．

1.2.2 適応型ルーティングにおける OSF

適応型ルーティングはルーティング機構と選択機構により構成される．ルーティング機構では適応型アルゴリズムが出力チャネル —物理，仮想の両方を含む— の候補の集合を決定する．一方，選択機構では OSF がその中から実際にパケットの出力チャネルを決定する．そのため，適応型ルーティングのスループットを向上させるためには，適応型アルゴリズムと同様に OSF についても基礎的な技術を確立させる必要がある．

しかし，既存の OSF のほとんど [BP89][DA93][SB97][L.S00] はトラフィックの状況を反映しない．そのため，これらの OSF は混雑している方向へのパケット転送を避けることが難しい点が問題がある．また，トラフィックの状態を把握する機能を持たせてあるもの [JFPJ00] もあるが，仮想チャネル間においてトラフィックを分散することができる利点がある一方，物理チャネルの使用状況に偏りが生じる可能性がある．このことは仮想チャネルの目的が物理チャネルを効率的に使用することであることと矛盾する．その結果，この OSF は高スループットを実現できていない．また，特定の適応型アルゴリズムを念頭に置いた方法 [FJ00] は複雑な仮想チャネルの使い方を想定していないため，仮想チャネル間のトラフィックの分散をできない．既存のこれらの OSF は単純であるが，安定して高性能を得るためにはシンプルさを維持しつつ，もう一工夫必要と考えられる．

そこで，OSF を，出力する物理チャネルの選択とその物理チャネル内での出力仮想チャネルの選択という2つの選択ステップに分けることにより効率的にトラフィックを分散する load-dependent 選択機構 (LDSF)[鯉渕 99] [鯉渕 00] [鯉渕 01], LRU 選択機構 [鯉渕 01] および minimal multiplexed and least recently used (MMLRU) 選択機構 [MAAH01b] を

[4] スイッチにおいてデッドロックの発生を防ぐために課すチャネル切り換えの制限

提案する．LDSF，LRU 選択機構および MMLRU 選択機構では各物理チャネル毎にそのチャネルの利用状況を反映するカウンタをおく．そして，各スイッチは自スイッチ内の物理チャネルと仮想チャネルの利用状況からトラフィックの混雑を判断し，混雑を迂回するように出力チャネル—物理，仮想の両方を含む— を選択する．

1.2.3 不規則なトポロジの SAN における仮想チャネルを用いた固定ルーティング

今世紀に入り，InfiniBand[I.T01] や QsNET[PFH01] などの固定ルーティングを採用した SAN が登場した．これらの SAN では固定ルーティングの欠点である物理チャネル利用率の低下を防ぐために仮想チャネル[5]を採用している点が特徴である．

しかし，現状ではトポロジ，サイズに制限がない SAN に対して Up*/Down* ルーティングを基にした固定ルーティング [Mae91] [JL99][JA00] を用いることが多い．また，L-turn ルーティングと R-turn ルーティングも仮想チャネルを想定したものではないため，効果的に仮想チャネルを利用することが難しい．

そこで，仮想チャネルを用いてネットワークを同一トポロジのサブネットワークの層に分割する固定ルーティングである descending layers (DL) ルーティングを提案する [鯉渕 02b] [鯉渕 02c] [MAH02a]．DL ルーティングはサブネットワーク間の切り換えにより (1) 非最短経路の割合を減らし，かつ (2) 経路を分散させることができる，という利点を持つ．また，DL ルーティングは仮想チャネル数によらず，あらゆるトポロジ，仮想チャネル数の SAN に適用することができる．

1.3 本論文の構成

本論文の構成は次の通りである (図 1.1)．

第 2 章では PC クラスタおよび SAN について説明した後，SAN で用いられるデッドロックフリールーティングについて説明する．そして，第 3 章では不規則なトポロジの SAN における 2 次元有向グラフを用いた適応型アルゴリズムである L-turn ルーティングと R-turn ルーティングを提案し，様々なトポロジで評価を行う．また，第 4 章では，適応型ルーティングにおける OSF について検討を行い，LDSF, LRU 選択機構 および MMLRU 選択機構を提案し, 評価を行う．また，第 5 章では不規則なトポロジの SAN における仮想チャネルを用いた固定ルーティングである DL ルーティングを提案し，様々なトポロジで評価を行う．最後に第 6 章にて結論を述べる．本研究で取り組んだ 3 つの研究の要約と対応する章を表 1.1 に示す．

[5] InfiniBand の規格では仮想レーンと呼んでいる．

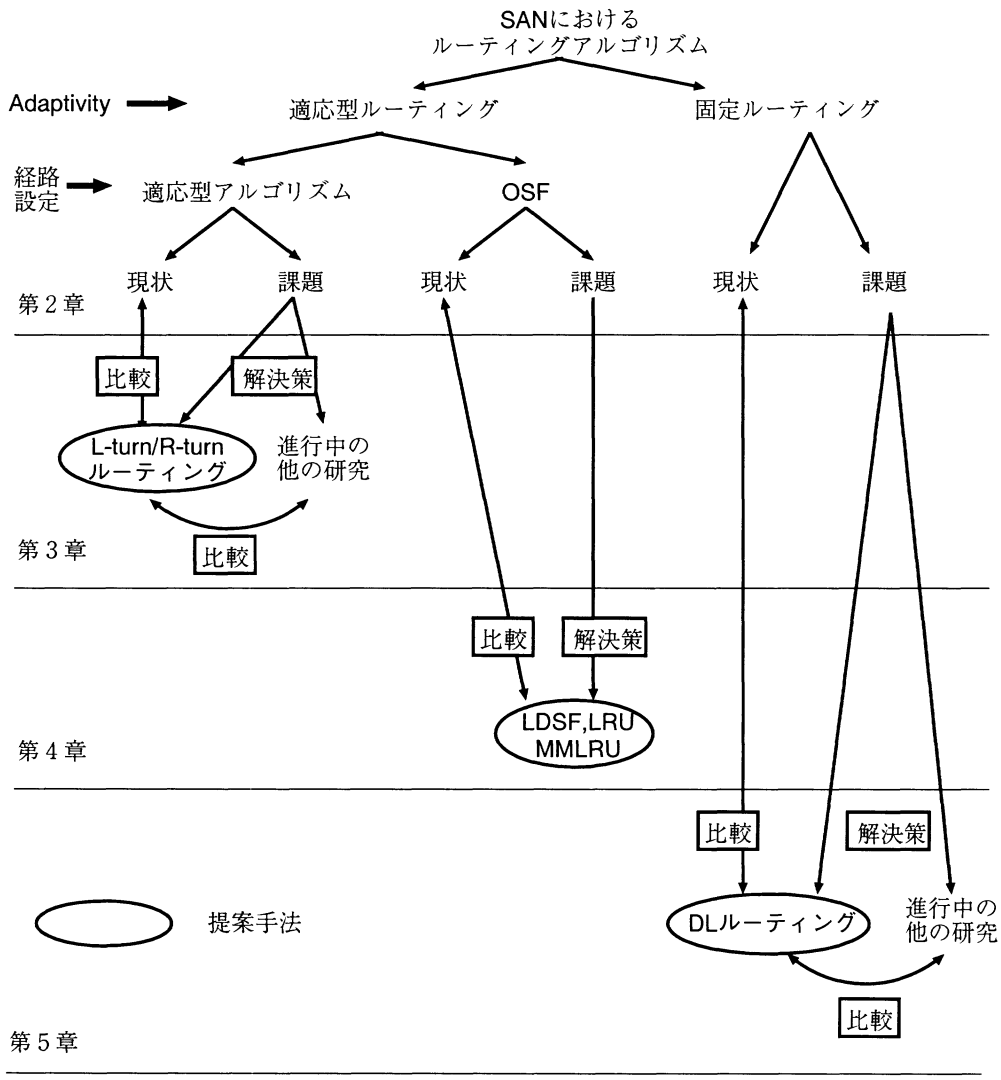


図 1.1: 本研究の位置付け

表 1.1: 本研究の要約

不規則な トポロジの SAN における 適応型 アルゴリズム	第3章	従来技術の 問題点	Up*/Down* ルーティングが用いられており， トラフィックが偏るため， バンド幅を生かせない．
		目的	あらゆるトポロジの SAN に適用でき， ハードウェアの付加なしに，トラフィックを 分散させる適応型アルゴリズムの提案
		提案技術	2次元有向グラフを用いてパケットの 転送禁止方向を分散させた L-turn ルーティングと R-turn ルーティング
		効果	スループットの向上
適応型 ルーティング における OSF	第4章	従来技術の 問題点	一部の適応型アルゴリズム，トポロジ に特化している，もしくは，トラフィックの 負荷に応じたチャネル (物理，仮想の両方を含む) 選択ができない．
		目的	適応型アルゴリズム，トポロジに依存せず， トラフィックの負荷に応じて出力チャネル (物理，仮想の両方を含む) を選択する OSF の提案
		提案技術	各物理チャネルにカウンタを 設置し，通過したトラフィック情報を 記録する．そして，カウンタの値によって 出力先を選択する OSF
		効果	高く安定したスループットの実現， および，低レイテンシの実現
不規則な トポロジの SAN における 仮想チャネル を用いた固定 ルーティング	第5章	従来技術の 問題点	Up*/Down* ルーティングを基にした手法が 用いられており，仮想チャネルを 生かしたパケット転送ができない．
		目的	あらゆるトポロジの SAN に適用でき， 仮想チャネルをスループット向上に用いる 固定ルーティングの提案
		提案技術	SAN を同一トポロジのサブネットワークの 層に分割する DL ルーティング
		効果	スループットの向上，低レイテンシ， および，パケットの平均ホップ数の削減

第2章　システムエリアネットワーク

本章では，まず，PC クラスタについて述べる．そして，PC クラスタ構築の鍵を握る SAN についてその他の相互結合網との比較を通して明確にする．次に，SAN におけるルーティングとその解決すべき課題について述べる．最後に SAN の実現例と歴史的展望を述べる．

2.1　PC クラスタの分類

PC クラスタは数十から数千台の PC を接続することで構築され，PC 間を結ぶ相互結合網の種類により次の 2 つに分類することができる．

- ベオウルフ型クラスタ
- 専用のネットワーク (SAN) を用いたクラスタ

ベオウルフ型クラスタについては様々な定義が行われているが，本論文では PC を TCP/IP を用いたネットワーク (ローカルエリアネットワーク: LAN) で接続した計算システムと定義する．ベオウルフ型クラスタは 90 年代中頃に登場し，汎用のパーツのみで構築されるため，並列計算機に代わる低コストな技術として注目を集めた．しかし，TCP/IP を用いる LAN は低レイテンシを保証することが難しい．そのため，ベオウルフ型クラスタは大規模なアプリケーション，リアルタイム画像処理および実時間トランザクション処理などを行うことが難しく (PC クラスタの用途については次節参照)，大規模な PC クラスタを構築することには向かない．

そのため，現在，LAN の代わりに TCP/IP を使わない専用のネットワークである SAN を用いた PC クラスタが主流になりつつある．そこで，本論文では，SAN を用いた PC クラスタに焦点を当てる．そして，本論文では特に指定しない限り，PC クラスタは SAN を用いた PC クラスタのことを指すことにする．

2.2　PC クラスタの目的

PC クラスタの用途は主に次の 2 つである．

- 大規模科学技術計算
- データベース，サーバー

2.2.1　大規模科学技術計算

航空宇宙学，気象予報などを含めた多くの科学技術分野ではTFLOPS (tera-FLOPS[1]) オーダーの計算処理が要求されるため，PC単体[2]では処理することができない．そこで，この計算処理を行うために，これまで大規模な並列計算機についての研究および商品化が行われてきた．大規模な並列計算機は数百から数千プロセッサ程度の規模を持つため，高速な通信機構が一つの鍵となる．そのため，並列計算機で用いられる相互結合網は各社独自に設計することで高速化を図ってきた (Cray T3D[Oed93]，Cray T3E[Oed93]，Intel Paragon[Int91], Stanford DASH[Dea92]，Stanford FLASH[Jea94], MIT Alewife[Aea90], MIT J-Machine[MDW93], および MIT Reliable Router[Wea94]．詳細は第2.8節および付録参照)．

しかし，大規模な並列計算機は強力な計算能力を提供するにも関わらず，次に挙げる4つの点でコストパフォーマンスが悪い．そのため，近年，大規模な並列計算機の研究，開発は減る傾向にある．

限定的な需要 大規模な並列計算機の市場 (大規模科学技術計算) が限られているため，PCのように大量生産されることはない．したがって，大規模な並列計算機は (1) ソフト (オペレーティングシステム, アプリケーション)，ハード共に開発コストの製品への高い転嫁率，および (2) 少量生産による製造コストの高騰，により割高になる．

高いメンテナンスコスト 大規模な並列計算機に特化したパーツはPCのパーツと異なり，各社独自に規格を設け，設計している．そのため，大規模な並列計算機をメンテナンスするための技術はPCのものに比べ熟練を要する．そのため，大規模な並列計算機のメンテナンススキルはPCに比べ割高である．

長い開発時間による相対的な性能低下 最新のテクノロジを用いた製品は，まず，大量生産されるPCの市場に投入される．一方で，汎用のパーツを拡張利用する大規模な並列計算機の開発は数年かかる．その結果，開発中の大規模な並列計算機が市場に登場する頃には，各パーツは最新のPCのパーツに比べ，相対的に低性能なものとなってしまう．

低い互換性 大規模な並列計算機は，各社独自にアーキテクチャを採用することにより，互換性に欠けるという問題がある．同じ会社であっても2つの世代の並列計算機間に互換性がないことすらある．さらに，Thinking Machines社のように大規模な並列計算機の製造会社は倒産することが多く，そうなると後続機が出ないため，他機へのアプリケーションの移植に多大な労力を費やすことになる．

一方，大量生産されているPCの性能は年々劇的に向上している．また，並行してコンポーネント (PCおよびクラスタ) 間の相互結合網であるSAN[N.J95] [Hor96] [PFH01]，高速通信ライブラリ (PM[住元00]，GAMMA[GG01] [G.C01] およびEMP[PPD01])，CPUの通信処理を軽減するためにアドレス変換機構やプロテクション機構等を搭載したネット

[1] floating-point operations per second の略で，プログラムの浮動小数点演算数を実行時間で割った値．FLOPSは計算システムの浮動小数点演算性能の目安となる測定値である．

[2] 2002年10月現在，PCは数GFLOPS程度である．

ワークインタフェースカード (NIC)[土屋 02]，および並列処理を支援するバリア同期などのマルチキャスト [田中 97] についての調査，研究も進んでいる．その結果，現在，数百から数千台の PC を用いた大規模なクラスタの構築が可能となった．そして，PC クラスタは (1) 開発リードタイムを必要としないため最新のチップ技術を利用することができる，(2) 汎用のパーツにより構成されるため，高い互換性を持つ，(3) Linux や MPI 通信ライブラリ [Mea96] などのメーカーに非依存なソウトウェア環境が整備されている，という利点から，現在，大規模科学技術計算システムの主流になりつつある．

2.2.2　データベース，サーバー

PC クラスタはメモリバスで結合する形式のマルチプロセッサと比較すると，速度が遅い I/O バスで結合されているため，通信コスト (時間) が高くつく傾向がある．一方で，PC クラスタは高い耐故障性，コストパフォーマンスおよび拡張性を持つ．この PC クラスタの特性は分散システムの分野であるトランザクション処理，サーチエンジンおよび電子メールサーバーなどのインタネットアプリケーションに適している．例えば，World Wide Web のサーチエンジンである Google は年々激増する Web ページインデックス[3]に対処するため，PC クラスタを使い，6,000 個のプロセッサと 12,000 個のディスクにより全体で 1 Pbyte (peta-byte) にのぼるディスクストレージを用いて運用されている．また，Google クラスタでは耐故障性を提供するために，RAID[4]を組んでクラスタ内で冗長性を持たせてはいるが，さらに，多くの冗長なサイトを置くことによりクラスタシステム自身の故障に備えている．このように，PC クラスタはデータベースやサーバー分野において現在使用されはじめている．

2.3　PC クラスタにおける相互結合網

2.3.1　SAN の登場

PC クラスタはローカルメモリを持つ安価な PC を数十から数千台接続することで構築される．そして PC 間の通信は相互結合網を介し，メッセージのやりとりにより交信を行う．そのため，相互結合網とその上でのメッセージの送受信は PC クラスタの構成，性能向上の鍵となる．

PC クラスタの相互結合網において重要な点は次の 2 つである．

(a) 大規模並列計算機の結合網と異なり，トポロジ，リンク長に対する制限が緩い．

(b) 大規模並列計算機の結合網と同様に高バンド幅，低レイテンシである．

(a) は，PC クラスタではマシンルーム内に集中配線する並列計算機と異なり，より広い範囲に分散している PC を用いることがあるため結合網の物理的配置にある程度の自由

[3]2000 年 12 月現在，約 13 億ページである．また，Google における検索数は月 20%の割合で増加している [HP02].

[4]redundant array of inexpensive (independent) disks の略で，ディスクを複数台並べることで信頼性を持たせたディスクシステム．RAID ではあるディスクが故障した場合に，他のディスクの差分，コピーにより，失われたデータを復元することができる．

度が必要となることに起因する．また，(b) は，PC クラスタが従来の並列計算機と同様に高速なダイレクトメモリ通信を必要とすることに起因する．

しかし，LAN では (b) の条件を満たすことが難しいため，2つの条件を満たす新たな相互結合網である SAN が登場した．SAN では (b) の観点から，複数の速度のネットワークを相互に接続して構築することはせず，同一の高速なスイッチ群と大容量リンクを用いて構築される [PFH01] [Hor96] [N.J95] [西 宏 00] [STH+00] [NKN+01]．また，PC-スイッチ間およびスイッチ間は point-to-point リンクにより接続する．通常，リンク長は数 km 程度までであり，バンド幅は数 Gbps 程度のものが多く用いられる．各リンクは，1つの双方向物理チャネルで構成され，場合によっては物理チャネルを時分割で共有する仮想的なチャネルが用いられる (仮想チャネルについては第 2.5 節で詳しく述べる)．

2.3.2　相互結合網の比較

前節で述べた通り，SAN と大規模並列計算機の結合網は，サイズとトポロジの2点で大きな違いがある．一方，SAN は特定の地域内において，ローカルメモリを持つ PC 間を接続する，という点でローカルエリアネットワーク (LAN) と似ているように見える．そこで，本節では改めて LAN との比較を行うことで SAN の特徴を明らかにする．

まず，代表的な SAN および LAN の比較を表 2.1[HP02] に示す．ただし，InfiniBand と Myrinet については第 2.7 節にて別個に詳しく述べる．

表 2.1: SAN および LAN の代表例の比較

	SAN		LAN	
	InfiniBand	Myrinet	10M/100M bit イーサネット	1000M bit イーサネット
長さ (m)	17/100	10/550/10000	500/2500, 200	100
クロック レイト (MHz)	2500	1000	10, 100	1000
スイッチ？	yes	yes	optional	yes
コネクション レス？	yes	yes	yes	yes
ノード数	$\leq \approx 1000$	$\leq \approx 1000$	≤ 254	≤ 254
ピークリンク バンド幅 (Mbps)	2000, 8000, or 24000	1300 to 2000	10, 100	1000
標準化	InfiniBand Trade Association	ANSI/VITA 26-1998	IEEE 802.3	IEEE 802.3 ab-1999

表 2.1 より，SAN と LAN は類似点が多いことがわかるが，次のように本質的な要求が異なるため，別々の標準化，開発が必要となる．

(a) SAN では LAN に比べてプロトコルオーバヘッドの低減が重要となる．

(b) LAN では SAN に比べて強力なプロテクションが必要となる.

(c) SAN ではトラフィック混雑時でも安定したメッセージ配信が必要となる.

(a) では例えば Gbit LAN において TCP/IP プロトコルを用いた場合, 0.8-1.0GHz の CPU が必要となることが知られている [HP02]. 一方, SAN の代表例である InfiniBand のプロトコルは非常に軽い処理ですむ. さらに InfiniBand ではホスト PC の通信コストを削減するために, ネットワークインタフェースコントローラがある程度の通信処理を行う. (b) は SAN ではサーバーもしくはクラスタ内でのデータ移動が主であるため, 強力なプロテクション機構が要らないことに起因する. (c) は, SAN では TCP/IP のようにトラフィック混雑時にパケットを廃棄して対処することが許されないことに起因する. つまり, ストレージアプリケーションを含めた PC クラスタのアプリケーションではパケットの廃棄により性能が著しく低下するため, パケットの廃棄を極力避けることが必要である.

このように, SAN は大規模並列計算機の相互結合網や LAN とは異なる特徴を持つ結合網であり, 一定の地位を築いている. 次に, この SAN の鍵を握る各技術—パケット転送方式, ルーティング— について述べる. PC 内のメモリ空間の管理, パケットの組み立て, 送受信を行う通信ライブラリやその肩代わりをする知的な NIC はルーティング, パケット転送方式と同様に性能に大きな影響を与えるが, 本論文の範囲を越えるため, 詳しくは扱わない. これらの通信ライブラリ, NIC の最近の動向は文献 [土屋 02] にまとめられている.

2.4　パケット転送方式

SAN においてメッセージはパケットの形で転送される. そして, 各スイッチは隣接するスイッチと通信を行うために物理チャネルもしくは仮想チャネル (第 2.5 節参照) を通じてパケットを送り, 同一スイッチの PC 間以外の通信は複数のスイッチを経由して行う. パケットの形式は, システムによって異なるが, 一般的には, 図 2.1 のように目的地 PC の番号, パケットの種類等を示すヘッダとデータ本体からなる. 多くのマシンでは, 物理チャネルは, 8bit から大きいもので 64bit 程度のデータ幅を持つが, 1 回の転送ではもちろんパケット全体を送り切れない. 物理チャネルに 1 クロックで挿入することができる単位をフリットと呼び, 1 つのパケットは, フリットを単位として転送される.

パケット長は固定の場合と可変を許す場合があるが, 可変長の場合でも最大長は決まっている. 可変長パケットは, ヘッダ内にパケット長を入れておき, 各スイッチ, PC でパケットの終わりを検出できるようにしておくのが普通である.

パケット転送方式は, 次の 3 方式に大別される.

- Store-and-Forward 方式 (SF 方式)
  各スイッチはパケット全体を格納することのできるチャネルバッファを持つ. 図 2.2(a) に示すように, 各スイッチはパケット全体をチャネルバッファに受けとってから, 順に次のスイッチに渡していく.

- ワームホール方式 (WH 方式)[DS87]
  各スイッチは基本的には, 1 フリット分を格納することのできるチャネルバッファを

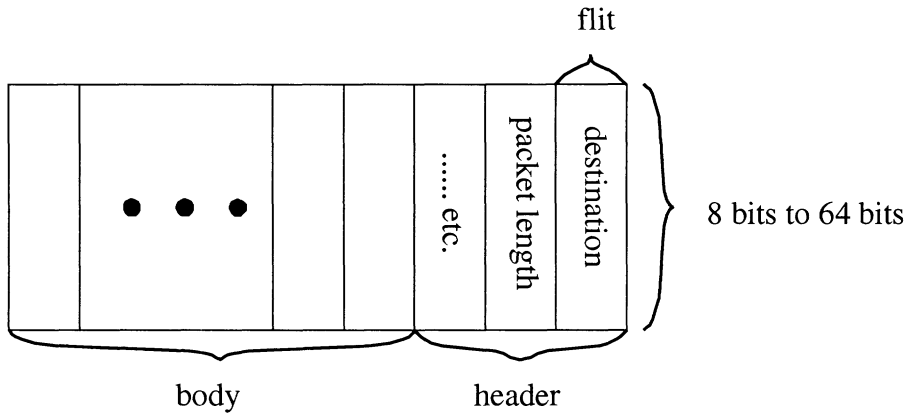


図 2.1: パケットの構成

持つ. 図 2.2(b) に示すように, パケットの先頭は, 送り先のフリットバッファが空いている限り, 次々と先のスイッチに進んでいく. パケットは複数のスイッチのチャネルバッファの列にまたがって格納され, 全体がいも虫のように前進する. 先頭が進もうとするバッファが, 他のパケットによって使われていた場合, パケットの進行はそこでストップし, チャネルバッファが空くのを待って前進を再開する.

- バーチャルカットスルー方式 (VCT 方式)[KK79]
  SF 方式同様, 各スイッチはパケット全体を格納することのできるチャネルバッファを持つ. しかし, ワームホール方式同様, パケットの先頭は, 本体の到着を待つことなしに次々と先のスイッチに進んでいく. パケットの先頭が, 他のパケットによってブロックされた場合, パケット本体の転送は停止することなしに, 先頭フリットのあるスイッチのチャネルバッファに格納される.

SAN ではパケットの転送を開始する場合, 専用のハンドシェーク線を使ってハンドシェークを取るが, 転送を開始した後, クロックに同期して 1 フリットごとに転送を行なっていく.

この際, WH 方式 の場合, 受信バッファのオーバフローを抑えるために, 先頭フリットがブロックされていないか, 専用のハードウェアで監視する必要がある. 一方 SF 方式の場合, パケットを受けとりつつ, 次のスイッチに送ることはせず, 受信バッファのオーバフローも起きないためソフトウェア処理が可能である.

SAN では, PC 間の高速なダイレクトメモリ通信をサポートするため, 通常 WH 方式もしくは VCT 方式が用いられる. これは WH 方式および VCT 方式のパケット転送時間が SF 方式に比べ小さいためである. これは次の式より明らかである.

ここで, 結合網においてスイッチ間の距離の最大値を示す直径を D, パケットヘッダのフリット数を F_h, 本体 (データ部) のフリット数を F_b とし, 1 フリットを 1 クロックで転送可能であるとする. SF 方式では, 全スイッチでひととおりパケットを格納する必要があるため, パケット転送に要する時間は

$$(F_h + F_b) * D$$

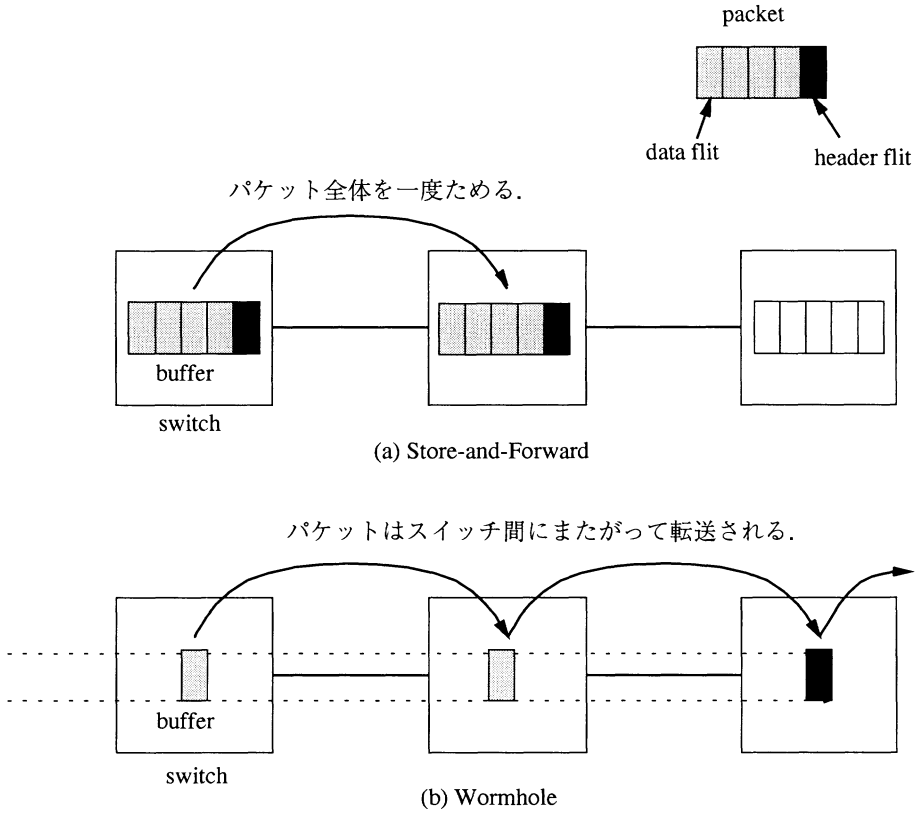


図 2.2: パケット転送方式

となる．これに対して WH 方式，VCT 方式では各スイッチはヘッダの格納のみが必要なので

$$F_h * D + F_b$$

となる．上記2式において通常ヘッダは1から2フリットですむため，F_h は小さく，このことは，直径 D が転送遅延にあまり影響を及ぼさないことを示している．

次に WH 方式と VCT 方式の比較を行う．VCT 方式は先頭フリットが他のパケットにブロックされた場合も，後続のフリットは停滞せずに進む．このため，VCT 方式はブロックされたパケットが他のパケットの進行を妨げることが少ない点で有利である．

一方，WH 方式は，スイッチの内部に次の出力経路を決定するために最低限必要となるパケットヘッダ分のバッファを用意すればよいため，バッファサイズを最小に抑えることができる．つまり WH 方式はスイッチのハードウェア量の点で有利である．

2.5　仮想チャネル

WH 方式の問題点は，パケットの先頭がブロックされるとそのパケットは複数のスイッチのバッファを占有しながら停止してしまう点にある．この場合問題なのは，図 2.3 のように停止したパケット A によりバッファが占有されるため，進行方向のバッファが空いているパケット B もブロックされてしまう点である．

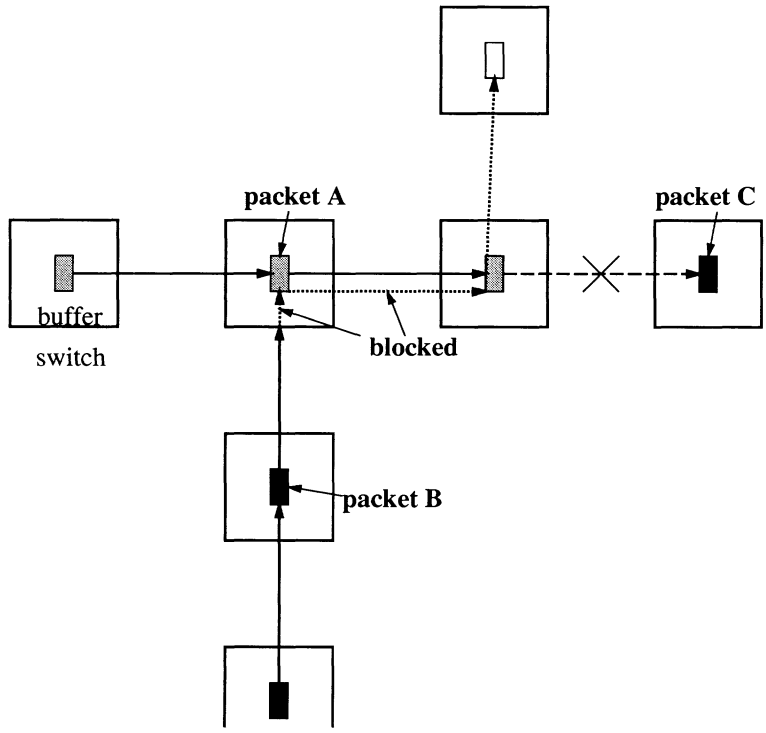


図 2.3: パケットのブロック

そこで，図2.4に示すようにスイッチ内に別のバッファを設け，そのバッファが空いているかどうかを判断するハンドシェーク線を独立に設ける．このようにすると，パケットBは空いている方のバッファを利用してブロックされることなしに先に進むことができるようになる．この方法は，ちょうど一車線しかない道路では，右折する車によって後続車がすべてブロックされてしまうのが，二車線にして右折レーンを設けることにより，ブロックがなくなるのに似ている．

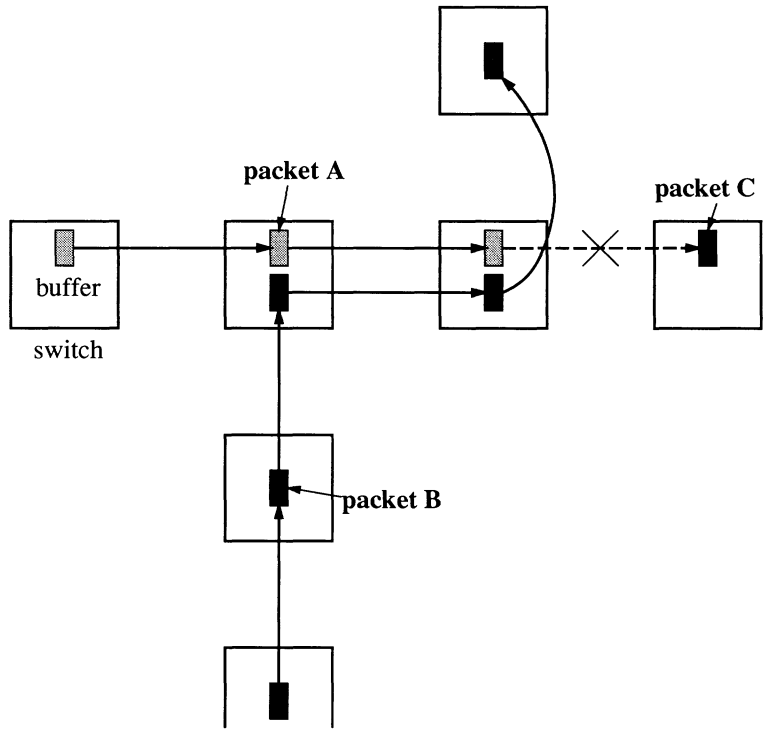


図 2.4: 仮想チャネルの利用によるブロックの回避

新たに設けられたバッファは，バッファの量を増やすだけでなく，独立のハンドシェーク線を用いて，独立にパケット転送を行なうことが必要である．この手法では，それぞれのバッファによってスイッチ間に仮想チャネルと呼ぶ仮想的な転送経路を作ることができる．仮想チャネルを利用したスイッチ間の転送制御を，仮想チャネルフロー制御と呼ぶ[Dal92]．仮想チャネルフロー制御を行なうことにより，複数の仮想チャネルで共有される物理チャネルの利用率は向上し，物理チャネル数を増やすことなしに，結合網の転送容量を飛躍的にあげることができる．図2.4では2本の仮想チャネルを使っているが，必要に応じて何本も設けることが可能である．

仮想チャネルは物理チャネル利用率の向上という長所と，仮想チャネルバッファとハンドシェーク線の増加という短所を抱えている．そのため，仮想チャネルは必ずしもすべてのSANで用いられるわけではない．

2.6　デッドロックフリールーティング

2.6.1　ネットワークモデル

ルーティングアルゴリズムはパケットを出発地 PC から目的地 PC へ送るときに使用する物理チャネル (仮想チャネルがある場合は，仮想チャネル) を決定する．同一スイッチの PC 間の通信はスイッチの構造がクロスバーであるため，ルーティングアルゴリズムに依存しない．よって，ルーティングアルゴリズムはスイッチ間のパケット転送経路を決定することが焦点となる．

したがって，ルーティングアルゴリズムを設計する場合，SAN はグラフ $G(N, C)$ で表すことができる [JSL02]．ただし，N はスイッチの集合，C はリンクの集合をそれぞれ表す．例えば，図 2.5 の PC クラスタは，図 2.6 に示したグラフで表すことができる．

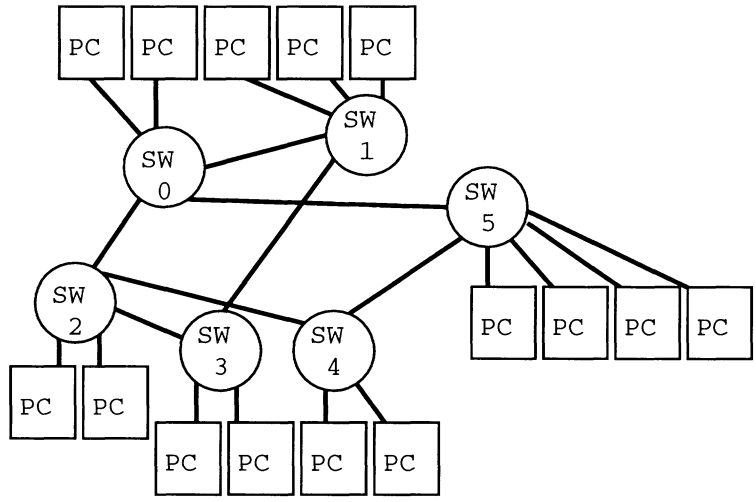


図 2.5: PC クラスタの例

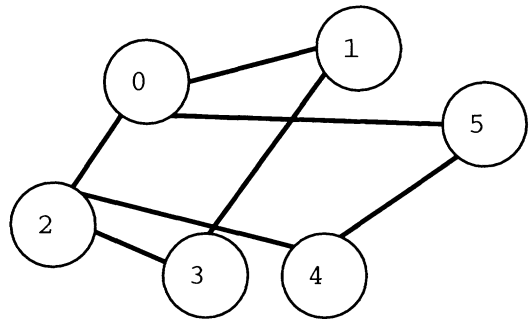


図 2.6: 図 2.5 に対応するグラフ G

SAN は同一の高速なスイッチ群と大容量リンクを用いて構成されるため，図 2.6 のように辺に重み付けをおこなわない単純なモデル化を行うことができる．

2.6.2　デッドロックの発生

SAN ではパケット転送方式として WH 方式，もしくは VCT 方式をとるため，ルーティングアルゴリズムはデッドロックに対する処理が重要となる [JSL02][天野 96].

デッドロックとは，ネットワークを通過中のパケットが，起こる可能性がない事象を待ち続けることにより，転送することが不可能となる状態のことをいう.

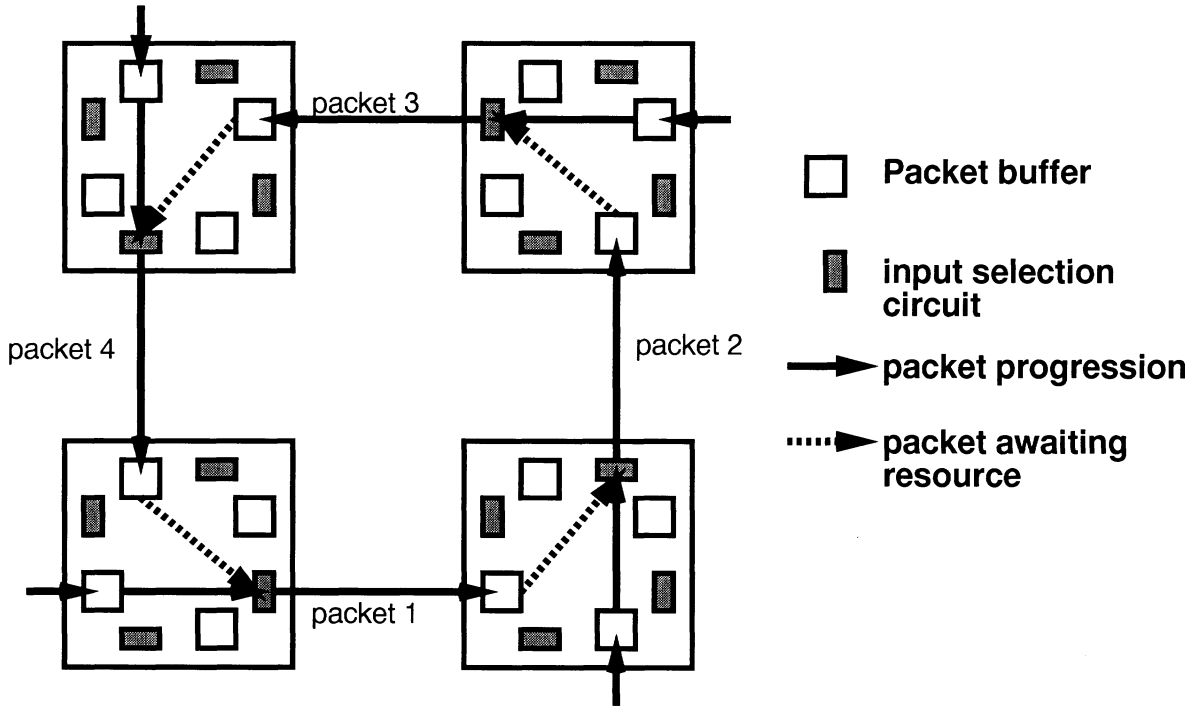


図 2.7: デッドロックの例

デッドロックが生じるのは，スイッチのチャネルバッファ間に循環依存があるためである．図2.7にデッドロックの例を示す．図2.7では，4つのパケットがそれぞれ行き先のパケットバッファが空くのを待っているが，互いにバッファを占有しあい，動きが取れなくなっている．デッドロックは WH 方式に限らず，バッファが有限で循環する限りは VCT 方式，SF 方式でも生じるが，バッファを占有してブロックする WH 方式では特に頻繁に生じる.

2.6.3　デッドロックリカバリー方式とデッドロックフリー方式

デッドロックの対応策として次の2つのルーティング方法がある.

- デッドロックリカバリー方式
- デッドロックフリー方式

デッドロックリカバリー方式はデッドロックが発生した場合，パケットの廃棄，再送処理等によりパケット転送を保証する方法である．しかし，一般的にデッドロックリカバリー方式は (1) デッドロックの検出，デッドロックからの回復機構などが複雑になりソフトウェアのオーバヘッドが大きくなる，かつ，(2) デッドロックが頻繁に発生すると性能の低下が大きくなる，という問題があるため，様々な研究が行われている [ST97][ST99][KT95b][KT95a][KTJ96] が，実装された例はほとんどない．

一方，デッドロックフリー方式はデッドロックが発生しないチャネルバッファの訪問順を設定し，その順にパケットを転送するルーティング法である．一般的に SAN ではパケットのレイテンシを削減するためにデッドロックフリー方式が用いられる．そのため，以後，本論文ではデッドロックフリールーティングについて扱う．

2.6.4　固定ルーティングと適応型ルーティング

デッドロックフリールーティングは固定ルーティングと適応型ルーティングの 2 種類に分類される．

固定ルーティングは，出発地の PC と目的地の PC が決まればパケットは必ず同じ経路を通る単純な方法である．固定ルーティングは次の 3 つの長所を持ち，一般的にスイッチの高速動作を重視する場合に用いられる．

- 動的な経路選択をする機能が不必要なため，スイッチ構造を単純にでき，高速動作が可能である．
- パケットが送り出した順番に必ず到着する性質 (FIFO 性) を持つ．
- 経路が固定されているため，パケット配送エラーの検出が容易である．

一方，適応型ルーティングは複数の経路の中から動的にパケットが使用する経路を選択する方法である．適応型ルーティングではある経路が混雑した場合，別の経路を使ってパケットを転送することにより，故障や混雑を回避することができる．ただし，ルーティングに関する情報がヘッダフリットに格納されているため，適応型ルーティングはフリット毎に異なる経路を選択することはできない．適応型ルーティングは，適応型アルゴリズムと出力選択機構 (output selection function: OSF) の 2 段階に分けられる．適応型アルゴリズムはデッドロックフリーな出力チャネルの候補の集合を求め，OSF によってその候補の中から実際のパケットの出力チャネルが決定される．OSF をルーティングポリシと名付けている場合もある [BP89][Wu96] [Wu99]．

適応型ルーティングは次の 2 つの長所を持ち，一般的に物理チャネルの利用率を重視する場合に用いられる．

- 使用可能な経路や物理チャネルを有効に用いることできる．
- 故障個所を迂回することにより耐故障性を得ることができる．

以後，既存の適応型ルーティングについて，適応型アルゴリズムと OSF を別々に説明し，その課題を指摘する．そして，固定ルーティングについても同様の議論を行う．

2.6.5　適応型アルゴリズム

SAN は規則的なトポロジで構築される場合もあるが，不規則なトポロジをとることが多い．また，規則的なトポロジに組んだ SAN における適応型アルゴリズムは従来の並列計算機で用いられたものと同様のものが使用でき，基礎的な技術が確立されている (規則的なトポロジにおける適応型アルゴリズムについて付録にまとめた)．

そこで，本節では不規則なトポロジの SAN における適応型アルゴリズムに焦点をあて，議論する．まず，不規則なトポロジの SAN における代表的な適応型アルゴリズムである Up*/Down* ルーティングと構造化チャネル法について述べる．そして，次に両者を比較し，解決すべき課題をまとめる．

2.6.5.1　Up*/Down* ルーティング

Up*/Down* ルーティングは不規則なトポロジの SAN における代表的な適応型アルゴリズムであり，Autonet[Mae91] や Myrinet[N.J95] (いずれも第 2.7 節参照) などのネットワークにおいて既に利用されている．

Up*/Down* ルーティングはデッドロックフリーと全スイッチ間の通信経路を保証するためにスパニングツリーのマッピングを基にする．スパニングツリーとはグラフ (ネットワーク) 内のすべての頂点 (スイッチ) を含むツリーのことである．Autonet で用いられたスパニングツリーの構築方法は breadth first search (BFS) に基づいている．BFS スパニングツリーの分散アルゴリズムとしては minimum depth スパニングツリー (MDST)[Mae91] および propagation order スパニングツリー (POST)[RS91] がある．これらは共にスパニングツリーの高さを最小とすることを念頭に置いている (POST では必ず最小となることが保証されないが，ほとんどの場合最小となることが Autonet 上では確認されている [RS91])．

ここでは POST の概念によるスパニングツリー構築アルゴリズムを簡単に示す．

1. 全スイッチの中から任意にスパニングツリーのルートを選択する．
2. ルートは，すべての隣接スイッチに join 要求メッセージを送信し，要求を受諾したスイッチをルートの子としてスパニングツリーに付け加える．
3. あるスイッチの子となったスイッチは，同様にしてすべての隣接スイッチに join 要求メッセージを送信し，要求を受諾したスイッチ (既にスパニングツリーに含まれているスイッチは要求を拒否する) を自身の子スイッチとしてスパニングツリーに付け加える．
4. 全スイッチがスパニングツリーに含まれるまで 3 の作業を繰り返す．

スパニングツリーの構築が完了した後，ネットワーク上のすべての物理チャネルに対して次に示す規則に基づいて up または down の方向を割当て，有向グラフを構築する．

1. up 方向を次の 2 つの条件のいずれかを満たす物理チャネルに対して割当てる．ただし，スイッチ数を n とすると，各スイッチには 0 から $(n-1)$ までの一意の整数の ID が割当てられているとする．

(a) 移動先のスイッチが移動元のスイッチよりもルートに近い.

(b) 移動先のスイッチと移動元のスイッチのルートからの深さが同一であり，移動先のスイッチ ID の方が移動元のスイッチ ID よりも小さい.

2. down 方向を残りのすべての物理チャネルに対して割当てる.

この作業により，図2.8のような有向グラフが構築される.

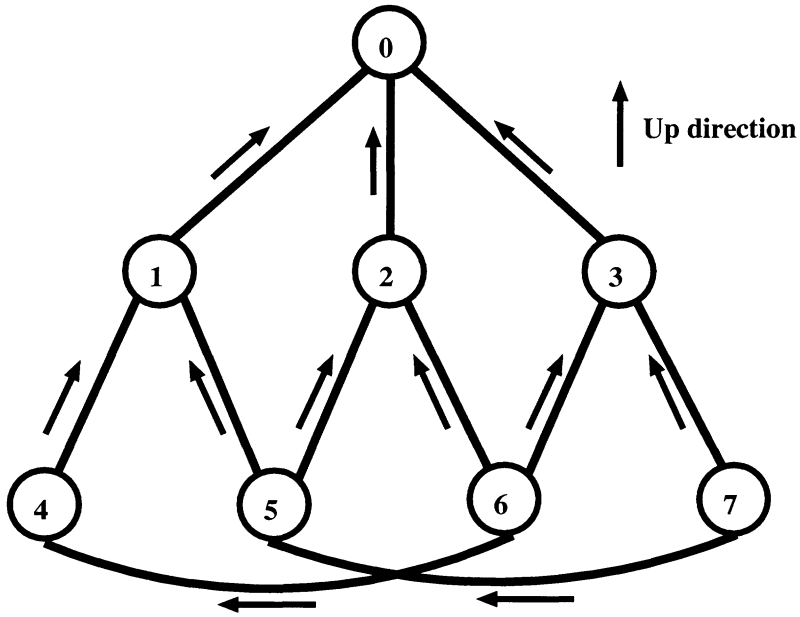


図 2.8: BFS スパニングツリーに基づいた有向グラフ

この方向割当てによりすべての循環は少なくとも1つ以上の up 方向と down 方向の物理チャネルを含む. Up*/Down* ルーティングはデッドロックフリーと任意のスイッチ間の経路を保証するために，次のようにルーティングを行う:

すべてのパケットは必ず0回以上 up 方向に移動した後に0回以上 down 方向に移動して目的地スイッチまで到達する.

この条件により，パケットは down 方向から up 方向への方向転換ができなくなり，すべての循環が除去される. Up*/Down* ルーティングは上記の条件を守る限りは経路を自由に選択できるが，常に最短経路を取るとは限らないので非最短型の適応型アルゴリズムである. 例えば，図2.8においてスイッチ5からスイッチ0へパケットを転送する場合には，スイッチ1またはスイッチ2を経由してスイッチ0まで到達することができるのですべての最短経路を選択することができる. これに対して，スイッチ3からスイッチ5へパケットを転送する場合には，down 方向から up 方向への移動が必要であるためスイッチ7を経由する最短経路は選択することができず，スイッチ0とスイッチ1またはスイッチ2を経由する非最短経路しか選択することができない.

2.6.5.2　構造化バッファ/チャネル法

構造化バッファ法はSF方式用に提案された方法で，あらゆるトポロジに適用できる最短型ルーティングである[MJ80]．この方法では結合網の直径よりも多いバッファ数がスイッチに必要となる．

ここで結合網の直径をDとし，バッファにクラス0からクラスDまでの番号が割当てられているとする．転送を開始したパケットはまず，番号0のバッファに格納される．そして，次に(格納されていたバッファのクラス+1)のバッファが空いていた場合にそのバッファに対して転送が行われる．この方法により，利用するバッファ番号はつねにあがる一方であり，全体で循環することはなくなる．

堀江らはこの考え方をWH方式に適用した構造化チャネル法[堀江92]を提案した．この方法はバッファを仮想チャネルにおきかえたもので，結合網の直径よりも多い仮想チャネル数を持つSANに対して適用することができる．ルーティング例を図2.9に示す．

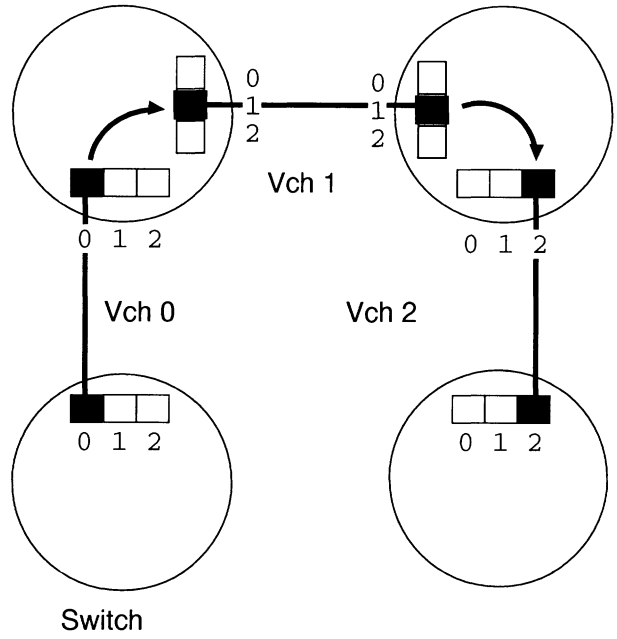


図 2.9: 構造化チャネル法

2.6.5.3　解決すべき適応型アルゴリズムの課題

本節にてSANにおけるルーティングアルゴリズムの評価基準，およびUp*/Down*ルーティングと構造化チャネル法の問題点を提示し，解決すべき課題について述べる．

SANは数十から千スイッチ程度の規模であるため，適応型アルゴリズムの計算量がスイッチ数Nに対し$O(N^3)$程度までであれば問題になることはほとんどない[JA00] [SLT02]．そのため，その範囲に収まる適応型アルゴリズム間の計算量の比較は，それほど重要ではない．なお，Up*/Down*ルーティングと構造化チャネル法はいずれもその範囲の計算量

に収まる．

SAN における適応型アルゴリズムの評価において最も重視される点は，各 PC が 1 クロックに受けとることができる平均フリット数の最大値を示すスループットを如何に向上させるか，という点である．また，さらにレイテンシが低ければなお，良い．

スループットに影響を与える要因としては，最短経路の割合とトラフィックの分散能力の 2 つが挙げられる．そこで，この観点を含めた Up*/Down* ルーティングと構造化チャネル法の比較を表 2.2 に示す．

表 2.2: Up*/Down* ルーティングと構造化チャネル法の比較

	Up*/Down* ルーティング	構造化チャネル法
トポロジフリー？	yes	yes
サイズ制限？	no	yes
最短型？	no	yes
仮想チャネルが必要?	no	yes
トラフィックの分散能力	low	high

表 2.2 より，Up*/Down* ルーティングは (1) SAN のトポロジ，ネットワークサイズに制限がなく，かつ，(2) 仮想チャネルが実装されていなくとも適用可能である，という長所を持つ．しかし，Up*/Down* ルーティングは (1) 非最短経路が発生する，(2) トラフィックに偏りが生じるため一部の物理チャネルがボトルネックとなる，ことが原因で，スループットが低い傾向にある．

Up*/Down* ルーティングにおけるトラフィックの偏りは，パケットの禁止ターン (down 方向から up 方向への方向転換) が同一スイッチ間に必ず 2 つ形成されるために発生する (図 2.10)．例えば図 2.10 においてスイッチ B に 2 個，スイッチ A に 6 個とすべての禁止ターンに偏りが生じている．

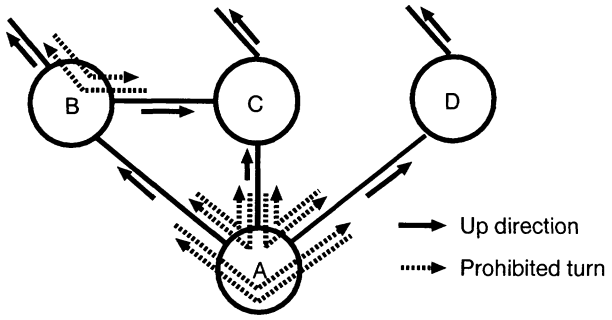


図 2.10: Up*/Down* ルーティングにおける禁止ターンの偏り

一方，構造化チャネル法は，結合網の直径よりも多い仮想チャネル数が必要となる．そのため，実質的に構造化チャネル法はネットワークサイズを制限することになる．したがって，構造化チャネル法は最短経路を保証することができるにも関わらず，限定的な使用に限られているのが現状である [西 宏 00] [STH+00] [NKN+01]．

このことから，現状では不規則なトポロジを取るSANでは低スループットであるUp*/Down*ルーティングを用いる場合が多いことが分かる．そこで，第3章にてUp*/Down*ルーティングと同様に高い汎用性を持つ適応型アルゴリズムであるL-turnルーティングとR-turnルーティングを提案する．なお，ここでの高い汎用性とは，(1) 仮想チャネルを必要とせず，(2) あらゆるトポロジに適用することができることを指す．

不規則なトポロジのSANにおいて仮想チャネルを用いずに最短型のデッドロックフリールーティングを開発することはほぼ不可能である．そこで，L-turnルーティングとR-turnルーティングは最短経路の保証をしない代わりに，スループット向上のもう一つの鍵である高いトラフィック分散能力を持つ点が特徴である．

2.6.6　OSF

前節では適応型アルゴリズムについて述べた．本節では適応型ルーティングのもう1つの構成要素であるOSFについて述べる．

大規模並列計算機の結合網やSANにおいてOSFの研究はこれまでほとんど行われていないため，幅広い検討が必要である．

そこで，包括的にOSFを扱うために，メッシュなどの次元を持つトポロジ[5]に特化した次元順選択機構 (dimension order selection function)，ジグザグ選択機構 (zigzag selection function) を含めた既存のOSFについてまとめ，解決すべき課題について指摘する．また，次元を持つトポロジについては，付録にまとめた．

2.6.6.1　ランダム選択機構

ランダム選択機構 (random selection function)[DA93] はあるスイッチから出力可能な物理チャネルおよび仮想チャネルが複数存在する場合，その中から出力チャネルをランダムに選択する方法である．この方法はOSFの中で最も単純であるが，ランダムに出力物理チャネルおよび仮想チャネルを選ぶ事によりトラフィックをある程度分散させる事ができる．

2.6.6.2　次元順選択機構

次元順選択機構はメッシュなどの次元を持つトポロジを対象としたOSFである．次元順選択機構は出力可能な物理チャネルが複数ある場合，その中で次元の一番低い[6]物理チャネルを選択する方法である．例えば，2次元メッシュにおいてx, y方向共に空いているチャネルがある場合，x方向を選択する (図2.11)．

[5]スイッチの位置を座標を用いて表すことができるトポロジ
[6]番号の一番小さい座標軸の方向

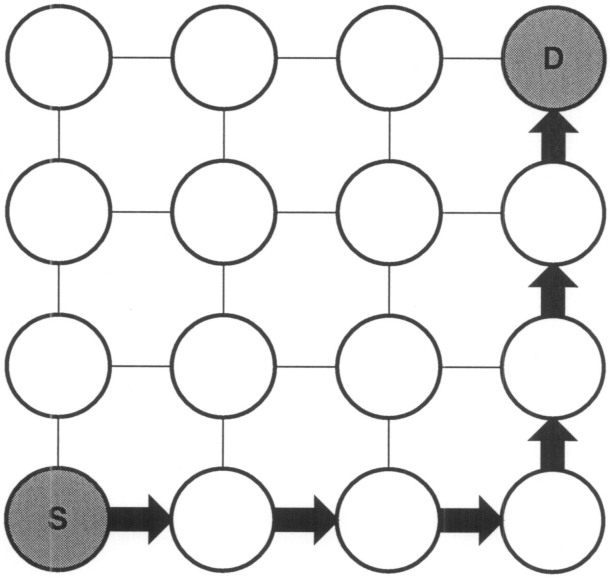


図 2.11: 2次元メッシュにおける次元順選択機構

2.6.6.3　ジグザグ選択機構

ジグザグ選択機構 [BP89] は次元順選択機構と同様にメッシュなどの次元を持つトポロジを対象とした OSF である．ジグザグ選択機構では，出力可能な物理チャネルが複数の方向にある場合，目的地までのホップ数が最大の次元の出力方向を選択する．例えば2次元メッシュにおいて，$s\,(x_s, y_s)$ から $d\,(x_d, y_d)$ にパケットを送る場合，x, y 方向共に出力可能な物理チャネルがあれば $|x_d - x_s|$ と $|y_d - y_s|$ の値のうち大きい方の次元方向を選択する (図 2.12)．つまり，ジグザグ選択機構はメッシュなどの結合網ではなるべく中心に向かって斜めにルーティングを行う OSF である．

2.6.6.4　LFU 選択機構

least frequently used(LFU) 選択機構 [JFPJ00] は出力可能な仮想チャネルが複数ある場合，ある一定期間の間で最も使用された頻度の低い仮想チャネルを選択する [JFPJ00].

2.6.6.5　SP 選択機構

static point (SP) 選択機構 [JFPJ00] は，優先度を交互に各物理チャネルの仮想チャネルに静的に割当て，その優先度を基に出力仮想チャネルを選択する．

優先度を交互に物理チャネルに割当てるため，各物理チャネルにおいて利用される仮想チャネル数をある程度抑えることができる．

Silla らが提案した minimal ルーティング (第 3.3 節参照) に対し SP 選択機構の優先度の割当てを行った例を図 2.13 に示す．

図 2.13 は 2 つの出力物理チャネルを持ったスイッチの図である．各方向には 3 本の仮

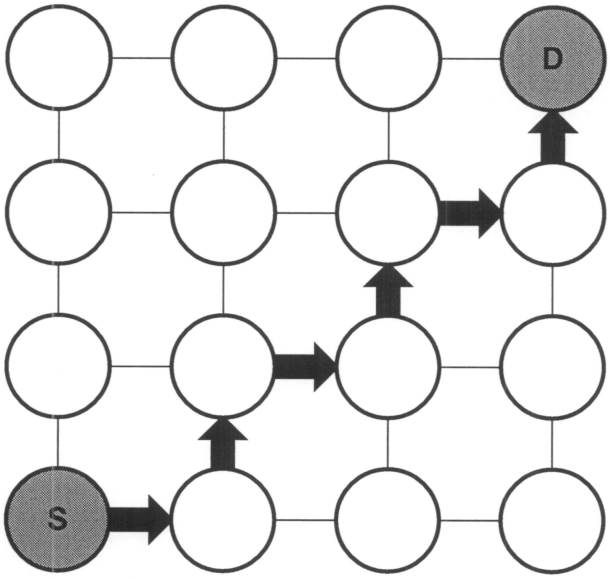


図 2.12: 2次元メッシュにおけるジグザグ選択機構

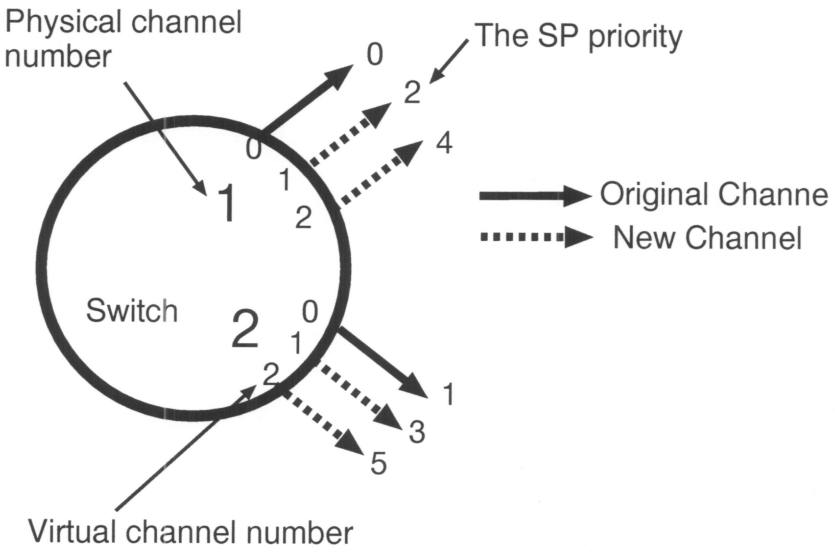


図 2.13: Minimal ルーティングにおける SP 選択機構

想チャネルがあり，番号0の仮想チャネルはoriginal channel, 番号1,2の仮想チャネルはnew channelである．各仮想チャネルの矢印の先にある数字が大きいほど優先度が高いことを示しており，original channelはnew channelが選択できない場合に限り，選択されるようになっている．

2.6.6.6　MM 選択機構

Sillaらは自身で提案したminimalルーティングを対象にして，時分割で共有しているパケット数が最も少ない出力可能な物理チャネルを選択するminimal multiplexation (MM)選択機構を提案した．また，MM選択機構は複数の出力可能な物理チャネル内のパケット数が同一の場合，SP選択機構と同様にして静的にわりふった優先度により出力可能な仮想チャネルを持つ物理チャネルを選択する[FJ00],[JFPJ00].

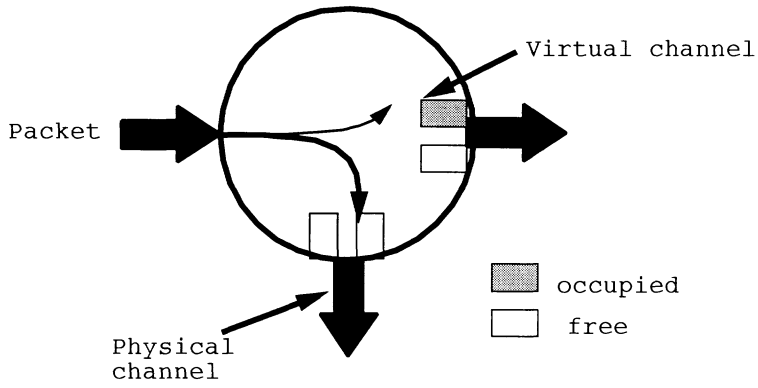


図2.14: 3本のリンクを持つスイッチにおけるMM選択

例えば図2.14のように2つの出力可能な物理チャネルがあり，各物理チャネルに仮想チャネルが2本ある場合，仮想チャネルの空いている本数が最も多い物理チャネル，すなわち，仮想チャネルが2つとも空いている物理チャネルを選択している．

文献[DA93]で提案されているminimum congestionポリシはMM選択に似ているが，これは最も空いている仮想チャネルが多い方向(物理チャネル)を選択するものであり，各方向(物理チャネル)の仮想チャネル数が同数の場合はMM選択機構と同一になる．

2.6.6.7　解決すべきOSFの課題

SANを含めた大規模計算システムの相互結合網においてスイッチは単純で高クロックに耐えられることが必要である．そのため，OSFは既存のもののように複雑な算術演算を行わず，単純にカウンタとその比較器を付加する程度で実装できることが望ましい．

また，OSFにおいて重要視される点は，スループットの高さである．さらにレイテンシが低ければなお，良い．OSFがスループットを高めるための手段としてはトラフィックの分散が挙げられるが，そのためには，各スイッチがトラフィックの混雑状況を判断する必要がある．そこで，スイッチがトラフィックの混雑状況を動的に判断して，トラフィッ

クが偏らないように出力物理チャネルと出力仮想チャネルの選択ができるか，という点に絞って以後検討する．

既存の OSF の中で，スイッチがトラフィックの混雑状況を把握することができるものは LFU 選択機構, MM 選択機構のみである．しかし，LFU 選択機構は仮想チャネル間のトラフィックの分散に主眼を置いており，物理チャネル間のトラフィックの分散を実現することが難しい．一方，MM 選択機構は逆に仮想チャネル間のトラフィックの分散を実現することが難しい点が問題である．

そこで，第4章にて，仮想チャネル間，物理チャネル間の両方においてトラフィックの分散を実現する OSF の提案を行う．また，これまで各 OSF のスループットの評価があまり行われていなかった．そこで，さらに第4章では，フリットレベルのシミュレーションにより各 OSF のスループットの比較を行うことで OSF が性能に与える影響について明らかにする．

2.6.7　固定ルーティングとその解決すべき課題

SAN では適応型ルーティングではなく，固定ルーティングが用いられる場合がある．そこで，本節では既存の固定ルーティングとして Up*/Down* ルーティングと構造化チャネル法を基にした手法について述べる．

第 2.6.5 節で述べた通り，Up*/Down* ルーティングと構造化チャネル法は本来，適応型アルゴリズムである．しかし，これらは同一 PC (スイッチ) 間の経路を 1 つに選択し，その経路のみを使用することにより固定ルーティングとして実装することができる．そのため，現状では不規則なトポロジの SAN において Up*/Down* ルーティング，もしくは構造化チャネル法が固定ルーティングとしても用いられている．特に，Up*/Down* ルーティングはネットワークサイズに制限がないため，頻繁に用いられる．

SAN における固定ルーティングの評価基準は適応型アルゴリズムと同様にスループットの高さである．しかし，固定ルーティングは適応型ルーティングに比べ，同速度で動作するスイッチを用いた場合，一般的に物理チャネルの利用率が劣ることがわかっている [DA93][JSL02]．そこで，最近の固定ルーティングを採用した SAN ではこの欠点を補うため，仮想チャネルを用いている [I.T01][PFH01][STH+00][NKN+01]．そのような SAN では仮想チャネルの使い方がスループット向上の鍵を握る．つまり，仮想チャネルを最短経路の割合の増加およびトラフィックの分散に活用することができるか，という点が重要である．

しかし，Up*/Down* ルーティングを用いた場合，仮想チャネルの使用を想定していないため，仮想チャネルを効果的に利用することが難しい．一方，構造化チャネル法は第 2.6.5 節で述べた通り，仮想チャネル数によりネットワークサイズが制限される問題を抱えている．さらに構造化チャネル法は仮想チャネルを単純にデッドロックの除去のみに用いているため，仮想チャネルを効果的にスループット向上に利用することが難しい．そのため，仮想チャネル数によらずあらゆるトポロジに適用することができる固定ルーティングの開発が急務となっている．

そこで，第5章にて仮想チャネル数によらず，あらゆるトポロジに適用できる固定ルーティングである DL ルーティングを提案する．DL ルーティングは仮想チャネルを最短経

路の割合の増加およびトラフィックの分散に用いることでスループット向上を達成する．

2.7　SAN の実現例

本節では SAN の基礎となった Autonet，および SAN の実現例である Myrinet, InfiniBand, RHiNET および QsNET について説明する．

2.7.1　Autonet

Autonet [Mae91] [RS91] はスイッチ，WS の再構成を行うことができる LAN であり，10 Mbps のイーサネットに代わる技術の確立を目的に提案された．Autonet はバス接続ではなく，スイッチ間を全2重の point-to-point リンクで結んだ形状を取る．スイッチは12ポートのフルクロスバーを持ち，cut-through 方式によりパケットを転送する．リンク長は同軸ケーブルでは 100 m, 光ファイバでは 2 km までサポートしている．2 km の光ファイバリンクの伝送遅延とブロードキャストパケットのデッドロックフリーを考慮して start-stop フロー制御において 4,096 byte の FIFO が必要となる．また，Autonet はトポロジの制限がなく，自動的なトポロジの状態の認識，再構成をすることができる．ルーティングは適応型アルゴリズムである Up*/Down* ルーティングを用いており，複数の経路が選択可能である．1990 年2月時点で米国 Digital's System Research Center において 100 ホスト，30 台のスイッチで動作している [Mae91].

2.7.2　Myrinet

Myrinet [N.J95] は CalTech Mosaic および，USC/ISI ATMIC LAN [RADG94] の研究成果により生まれた SAN 構築用のネットワークシステムで，ホストインタフェースとスイッチにより構成されている．Myrinet スイッチはパイプライン化されたクロスバーチップとリンクレベルのプロトコルを持つインタフェースチップの2つチップに分けて実装されている．

Myrinet は WH 方式を用いたソースルーティングであり，パケット長は任意で最初の3フリットはヘッダとなっている．ソースルーティングとは，出発地の PC で目的地の PC までの経路をパケット内に持たせる方式で固定ルーティングの実装方法の1つである．リンクは全2重であり，9bit 幅，80MHz による駆動となっている．9bit のリンクは，データとともに，流量制御にも用いられる．Myrinet スイッチは流量制御を行うために，STOP キャラクタおよび，GO キャラクタを往路に混入させることができる．受信部のパケットバッファは slack buffer により構成されている．slack buffer は waveform pipeline を行うスイッチにおいて有効な流量制御方式である．

Myrinet スイッチはパケットバッファにパケットが投入されている途中で，蓄えられたパケットの量が決められた STOP ラインを過ぎると送信元の隣接スイッチに STOP キャラクタを送る．STOP キャラクタが相手に届くと転送を即座に停止させるが，STOP キャラクタの送出と受理に必要な遅延を含めて，既にネットワークに投入されたパケットを吸収するに十分な余裕が STOP ラインに設けられている．そして，Myrinet スイッチはバッ

ファ内部のパケットが次の目的地に送られ，バッファに余裕が生まれるとGOキャラクタを送り，パケットを再び送り出すように要求する．ケーブルの長さの制限から，往路復路ともに32クロック分のパケットを一度に送るようにしているため，slack bufferの長さはGOとSTOPの処理にかかる16クロックを含んで全体で$(32 \times 2 + 16)$クロック分の幅を持っている．

Myrinetのもう一つの特徴はホストインタフェースにある，LANaiチップである．LANaiチップは，外部のローカルバスインタフェースとメモリ，およびリンクに結合されている．チップ内部には，Myrinetインタフェース，DMAコントローラ，プロセッサが実装されており，Myrinet Control Program (MCP) を記述することによって制御することができる．このMCPによりネットワークアドレスへのマッピングを行なわせることが可能である．

2.7.3 RHiNET

RWCP High Performance Network (RHiNET) [西 宏 00] [STH+00] [NKN+01] [TSJ+99]はRWCPと当研究室により開発されたSAN[7]である．RHiNETはマシンルーム内のPC間接続のみならず，オフィス，もしくはビルのフロア内のPC間接続に焦点を当てている．

2002年10月時点でRHiNETは構造化チャネル法やその応用である縮約構造化チャネル法を用いるために64本の仮想チャネルを持つ．しかし，64本の仮想チャネル分のバッファを用意することは困難であるため，仮想チャネルキャッシュという機能を導入してバッファ量を削減している．また，1kmのリンク長をサポートするため，フロー制御としてcredit based方式を採用している [NKN+01].

現在，RHiNETはネットワークインタフェースのコントローラであるMartiniとスイッチであるRHiNET-3/SWにより構成される．Martiniではユーザレベルゼロホストコピー通信[8]をサポートするためにユーザメモリ領域のプロテクション，アドレス変換機構などの機能をすべてハードウェアで高速に処理する．また，ハードウェアで実装されていない通信処理をコアプロセッサのソフトウェアで実現するといった高い柔軟性も併せ持つ．一方，RHiNET-3/SWは0.14 um CMOSエンベッデッドアレイで構成される1チップスイッチであり，10 Gbpsのリンクバンド幅を持つ．また，リンクレベルのエラー検出と修正，再送機構を搭載し，エラーレートの高い安価な媒体を用いた場合にもハードウェアのレベルで信頼性を確保し，通信のソフトウェアオーバヘッドを削減する [NKN+01].

2.7.4 InfiniBand

InfiniBand[I.T01] はPCクラスタおよび高速I/Oネットワークの標準化を目指しているSANであり，Compaq, Dell, Hewlett-Packard, IBM, Intel, MicrosoftおよびSun

[7]提案者はSANをマシンルームなどで，トポロジに制限を与え，短いリンク長で集中配線したネットワークであると狭義し，RHiNETがこのSANとLANの特徴を持つという点でLocal Area System Network (LASN) と呼んでいる．

[8]PC間の通信の際，通常，ホスト内ではシステムコールを介してカーネルが主記憶からネットワークインタフェースカード (NIC) へ複数回のコピーを通してデータを転送する．これに対し，システムコールなどの遅延を避けるためにユーザレベルでの通信を行う方法のことをユーザレベル通信と呼ぶ．また，ホスト内のデータコピーの回数を減らすために主記憶からNICへ直接データを転送する方法のことをゼロコピー通信と呼ぶ．ユーザレベルゼロホストコピー通信とはユーザレベルで実現するゼロコピー通信のことである．

Microsystems などの多くの会社が規格に参加している．InfiniBand は，point-to-point リンクを用いたスイッチベースのネットワークであり，スイッチトポロジの制限はない．

InfiniBand は最大 15 本の仮想レーンをデータトラフィックに使用することができる．仮想レーンとは，仮想チャネルと同様に複数の論理的なデータの流れを同一物理チャネルに実装する技術であり，他の論理的なデータの流れの影響なしに，各仮想レーンのリンクレベルのフロー制御をすることができる．仮想レーンは主に Quality of Service (QoS) とトラフィックの優先度処理を意図したものであるが，デッドロック除去のために使うこともできる．また，パケットのヘッダに付加されているサービスレベル (SL) により使用する仮想レーンを決定するため，各スイッチは，このマッピングテーブルを持つ．また，ルーティングは，各スイッチのもつルーティングテーブルを参照する分散方式の固定ルーティングである．ただし，InfiniBand はサブネット内の channel adapter (CA) 間において複数経路が選択可能である．しかし，各経路は目的地のポートに割当てられた local identifier (LID) により識別され，出発地の CA が LID の 1 つを選択するために経路が一意に定まる．よってこの点で InfiniBand は固定ルーティングに分類される．また，ルーティングテーブルは，パケットの入力仮想レーンを参照せずに，パケットの目的地のみをインデックスにして出力方向を決定する．しかし，既存の固定ルーティングのほとんどは入力チャネル—物理チャネルおよび仮想チャネル— と目的地をインデックスにしている．そのため，現状では，InfiniBand に Up*/Down* ルーティングなどの既存の固定ルーティングを実装するためには，(1) 最短経路の割合をある程度犠牲にする方式 [JAJ01]，もしくは，(2) destination renaming[9] を用いて元の固定ルーティングをそのまま実装する方式 [PJJ01]，のどちらかを用いる必要がある．

このように InfiniBand アーテキテクチャ (IBA) はやや特殊なルーティングの実装を定めている．InfiniBand は 2001 年 6 月に Specification Volumen 1, Release 1.0.a が発表され，RedSwitch[10] などの発表，製品化が進んでいる．

2.7.5　QsNET

Quadrics ネットワーク (QsNET) [PFH01] [FFA+02] は Compaq Alpha Server SC で用いられる SAN であり，各 PC の仮想アドレス空間の統合やリンクレベルの end-to-end プロトコルによるフォールトトレランスといった特徴を持つ．QsNET はネットワークインタフェースのコントローラ Elan とスイッチ Elite により構成される．Elan ネットワークインタフェースは DMA やホストプロセッサのユーザレベルでの要求の処理などを行うマイクロコードプロセッサ，高位の通信ライブラリをホストプロセッサの介在無しに処理する 32bit の RISC スレッドプロセッサ，MMU (メモリマネージメントユニット)，ルーティングテーブル，8kbyte キャッシュメモリおよび 64MB SDRAM で構成される．Elite スイッチは 2 本の仮想チャネルを持つ 8 本の双方向リンクを接続でき，16×8 のクロスバースイッチで構成される．QsNET は fat ツリートポロジ (詳しくは付録参照) で構成され，パケット転送方式として WH 方式が用いられている．

[9] スイッチの経路選択を柔軟に行うために同一の目的地に対して複数の識別子を与え，経路制御をおこなう方法

[10] http://www.redswitch.com

2.7.6　既存の SAN の比較

既存の SAN と Autonet について，ルーティングに関する項目を表2.3に示す．

表 2.3: 既存の SAN の比較

	Autonet	Myrinet	RHiNET	InfiniBand	QsNET
トポロジフリー？	yes	yes	yes	yes	no
仮想チャネル数	1	1	64	15	2
適応型アルゴリズム？	yes	no	no	no	no
本論文との関連	第3章，第4章	第3章	第5章	第5章	—

表2.3において，Myrinet はルーティングアルゴリズムがチャネルバッファの切り換えを指定することができない点で仮想チャネル数を1本，つまり仮想チャネルメカニズムを使えないとした．また，QsNET は文献 [PFH01] において，適応型ルーティングを採用していると述べられているが，QsNET はソースルーティングを用いているため中間スイッチで経路を変更することはできない．そこで本論文では QsNET を固定ルーティングと分類した．

表2.3より，QsNET を除くすべての SAN は不規則なトポロジをサポートしていることがわかる．また，QsNET で採用されている fat ツリーも形状に柔軟性 —ツリーのルート方向へのリンク数 p，ツリーのリーフ方向へのリンク数 q，及び階層数 r の組 (p,q,r) の設定— をもつトポロジである．

不規則なトポロジにおけるデッドロックフリールーティングの開発はメッシュなどの規則網の場合に比べて難しいが，これらの SAN が従来の並列計算機と同等の通信性能を提供するためには避けて通れない課題である．また，表2.3に各 SAN と関連する章についても示した．Myrinet では仮想チャネルを必要としない適応型アルゴリズムを一部の経路を削除することにより固定ルーティングとして実装することができる．つまり Myrinet には L-turn ルーティングおよび R-turn ルーティングを実装することが可能である．そのため，表2.3において第3章を Myrinet と関連づけた．これらの既存の SAN に対するルーティングアルゴリズムの移植は比較的容易である場合が多い．例えば，Myrinet では MCP を更新することにより実装することができる．

2.8　大規模計算システムにおける相互結合網の歴史的展望

大規模並列計算機，PC/WS クラスタで用いられてきた相互結合網の歴史，情勢，今後の展望について述べる．

1980 年代

大規模並列計算機 初期の相互結合網としてはハイパーキューブ (付録参照) が頻繁に用いられた [天野 96] [HP02]．当時の並列計算機は主として物理計算を目的としており，

様々な並列アルゴリズムが開発された．これらのマシンでは専用のスイッチではなく，ソフトウェアにより SF 方式でパケットを転送していた．そのため，結合網の直径の削減が研究課題となり，80 年代後半はトポロジの構成と，その上でのプログラムの動作に関して様々な議論が行われた．

PC/WS クラスタ 専用の並列計算機に代わる技術として，ワークステーション (WS) を用いたクラスタ計算機である WS ファーム/サイクルハーベスティングについての検討が行われた．WS ファーム/サイクルハーベスティングは LAN で接続された複数の WS をユーザが使用していない時に並列計算機として使う計算システムであり，遊休 WS をどう活用するか，という視点に基づいた技術である [D.A87] [MMM88]．しかし，これらは LAN を用いることが前提であるため，専用の相互結合網についての研究はほとんど行われていない．

1990 年代前半

大規模並列計算機 WH 方式が確立され，直径の大きさによる各トポロジ間のパケットレイテンシの差が隠蔽できるようになった．そのため，トラフィックの偏りがでにくい対称的なトポロジを用いることで並列計算機のトポロジについての議論は一段落ついた格好となった．そして，その後，実装が楽な結合網である k-ary n-cube (付録参照) を対象として，Turn モデルなどの様々な適応型アルゴリズムについての研究が盛んに行われた．

PC/WS クラスタ 1993 年頃より，PC と LAN を用いたベオウルフ型クラスタの構築が現実味を帯びてくる．これは，(1) Linux の発展による PC の高品質なソフトウェア環境の実現，(2) PC 用 10Mbit イーサネットの普及，(3) Intel 80386 プロセッサの登場による PC の性能向上，が主な要因である．そして，1993 年秋にベオウルフプロジェクトが，1 GFLOPS コンピュータを$50,000 以下で NASA に提供することを目的に始まり，1994 年に 80486 CPU を搭載した 16 台の PC のクラスタでこの目的を達成した [GJ01]．その後，PC クラスタの関心の高まりと共に LAN にかわる PC クラスタの相互結合網についても注目されるようになった．

1990 年代後半

大規模並列計算機 適応型アルゴリズムである Duato's protocol [Dua95] により k-ary n-cube などの規則的なトポロジにおいて一部の仮想チャネル間の循環を除去することなくデッドロックフリーを実現することに成功した．そして，Duato's protocol により規則的なトポロジにおける適応型アルゴリズムの議論は一段落した．

それ以降は，k-ary n-cube やメッシュなどのトポロジにパケット転送方式として WH 方式もしくは VCT 方式を用いた数千プロセッサ規模の並列計算機が多数実装され，Duato's protocol などがその上で用いられた [Wea94] [ST96]．

PC/WS クラスタ PC クラスタは，マシンルームに集中配線し，トポロジ，リンク長に制限を与えることで高スループット，低レイテンシを提供する初期の Myrinet の登

場により高性能化が進んでいった．その後，開発が進んだ Myrinet はトポロジ，リンク長の制限が緩くなり，それと共に SAN の分野が形成されていった．

2000 年以降

大規模並列計算機 PC の飛躍的な性能向上に伴い，並列計算機の市場に PC クラスタが台頭してきた．しかし，今後も並列計算機の需要がなくなることはないと考えられる．例えば，世界のトップ 500 のスーパーコンピュータ[11]の中で大規模並列計算機が上位を占めている．今後は，これまで並列計算機によって行われてきた科学技術計算のうち，PC クラスタで処理可能な分野については並列計算機が用いられる機会は減少するであろうが，コストパーマンスよりも計算能力を重要視せざるを得ない分野では並列計算機が使われ続けるであろう．また，それに伴い，並列計算機の開発も続いている．例えば，Cray は 2010 年までに高度で多様なアプリケーション処理における実効性能で PFLOPS (peta-FLOPS) を達成するスーパーコンピュータ (Cray X1) を出荷すると発表している[12]．

また，パケット転送方式と規則網における適応型アルゴリズムが確立されたこともあり，並列計算機の相互結合網は，今後，実装に関連した技術に重点がおかれると考えられる．

PC/WS クラスタ 並列計算機の市場に PC クラスタが台頭してきた．しかし，CPU のクロックが 1 GHz を越えるようになり，汎用の LAN を用いるベオウルフ型クラスタのネットワーク部の非力さが目立ちはじめる．そのため，SAN を用いた PC クラスタが目立つようになる．また，それに伴い，現在，SAN におけるルーティングに関する研究も盛んに行われている．この研究の特徴は並列計算機の相互結合網のルーティングと異なり単純な固定ルーティングが注目されている点である．これは (1) SAN が複雑なトポロジをとる場合が多い，(2) 現在，スイッチの高速動作が重要視される，ことに起因する．しかし，並列計算機の相互結合網におけるルーティングは，固定ルーティングに関する議論から適応型ルーティングに関する研究に発展していった経緯がある．そのため，SAN におけるルーティングも同様に，将来的には適応型ルーティングに落ちつく可能性がある．

今後，PC クラスタは従来の並列計算機の市場のみならず，サーバーの市場において発展，普及していくであろうが，PC クラスタの相互結合網の中長期的な予測は難しい．例えば，将来的には並列処理をサポートするための通信ライブラリを移植した 10 Gbps イーサネット[13]が用いられる可能性がある．そうなると，今後，InfiniBand は一部の特殊用途のクラスタを除いて用いられなくなり，PCI バスなどの I/O バスに代わる storage area network としての利用に留まる，ということもありうる．逆に，InfiniBand が PC クラスタの用途のみならず，LAN の領域に食い込む可能性もある．現在，SAN と LAN はアプローチこそ異なるが，徐々に重複する部分が増えてきている．そして，もはや，両者は TCP/IP を利用するかどうか，という点で区別する以外，明確な判定基準がない状況になりつつある．

[11] http://www.top500.org
[12] http://www.cray.com/news/0211/x1announce.html
[13] http://www.10gea.org

この勝者がどちらになるか，住み分けができるのかは，数年待たなければ分からない．しかし，PC クラスタでは今後とも低レイテンシなダイレクトメモリ通信が必要である．したがって，TCP/IP を用いる LAN の通信レイテンシが劇的に改善されなければ，SAN は PC クラスタで使われ続けるであろう．そうなると，今後もデッドロックフリールーティング技術が用いられることになり，本論文で提案するルーティング技術が将来に渡って PC クラスタを含めた大規模計算システムに活用することができるといえる．

第3章　不規則なトポロジの SAN における適応型アルゴリズム

PC クラスタではシステムの拡張性および各部の故障時の可用性が重視されるため，多くの SAN は不規則なトポロジをサポートする．しかし，不規則なトポロジの SAN では，並列計算機で一般的に用いられているメッシュやトーラスなどの固定トポロジに比べてルーティングアルゴリズムが経路保証とデッドロックフリーを両立させることが難しい．

そのため，多くの場合，ツリー構造の持つ非循環性と連結性を利用したスパニングツリーベースのルーティングアルゴリズム [Mae91] [JPJ+02] [SLT02] [JL99] が用いられる．この中で最も代表的なものは Up*/Down* ルーティング [Mae91] である．Up*/Down* ルーティングはスパニングツリーのマッピングを基に各物理チャネルに up 方向もしくは down 方向を割当てる．そして，Up*/Down* ルーティングは物理チャネル間の循環依存を除去することによりデッドロックフリーを保証する．Up*/Down* ルーティングの問題点の 1 つは，トラフィックがツリーのルート付近に偏りやすく，ネットワークのバンド幅を生かすことが難しい点である．

本章では，この問題点を解決するために，H/V グラフ [AMAH02] という 2 次元有向グラフを導入する．H/V グラフは Up*/Down* ルーティングで用いた 1 次元有向グラフを拡張することにより構成される．そして，次に H/V グラフを用いた適応型アルゴリズムである L-turn ルーティングと R-turn ルーティングを提案する．L-turn ルーティングと R-turn ルーティングはデッドロック除去のためのパケット転送制限をネットワーク全体に分散させている点が特徴であり，あらゆるトポロジの SAN に付加仮想チャネルなしで適用することができる．

以降，第 3.1 節で L-turn ルーティングと R-turn ルーティングを提案する．そして，第 3.2 節でシミュレーションによる評価結果を示し，第 3.3 節で同時並行的に進んでいる他の研究との比較検討を行う．最後に結論を第 3.4 節で述べる．

3.1　L-turn/R-turn ルーティング

Up*/Down* ルーティングは1次元の単純な方向割当てが原因となり禁止ターンの偏りが発生した.

そこで，本節ではまず，禁止ターンを分散するために Up*/Down* ルーティングで用いた有向グラフの次元数を増やす．この拡張されたグラフを H/V グラフと呼ぶが，H/V グラフは2次元であるため4つの方向と 12 個のパケットターン (同方向ターンを除く) を持つことになる．そのため，H/V グラフを用いることで禁止ターンの配置を分散させることが可能となる.

次に，H/V グラフを用いた L-turn ルーティングと R-turn ルーティングを提案する．L-turn ルーティングと R-turn ルーティングは異なる禁止ターンの集合を持ち，禁止ターンをできるかぎり分散するように設定している点が特徴である.

3.1.1　H/V グラフの構築

horizontal direction と vertical direction の2つの方向を持つ H/V グラフは，次の3つのステップにより構成される.

(1) BFS スパニングツリーを構築する.

(2) 各スイッチに2次元座標を割当てる.

(3) 各物理チャネルに2次元方向を割当てる.

3.1.1.1　BFS スパニングツリーの構築

まず，Up*/Down* ルーティングの場合と同様に BFS スパニングツリーを構築する．次に，BFS スパニングツリーを構築した後，*depth* を各スイッチに割当てる．*depth* はルートからの最短距離とし，そのツリーの階層構造を示す.

BFS スパニングツリーでは同階層のスイッチが複数存在するため，複数のスイッチが同一の値の *depth* を持つことができる．図 3.1 は9スイッチの不規則なトポロジの SAN に対して BFS スパニングツリーを構築し，*depth* を割当てた例である．図 3.1 においてスイッチ ID とは各スイッチを識別するために静的に割当てられている一意の整数のことであり *depth* や次節で定義する *horizontal spread* とは無関係である.

3.1.1.2　各スイッチに対する2次元座標の割当て

次に次元を追加するために各スイッチに *horizontal spread* を割当てる．*horizontal spread* はルートを起点として前順走査を行ない，各スイッチに対して走査を行なった順に図 3.2 (図 3.1 と同じネットワーク) のように0から昇順で整数を割当てる．そして，各スイッチに *horizontal spread* と *depth* を割当てた後に *horizontal spread* h, *depth* d を持つスイッチに対し，2次元座標 (h, d) を割当てる.

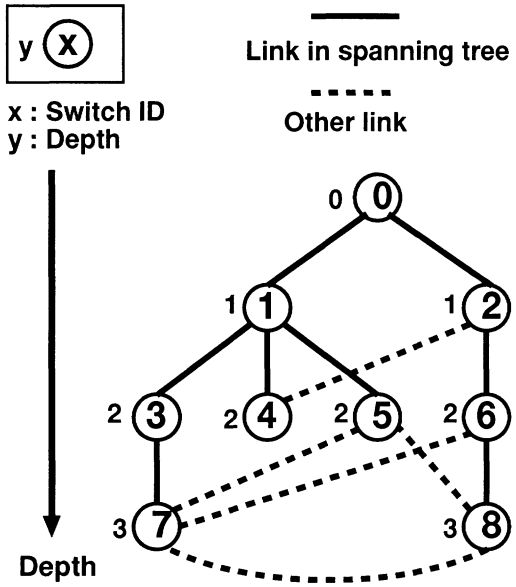


図 3.1: Depth の割当て

horizontal spread は *depth* と違い，一意であるため，*horizontal spread* と *depth* から成る 2 次元座標も同様に一意であることが保証される．

3.1.1.3　各物理チャネルに対する 2 次元方向の割当て

次にスイッチの 2 次元座標を基にして各物理チャネルに方向を割当てる．まず，各物理チャネルに対して次の条件に基づいて vertical direction を決める．ここで，物理チャネルの接続元および接続先のスイッチの座標をそれぞれ (S_h, S_d) および (D_h, D_d) とする:

- $(S_d > D_d) \vee ((S_d = D_d) \wedge (S_h < D_h))$ ならば up 方向を割当てる．
- $(S_d < D_d) \vee ((S_d = D_d) \wedge (S_h > D_h))$ ならば down 方向を割当てる．

次に，各物理チャネルに対して次の条件に基づいて horizontal direction を決める:

- $S_h > D_h$ ならば left 方向を割当てる．
- $S_h < D_h$ ならば right 方向を割当てる．

そして，horizontal direction h および vertical direction v を持つ物理チャネルの H/V direction $HV(h, v)$ を次のように定める:

- $HV(left, up)$ ならば left-up(LU) 方向を割当てる．
- $HV(left, down)$ ならば left-down(LD) 方向を割当てる．
- $HV(right, up)$ ならば right-up(RU) 方向を割当てる．

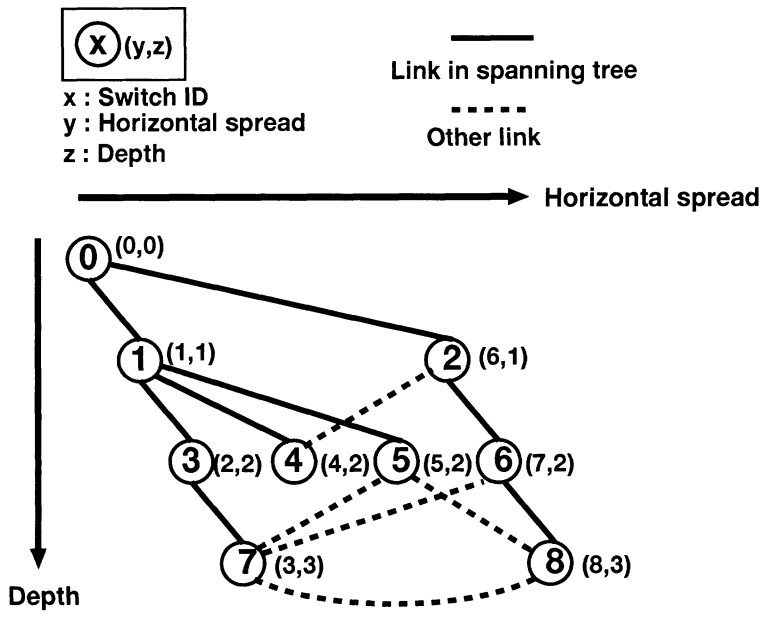


図 3.2: Horizontal spread の割当て

- $HV(right, down)$ ならば right-down(RD) 方向を割当てる.

これより，物理チャネルの接続元および接続先のスイッチの座標をそれぞれ (S_h, S_d) および (D_h, D_d) とした場合の物理チャネルの H/V direction は表 3.1 のようになる.

表 3.1: H/V direction

	$S_d > D_d$	$S_d = D_d$	$S_d < D_d$
$S_h > D_h$	Left-up(LU)	Left-down(LD)	Left-down(LD)
$S_h < D_h$	Right-up(RU)	Right-up(RU)	Right-down(RD)

この作業により，物理チャネルに H/V direction が割当てられ，H/V グラフが構築される．例として図 3.2 に示したグラフから生成された H/V グラフを図 3.3 に示す.

3.1.2　H/V グラフにおける循環除去

本節ではすべての循環を除去するための禁止ターンを次の手順で分散させる.

(1) すべてのパケットのターンを列挙する.

(2) 循環構造のパターンを列挙し，すべての循環を除去するように禁止ターンを設定する.

(3) 循環が生じないことを保持しながら，禁止ターン数を減らす.

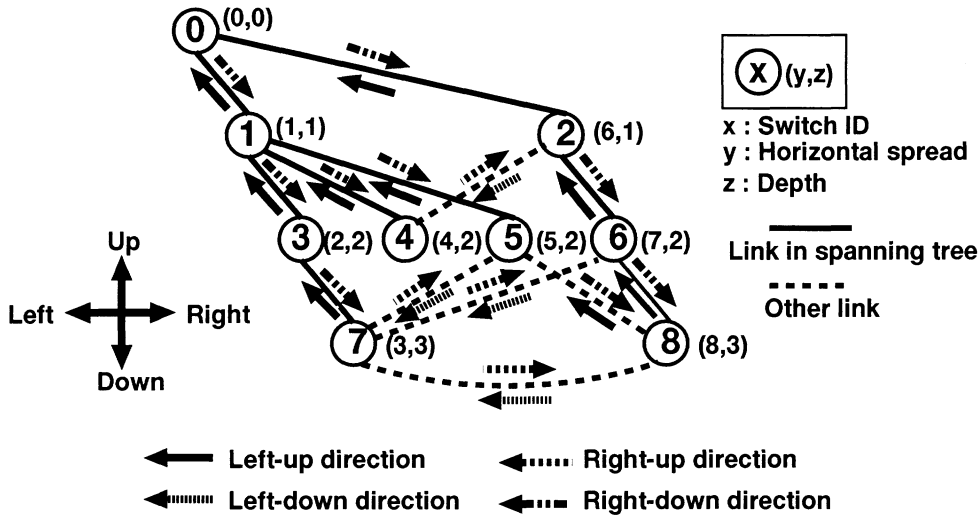


図 3.3: H/V グラフ

3.1.2.1　ターンの列挙

以後，スイッチ通過前のパケット移動方向 $prev$ とスイッチ通過後のパケット移動方向 $next$ により形成されるターンを $T_{prev,next}$ と表すことにする．

H/V グラフでは 4 つの方向があるため，異なる方向へのターンは，図 3.4 に示す 12 通りとなる．

3.1.2.2　循環構造のパターンの列挙および禁止ターンの選択

H/V グラフにおいて発生しうるすべての循環構造を 12 個のターンを基にして列挙し，循環構造を防ぐための禁止ターンの集合を決定する．

ここで，パケット転送時にターン T_i の後にターン T_j が発生した場合のターン間の依存関係を $D(T_i, T_j)$ と表し，$\{D(T_i, T_j) \mid j = (i+1) \bmod n, i = 0, 1, ..., n-1\}$ となる循環関係を持つ n 個のターンの集合を $L(T_0, T_1, ..., T_{n-1})$ と表す．

まず，トポロジがツリーの場合を考えると，その H/V グラフに循環構造は発生しない．しかし，このツリーに対して 1 本のリンクが付加されると，このリンクが接続する 2 つのスイッチとそれらの共通の祖先スイッチ[1]を介する 2 つの循環が発生する．

図 3.5 および図 3.6 はその 2 つの循環を示している．図 3.5 と図 3.6 はリンクをスイッチ B,C 間に付加した例だが，スイッチ B とスイッチ C の相対的な位置関係の違いにより付加リンクの向きが異なる．

図 3.5 に示した部分グラフ 1 における循環は次の 2 つである:

- 循環 $L_a(T_{LU,RD}, T_{RD,RU}, T_{RU,LU})$
- 循環 $L'_a(T_{LU,RD}, T_{RD,LD}, T_{LD,LU})$

[1]ツリーの頂点 (スイッチ) 間の関係を表すため，親族関係の用語を流用する．ある頂点から見て，親，親の親，... をまとめて祖先と呼ぶ．

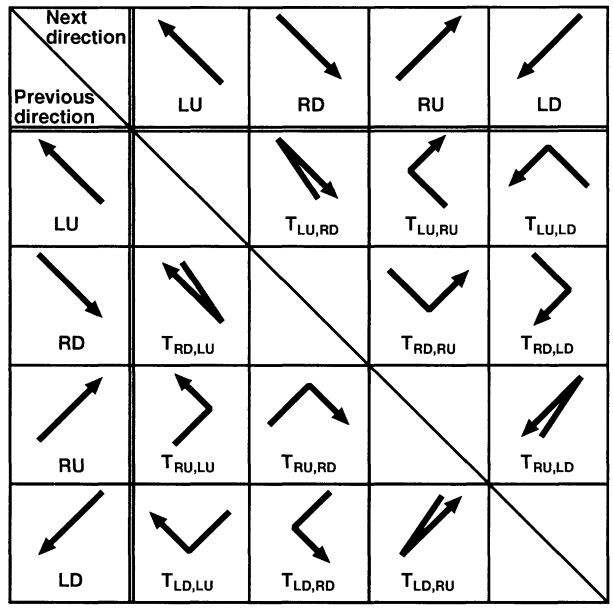


図 3.4: H/V グラフにおけるすべてのターン

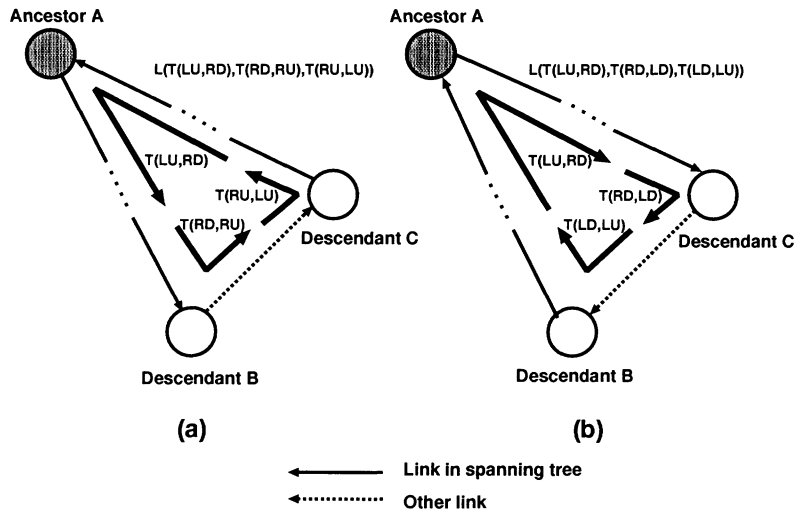


図 3.5: 部分グラフ 1 (a) 循環構造 (左回り) (b) 循環構造 (右回り)

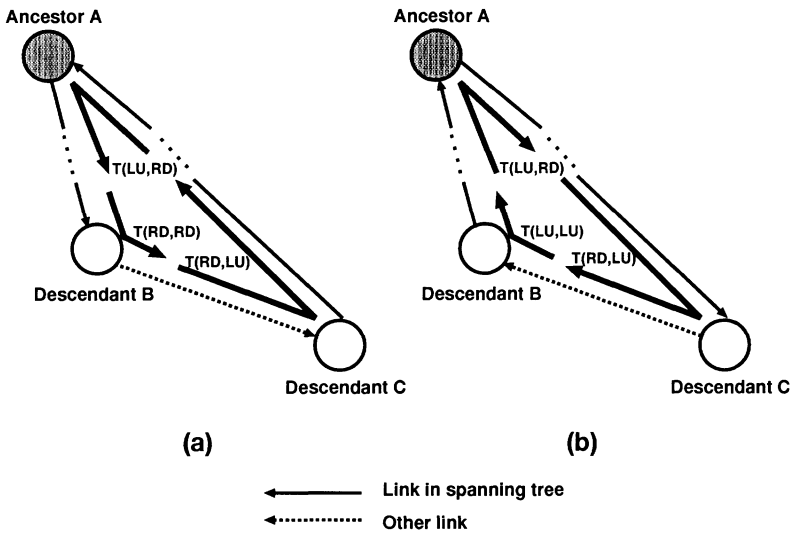


図 3.6: 部分グラフ 2 (a) 循環構造 (左回り) (b) 循環構造 (右回り)

また，同様にして図 3.6 に示した部分グラフ 2 における循環は次の 2 つである:

- 循環 $L_b(T_{LU,RD}, T_{RD,LU})$
- 循環 $L'_b(T_{LU,RD}, T_{RD,LU})$

となる．循環 L_b, L'_b は論理的に同一である．

循環構造を無くすためには各々の循環構造を構成しているターンの一つをそれぞれ禁止する必要がある．ここで，禁止ターンの集合を次のポリシに従って決める:

(a) ターン $T_{LU,RD}$ は禁止しない，かつ，

(b) (できる限り) 図 2.10 のような禁止ターンの偏りが生じないようにする．

スパニングツリーベースのルーティングが H/V グラフにおいて経路保証を実現するためには (1) 任意のスイッチから LU 方向のスパニングツリー構成チャネルを 0 回以上用いて任意の目的地スイッチの祖先スイッチに到達可能であり，かつ，(2) 祖先スイッチに到達後に RD 方向のスパニングツリー構成チャネルを 0 回以上用いて任意の目的地スイッチに到達可能である，という必要があるので, $T_{LU,RD}$ のターンを禁止することはできない．

このポリシより，禁止するターンの集合は次に示す 2 通りとなる:

(a) ターン集合 $P_1 = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}\}$

(b) ターン集合 $P_2 = \{T_{RD,RU}, T_{RD,LD}, T_{RD,LU}\}$

図 3.7(a) および (b) の場合，禁止ターン集合 P_1 および P_2 は分散していることが分かる．しかし，図 3.7(c) の場合，このターンは同一リンク間に偏っている．これはターン $T_{LU,RD}$ を禁止できないためやむを得ない．

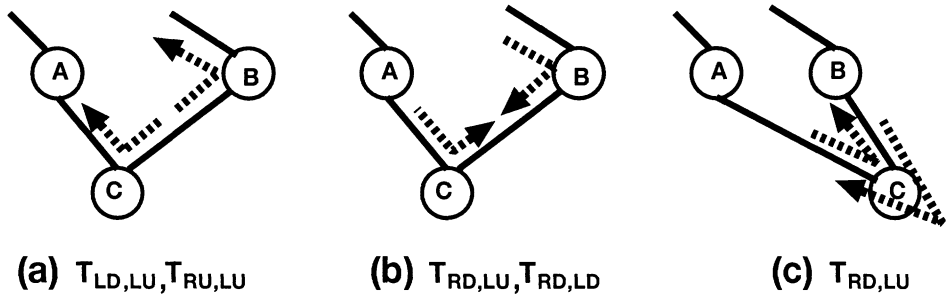


図 3.7: H/V グラフにおける禁止ターン

次に，ターン集合 P_1 もしくは P_2 を禁止した時に発生する可能性のある残りの循環について列挙する．ここで，禁止ターン集合 P_1 のターンを含まない循環を検出するために，残りの 9 つのターンをターン集合 $Q_1 = \{T_{LU,i} \mid i \in \{LD, RU, RD\}\}$ およびターン集合 $Q_2 = \{T_{i,j} \mid i, j \in \{LD, RU, RD\}, i \neq j\}\}$ の 2 種類に分類する．

同様に，禁止ターン集合 P_2 のターンを含まない循環を検出するために，残りの 9 つのターンをターン集合 $Q_3 = \{T_{i,RD} \mid i \in \{LU, LD, RU\}\}$ およびターン集合 $Q_4 = \{T_{i,j} \mid i, j \in \{LU, LD, RU\},\ i \neq j\}$ の 2 種類に分類する．

定理 1 ターン集合 Q_1 のターンを含む循環はターン集合 P_1 のターンを含む．また，ターン集合 Q_3 のターンを含む循環はターン集合 P_2 のターンを含む．□

証明　もし，ターン集合 Q_1 のターン T_q を含み，かつ，ターン集合 P_1 を含まない循環が存在するとする．ターン T_q はパケットが LU 方向からその他の方向へ進む場合に形成されるため，T_q の前のターンは $\{T_{i,LU} \mid i \in \{LD, RU, RD\}\}$ となる．しかし，これはターン集合 P_1 と等しく，仮定と反する．したがって，ターン集合 Q_1 のターンを含むすべての循環はターン集合 P_1 のターンを含む．同様にして，ターン集合 Q_3 のターンを含む循環はターン集合 P_2 のターンを含むことが導かれる．□

定理 1 より，もし，ターン集合 P_1 を禁止した場合，ターン集合 Q_1 は循環を形成しない．よって，もし，ターン集合 P_1 を禁止した場合，すべての循環はターン集合 Q_2 のターンのみで形成されることになる．

また，同様に定理 1 より，もし，ターン集合 P_2 を禁止した場合，ターン集合 Q_3 は循環を形成しない．よって，もし，ターン集合 P_2 を禁止した場合，すべての循環はターン集合 Q_4 のターンのみで形成されることになる．

定理 2 ターン集合 Q_2 における循環は 4 つの循環 $L_1(T_{RU,RD}, T_{RD,LD}, T_{LD,RU})$, $L_2(T_{RD,RU}, T_{RU,LD}, T_{LD,RD})$, $L_3(T_{LD,RU}, T_{RU,LD})$, および $L_4(T_{RD,RU}, T_{RU,RD}, T_{RD,LD}, T_{LD,RD})$ のうち 1 つを含む．□

証明　循環を列挙するため, turn dependency graph (TDG) を導入する．TDG はターン間の依存関係を示すグラフで頂点はパケットターンを示し，方向を持つ辺は 2 つのターン間の依存を示す．図 3.8 はターン集合 Q_2 に対する TDG であり，頂点はターン集合 Q_2 の各ターンを示している．

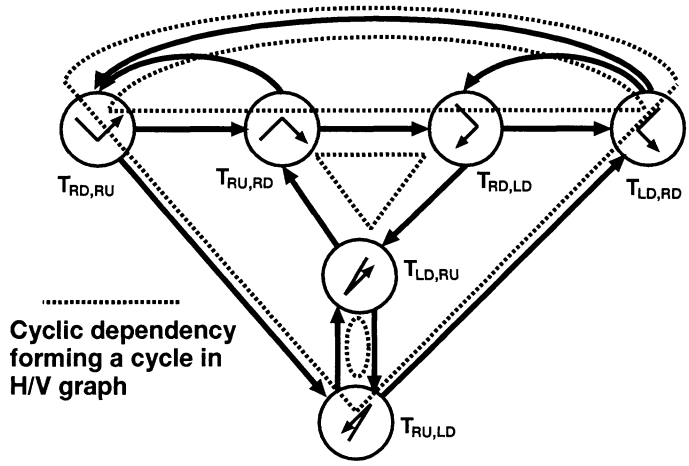


図 3.8: ターン集合 Q_2 に対する TDG

TDG から循環構造を形成するターンの連鎖を列挙するためには, 次の 2 つの条件を満たすターンの連鎖を探せばよい.

- ターンの連鎖を形成する任意のターンを示す頂点から出発し, 矢印を辿ってターンの連鎖を形成する残りのすべての頂点のみを訪問した後に出発頂点に戻ってくる.
- 上記の訪問の際に, vertical direction と horizontal direction においてそれぞれ対となるターンを計 2 回以上行なっている.

図 3.8 より，ターン集合 Q_2 に対する TDG において，この条件を満たすすべての循環は点線で示した 4 つのいずれかを含むことは明らかである．よって，ターン集合 Q_2 における循環は循環 L_1, L_2, L_3 および L_4 の 1 つを含む．□

循環 L_1, L_2, L_3 および L_4 を図 3.9 に示す.

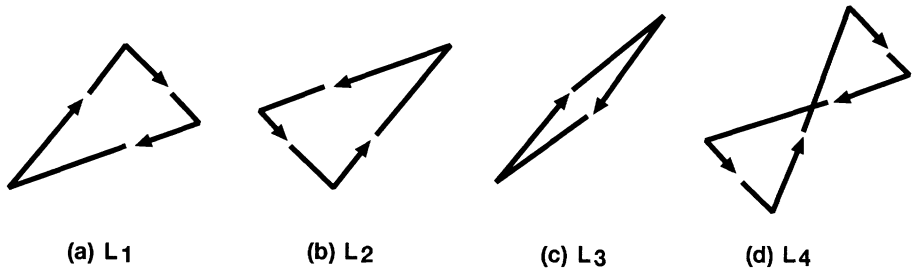


図 3.9: ターン集合 Q_2 における 4 つの循環

この 4 つの循環を除去するために禁止すべきターン集合は次の 2 通りとなる:

$$\text{ターン集合 } P_1' = \{T_{LD,RU}, T_{LD,RD}\}$$
$$\text{ターン集合 } P_1'' = \{T_{RU,LD}, T_{RU,RD}\}$$

ターン集合 $\{T_{LD,RU}, T_{RD,RU}\}$ および $\{T_{RU,LD}, T_{RD,LD}\}$ を禁止することも同様に考えられるが，ターン集合 P_1 を禁止した場合，これらは禁止ターン $T_{LD,LU}$ および $T_{RU,LU}$ と各々禁止ターンの偏りを生成してしまう．

このことより，ターン集合 P_1 を禁止した場合，すべての循環を除去するために禁止すべきターン集合は次の 2 通りとなる:

(a) ターン集合 $P_{1a} = P_1 + P_1' = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}, T_{LD,RU}, T_{LD,RD}\}$

(b) ターン集合 $P_{1b} = P_1 + P_1'' = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}, T_{RU,LD}, T_{RU,RD}\}$

定理 3 ターン集合 Q_4 における循環は 4 つの循環 $L_5(T_{LD,RU}, T_{RU,LU}, T_{LU,LD})$, $L_6(T_{LD,LU}, T_{LU,RU}, T_{RU,LD})$, $L_7(T_{LD,RU}, T_{RU,LD})$ および $L_8(T_{LD,LU}, T_{LU,RU}, T_{RU,LU}, T_{LU,LD})$ のうち 1 つを含む．□

証明　ターン集合 Q_4 に対する TDG を図 3.10 に示す．定理 2 の証明と同様に考えると，

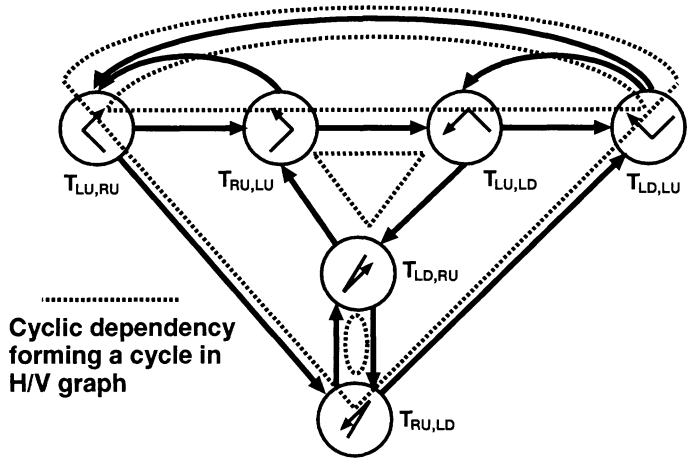


図 3.10: ターン集合 Q_4 に対する TDG

ターン集合 Q_4 における循環は循環 L_5, L_6, L_7 および L_8 の 4 つのうち 1 つを含む．□

循環 L_5, L_6, L_7 および L_8 を図 3.11 に示す．

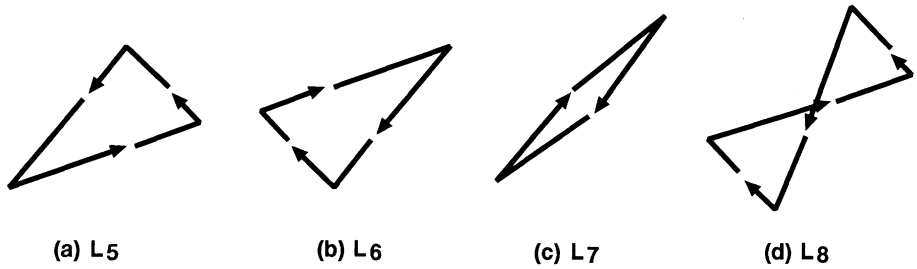


図 3.11: ターン集合 Q_4 における 4 つの循環

ターン集合 P_1', P_1'' を決定した時と同様に考えて，この 4 つの循環を除去するために禁止すべきターン集合は次の 2 通りとなる:

$$\begin{aligned} \text{ターン集合 } P_2' &= \{T_{LD,RU}, T_{LU,RU}\} \\ \text{ターン集合 } P_2'' &= \{T_{RU,LD}, T_{LU,LD}\} \end{aligned}$$

よって，ターン集合 P_2 を禁止した場合，すべての循環を除去するために禁止するターン集合は次の 2 通りとなる:

(a) ターン集合 $P_{2a} = \{T_{RD,LD}, T_{RD,RU}, T_{RD,LU}, T_{LD,RU}, T_{LU,RU}\}$

(b) ターン集合 $P_{2b} = \{T_{RD,LD}, T_{RD,RU}, T_{RD,LU}, T_{RU,LD}, T_{LU,LD}\}$

3.1.2.3　禁止ターン数の削減

前節にて，H/V グラフにおける禁止ターン集合 $P_{1a}, P_{1b}, P_{2a}, P_{2b}$ を定義した．ターン集合 $P_{1a}, P_{1b}, P_{2a}, P_{2b}$ を禁止した場合，各々ターン集合 P_1', P_1'', P_2', P_2'' は禁止される．しかし，ターン集合 P_1', P_1'', P_2', P_2'' は禁止しなくとも循環が生じない場合がある．例えばターン集合 P_{1a} を禁止した場合，図 3.12 においてスイッチ 4 におけるターン $T_{LD,RD}$，およびスイッチ 7 におけるターン $T_{LD,RU}$ は禁止される．しかし，スイッチ 4 におけるターン $T_{LD,RD}$ およびスイッチ 7 におけるスイッチ 5 からスイッチ 6 へのターン $T_{LD,RU}$ を許した場合においても循環は生じない．

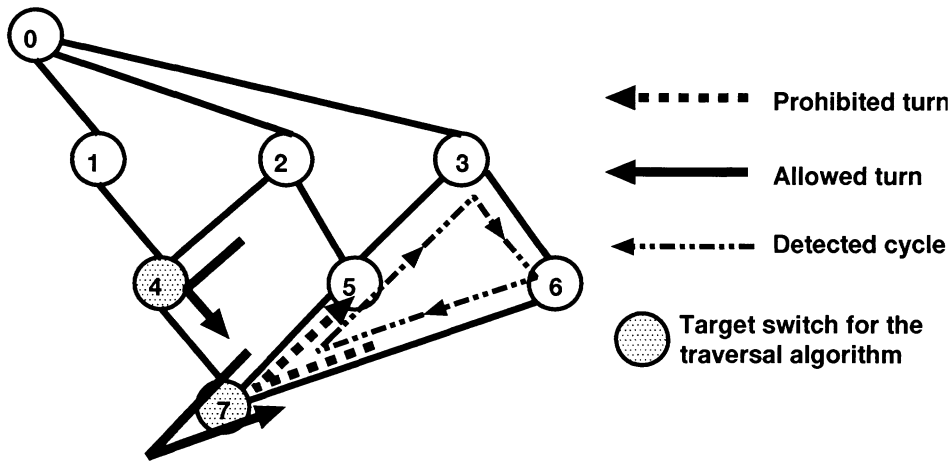


図 3.12: H/V グラフにおける冗長な禁止ターン

そこで，ターン集合 P_1, P_2 を各々禁止した場合，各々ターン集合 P_1', P_1'', P_2', P_2'' のターンが循環を形成するか否かを判定する循環検出アルゴリズム (cycle detection algorithm) を導入して，すべての循環を除去しつつ，禁止するターン数を減らす．

次の説明ではターン集合 P_1 を禁止した場合におけるターン集合 P_1' のターンが循環を形成するか否かを判定する循環検出アルゴリズムについて述べるが，その他のターン集合を選択した場合についても同様に考えることが可能である．

循環検出アルゴリズムは H/V グラフにおける探索アルゴリズムである．循環構造検出のための探索は, H/V グラフにおいてターン集合 P_1' に含まれる 2 つのターンのいずれかが発生しうるすべてのスイッチに対して行なう．この際，該当スイッチは次のいずれかの条件を満たすものとなる:

(a) 1 本以上の RU 方向の物理チャネルと 1 本以上の RD 方向の物理チャネルを持つ (ターン $T_{LD,RD}$ を生成);

(b) 2 本以上の RU 方向の物理チャネル を持つ (ターン $T_{LD,RU}$ を生成).

循環構造の検出は, 上記の該当スイッチにおける該当物理チャネルを起点として depth first search (DFS) を行ない, 禁止ターンを行なわずに特定の物理チャネルを通って該当スイッチに戻ってくるかを調べることにより行なわれる.

探索の具体的な手順は次のようになる.

1. (a) の条件に該当するスイッチからの探索　まず, 該当スイッチから RD 方向の物理チャネルを辿って隣接スイッチを訪問する. その後, DFS の方針に基づいて, 訪問先のスイッチにおいて選択可能な出力物理チャネルが存在する限り次々と隣接スイッチを訪問していく. そして, 冗長な探索を避けるために一度訪問した出力物理チャネルには使用済のチェックをして二度と使われないようにしておく. この際，次の条件に該当する出力物理チャネルは選択不可とする:

- ターン集合 P_1 に含まれる禁止ターンが発生する,
- 現在の探索より以前に行なわれた別のスイッチにおける探索の結果として利用することが禁止されている，もしくは,
- 使用済みのチェックがされている.

訪問先のスイッチにおいて選択可能な出力物理チャネルが1つも存在しない場合，1 つ手前のスイッチに戻ってから探索を続ける. 探索において, 隣接スイッチから LD 方向の物理チャネルを通って該当スイッチに戻ってきた場合には, その LD 方向の物理チャネルと探索の開始時に利用した RU 方向の物理チャネルとの間で発生するターン $T_{LD,RU}$ のターンを含む循環構造が検出されたことになる. そのため，該当する LD 方向の物理チャネルの利用後に RU 方向の物理チャネルを利用することを禁止する. 探索は, 選択可能な出力物理チャネルが一つも存在しなくなり出発スイッチに戻ってきた時点で終了する. なお, 探索は該当スイッチに存在するすべての RD 方向の物理チャネルを起点としてそれぞれ行なわれる.

2. (b) の条件に該当するスイッチからの探索　(b) の条件に該当するスイッチからの探索は, (a) の条件に該当するすべてのスイッチにおける探索が終了した後に行なう. これはターン $T_{LD,RD}$ とターン $T_{LD,RU}$ が同じ循環に含まれる場合にターン $T_{LD,RD}$ を優先的に禁止するためである. ターン $T_{LD,RU}$ は同一スイッチにおける禁止ターンの偏りを生成しうるため，禁止数をなるべく減らすことが禁止ターンの分散につながる.

探索の方法は (a) の場合とほぼ同様にして行なわれ, 異なることは (1) 探索の開始の際に辿るべき物理チャネルが RU 方向の物理チャネルである，および，(2) 禁止するターンがターン $T_{LD,RU}$ である，の2点である．

上記の循環構造検出アルゴリズムに必要とされる計算量は, スイッチの数を N, スイッチあたりのリンク数を L とすると $O(N^2 * L)$ となるので, SAN では現実的な範囲内である．

3.1.3　L-turn/R-turn ルーティングアルゴリズム

本節にて L-turn ルーティングと R-turn ルーティングを定める．

left-up first turn (L-turn)/α ルーティングは禁止ターン集合 P_1 のターンをすべて禁止し，ターン集合 P_1' のターンの一部を循環検出アルゴリズムにより禁止したものとする．そして，L-turn/β ルーティングは禁止ターン集合 P_1 のターンをすべて禁止し，ターン集合 P_1'' のターンの一部を循環検出アルゴリズムにより禁止したものとする．また，L-turn ルーティングとは L-turn/α ルーティングと L-turn/β ルーティングの2種類を指すことにする．L-turn ルーティングの名称は LU 方向の目的地スイッチにパケットを転送する場合，はじめに必要ホップ数 LU 方向に移動しなければならないことによる．

一方，right-down last turn (R-turn)/α ルーティングは禁止ターン集合 P_2 のターンをすべて禁止し，ターン集合 P_2' のターンの一部を循環検出アルゴリズムにより禁止したものとする．そして，R-turn/β ルーティングは禁止ターン集合 P_2 のターンをすべて禁止し，ターン集合 P_2'' のターンの一部を循環検出アルゴリズムにより禁止したものとする．また，R-turn ルーティングとは R-turn/α ルーティングと R-turn/β ルーティングの2種類を指すことにする．R-turn ルーティングの名称は *RD* 方向への移動を最後に行なわなければならないことによる．

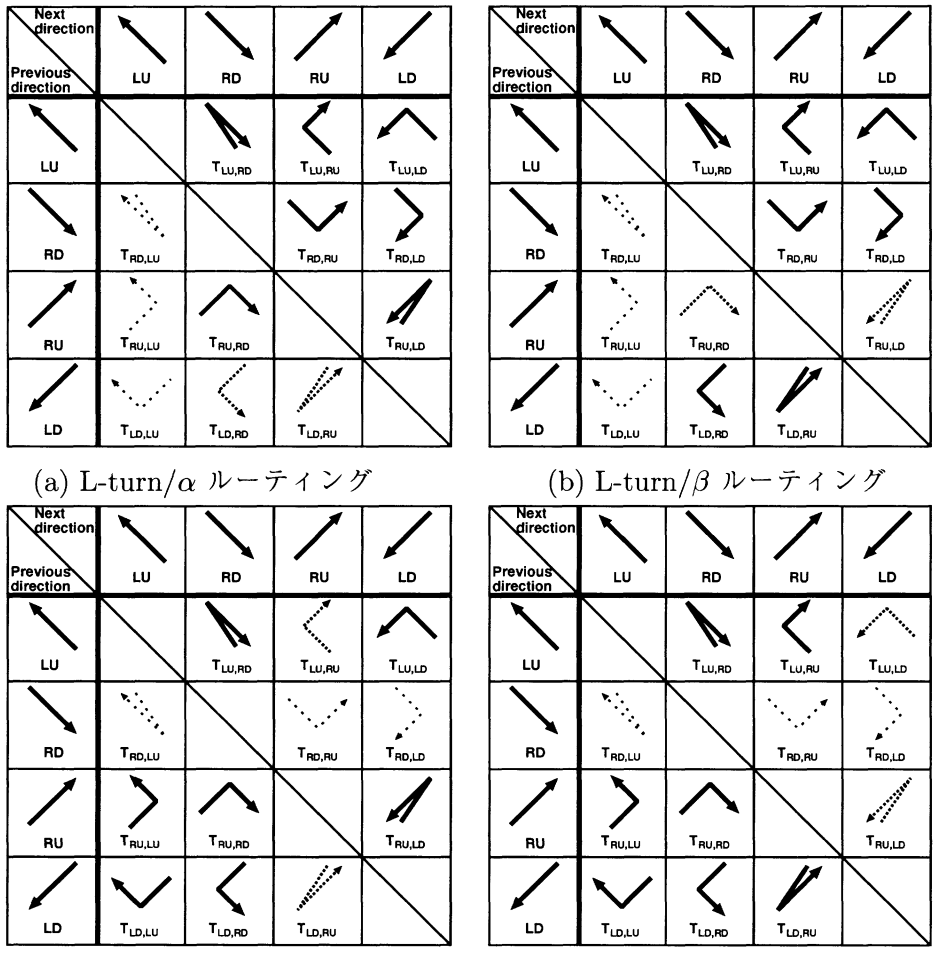
この4つの適応型アルゴリズムのパケットの転送制限を図 3.13 に示す．図 3.13 において，薄い点線は禁止ターン，濃い点線は循環検出アルゴリズムにより禁止される制限ターンを示す．

一般的に非最短経路を許す場合，各パケットが消費する物理チャネル数が増えることにより，スループットが低下する傾向がある．そこで，L-turn ルーティングと R-turn ルーティングは，選択可能な経路の中から最短である経路のみを用いる．

定理 4 *L-turn* ルーティングと *R-turn* ルーティングはデッドロックフリーである．□

証明　L-turn ルーティングにおいて，LU 方向を含むすべての循環はターン集合 P_1 によって除去される．また，他の循環はターン集合 P_1' もしくは P_1'' によって除去される．したがって，L-turn ルーティングにおいてすべての循環は除去される．同様にして，R-turn ルーティングにおいてもすべての循環は除去される．したがって，L-turn ルーティングと R-turn ルーティングはデッドロックフリーである．□

定理 5 *L-turn* ルーティングと *R-turn* ルーティングはあらゆるトポロジに対して，すべてのスイッチ間の経路が存在することを保証する．□



(a) L-turn/α ルーティング　(b) L-turn/β ルーティング

(c) R-turn/α ルーティング　(d) R-turn/β ルーティング

図 3.13: L-turn ルーティングと R-turn ルーティング

証明　スパニングツリー構成リンク間のターンはターン $T_{LU,RD}$ のみであることは自明である．また，L-turn ルーティングと R-turn ルーティングにおいて，ターン $T_{LU,RD}$ は許されている．そのため，パケットはスパニングツリー構成チャネルを必要数利用することにより，任意の出発地スイッチから任意の目的地スイッチへ到達することが可能となるため経路保証が実現される．□

3.2　評価

3.2.1　シミュレーション方式

高性能大規模計算機のアーキテクチャのシミュレーションは，確率モデルシミュレーション，トレースドリブンシミュレーション，エグゼキューションドリブンシミュレーションおよび命令レベルシミュレーションに分類することができる [上樂 99]．

確率モデルシミュレーションは，メモリアクセスのパターンなどを乱数モデルに基づいて発行して評価を行う方法であり，相互結合網やメモリシステムなどの評価を行なう際によく用いられる．一方，トレースドリブンシミュレーション [JP92] は，実機上でアプリケーションプログラムを実行してメモリ参照のアドレスのトレースデータをとり，それをシミュレータに入力して評価を行なう方法である．また，エグゼキューションドリブンシミュレーション [JMPJ99] では，アプリケーションプログラムを解析し，同期操作や共有メモリアクセス等のプロセス間のインタラクションが起きる時には，そこにシミュレータを呼び出すコードを埋め込む．そして，このプログラムをコンパイルしてシミュレータをリンクし，計算機上で複数のプロセスを生成して実際に並列 (並行) に実行する．メモリ参照の順序は，順序の保証が必要なアクセス時にこれらのプロセスを再スケジューリングすることで保証する．また，命令レベルシミュレーションは，CPU のインストラクションのレベルまでソフトウェアでシミュレートして，実機と同様の環境を構築して評価する方法である．

この中で，大規模な SAN の性能評価にはほとんどの場合，確率モデルシミュレーションを用いる．これは後者になるほど正確な検証を行うことができる一方，シミュレータの開発が困難を極め，プログラムの実行時間が莫大になるためである．本研究においても，性能評価に確率モデルシミュレーションを用いた．

3.2.2　シミュレーション条件

L-turn ルーティング，R-turn ルーティングおよび Up*/Down* ルーティングの各適応型アルゴリズムをフリットレベルシミュレータを用いて評価を行った．フリットレベルシミュレーションは確率モデルシミュレーションの中でフリットの動作単位で検証を行う精度の高い方法である．シミュレータは C++言語により約 12,000 行で記述されており，パケット転送方式に VCT 方式および WH 方式を選択することができる．また，本シミュレータにおいてネットワークサイズ，パケット長等はパラメータを変更することにより設定することができる．このシミュレータの実行速度は，Pentium3 1.2GHz を搭載し，FreeBSD 4.6.1-RELEASE をインストールしたマシンにおいて 64 台のスイッチを用いた SAN の

1,000,000 クロックの動作が約 8 分で完了する程度である．

本シミュレーションにおいて目的地の PC は次のトラフィックパターンにより決定した．

- uniform
  すべての目的地はランダムに決定され，均一に分散される．

- bit reversal
  まず，各 PC に 0 から $(n-1)$ (ただし，n は PC 数) までの一意の 2 進の番号を割当てる．
  そして，2 進の番号 $(a_0, a_1, \cdots, a_{n-2}, a_{n-1})$ を持つ PC は自分の番号のビット列を逆順に並べた番号 $(a_{n-1}, a_{n-2}, \cdots, a_1, a_0)$ の PC へパケットを送る．

シミュレーションに用いたその他の条件を表 3.2 に示す．

表 3.2: 適応型アルゴリズムのシミュレーションパラメータ

実行時間	1,000,000 クロック (初めの 50,000 クロックは無視)
トポロジ	不規則トポロジもしくは 2D トーラス (付録参照)
サイズ	16 もしくは 64 スイッチ
仮想チャネル数	1
パケット長	128 フリット
パケット転送方式	VCT 方式
フリット転送時間	最低 3 クロック

シミュレーションにおいて初めの 50,000 クロックはネットワークが安定せず，想定した負荷に達していないと考えられるため評価の対象外とした．

シミュレータのスイッチのポート数は，現実の SAN である RHiNET-2/SW[STH+00], Myrinet スイッチ M3F-SW8，および，M3F-SW8M [2]が 8 個である点をふまえ，8 個とした．そして，内 4 ポートは各々異なる PC に直結した．スイッチの残りの 4 ポートは他のスイッチとの接続に利用される．ネットワークのトポロジはトーラス以外に同一スイッチ対にリンクを 2 本以上接続しない，という制約を課した上でランダムに生成した．スパニングツリーの構築アルゴリズムとしては，親子関係を結ぶスイッチ対を決定する付加リンクの選択アルゴリズムに (1)Autonet で用いられた minimum depth スパニングツリー (MDST) を基にした breadth first search (BFS) と (2)Sancho らが提案したヒューリスティックルールによる depth first search (DFS) を用いた [JA00]．(2) のヒューリスティックルールは詳細を第 3.3 節で述べるが，既にスパニングツリーに組み込まれているスイッチとの接続数の最も多いスイッチへのリンクを選択するものである．ルートは BFS スパニングツリーの場合，Autonet と同様に ID 0 のスイッチとし [Mae91]，DFS スパニングツリーの場合，crossing path, average distance の値により決定するヒューリスティックルール [JA00] により選択した．また，選択可能な物理チャネルの中からランダムに出力物理チャネルを選択するランダム選択機構を OSF として使用した．

[2]http://www.myrinet.com/myrinet/product_list.html

フリット転送時間はルーティング，スイッチ内のクロスバーの移動，スイッチ間の移動に各1クロックとした．実際のスイッチでは高速で動作させるために1クロックあたりの処理内容を細分化する．そのため，スイッチ内のパケット転送は多くの場合 2 クロックではできない．しかし，(1) スイッチ内のパケット転送クロック数はスイッチの設計方針に依存するため特定できない，(2) スイッチ内の動作は上記2つに大別することができる，という2点をふまえ2クロックとした．また，相互結合網のシミュレーションでは，可変長のパケット長を用いる場合と固定長のパケット長を用いる場合がある [JSL02]．しかし，パケット長の分散はアプリケーションに大きく依存するため，本シミュレーションでは固定長とした．また，パケット長については数十スイッチ規模の相互結合網の評価に頻繁に用いられる 128 フリット (具体的には，例えば，スイッチが1クロックに 2 byte を送信可能とすると，ヘッダを含めたパケット長は 128×2 = 256 byte となる．) とした [JSL02]．この他に4種類のパケット長 (64, 128, 256, 512 フリット) を用いた場合や他の固定パケット長を用いた場合についても一部評価を行ったが，結果の傾向は変わらなかった．L-turn ルーティングと R-turn ルーティングは仮想チャネルを持たない SAN を主な対象としているため，シミュレーションにおいて仮想チャネルは1本，つまり物理チャネルのみの利用とした．また，次の指標について評価を行った．

ネットワークレイテンシ

ある PC p がパケットの最初のフリットを NIC の入力バッファに挿入した時刻を t_0，目的地の PC q の NIC がパケットの最後のフリットを受け取った時刻を t_1 とする．ここで，$T_{lat}(p,q) = t_1 - t_0$ をネットワークレイテンシと呼び，ネットワークの性能を測る指標とした．また，PC の入力バッファサイズは5パケット分とした．

スループット

スループットは，全 PC が毎クロックに 1 フリット受信する場合を 1.00 として，受信トラフィックの最大値とした [JSL02]．

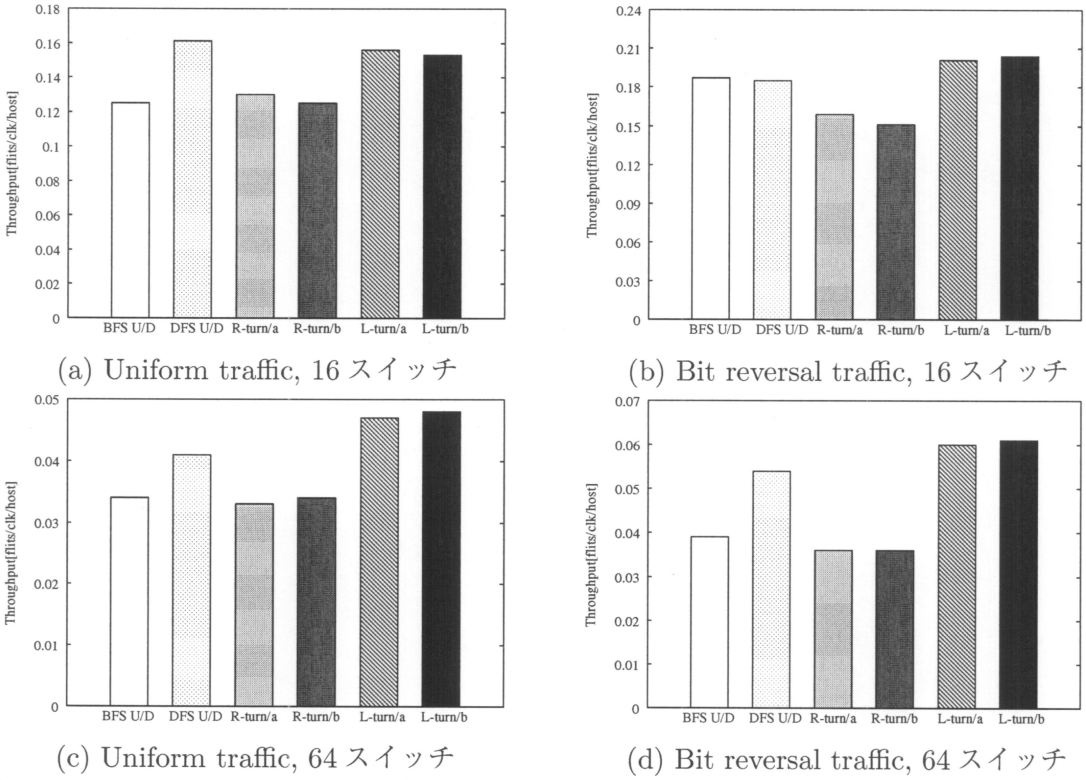
平均ホップ数

パケットが目的地の PC に到達するまでに通過したスイッチ数とした．

3.2.3　不規則なトポロジの SAN における評価結果

L-turn ルーティング，R-turn ルーティングおよび Up*/Down* ルーティングの不規則なトポロジの SAN におけるシミュレーション結果を図3.14，表3.3，および表3.4に示す．図3.14において縦軸は10個のトポロジでの平均スループットを表している．また，各適応型アルゴリズムにおけるパケットの平均ホップ数を表3.5に示す．

表3.5より R-turn ルーティングは平均ホップ数が BFS Up*/Down* ルーティングに比べ大きいことがわかる．このことから，R-turn ルーティングは上記の2つの理由以外に



(a) Uniform traffic, 16 スイッチ　(b) Bit reversal traffic, 16 スイッチ

(c) Uniform traffic, 64 スイッチ　(d) Bit reversal traffic, 64 スイッチ

図 3.14: 不規則なトポロジの SAN における平均スループット

平均ホップ数の増加も低スループットの原因といえる．

図 3.14，表 3.3，および表 3.4 より，uniform traffic を用いた 16 スイッチの SAN の場合のみ，L-turn ルーティングはヒューリスティックルールを用いた DFS Up*/Down* ルーティング (詳細は第 3.3 章) に比べ若干スループットが低いが，その他の状況では最大 17%のスループット向上を筆頭に最も優れたスループットを示している．16 スイッチの SAN において uniform traffic の場合 L-turn ルーティングのスループットが DFS Up*/Down* ルーティングのスループットに比べ低い理由は，(1) 表 3.5 に示す通り，L-turn ルーティングは DFS Up*/Down* ルーティングに比べパケットの平均ホップ数が大きく，かつ (2) 16 スイッチの uniform traffic の場合トラフィックを分散する余地が小さいためである．つまり，この条件の場合のみ DFS Up*/Down* ルーティングのパケットホップ数の削減効果の方が，L-turn ルーティングのトラフィック分散効果よりもスループットに効果的であったためと考えらる．

しかし，その他のすべての場合において L-turn ルーティングが最も優れているため，Up*/Down* ルーティングを改良することよりも，禁止ターンの分散配置を実現する我々の手法の方がスループット改善に効果的であることがわかった．

また，R-turn ルーティングは L-turn ルーティングに比べて常にスループットが低い．これは次の 2 つの理由による．

- R-turn ルーティングはターン $T_{RD,LU}$ を除くすべての LU 方向へのターンが許可さ

れている．また，R-turn ルーティングはルートから離れる方向である *RD* 方向から他の方向へのターンが全て禁止されている．そのため，パケットがルート方向に転送される傾向があり，L-turn ルーティングに比べトラフィックが偏りやすい．

- R-turn ルーティングのパケットのホップ数が L-turn ルーティングのパケットのホップ数に比べ大きい．

また，不規則なトポロジの SAN ではスループットの分散は，表 3.3 および表 3.4 より，各適応型アルゴリズムで大差がないことが分かった．

次に，スループットの次に重要な評価項目であるレイテンシについて図 3.15 および図 3.16 に示す．図 3.15 および図 3.16 は DFS Up*/Down* ルーティングに対する L-turn/α ルーティングのスループット向上の割合が平均的であったトポロジにおける受信トラフィックとレイテンシの関係を示している．図 3.15 および図 3.16 より，L-turn ルーティングは DFS Up*/Down* ルーティングに比べ 16 スイッチの場合はレイテンシが若干大きいが，64 スイッチの場合はレイテンシが小さいことがわかる．また，図 3.15 および図 3.16 より R-turn ルーティングと BFS Up*/Down* ルーティングは L-turn ルーティングと DFS Up*/Down* ルーティングに比べトラフィックが少ない場合においてレイテンシが大きいことがわかった．これは，R-turn ルーティングと BFS Up*/Down* ルーティングはトラフィックが偏ることによりパケットの衝突が頻繁に起きたためと考えられる．

表 3.3: 16 スイッチの不規則なトポロジの SAN におけるスループットとその分散

	Uniform				Bit reversal			
	平均	分散	最小	最大	平均	分散	最小	最大
BFS Up*/Down*	0.125	0.017	0.105	0.155	0.187	0.039	0.127	0.281
DFS Up*/Down*	0.161	0.012	0.149	0.182	0.185	0.023	0.145	0.230
R-turn/α	0.130	0.024	0.098	0.170	0.159	0.031	0.106	0.221
R-turn/β	0.125	0.021	0.091	0.155	0.151	0.023	0.107	0.186
L-turn/α	0.156	0.020	0.112	0.186	0.201	0.031	0.148	0.263
L-turn/β	0.153	0.015	0.123	0.174	0.204	0.033	0.145	0.260

表 3.4: 64 スイッチの不規則なトポロジの SAN におけるスループットとその分散

	Uniform				Bit reversal			
	平均	分散	最小	最大	平均	分散	最小	最大
BFS Up*/Down*	0.034	0.004	0.027	0.040	0.039	0.007	0.030	0.053
DFS Up*/Down*	0.041	0.004	0.033	0.048	0.054	0.007	0.043	0.064
R-turn/α	0.033	0.003	0.028	0.040	0.036	0.005	0.030	0.045
R-turn/β	0.034	0.003	0.031	0.038	0.036	0.005	0.027	0.047
L-turn/α	0.047	0.005	0.038	0.055	0.060	0.006	0.048	0.070
L-turn/β	0.048	0.004	0.041	0.053	0.061	0.005	0.052	0.068

表 3.5: 不規則なトポロジの SAN における平均ホップ数

	16 スイッチ		64 スイッチ	
	Uniform	Bit reversal	Uniform	Bit reversal
BFS Up*/Down*	2.03	2.01	3.82	3.68
DFS Up*/Down*	1.95	2.01	3.53	3.45
R-turn/α	2.04	2.05	3.87	3.73
R-turn/β	2.05	2.07	3.86	3.76
L-turn/α	2.01	2.00	3.67	3.61
L-turn/β	2.01	2.02	3.67	3.65

また，表3.3および表3.4よりL-turn ルーティングは uniform traffic に比べトラフィックに偏りが生じる bit reversal traffic において特に効果的であるといえる．

3.2.4　規則的なトポロジの SAN における評価結果

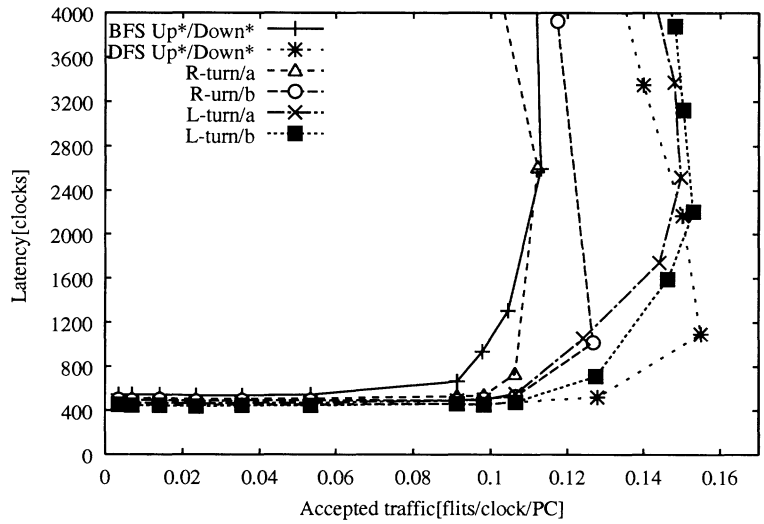
規則的なトポロジである 8 × 8 2D トーラスを用いた場合の各適応型アルゴリズムの評価結果を図3.17に示す．横軸は受信トラフィック，縦軸はレイテンシを示す．また，表3.6に 8 × 8 2D トーラスにおけるパケットの平均ホップ数を示す．

表 3.6: 8 × 8 2D トーラスの SAN における平均ホップ数

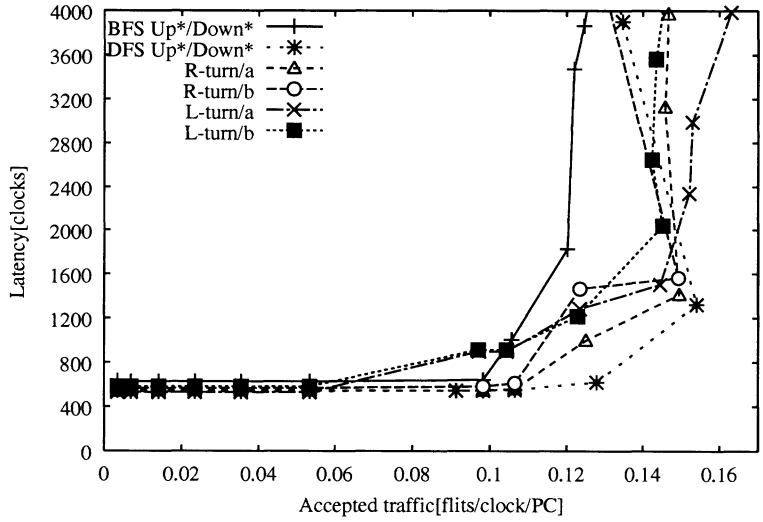
	Uniform	Bit reversal
BFS Up*/Down*	4.47	4.41
DFS Up*/Down*	4.30	3.71
R-turn/α	4.32	3.77
R-turn/β	4.31	3.70
L-turn/α	4.32	3.82
L-turn/β	4.32	3.74

図3.17よりL-turn ルーティングは既存の両Up*/Down* ルーティングに比べ低レイテンシかつ，54%～80%のスループット向上を達成していることがわかる．また，表3.6より不規則なトポロジの場合と同様にL-turn ルーティングは平均ホップ数の削減効果はDFS Up*/Down* ルーティングに比べ小さい．このことからトーラスにおいては禁止ターンの分散が不規則なトポロジの場合に比べより大きくスループットに影響するといえる．

実際の SAN では規則性や階層性がある程度見られるため，規則網であるトーラスにおける評価は，不規則なトポロジの場合と同様に重要である．L-turn ルーティングは，このトーラスでも最も高い性能を達成したことから 2 つの Up*/Down* ルーティングに比べ，様々なトポロジで安定しているといえる．

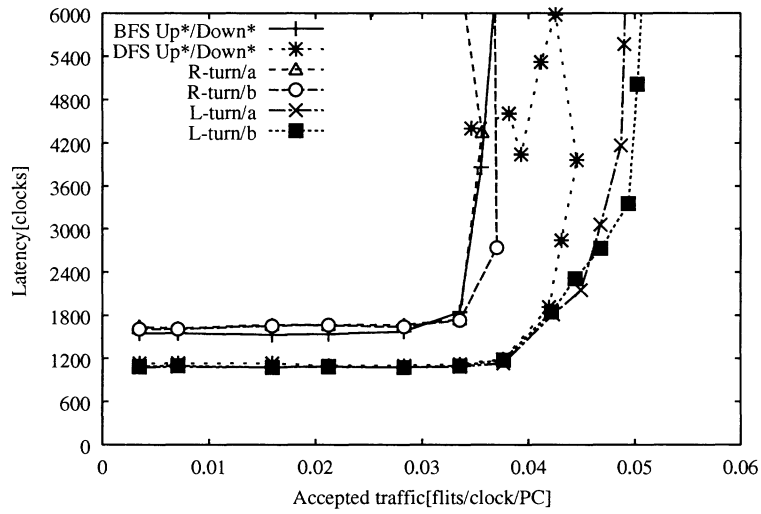


(a) Uniform traffic

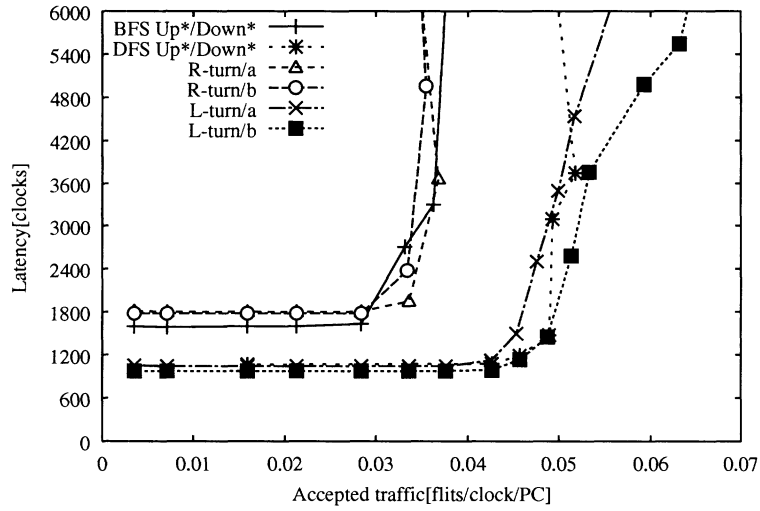


(b) Bit reversal traffic

図 3.15: 不規則なトポロジの SAN におけるスループットとレイテンシ (16 スイッチ)

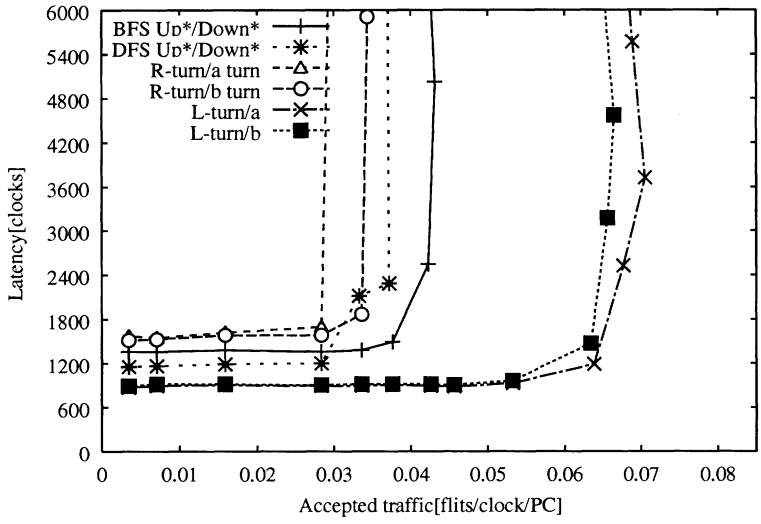


(a) Uniform traffic

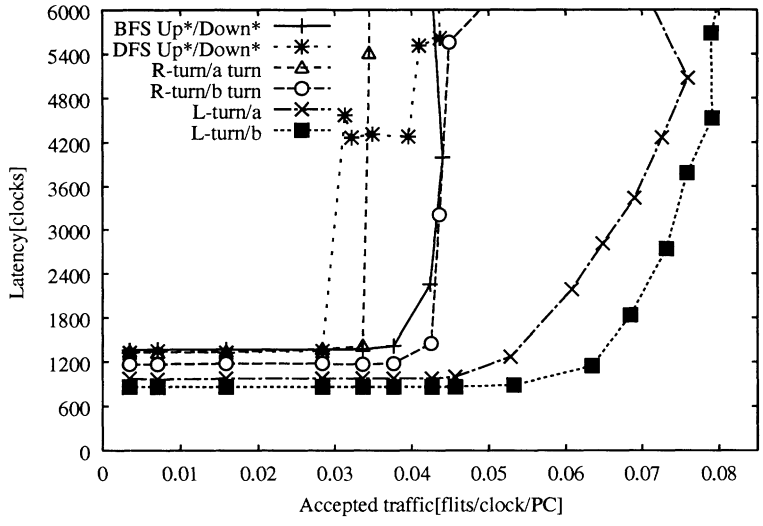


(b) Bit reversal traffic

図 3.16: 不規則なトポロジの SAN におけるスループットとレイテンシ (64 スイッチ)



(a) Uniform traffic



(b) Bit reversal traffic

図 3.17: 8 × 8 2D トーラスの SAN におけるスループットとレイテンシ

3.3　その他の解決策

Up*/Down* ルーティングは禁止ターンが偏ることによりネットワークバンド幅を生かすことが難しい点が問題である．この問題を解決するために本章では L-turn ルーティングと R-turn ルーティングを提案した．

これに対し，他の解決策を目指した研究も同時並行的に進んでいる。本節では他の解決策について紹介し，L-turn/R-turn ルーティングとの違いについて説明する．

3.3.1　ヒューリスティックルールを用いた Up*/Down* ルーティング

3.3.1.1　BFS スパニングツリーと DFS スパニングツリー

Up*/Down* ルーティングの性能は，各物理チャネルに対する方向の割当て方に大きく影響されるため，スパニングツリーの構築アルゴリズムが重要になる．そこで，ヒューリスティックルールによる depth first search (DFS) のスパニングツリーの構築方法に関する研究が行われた [JAJ00][JA00]．これは，第 2.6.5 節で述べた breadth first search (BFS) のスパニングツリー構築アルゴリズムと大幅に手続きが異なる．

まず，BFS スパニングツリーを用いた Up*/Down* ルーティングにおける冗長な禁止ターンについて検討する．BFS スパニングツリーを基にした場合，第 2.6.5 節で述べた通り，スパニングツリーの構築はシンプルであるが，同階層の物理チャネルの方向の割当て方の曖昧さにより同階層の物理チャネル間に冗長な禁止ターンが発生する [JAJ00] 問題が存在する．この問題の具体例として Up*/Down* ルーティングを 9 スイッチの不規則なトポロジの SAN に適用した図 3.18 を示す．

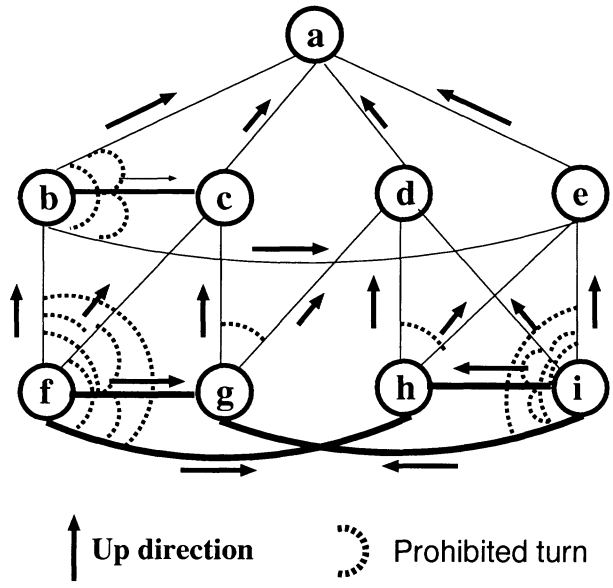


図 3.18: 同階層の物理チャネルによる冗長な禁止ターン

図 3.18 において，循環を形成する太線で示された一連の同階層の物理チャネルは一箇所

で循環を切断することによりデッドロックは防げるにも関わらず，スイッチ**f**とスイッチ**i**の2個所で循環を切断している．この冗長な制約が発生する問題は，同階層の物理チャネル間に対する方向の割当てを変えることにより解決が可能である．例えばスイッチ**h**からスイッチ**f**への物理チャネルの向きを up 方向に変更することにより禁止ターン数を減らすことができる．しかし，BFS スパニングツリーでは物理チャネルの方向をスイッチ ID を基に割当てるため，この問題を解決することは難しい．

そこで，この問題を解決するために DFS スパニングツリーが Sancho らにより提案された [JA00]．DFS スパニングツリーの構築手続きを次に示す．

procedure Depthfirst(v_k)
begin
while all nodes have not been visited yet *do*
　for i = 1 *to* links(v_k) *do*
　　select output channel in node $v_k :\rightarrow v_q$
　　acording to heuristic.
　　if node v_q has not been visited yet *then*
　　　add the channel $v_k \rightarrow v_q$ to the tree.
　　　add the node v_q to the tree.
　　　mark v_q as visited.
　　　call to Depthfirst(v_q)
　　endif.
　endfor.
endwhile.
endprocedure.

(J.C.Sancho ら，[JAJ00] より)

DFS スパニングツリーは再帰的な手続きで構築されるが，手続き開始スイッチからはじめて再帰手続きから戻るスイッチまでの path(以後，メインブランチと呼ぶ) がすべてのスイッチを含むとは限らない．すべてのスイッチを含まない場合，残りのスイッチが接続されている path(以後，セカンダリブランチと呼ぶ) が存在する．

次に構築した DFS スパニングツリーを基に各物理チャネルに up 方向，もしくは down 方向を割当てる．ここでは，禁止ターン数を削減するために各スイッチにラベリングを行い，このラベルを基に方向を定める [JAJ00]．

ラベリングはメインブランチに含まれるスイッチでは DFS スパニングツリーを構築する際に訪問した順に 0 からの昇順の整数を割当てる．一方，セカンダリブランチに含まれるスイッチでは，訪問した順に降順の整数を割当てる．つまり，セカンダリブランチのリーフに対しては，そのブランチの中で最も小さい値が割当てられる．

図 3.19 に DFS スパニングツリーにおけるラベリングの例を示す．

物理チャネルの方向については $L(x)$ をスイッチ x に割当てられたラベルを返す関数とすると，$\{L(y) > L(x), x, y \in V\}$ の場合，スイッチ y からスイッチ x への物理チャネル (y, x) に down 方向，スイッチ x からスイッチ y への物理チャネル (x, y) に up 方向を割当てる．

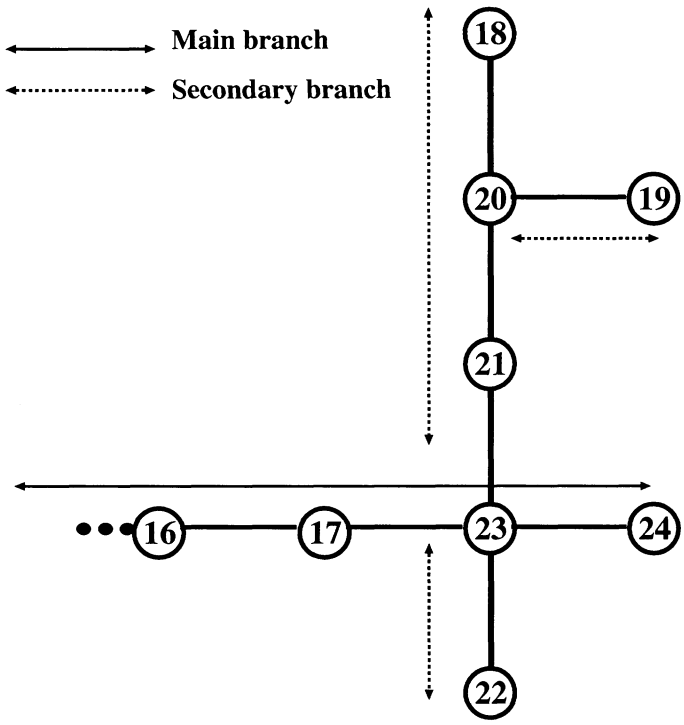


図 3.19: メインブランチとセカンダリブランチのラベリング

この方向割当てを基に Up*/Down* ルーティング を適用すると，$L(y) > L(x) < L(z)$ が成立する場合，物理チャネル (x, y),(x, z) の間の循環が除去される．これによりパケットはラベルに対し昇順で必要ホップ数転送された後，降順で必要ホップ数転送されるため，connectivity も保証される．この方法では，各スイッチに対し，一意のラベルを割当てるため，同階層物理チャネル間の冗長な禁止ターンが発生する問題は生じない．

BFS Up*/Down* ルーティング と DFS Up*/Down* ルーティング との比較を図 3.18，図 3.20 および図 3.21 に示す．図 3.20 は図 3.18 と同一ネットワークに対し DFS スパニングツリーを基にしたグラフであり，図 3.21 は各スイッチに対しラベリングを行い Up*/Down* ルーティングを適用したものである．図 3.21 では図 3.18 で発生した同階層の物理チャネル間における冗長な禁止ターンは存在しない．また，Up*/Down* ルーティングで禁止されるターン数が図 3.18 の BFS の場合 34 個 (17pair) あるのに対し，図 3.21 の DFS の場合 30 個 (15pair) である．

3.3.1.2　ヒューリスティックルールによる DFS スパニングツリー

不規則なトポロジの SAN において BFS スパニングツリーおよび DFS スパニングツリーを構築する際，物理チャネルの向きの割当てに影響を与える要因を表 3.7 に挙げる．

表 3.7 において付加リンクの選択とは，スパニングツリー構築時における親子関係を結ぶスイッチ対の選択を表す．例えば図 3.22 においてスイッチ 4 は type A においてスイッチ 2 と親子関係を結んでいるが，type B においてはスイッチ 1 と親子関係を結んでいる．

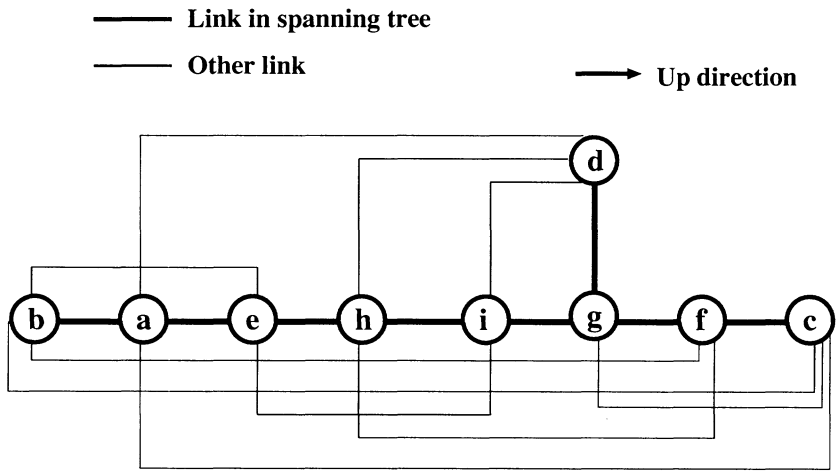


図 3.20: 図 3.18 と同一ネットワークにおける DFS スパニングツリー

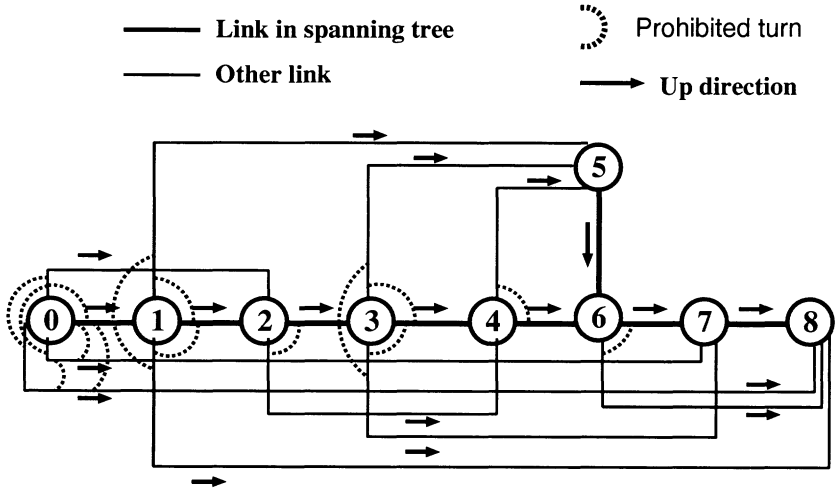


図 3.21: 図 3.20 におけるラベリングと Up*/Down* ルーティングの禁止ターン

表 3.7: 物理チャネルの向きに影響を与える要因

	BFS	DFS
付加リンクの選択	×	○
ルートの選択	○	○

BFS を基にした場合，付加リンクの選択に依らず図 3.22 のように各スイッチからルートまでの距離が一定であるため，付加リンクの選択は物理チャネルの向きに影響しない．図 3.22 において 2 つのスパニングツリーは同一トポロジに対し，同一ルートで作成されたスパニングツリーを基にした有向グラフであり，付加リンクの選択のみが違う．

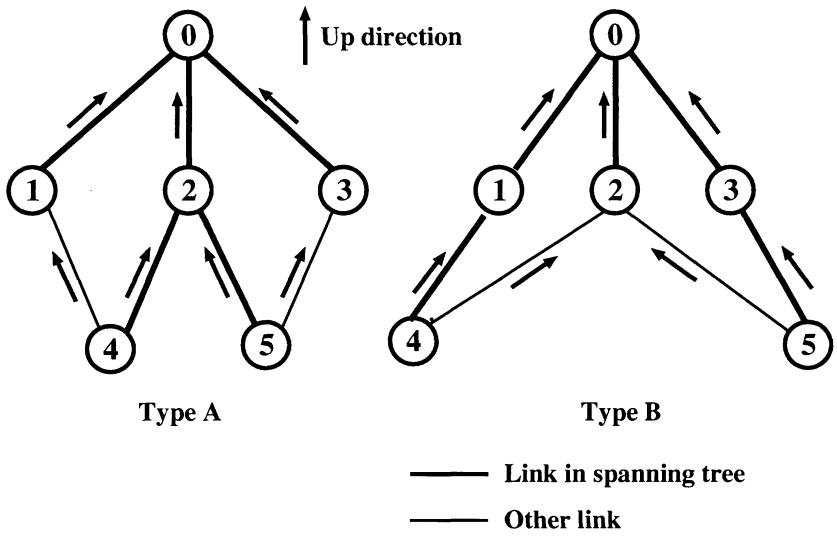


図 3.22: 異なるリンクの付加順により生成された BFS スパニングツリー

表 3.7 に示した通り，BFS スパニングツリーはヒューリステックアルゴリズムにより性能向上を狙うことができる範囲がルートの選択のみであるのに対し，DFS スパニングツリーでは付加リンクの選択，ルートの選択の 2 つである．

以後，改善の余地が大きい DFS スパニングツリーのために Sancho らにより提案されたヒューリスティックルールを説明する [JA00].

付加リンクの選択 (path heuristic): 2 つのヒューリスティックルールを次に示す．

H1 隣接スイッチの中で，残りのスイッチとの average topological distance(スイッチ間のトポロジ的な最短距離) が最も大きいスイッチを選択する．

H2 既にスパニングツリーに組み込まれているスイッチとの接続数の最も多いスイッチを選択する．また，同数の場合は H1 により選択する．

H2 の狙いは図 3.23 において，各スイッチに対して禁止ターン数が 0 である (a) や禁止ターン数が 2 である (b) のように down 方向により接続されるリンク数を増やし，禁止ターンが集中することを減らすことである．多くの場合 H2 の方が H1 に比べ高性能であることが報告されている [JA00].

ルートの選択 (root heuristic): Up*/Down* ルーティングの制約下において全スイッチ対の最短経路の平均距離を示す average distance, 1 つの物理チャネルを通過する全スイッチ対の経路数の中の最大値を示す crossing path を用いて，次のようにしてルート，つまりスパニングツリーを選択する．

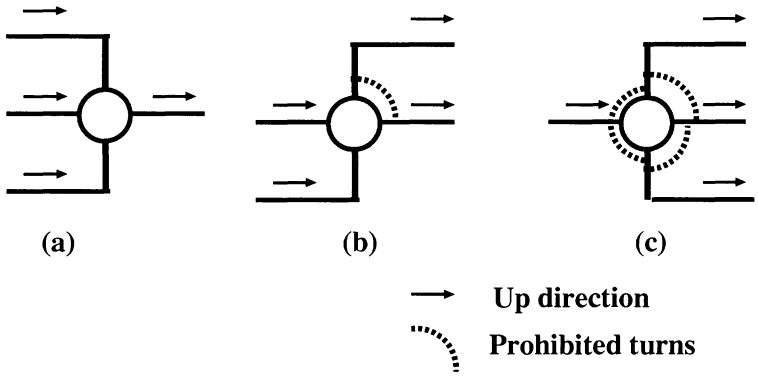


図 3.23: スイッチに接続されるリンクの方向

(a) すべてのスイッチに対して，各々をルートとした場合の有向グラフの average distance, crossing path を計算する．

(b) (a) において crossing path が最小値である有向グラフを選択する (ルートを選択する)．この際，もし，crossing path の最小値が同数の場合，average distance の最も小さい有向グラフを選択する．

このヒューリスティックルールの狙いは average distance を使うことにより多くのパケットが最短経路をとることを考慮し，かつ crossing path を使うことによりトラフィックのバランスをとることであり，これに必要な計算時間はスイッチ数を N とすると，$O(N^3)$ である．

本来，Sancho らはこのヒューリスティックルールを Myrinet などで固定ルーティングとして実装すること[3]を想定している [JPMJ02]．しかし，Up*/Down* ルーティングを基に拡張しているため，適応型アルゴリズムとしての利用は可能である．また，文献 [SRD01] にて，このヒューリスティックルールを用いた Up*/Down* ルーティングを適応型アルゴリズムとして使用した場合について議論されている．

3.3.2　複数のスパニングツリーを用いる方法

Flich らが提案した複数のスパニングツリーを用いる方法は複数の仮想チャネルを導入することにより，Up*/Down* ルーティングの問題点を解決する [JPMJ02]．

この方法は仮想チャネル番号毎に異なるルートを持つ Up*/Down* ルーティングを適用する．そして，各パケットは 1 つの番号の仮想チャネルのみを通して転送される．つまり，パケットは最も短い経路を取る Up*/Down* ルーティングの番号の仮想チャネルを利用する．この方法では，あるルートを用いた Up*/Down* ルーティングでは最短型経路を取ることができない場合に，他のルートを用いた Up*/Down* ルーティングを用いることができる．そのため，この方法はパケットの平均ホップ数を減らすことができる．この方法で

[3]第 2.7 節および第 5 章参照

は同一番号の仮想チャネル間のパケット転送が Up*/Down* ルーティングであり，異なる番号の仮想チャネル間のパケット転送が禁止されているためデッドロックフリーである．

3.3.3 LASH ルーティング

第 2.6.5 節で述べた構造化チャネル法は結合網の直径よりも多い仮想チャネル数を必要とした．これに対し，LASH ルーティングは最短経路を取るために必要な仮想チャネル数を減らすことができる [SLT02]．次に手順を示す．

まず，routing function R を 2 つの sub-function R_{phys}, R_{virt} に分割する．

- sub-function R_{phys} は各スイッチ対に対し，1 つの物理的 (トポロジ的) 最短経路を決める．
- sub-function R_{virt} は各スイッチ対の経路が使用する virtual layer (VL) を決める．つまり，各スイッチ対の経路は，1 つの VL に割当てられる．VL とは同一番号の仮想チャネルの集合のことを指す．

(1) ネットワークを各物理チャネルに対し 1 つの仮想チャネルで構成される VL の層に分割する．

(2) 全スイッチ対の最短経路を検索し，sub-function R_{phys} を得る．

(3) まだ VL が割当てられていないスイッチ対 sd を選び，スイッチ対 sd を加えても循環が生じない (他のスイッチ対が既に割当てられている) VL i を探す．そして，スイッチ対 sd を sub-function R_{virt_i} に追加する．

(4) step (3) が成立しない場合，新たに VL j を作り，sub-function R_{virt_j} にスイッチ対 sd を追加する．

(5) まだ VL が割当てられていないスイッチ対が存在すれば step (3) に戻る．

LASH ルーティングを用いる場合に必要となる仮想チャネル数については，統計的には 32 スイッチでは最大 3 本，64 スイッチでは最大 5 本および 128 スイッチでは最大 6 本あれば良いが，理論的にはスイッチ数 N に対して最悪の場合，$N/2$ であるということが報告されている [SLT02]．

また，Sancho らは LASH ルーティングと同様のアイデアを基に InfiniBand を対象としたルーティングを提案している [JAJ+02]．Sancho らの InfiniBand ルーティングは仮想ネットワーク (LASH ルーティングにおける VL と同義) 内の循環除去に Up*/Down* ルーティングを用いる点が LASH ルーティングと異なる．

3.3.4 Silla らの Minimal ルーティング

Silla らはすべての循環を除去することなしに，デッドロックフリーを実現する適応型アルゴリズムを提案した [SD00]．概要を次に示す．

まず，2本の仮想チャネル (original channel, new channel) を用意する．そして，ネットワークにパケットを注入する時には必ず new channel へ転送し，ルーティングは原則として new channel を用いて行なう．new channel の使用条件は，(トポロジ的な) 最短経路を取ることである．ただし，各スイッチにおいて選択可能な new channel がすべて使用されている場合，original channel に切り換えてルーティングを行う．original channel では既存の適応型アルゴリズムを用いてパケット転送を行う．一度 original channel を使用したパケットは二度と new channel を使用することはできない．

具体例として original channel に Up*/Down* ルーティングを用いた場合の minimal ルーティングの概要を図 3.24 に示す．

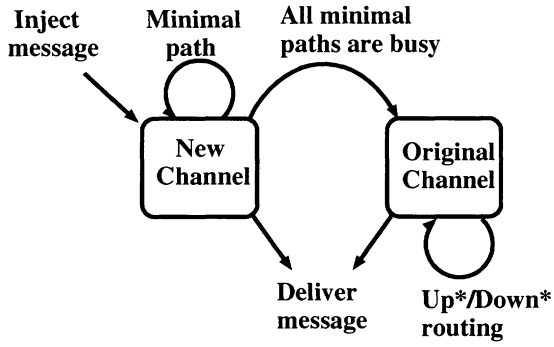


図 3.24: Up*/Down* ルーティングを用いた minimal ルーティングの概要図

original channel は，既存のルーティングアルゴリズムを用いているためデッドロックフリーが保証されており，ネットワーク全体に渡る逃げ道 (escape path) になる．

この minimal ルーティングはパケットが new channel を使用した場合最短経路をとることができるため，既存の非最短型のルーティングアルゴリズムに比べて大幅な性能向上を実現することが報告されている [SD00]．

3.3.5　In transit バッファ

Up*/Down* ルーティングは循環を除去するために down 方向から up 方向へのパケット転送を禁止した．そのため，Up*/Down* ルーティングは非最短経路の発生と禁止ターンの偏りが発生した．

そこで，Flich らは down 方向から up 方向へのパケット転送を行う場合，パケットを一旦そのスイッチに接続しているホスト PC (文献 [JPMJ02] では in transmission host と呼んでいる) に格納し，その後再注入することで最短型ルーティングを提案した [JMPJ02]．

この方法はホスト PC が専用のバッファを持つことができる場合，あらゆるトポロジの SAN に適用でき，仮想チャネル数を増やすことなく全パケットの最短経路を取ることができる．また，本来，この方法は Myrinet 上での実装を目的として提案されており，固定ルーティングとして使用する (第 2.7 節および第 5 章参照) ことを想定している [JPMJ02]．しかし，Up*/Down* ルーティングを基に拡張しているため，適応型アルゴリズムとしての利用も可能である．

3.3.6　L-turn/R-turn ルーティングとその他の解決策の比較

L-turn/R-turn ルーティングおよびその他の解決策の比較を表 3.8 に示す.

表 3.8: 適応型アルゴリズムの比較

	DFS Up*/Down*	複数のツリー	LASH	Silla らの minimal	In-transit バッファ	L-turn R-turn
トポロジフリー？	yes	yes	yes	yes	yes	yes
サイズ制限？	no	no	yes	no	no	no
最短型?	no	no	yes	almost yes	yes	no
仮想チャネルが必要？	no	yes	yes	yes	no	no
ホスト PC のバッファが必要？	no	no	no	no	yes	no

表 3.8 より，ヒューリスティックルールを用いた Up*/Down* ルーティング以外のものは仮想チャネルもしくはバッファを付加することで，パケットのホップ数を削減することを達成している．第 2.6.5 節で述べた通り，パケットのホップ数の削減は，各パケットが使用するネットワーク資源を削減するためにスループット向上につながる．また，現在，LASH ルーティングや Sancho らの InfiniBand ルーティングはトラフィックを分散するための研究が進んでいる [JAJ+02]．しかし，仮想チャネルもしくはバッファを追加できない場合，複数のスパニングツリーを用いる方法，LASH ルーティング，Silla らの minimal ルーティングおよび In transit バッファを用いることができない．つまり，複数のスパニングツリーを用いる方法，LASH ルーティング，Silla らの minimal ルーティングおよび In transit バッファはスループットを向上させるために使用用途 (対象とする SAN) を限定した手法といえる．しかし，現在，適応型ルーティングを用いることができる SAN は付加仮想チャネルおよび付加バッファが許されていないものが多く，複数のスパニングツリーを用いる方法，LASH ルーティング，Silla の minimal ルーティングおよび In transit バッファは先を見据えた研究といえる．

一方，ヒューリスティックルールを用いた Up*/Down* ルーティング，L-turn ルーティングおよび R-turn ルーティングはあらゆるトポロジ，サイズの SAN に仮想チャネルやバッファなしで適用できるという点で現状を踏まえた方法といえる．しかし，そのために最短経路を保証することができない欠点を持つ．そこで，L-turn ルーティングと R-turn ルーティングはその高いトラフィック分散能力もつことで対処しており，第 3.2 節にて L-turn ルーティングはスループット向上を達成することがわかった．

このように本節で述べたヒューリスティックルールを用いた DFS Up*/Down* ルーティング以外の解決手法は対象としている SAN が異なるため単純な性能面の比較は行っていないが，各々異なる特色を持つ．

3.3.7　Turn モデルの視点

Up*/Down* ルーティング，L-turn ルーティングおよび R-turn ルーティングについて第 3.2 節で性能面の比較を行ったが，Turn モデルの概念により論理的に比較することも可能である．

Glass と Ni により提案された Turn モデル [GN92] はメッシュなどの次元を持つ規則網における適応型アルゴリズムの設計モデルである．Turn モデルはパケットのスイッチにおけるターンの方向を解析する．詳細は付録にまとめてある．

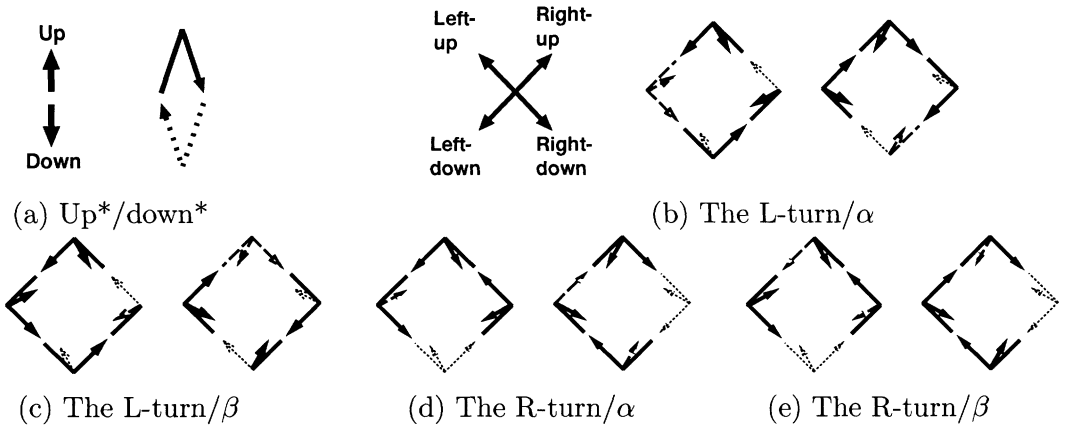


図 3.25: Up*/Down* ルーティング，および，L-turn/R-turn ルーティングの Turn モデル

Up*/Down* ルーティング，L-turn ルーティングおよび R-turn ルーティングは次元を持つ仮想的な有向グラフを構築することを利用して図 3.25 のような Turn モデルで示すことができる．

図 3.25 において薄い点線が禁止ターン，濃い点線が循環検出アルゴリズムにより場合によっては禁止されるターンを各々示す．図 3.25 より L-turn ルーティングと R-turn ルーティングはターンが細分化され，かつ，禁止ターンの分散配置を実現していることがわかる．

3.4　まとめ

本章では，デッドロック除去のために課す禁止ターンを分散させる適応型アルゴリズムである L-turn ルーティングと R-turn ルーティングを提案し，評価を行った．これらは 2 次元有向グラフである H/V グラフの構築を行う．そして，これらは H/V グラフにおけるすべての循環のパターンを列挙，解析することで，禁止ターンの分散配置を実現する．また，これらは循環検出アルゴリズムを導入することで，禁止ターン数の削減を実現する．確率モデルシミュレーションの結果，L-turn ルーティングは Up*/Down* ルーティングに比べ，不規則なトポロジの SAN において最大 17%のスループット向上，および，トーラストポロジの SAN において最大 80%のスループット向上が確認された．

L-turn ルーティングと R-turn ルーティングが SAN において高性能であることを正確に示すためには，エグゼキューションドリブンシミュレーション，命令レベルシミュレーション，そして実機での評価と進めていく必要がある．しかし，確率モデルシミュレーションの結果は，通常エグゼキューションドリブンシミュレーション [JMPJ99][SD00] や命令レベルシミュレーション [上樂 99][上樂 00] の結果と傾向が一致することがわかっており，今後，L-turn ルーティングの有効性がこれらの評価を通して確固たるものにできると考えられる．L-turn ルーティングは付加仮想チャネルなしにすべてのトポロジの SAN に適用できる点で，今後の基盤となりうるルーティングである．

第4章　適応型ルーティングにおける OSF

第 2.6.4 節で述べたように適応型ルーティングは適応型アルゴリズムと出力選択機構 (output selection function: OSF) により構成される．そのため，適応型ルーティングのスループットを向上させるためには，適応型アルゴリズムと同様に OSF についても基礎的な技術を確立させる必要がある．

大規模並列計算機および SAN で用いられる規則網における適応型アルゴリズムは，これまでに物理チャネルおよび仮想チャネルの動的な切り換えを行うことができる自由度の高いものが提案されてきた (個々の適応型アルゴリズムについては付録にまとめた)．また，現在でも適応型アルゴリズムに関する議論は幅広く行われている．しかし，OSF は注目度が低く，その役割であるトラフィックの分散を仮想チャネル間，物理チャネル間の両方で実現するものがないのが現状である．

例えばメッシュなどの次元を持つトポロジを対象にした次元順選択機構やあらゆるトポロジに適用することができるランダム選択機構はトラフィックの状態を全く反映せずに，次元順およびランダムに転送しようとする．そのため，これらは混雑している方向にパケットを転送してしまう確率を下げることが難しく，性能がトラフィックパターンやトポロジに依存してしまう問題がある [L.S00][WK97]．一方，トラフィックの状態を把握する機能を持たせたものもある．LFU 選択機構は最近一定期間で最も使われていない仮想チャネルを選ぶため，仮想チャネル間においてトラフィックを分散させることができる [JFPJ00]．しかし，LFU 選択機構は物理チャネルの使用状況に偏りが生じる可能性がある．これは，仮想チャネルの目的が物理チャネルを効率的に使用することであることと矛盾する．逆に MM 選択機構は Silla らの minimal ルーティングのために提案されたため，仮想チャネル間の切り換えを頻繁に行う適応型アルゴリズムにおける性能向上は難しい [FJ00][JFPJ00]．

よって，自由度の高い適応型アルゴリズムが提案された現在，OSF を改良することが SAN を含む相互結合網の性能向上の鍵となる可能性がある．

そこで，本章では OSF を出力する物理チャネルの選択とその物理チャネル内での出力仮想チャネルの選択という 2 つの選択ステップに分けることにより効率的にトラフィックを分散する load-dependent 選択機構 (LDSF), LRU 選択機構，および minimal multiplexed and least recently used (MMLRU) 選択機構を提案する．これらは既存の OSF と同様に単純さを維持しつつ，物理チャネル間，仮想チャネル間の両方においてトラフィックの分散を狙う点が特長である．この 3 つの OSF はトラフィックを分散させるためのアプローチは異なるが，いずれもトポロジ，トラフィックパターンおよび適応型アルゴリズムに依存しない．LDSF，LRU 選択機構および MMLRU 選択機構では各物理チャネル毎にそのチャネルの利用状況を反映するカウンタをおく．そして，各スイッチは自スイッチ内の物理チャネル，仮想チャネルの利用状況によりトラフィックの混雑を判断し，混雑を迂回するように出力物理チャネルおよび仮想チャネルを選択する．

以降，第 4.1 節にて LDSF, 第 4.2 節にて LRU 選択機構，第 4.3 節にて MMLRU 選択機構を各々提案する．また，第 4.4 節で評価を行い，第 4.5 節で結論を述べる．

4.1 LDSF

4.1.1 LDSF の概要

本節では出力する物理チャネルの選択とその物理チャネル内での出力仮想チャネルの選択という2つの選択ステップに分けることにより効率的にトラフィックを分散する LDSF を提案する (図 4.1).

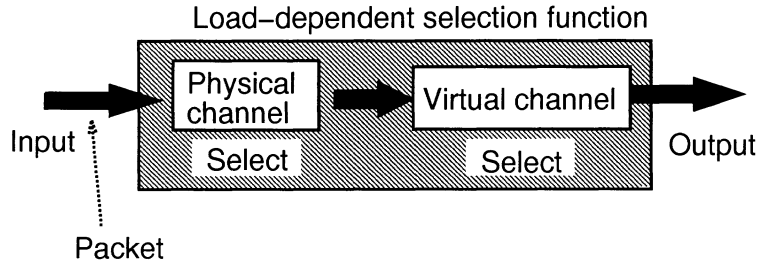


図 4.1: LDSF の概要

現在，相互結合網のスループットを上げるために，スイッチはよりシンプルで高クロックに耐えられるものが必要である．実装されている SAN や並列計算機の相互結合網で WH 方式が多く利用されているのはその典型である [Oed93][ST96] ．LDSF は実装上の容易さを考慮しており，スイッチ内の各方向の物理チャネルに対して1つカウンタを用意する事により実現している．また，LDSF はトポロジを選ばないため，すべての結合網に対して適用できる．

LDSF の概要を次に示す．

(a) スイッチ内の各物理チャネルの出力毎にカウンタを1つ設け，値0で初期化する．

(b) フリットが物理チャネルを通過する毎にそのカウンタを1増加させ，フリットの出力要求がないクロック毎にカウンタを1減算する．

(c) ルーティングの際，出力物理チャネルが2つ以上ある場合，各出力可能な物理チャネルのカウンタを比べ，値が一番小さい出力物理チャネルを選択する．そして，その中で適応型アルゴリズムによる制限の最も厳しい仮想チャネルを選択する．

LDSF は (1) 以前パケットを送出した物理チャネルをなるべく利用しない，かつ，(2) 一定時間経過したルーティング情報は消滅する，という特徴を持つ．

物理チャネルの選択および仮想チャネルの選択について次に詳しく述べる．

4.1.2 物理チャネルの選択

物理チャネルの選択を行う選択機構においてトラフィックを分散させるためには，混雑している地域を特定し，把握する必要がある．実装を想定すると結合網中のすべてのスイッチにおける混雑状況をクロック毎にどこかで集計して把握することは不可能である．また，隣接スイッチの混雑情報のみを把握する方法も，(1) 実装が複雑になる，(2) ネット

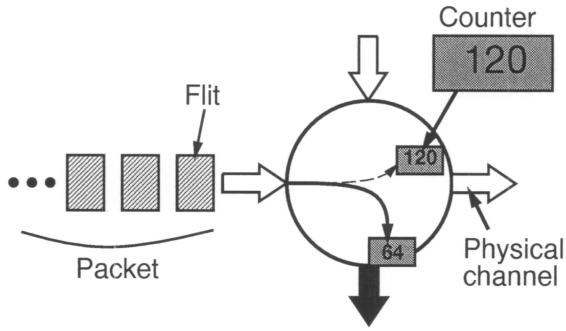


図 4.2: 2次元トーラスにおける LDSF(2つの出力物理チャネルが選択可能な場合)

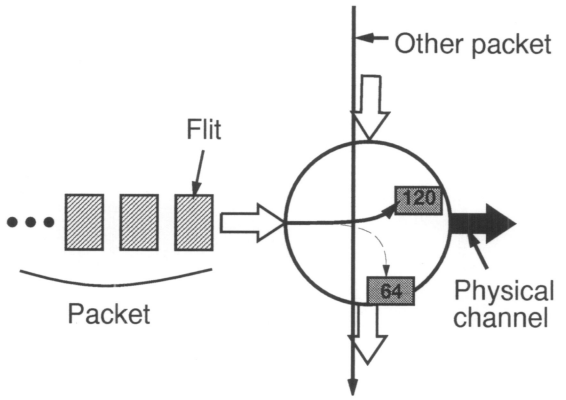


図 4.3: 2次元トーラスにおける LDSF(片方の出力物理チャネルが塞がっている場合)

ワーク資源を消費してしまう，などの問題がある．そこで LDSF では各スイッチが各自の物理チャネルをいつ利用したかという情報をある一定期間保存することでトラフィックの混雑状況をある程度把握する．

このために，LDSF ではスイッチ内の各物理チャネルに値0に初期化されたカウンタを1つ設け，フリットが通過する毎にカウントアップする．つまり，あるパケットが到着した時にカウンタの値が0であった場合，そのパケットの一番最後のフリットの転送が終了した時点でのカウンタの値はそのパケット長となる．また，クロック毎にフリットの出力要求がない方向のカウンタは0でなければ，デクリメントを行う．そして，ルーティングの際，出力可能な方向が2つ以上ある場合，各出力可能な物理チャネルのカウンタを比べ，値が一番小さい出力方向を選択しパケットを転送する．LDSF の例として2次元トーラスの場合を図4.2および図4.3に示す．図4.2ではカウンタ値が120と64でどちらも選択可能なので64の出力方向を選択している．なお，片方の出力物理チャネルが既に他のパケットにより塞がっている場合は，当然，利用可能なもう一方の出力物理チャネルを選択する(図4.3)．

また，LDSF は WH 方式においてブロックされたパケットが同時に複数の物理チャネルを占有し，停滞した場合，カウンタ操作を行わない．従ってこのパケットが存在するス

イッチでその混雑情報を保持することができる．

パケット長が結合網の直径に比べて十分に大きい場合，通過フリット数を通してネットワークの状況を把握する方式は一つの有効な方法である．

4.1.3　仮想チャネルの選択

仮想チャネルの選択を行う選択ステップではすべての仮想チャネルを効果的に使うことを目的とする．そのため，後者の選択ステップでは同一物理チャネル内の複数の仮想チャネルが選択可能な場合，その中で一番制限の厳しい仮想チャネルを選択する．ただし，ここでの制限とは，デッドロック除去のために課されたパケットの転送制限のことを指す．これは，使用条件の厳しい仮想チャネルは制限の緩い仮想チャネルに比べ利用率が低い場合が多いことを考慮している．もし，制限の緩い仮想チャネルの利用を優先した場合，どのようなパケットも制限の緩い仮想チャネルを利用することになり，制限の緩い仮想チャネルしか利用できないパケットの転送効率が下がってしまう．そこで，LDSF は使用制限の緩い仮想チャネルを残すことにより他の一部のパケットがその仮想チャネルを選択可能になり，その結果全体でのパケットの選択可能な経路数が増える．なお，物理チャネル内のすべての仮想チャネルの使用条件が同一の場合，選択方法に関わらずルーティングの結果は同じになるので，検討を行なう必要がない．

4.2 LRU 選択機構

LDSF はトラフィックの混雑状況に応じて出力物理チャネルとその中の出力仮想チャネルを選択できる点で従来の方法と異なっている．しかし，パケット長が大きい場合，LDSF の通過フリット数を記録するカウンタは大容量なものになってしまい，ハードウェア量が無視できないレベルになる可能性がある．

そこで本章では LDSF の機能を大幅に削減することによりハードウェア量を抑え，なおかつ性能をほぼ維持することができる OSF である LRU(Least Recently Used) 選択機構を提案する．

LRU 選択機構は LDSF と同様に物理チャネルの選択と仮想チャネルの選択の 2 つの選択機構に分ける．前者の選択機構では各スイッチにおいて使用されずにいた時間の最も長い出力物理チャネルを選択する．そして，後者の選択機構では LDSF と同様に適応型アルゴリズムによる制限の最も厳しい仮想チャネルを選択する．

LRU 選択機構は利用されていない物理チャネルを優先的に選択するため，LDSF と同様にトラフィックパターンに因らずトラフィックを分散させる効果が期待できる．

LDSF では各出力物理チャネルを一定時間内に通過したフリット数に応じて出力チャネルを選択したのに対し，LRU 選択機構は各出力物理チャネルを最後に通過したパケットの経過時間のみに応じて出力物理チャネルを選択する．

LDSF と LRU 選択機構の主な違いは次の 2 点である．

- LDSF は LRU 選択機構と違い，パケット長を考慮して出力チャネルを選択する．
- LDSF はパケットがブロックされて停滞した場合の混雑情報が記録されない (カウンタ値が変わらない)．これに対して LRU 選択機構では停滞している方向の優先度を最も低く設定する．

LRU 選択機構のアイデアは仮想記憶におけるページの追い出しアルゴリズムである LRU(Lest Recently Used) と同様である．LRU はメモリアクセスの局所性を最大限利用できるためヒット率を上げることができる．これに対して LRU 選択機構はトラフィックの局所性をできるだけ排除し，分散させるため，使用されずにいた時間の最も長い出力物理チャネルを優先する．

4.3 MMLRU 選択機構

LDSF および LRU 選択機構は物理チャネル間のトラフィックの分散を実現するために，物理チャネル毎のトラフィックの統計情報を利用した．

一方 MMLRU 選択機構では MM 選択機構と同様に仮想チャネルの使用状況が物理チャネルの転送遅延を決定する点に着目する．

4.3.1 時分割で共有するパケット数削減の重要性

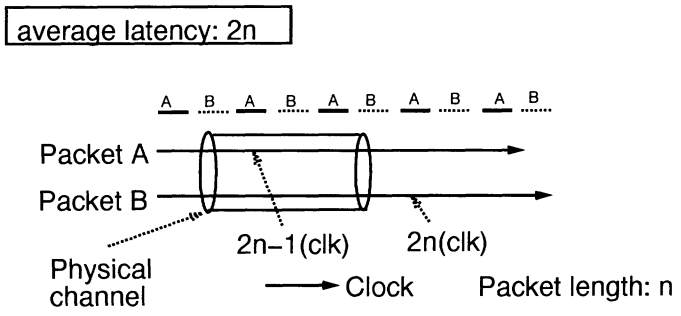


図 4.4: ラウンドロビンによる仮想チャネルフロー制御

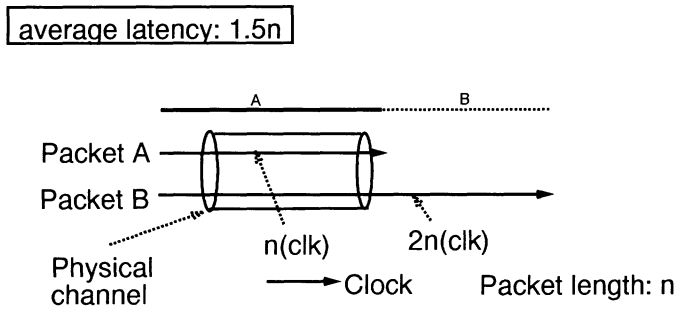


図 4.5: パケット単位の仮想チャネルフロー制御

仮想チャネルは多数のパケットが物理チャネルを時分割で共有している場合，各パケットが物理チャネルの通過に大きなレイテンシを伴うという問題を持つ．そのため，特定の物理チャネルにパケットが集中してしまうと，その物理チャネルを中心に局所的な混雑が発生し，ひいては結合網全体の性能に悪影響を及ぼす可能性がある．従って物理チャネルを時分割で共有するパケット数をできる限り削減してパケットのレイテンシを抑えることが，結合網の性能を引き出す一つのポイントになるということがいえる．物理チャネル移動時におけるパケットの転送遅延は仮想チャネルフロー制御の影響をうける．仮想チャネルフロー制御には代表的なものとして次の 2 つがある．

- フリット毎にラウンドロビンを行う．

- 物理チャネルを獲得しているパケットがブロックされるまで使い続ける.

前者の仮想チャネルフロー制御はある物理チャネルに長短2つのパケットが(仮想チャネル上に)存在する場合，短いパケットが長いパケットの通過を待ち，遅延が大きくなることを防ぐ効果がある．しかし，サイズが等しい2つのパケットが同時に同じ物理チャネルに転送された場合には，物理チャネルの通過に最低2倍の時間がかかってしまう (図 4.4).

後者は Network of Workstations(NOWs) での slack buffer[CW93] や credit based flow control[NKN+01] などで用いられる．後者の仮想チャネルフロー制御ではサイズが等しい2つのパケットを同時に同一の物理チャネルに転送した場合には，物理チャネルの通過にかかる時間は平均 1.5 倍程度で済む (図 4.5)．このように，仮想チャネルフロー制御がパケットのレイテンシに与える影響は少なくない．しかし，いずれの仮想チャネルフロー制御も，2つ以上のパケットが同一の物理チャネルに転送された場合には，大幅な遅延の発生を防ぐことはできない.

この検討結果より，仮想チャネルを用いる結合網では，OSF を用いて各物理チャネルにおいて同時に使用される仮想チャネル数を減らすことが，パケットの流れをスムーズにし，結合網の性能を引き出すために重要であるといえる.

4.3.2　MMLRU 選択機構アルゴリズム

MMLRU 選択機構は LDSF および LRU 選択機構と同様に次の2つの選択ステップにわけた構成をとる.

(1) 出力物理チャネルの選択

(2) 出力物理チャネルにおける仮想チャネルの選択

前者の選択ステップでは，物理チャネルを時分割で共有するパケット数の削減と物理チャネル間におけるトラフィックを分散させる役割を担う．一方，後者の選択ステップ では物理チャネルを時分割で共有するパケット数には影響せず，仮想チャネル間においてトラフィックを分散させる役割のみを担う.

MMLRU 選択機構は前者の選択ステップにおいて選択可能な物理チャネルの中で，空いている仮想チャネル数の最も多いものを選択する．選択可能な出力物理チャネルにおける空き仮想チャネル数が同じ場合には，パケットのヘッダが通過した最近の時刻が最も古い物理チャネルを選択する (パケットのヘッダを基にした LRU(Least Recently Used) ポリシ)．パケットのヘッダを基にした LRU の狙いは，物理チャネル内でパケットがマルチプレクスされる時間をできる限り短くし，レイテンシを抑えることである.

また，MMLRU 選択機構は後者の選択ステップにおいて LDSF および LRU 選択機構と同様に同一物理チャネル内の複数の仮想チャネルが選択可能な場合，その中で適応型アルゴリズムによる制限の最も厳しい仮想チャネルを選択する.

4.4　評価

次元順, ランダム, ジグザグ, LRU, LDSF および MMLRU 選択機構についてフリットレベルシミュレータにより評価を行った.

4.4.1　シミュレーション条件

フリットレベルシミュレータは第3.2節で用いたものを基にしたものである.
シミュレーションに用いた条件を表4.1に示す.

表 4.1: OSF のシミュレーションパラメータ

適応型アルゴリズム	Duato's protocol
実行時間	50,000 クロック (初めの 5,000 クロックは無視)
トポロジ	2D トーラス, もしくは 3D トーラス (付録参照)
サイズ	16×16 (256 スイッチ), 32×32 (1024 スイッチ) もしくは $8 \times 8 \times 8$ (512 スイッチ)
仮想チャネル数	3
パケット長	128 フリット
パケット転送方式	WH 方式
フリット転送時間	3 クロック

適応型アルゴリズムとしては規則的なトポロジにおいて自由度が高い Duato's protocol [Dua95] を用いた. Duato's protocol はネットワーク全体に渡る循環を含まない逃げ道 (escape path) を用意することで, 循環を含む経路のデッドロックを除去する手法である. この理論は複雑であるため, 詳細を付録に記した. なお, Duato's protocol はトーラスに適用する場合, 仮想チャネルが3本必要となる.

トポロジとして不規則なトポロジではなく, トーラスを用いる理由は次の2つである: (1) 次元を持つトポロジを対象としたジグザグ選択機構や次元順選択機構との比較を行うことができるため, MMLRU 選択機構の性能をより客観的に判断することができる: (2) これまでにトーラスにおける適応型アルゴリズムの研究は進んでおり, 適応型アルゴリズムの自由度をほぼ限界まで引きあげることができている. 一方で, OSF に関する評価はこれまでにあまり行われていないため, 基本的な結合網であるトーラスにおいて OSF が性能にどの程度性能に影響を与えるのかわかっていない. そこで, まず諸性質が分かっているトーラスにおける OSF の性能を調べ, 基本的な効果を把握する必要がある.

パケットの目的地 PC は次のトラフィックパターンを用いた.

- uniform
  すべての目的地はランダムに決定され, 均一に分散される.

- bit reversal
  まず，各 PC に 0 から $(n-1)$ (ただし，n は PC 数) までの一意の 2 進の番号を割当てる．
  そして，2 進の番号 $(a_0, a_1, \cdots, a_{n-2}, a_{n-1})$ を持つ PC は自分の番号のビット列を逆順に並べた番号 $(a_{n-1}, a_{n-2}, \cdots, a_1, a_0)$ の PC へパケットを送る．

シミュレーションで初めの 5,000 クロックはネットワークが安定せず，想定した負荷に達していないと考えられるため評価の対象外とした．また，次の指標について評価を行った．

ネットワークレイテンシ

ある PC p がパケットの最初のフリットを NIC の入力バッファに挿入した時刻を t_0，目的地の PC q の NIC がパケットの最後のフリットを受け取った時刻を t_1 とする．ここで，$T_{lat}(p,q) = t_1 - t_0$ をネットワークレイテンシと呼び，ネットワークの性能を測る指標とする．

スループット

スループットは，全 PC が毎クロックに 1 フリット受信する場合を 1.00 としており，受信トラフィックの最大値を示している [JSL02]．

4.4.2　評価結果

4.4.2.1　Uniform traffic

uniform traffic での評価を図 4.6 に示す．横軸は受信トラフィック，縦軸はレイテンシを表している．

uniform traffic ではトラフィック自体が分散されており OSF 間の性能差が表れにくい状況にも関わらず，図 4.6 の各図より MMLRU 選択機構は既存のものに比べ若干レイテンシを抑えていることがわかる．これは MMLRU 選択機構が物理チャネルを時分割で共有するパケットの平均数を抑えたことに起因すると考えられる．また，図 4.6 より $8\times8\times8$ 3D トーラスにおいて MMLRU 選択機構および LRU 選択機構は低レイテンシのみならず，高スループットを達成していることがわかる．

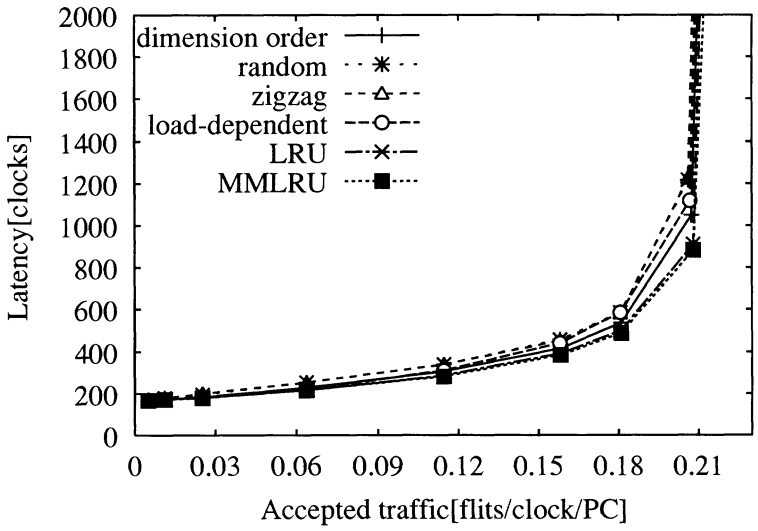
4.4.2.2　Bit reversal traffic

次に，bit reversal traffic での評価を図 4.7 に示す．図 4.7 より，次元順選択機構が最もスループットが低く，一方で，MMLRU 選択機構が最も高い性能を示していることがわかる．同じ Duato's protocol を用いているにも関わらず，OSF による性能差が uniform traffic に比べ著しいが，これは bit reversal traffic のような小規模な偏りが多数発生するトラフィックパターンでは混雑に応じてパケットを分散することができる OSF が効果的であることを示している．例えば図 4.7 において $8\times8\times8$ 3D トーラスでは提案した 3 つ

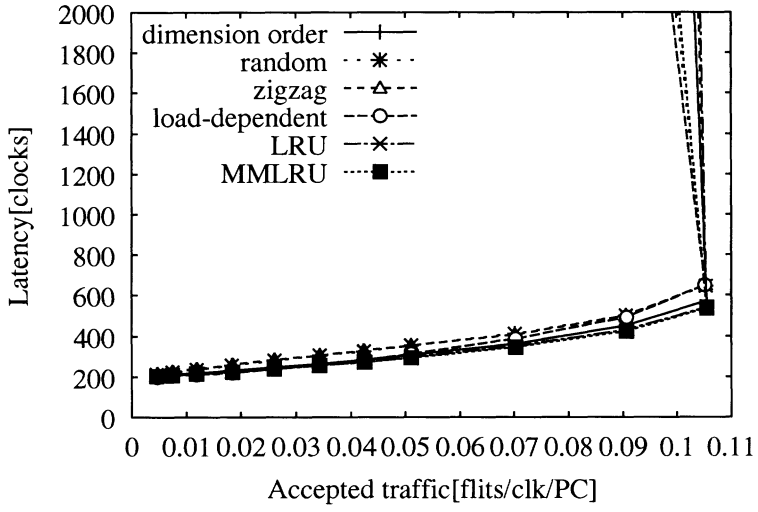
の OSF は既存の OSF に比べるとスループットを大幅に改善していることがわかる．また，図 4.7 より，各次元のスイッチ数が少ないほど，またトーラスの次元数が増えるほど OSF の性能差が大きくなる．これは，次元数が増えることにより迂回経路が増え，また，各次元のスイッチ数が少ないほど，1 ホップあたりの経路数が相対的に増えるため，OSF の影響が大きくなるためと考えられる．

また，選択可能な経路数を最大にするジグザグ選択機構が最適であるという理論的な報告 [BP89][Wu99] もあるが，動的に経路選択を行う MMLRU 選択機構の方が高いスループットを達成することがわかった．

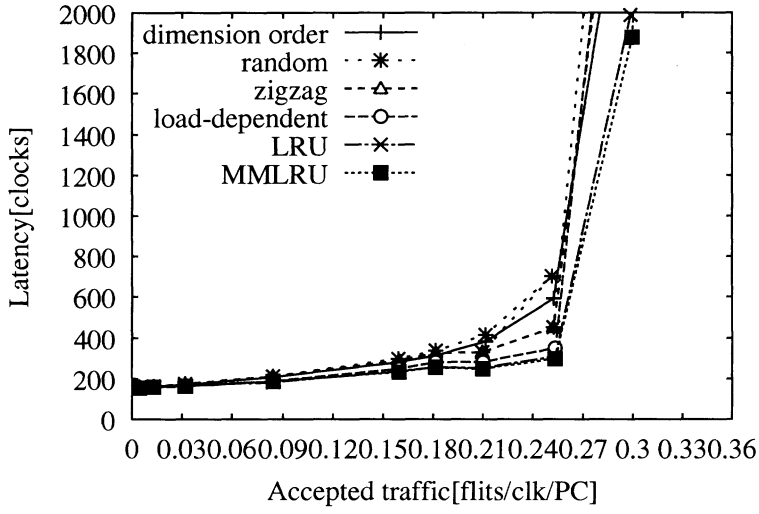
最後に図 4.6 と図 4.7 を比較すると一部の OSF は uniform traffic の場合に比べて bit reversal traffic の方が高い性能を示していることがわかる．bit reversal traffic は各スイッチから発生したパケットは各々異なる目的地であるため PC の consumption channel において衝突が発生しない．一方，uniform traffic では異なる出発地のパケットが同一の目的地である場合があるため，PC の consumption channel において衝突が発生する．そのため，この衝突が unifotm traffic における各 OSF の性能を落としていると考えられる．



(a) 16x16 2D トーラス

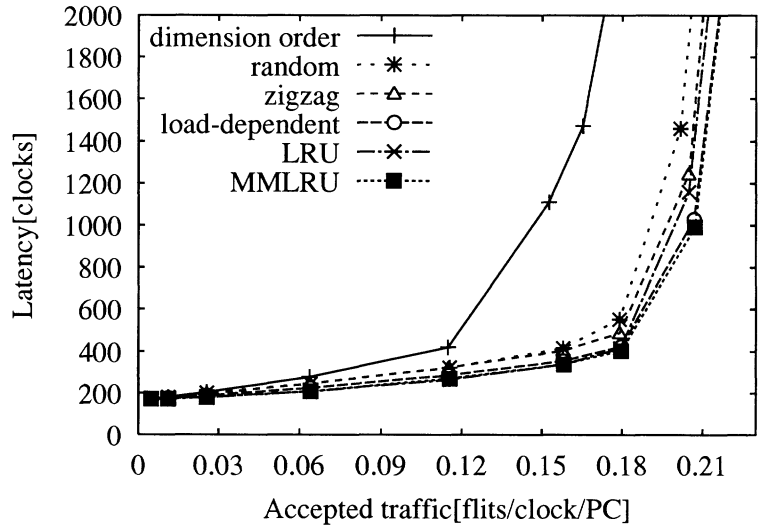


(b) 32x32 2D トーラス

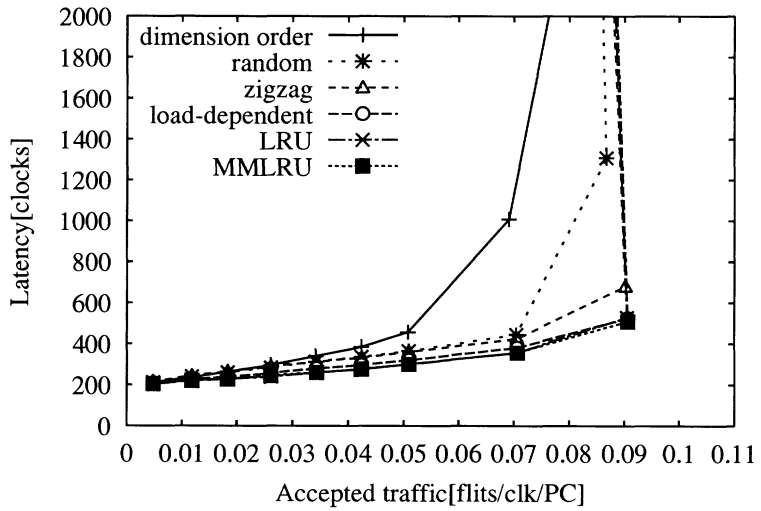


(c) 8x8x8 3D トーラス

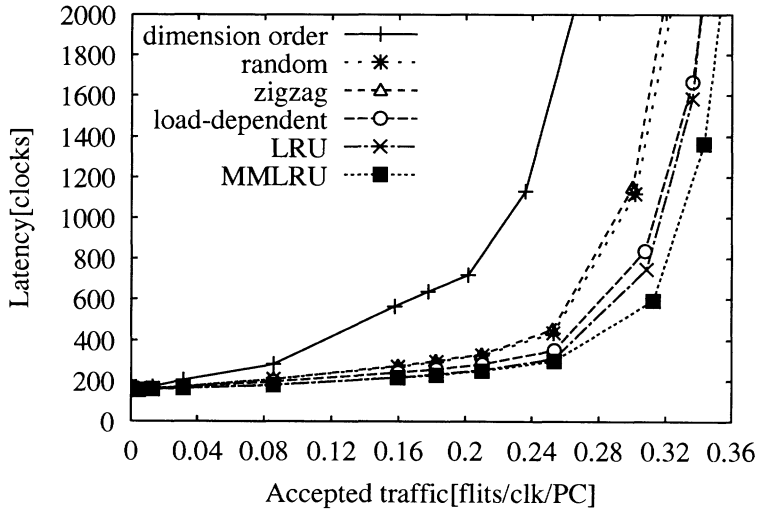
図 4.6: Uniform traffic におけるスループットとレイテンシ



(a) 16x16 2D トーラス



(b) 32x32 2D トーラス



(c) 8x8x8 3D トーラス

図 4.7: Bit reversal traffic におけるスループットとレイテンシ

4.5　まとめ

トラフィックの混雑状況を反映して出力物理チャネルと出力仮想チャネルを選択する OSF (LDSF, LRU 選択機構および MMLRU 選択機構) を提案し，評価を行った．

この3つの OSF は出力物理チャネルの選択と出力仮想チャネルの選択の2つの選択ステップにわける点が特徴である．この3つの OSF は各物理チャネルにカウンタを用意する必要がある．LDSF では物理チャネルの通過フリットをカウンタに記録し，その物理チャネルの利用状況を把握する．一方，LRU 選択機構はパケット単位で使用されずにいた時間の最も長い物理チャネルをカウンタで把握する．また，MMLRU 選択機構はこれらとは異なり，各物理チャネルを利用しているパケット数を減らすことでリンク遅延を抑える．MMLRU 選択機構では同数のパケットを持つ物理チャネル間の選択をパケット単位の LRU で行うため，少量のカウンタが必要となる．

出力仮想チャネルの選択では，3つの OSF とも適応型アルゴリズムによる制限が最も厳しいものを優先する．

これら3つの OSF はトラフィックを分散するためのアプローチは異なるが，適応型アルゴリズム，トラフィックパターンおよびトポロジに依存しない単純な手法である．シミュレーションの結果より，3つの手法はトラフィックパターン，トポロジにおいて従来の OSF より安定した性能を実現することがわかった．また，選択可能な経路数を最大にするジグザグ選択機構が最適であるという理論的な研究 [BP89][Wu99] もあるが，実際にはトラフィックの分散を動的に行う OSF の方が効果的であることがわかった．

OSF に関する研究は世界的にあまり進んでおらず，OSF が各結合網のパフォーマンスにどの程度影響を与えるかという研究はあまり行われていない．また，決め手となる OSF も未だ提案されていない．今後，適応型アルゴリズムと同様に OSF の研究は進展していくと考えられる．この際，本研究がその基礎となるものと期待している．

第5章　不規則なトポロジの SAN における仮想チャネルを用いた固定ルーティング

第3章および第4章にて適応型ルーティングに関する技術を提案した．リンクのバンド幅の使用効率を重視する場合，適応型ルーティングは効果的な手法であるが，スイッチには動的に空いているチャネル—物理，仮想の両方を含む— を選択する機能が必要となる．

本章では，この選択機能を省いてスイッチの高速動作を実現する固定ルーティングについて扱う．固定ルーティングはパケットの経路が出発地と目的地の組により一意に定まる．そのため，固定ルーティングは (1) パケット配送エラーの検出が容易であり，かつ，(2) 目的地におけるパケットのソートなしにパケット配達の FIFO 性を保証することができる，という利点を持つため近年再び注目を浴びている．しかし，固定ルーティングは適応型ルーティングに比べ，同速度で動作するスイッチを用いた場合，一般的に物理チャネルの利用率が劣ることがわかっている [DA93][JSL02]．そこで，最近の固定ルーティングを採用した SAN ではこの欠点を補うため，仮想チャネルを用いている [I.T01][PFH01][STH+00][NKN+01]．

昨年度策定された InfiniBand の規格 [I.T01] では最大 15 本の仮想チャネル[1]と固定ルーティングを定めている．また，QsNET[PFH01] は仮想チャネル 2 本を持ち，固定ルーティングを採用している．一方，オフィスなどに分散配置された PC を使った並列分散システムである RHiNET スイッチ 3[NKN+01] では最大 64 本の仮想チャネルを持っている．しかし，あらゆるトポロジの SAN に適用できるデッドロックフリー固定ルーティングを開発することは難しく，現状では，仮想チャネルを想定していない Up*/Down* ルーティングや多数の仮想チャネルが必要となる構造化チャネル法を基にした手法[2]に限られており満足な性能が得られていない．

そのため，トポロジを限定することで性能の高い SAN を提供する方法も提案されている．QsNet[PFH01] では，fat ツリートポロジを利用することにより，ある程度の柔軟性—ツリーのルート方向へのリンク数 p，ツリーのリーフ方向へのリンク数 q，及び階層数 r の組 (p, q, r) の設定— と効果的な固定ルーティング [PFH01] を両立している．

本章ではネットワークを仮想チャネルを用いて同一トポロジのサブネットワークの層に分割するアイデア [鯉渕 02b] [鯉渕 02c] [鯉渕 02a] [MAH02a] を提案し，仮想チャネル数によらず，すべての SAN に適用することができるデッドロックフリー固定ルーティングである descending layers (DL) ルーティングを提案する．

---

[1]規格 [I.T01] では仮想レーンと呼んでいる．

[2]これらは本来適応型ルーティングであるが，各スイッチ間の経路を 1 つに選択するアルゴリズム-以後，われわれは経路選択アルゴリズム (path selection algorithm) と呼ぶ- により固定ルーティングとして実装することができる．

DL ルーティングは

(a) サブネットワークの生成

(b) サブネットワーク内のデッドロックの除去及びサブネットワーク間のデッドロックの除去の設定

(c) 経路の生成

の3段階により構成される．(b) ではデッドロックフリーを満たす経路の候補を求め，(c) では，各スイッチ対の経路を1つに決定する経路選択アルゴリズムを定める．

以降，第5.1節にて DL ルーティングを提案し，第5.2節で評価を行う．そして，第5.3節にて同時並行的に進んでいる他の研究との比較検討を行う．最後に第5.4節で結論を述べる．

5.1　DL ルーティング

descending layers (DL) ルーティングは仮想チャネルをスループット向上に利用する点で Up*/Down* ルーティングとは異なる．

5.1.1　DL ルーティングの構成

DL ルーティングは次の 3 つの手続きにより定まる．

(1) サブネットワークの生成

(2) デッドロックの除去

(3) 経路の生成

5.1.1.1　サブネットワークの生成

まず，ネットワークを仮想チャネルを用いて 0 から $(r-1)$ の一意の整数番号がついた r 個のサブネットワークの層に分割する．各サブネットワークはネットワークのトポロジを物理的に分割したものではなく，対象とするネットワークと同一トポロジの仮想的なネットワークを指す．ただし，r は仮想チャネル数である．

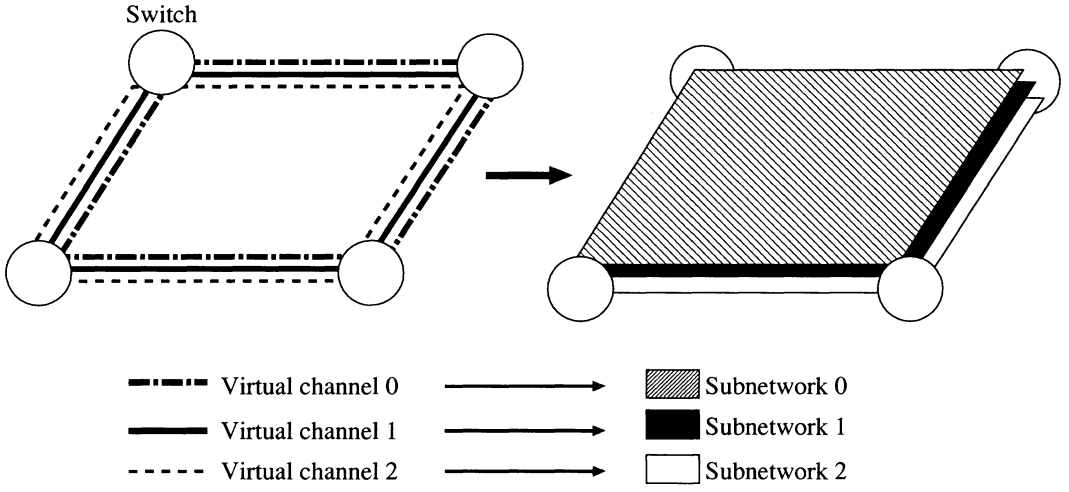


図 5.1: サブネットワークの構成例 (仮想チャネル数が 3 本の場合)

図 5.1 は仮想チャネルが 3 本の場合のサブネットワークの構成例である．この例のように仮想チャネル番号毎に異なるサブネットワークを割当てるため，複数のサブネットワークが 1 つの仮想チャネルを共有することはない．

5.1.1.2　デッドロックの除去

次にサブネットワーク内，及びサブネットワーク間のパケット転送制限を次のように課すことでデッドロックを除去する．

サブネットワーク内のデッドロックの除去 各サブネットワーク内において，パケットの循環依存が起きないようにパケット転送制限を課す．特に，番号0のサブネットワーク内では任意のスイッチ対の経路が必ず存在するように転送制限を課す．

サブネットワーク間のデッドロックの除去 サブネットワーク $i(0 < i < r)$ 内において転送が禁止されている方向へパケット転送をする場合，サブネットワーク i からサブネットワーク $(i-1)$ へ切り換える．この他のサブネットワーク間の切り換えはすべて禁止する．

5.1.1.3　経路の生成

各スイッチ対の経路は前述のデッドロック除去のための制限を満たす経路候補の中から最短経路を1つに決定することで決定する．つまり，ここでは経路選択アルゴリズムにより固定ルーティングを生成する．

5.1.2　実装アルゴリズム

前節にて DL ルーティングの構成を述べたが，DL ルーティングを実装する場合，その手順において

(a) 各サブネットワーク内のデッドロック除去，および，

(b) 経路の生成 (経路選択アルゴリズム)

の2段階については，何らかのアルゴリズムを定める必要がある．そこで，本節ではこの実装アルゴリズムについて述べる．

5.1.2.1　サブネットワーク内のデッドロック除去

サブネットワーク内のデッドロックフリーを実現する方法として複雑な計算やアルゴリズムを避け，Up*/Down* ルーティングを基に考える．つまり，各仮想チャネルに割当てられた up もしくは down の方向を基にしてデッドロック除去を考える．

UD* サブネットワーク内のデッドロック除去を行うためのパケット転送制限として簡単なものは Up*/Down* ルーティングをすべてのサブネットワーク内で用いることである．この各サブネットワーク内に Up*/Down* ルーティングを用いるアルゴリズムを UD* と呼ぶ[3]．

[3]名称はすべてのサブネットワーク内において up channel から down channel の転送制限がないことによる

UD* を用いた DL ルーティングでは，同一サブネットワーク内において down 方向から up 方向へのパケット転送を設定できない．しかし，この方法では，小さい番号のサブネットワークに切り換えることによりサブネットワーク間において down 方向から up 方向へのパケット転送を実現する．

例えば図5.2においてスイッチ**7**からスイッチ**13**へパケットを転送する場合，Up*/Down* ルーティングでは，仮想チャネル数に関係なく，7ホップ (**7**→**4**→**1**→**0**→**3**→**6**→**10**→**13**) 必要になる．一方，UD* を用いた DL ルーティングでは仮想チャネルが2本 (サブネットワーク $sn.0, sn.1$ に各々属するものとする) の場合，サブネットワーク番号を1回減少させることにより5ホップ (**7**→ ($sn.1$) →**11**→ ($sn.0$) →**9**→ ($sn.0$) → **6** → ($sn.0$) → **10** → ($sn.0$) →**13**) で到達することができる．さらに，仮想チャネルが3本 (サブネットワーク $sn.0, sn.1, sn.2$ に各々属するものとする) の場合，サブネットワーク番号を2回減少させることにより4ホップ (**7**→ ($sn.2$) →**11**→ ($sn.1$) →**9**→ ($sn.1$) → **12**→ ($sn.0$) →**13**) で到達可能である．

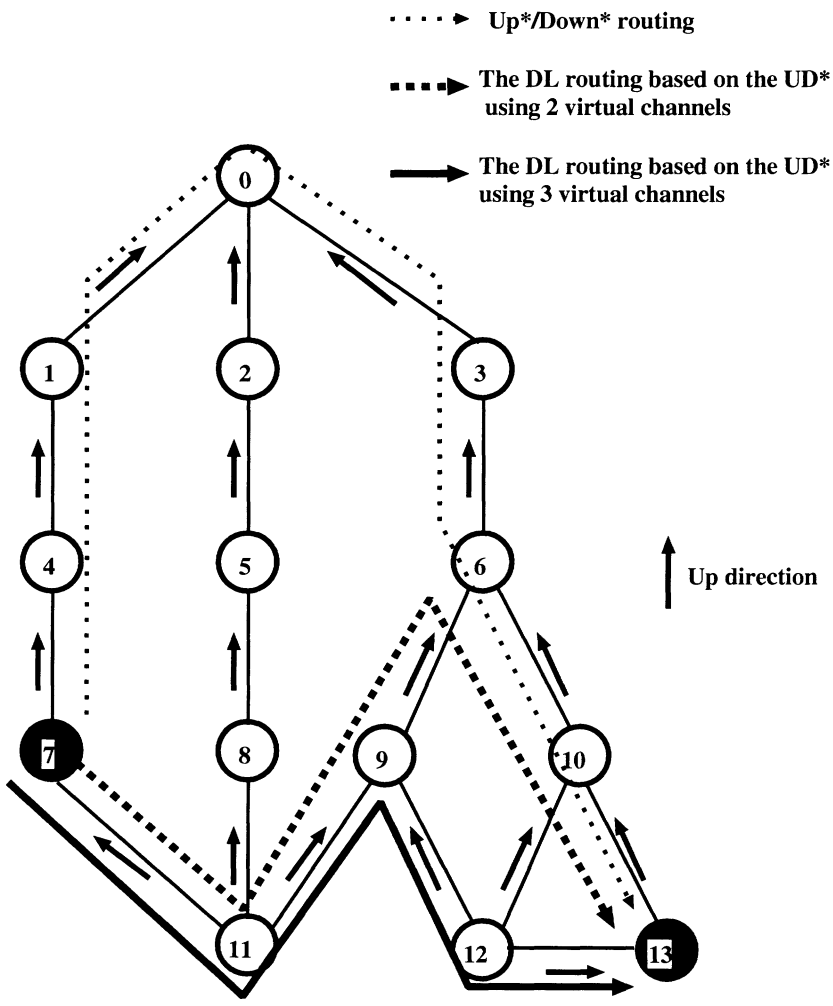


図 5.2: サブネットワークを用いたルーティング例

Mutual UD-DU　DL ルーティングは，各サブネットワーク内のパケット転送制限を基に経路を生成する．そのため，各サブネットワーク内で同一のルーティングを用いる UD* では同一の物理チャネルを使う経路が各サブネットワーク内に生成されることになる．しかし，この場合，物理チャネル間の経路分布に偏りが生じる可能性がある．

そこで，サブネットワーク内のデッドロック除去としてサブネットワーク毎に異なるパケット転送制限を課すアルゴリズムである mutual UD-DU を提案する．

mutual UD-DU は偶数番号のサブネットワークは up 方向から down 方向へのパケット転送を禁止する．また，奇数番号のサブネットワークは down 方向から up 方向へのパケット転送を禁止する[4]．なお，番号 0 のサブネットワーク内では Up*/Down* ルーティングとなるため，すべてのスイッチ対の経路が保証される．

UD-DU*　mutual UD-DU はパケット転送制限を全体に分散させることを重視している．しかし，そもそも Up*/Down* ルーティングはトラフィックがルート付近に偏るという問題を抱えており，この解決が性能向上につながる可能性がある．そこで，ルート付近のトラフィックの分散に重点を置くアルゴリズムである UD-DU* を提案する[5]．

UD-DU* ではサブネットワーク 0 において down channel から up channel へのパケット転送を禁止する．つまり Up*/Down* ルーティングを用いる．そして，その他のサブネットワークでは up channel から down channel へのパケット転送を禁止する．

5.1.2.2　経路選択アルゴリズム

DL ルーティングは，経路の生成において複数の候補の中から 1 つの最短経路を決定する．これを決定するポリシである経路選択アルゴリズムは経路の分散を行う役割を担う．経路選択アルゴリズムの中で単純なものはランダムに選択する，もしくは出力ポート番号が最も小さい経路を選択することである．しかし，これらは経路の分散を考慮していない．一方，Up*/Down* ルーティングのために考えられた経路選択アルゴリズムである Sancho's algorithm[JA00] は経路の静的解析を用いた方法であり，DL ルーティングにも適用できる．手順を次に示す．

1 すべての物理チャネルにカウンタを用意する．そして，すべてのスイッチ対の経路を計算し，カウンタ毎にその物理チャネルを通過する経路数で初期化する．Up*/Down* ルーティングのカウンタの初期化の例を図 5.3 に示す．ここで，図 5.3 においてスイッチ x からスイッチ y への経路を (x, y) とすると，スイッチ **a** から **c** への物理チャネルを 4 つの経路 $(a, e), (a, c), (b, c), (d, c)$ が通過するため，そのカウンタ値は 4 となる．

2 削除可能な経路候補を持つ物理チャネルの中で最大値のカウンタを持つ物理チャネルを選ぶ．ただし，削除可能な経路候補とは，その経路の代替経路が存在するもの

[4]名称はサブネットワーク内の up channel と down channel 間のパケット転送制限が，サブネットワークの番号毎に交互に変わることによる．

[5]名称はサブネットワーク 0 内において up channel から down channel の転送制限がなく，他のサブネットワーク内において down channel から up channel への転送制限がないことによる．

を指す．例えば，図 5.3 ではスイッチ **a** と **b** の間の物理チャネルのどちらかが選ばれる．

3 手順 2 で選択した物理チャネルを通過する 1 つの経路候補を削除する．この際，複数の経路候補が削除可能だった場合，同一スイッチ対の経路候補数が多い方の経路候補を削除する．例えば，図 5.3 においてスイッチ **a** からスイッチ **b** への物理チャネルを通過する経路 $(a,d),(a,b),(a,e),(c,b),(c,d)$ の中で経路 (a,e) のみ代替経路 (スイッチ **c** を通過する経路 (a,e)) が存在する．そのため，この物理チャネル上では経路 (a,e) を削除する．

4 経路を削除した後，削除した経路上の物理チャネルのカウンタ値を更新する．

5 すべてのスイッチ対の経路が 1 つに定まっていない場合，手順 2 に戻る．

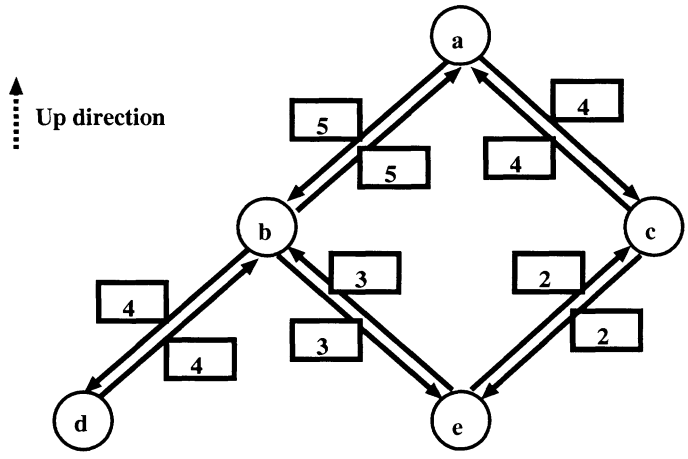


図 5.3: Up*/Down* ルーティングにおけるカウンタの初期化の例

Sancho's algorithm の計算量は，スイッチ数を N，直径を D とすると $O(N^2 * D)$ である．Sancho's algorithm では，その解析結果からトラフィックが偏る物理チャネルが発生しないように経路を選択する [MAH02b]．しかし，Sancho's algorithm は Up*/Down* ルーティングを対象に考えられたため，仮想チャネルを想定していない．もちろん，Sancho's algorithm はその手続きにおいて，物理チャネルを仮想チャネルに置き換えることにより仮想チャネルを持つ SAN に応用することが可能である．しかし，仮想チャネルを使う SAN の場合，経路の分散を物理チャネル間と仮想チャネル間の両方で考えるべきである．そこで Sancho's algorithm を仮想チャネルを用いる DL ルーティングに合わせて次のように改良する．このアルゴリズムを以後 high physical channel first と呼ぶ．

1 すべての仮想チャネルにカウンタを用意する．そして，すべてのスイッチ対の経路を計算し，カウンタ毎にその仮想チャネルを通過する経路数で初期化する．

2 削除可能な経路候補を持つ物理チャネルの中で (仮想チャネルの) カウンタの総和の最大値を持つ物理チャネルを選ぶ．

3 手順 2 で選択した物理チャネルにおいて削除可能な経路候補を持つ仮想チャネルの中で最大値のカウンタを持つ仮想チャネルを選ぶ.

4 手順 3 で選択した仮想チャネルを通過する 1 つの経路候補を削除する. この際, 複数の経路候補が削除可能だった場合は, 同一スイッチ対の経路候補数が多い方の経路候補を削除する.

5 経路を削除した後, 削除した経路上のチャネルのカウンタ値を更新する.

6 すべてのスイッチ対の経路が 1 つに定まっていない場合, 手順 2. に戻る.

high physical channel first の計算量は Sancho's algorithm と同様にスイッチ数を N, 結合網の直径を D とすると, $O(N^2 * D)$ である.

5.1.3 DL ルーティングの特徴

ルーティングは本質的にすべてのスイッチ対の経路を保証するようにパケット転送制限を設定しなければならない. しかし, DL ルーティングでは, 番号が 1 以上のサブネットワーク内において, 必ずしもすべてのスイッチ対の経路を保証する必要は無く, 経路保証の問題と切り離してパケット転送制限を課すことができる. これは, 番号 0 のサブネットワークが経路保証をしているためである.

定理 6 *DL* ルーティングはデッドロックフリーである. □

証明

(a) パケット転送においてサブネットワーク番号の切り替えは降順のみで行なわれる. したがって異なるサブネットワーク間においてデッドロックは発生しない.

(b) 各サブネットワーク内において循環が生じないようにパケット転送制限を課している. 従って各サブネットワーク内においてデッドロックは発生しない.

(a), (b) より, DL ルーティングはデッドロックフリーである. □

DL ルーティングは, その実装アルゴリズム (経路選択アルゴリズム及びサブネットワーク内のデッドロック除去アルゴリズム) に関わらず, 本質的に次の 3 つの特長を持つ.

仮想チャネル数に非依存: DL ルーティングは仮想チャネル数に関係なく, あらゆるトポロジ, スイッチ数の SAN に対して適用することができる.

最短経路を取るパケットの増加: Up*/Down* ルーティングを基にした固定ルーティングの場合 [JA00], すべてのチャネル間の循環依存を除去するために, 非最短経路が発生する. しかし, DL ルーティングでは, サブネットワークの切り換えにより仮想チャネル間における循環依存のみが除去される. そのため DL ルーティングでは物理チャネル間の循環依存が許される. 従って, DL ルーティングは, Up*/Down* ルーティングを基にした手法に比べ最短経路を取るパケットが増加する.

トラフィックの分散: Up*/Down* ルーティングを基にした固定ルーティングの場合，ツリー構造が本質的に持つ1次元的な up/down の概念をそのまま利用している．そのため Up*/Down* ルーティング では選択可能な経路の分布に偏りが生じ，効率的にネットワークのバンド幅を利用することが難しい．一方，DL ルーティングは物理チャネル間の循環依存が許されるため選択可能な経路が増加する．そのため high physical channel first などの経路の分散を図る経路選択アルゴリズムを用いることにより，より均等な経路の分布を実現することができる．よって，DL ルーティングは Up*/Down* ルーティングを基にした固定ルーティングに比べトラフィックの分散をはかることができる．

5.2　評価

DL ルーティングおよび Up*/Down* ルーティングについてフリットレベルの確率モデルシミュレーションにより評価を行った．

5.2.1　シミュレーション条件

フリットレベルシミュレータは第 3.2 節で用いたものを基に開発した．適応型ルーティングをサポートしたスイッチと固定ルーティングを用いたスイッチの違いは動作周波数や調停回路の複雑さ等であり，論理的なルーティングの手順に違いはない．したがって，本シミュレーションモデルでは両者の違いは小さく，第 3.2 節で用いたシミュレータの多くのモジュールを再利用することができた．

5.2.1.1　固定ルーティング

Up*/Down* ルーティングを DL ルーティングと同様に (パケット配達において) FIFO 性を持つ固定ルーティングとして実装した．この際，Up*/Down* ルーティングはネットワークへパケットを注入する際にランダムに設定した番号の仮想チャネルを使用する．一方，ネットワーク内では異なる番号の仮想チャネル間の移動は行わない．

DL ルーティングおよび Up*/Down* ルーティングは経路選択アルゴリズムとして low port first, Sancho's algorithm, 及び high physical channel first の 3 パターンを用いた場合について評価した．low port first は複数の経路が選択可能である場合，そのスイッチにおいて小さいポート番号を使う経路を選択する．

DL ルーティングおよび Up*/Down* ルーティングともにスパニングツリーのマッピングアルゴリズムを指定する必要がある．このアルゴリズムとして，シミュレーションでは minimum depth スパニングツリー (MDST) を基にした breadth first search (BFS) を用いた [Mae91]．また，ルートノードは Autonet[Mae91] と同様に ID 0 のノードを選択した．

5.2.1.2　パラメータ

スイッチは 8 ポートを持ち，内 4 ポートは各々異なる PC に直結した．そして，スイッチの残りの 4 ポートは他のスイッチとの接続に利用される．ネットワークのトポロジはト一

表 5.1: 固定ルーティングのシミュレーションパラメータ

実行時間	1,000,000 クロック (初めの 50,000 クロックは無視)
トポロジ	不規則トポロジ，もしくは 2D トーラス (付録参照)
サイズ	16, 32 もしくは 8×8 (64) スイッチ
仮想チャネル数	3
パケット長	128 フリット
パケット転送方式	VCT 方式
フリット転送時間	3 クロック

ラス以外に第 3.2 節と同様に同一スイッチ対にリンクを 2 本以上接続しない，という制約を課した上でランダムに生成した．不規則なトポロジの SAN におけるルーティングはトポロジにより性能が大きく左右する [MAAH01a]．そこで，不規則なトポロジでは 1 つの条件につき，10 個のトポロジにおいて評価を行った．目的地の PC は次のトラフィックパターンにより決定した．

- uniform
  すべての目的地はランダムに決定され，均一に分散される．

- bit reversal
  まず，各 PC に 0 から $(n-1)$ (ただし，n は PC 数) までの一意の 2 進の番号を割当てる．
  そして，2 進の番号 $(a_0, a_1, \cdots, a_{n-2}, a_{n-1})$ を持つ PC は自分の番号のビット列を逆順に並べた番号 $(a_{n-1}, a_{n-2}, \cdots, a_1, a_0)$ の PC へパケットを送る．

シミュレーションで初めの 50,000 クロックはネットワークが安定せず，想定した負荷に達していないと考えられるため評価の対象外とした．また，Up*/Down* ルーティングを用いた場合，経路の偏りと非最短経路の割合は仮想チャネル数に依存しない．したがって，仮想チャネル数により評価の傾向はかわらない．(例えば，2 本の場合と 3 本の場合の評価結果の傾向は同一であった [鯉渕 02b] [鯉渕 02c]．) そこで，本シミュレータでは QsNET が仮想チャネルを 2 本，InfiniBand においても 15 本以下であることをふまえ，仮想チャネル数を 3 本に固定した．

ネットワークレイテンシ

ある PC p がパケットの最初のフリットを NIC の入力バッファに挿入した時刻を t_0，目的地の PC q の NIC がパケットの最後のフリットを受け取った時刻を t_1 とする．ここで，$T_{lat}(p,q) = t_1 - t_0$ をネットワークレイテンシと呼び，ネットワークの性能を測る指標とした．

スループット

スループットは，全PCが毎クロックに 1 フリット受信する場合を 1.00 として，受信トラフィックの最大値とした [JSL02].

平均ホップ数

パケットが目的地の PC に到達するまでに通過したスイッチ数とした.

5.2.2　不規則なトポロジの SAN における評価結果 (uniform traffic)

5.2.2.1　デッドロック除去アルゴリズムの比較

サブネットワーク内のパケット転送制限毎の DL ルーティングと Up*/Down* ルーティングのシミュレーション結果を図 5.4, 表 5.2 および表 5.3 に示す．図 5.4 において縦軸は 10 個のトポロジでの平均スループットを表している.

表 5.2: 不規則なトポロジの SAN におけるスループットとその分散
(16 スイッチ, uniform traffic)

	平均	分散	最小	最大
Up*/Down* (low port)	0.161	0.032	0.112	0.207
Up*/Down* (Sancho's algorithm)	0.177	0.033	0.134	0.225
Up*/Down* (high p-ch first)	0.176	0.032	0.130	0.222
UD* (low port)	0.169	0.008	0.159	0.184
UD* (Sancho's algorithm)	0.289	0.022	0.253	0.326
UD* (high p-ch first)	0.290	0.024	0.290	0.334
Mutual UD-DU (low port)	0.180	0.016	0.160	0.205
Mutual UD-DU (Sancho's algorithm)	0.295	0.022	0.272	0.332
Mutual UD-DU (high p-ch first)	0.289	0.025	0.258	0.335
UD-DU* (low port)	0.169	0.008	0.159	0.184
UD-DU* (Sancho's algorithm)	0.286	0.023	0.254	0.320
UD-DU* (high p-ch first)	0.282	0.020	0.251	0.324

図 5.4，表 5.2 および表 5.3 より，UD*，mutual UD-DU，及び UD-DU* の各 DL ルーティングは同一の経路選択アルゴリズムを用いた場合，Up*/Down* ルーティングに比べスループットが大幅に向上していることがわかる．また，図 5.4 において，各 DL ルーティングは経路の分散を考慮する Sancho's algorithm および high physical channel first を用いた場合，Up*/Down* ルーティングとの性能差が大きくなる．これは，DL ルーティングにおける経路選択アルゴリズムの選択可能な経路数が Up*/Down* ルーティングの場合に比べ多いことにより，これら 2 つがより効果的に経路を分散できたためと考えられる．また，表 5.2 および表 5.3 より DL ルーティングは分散が小さいことから Up*/Down* ルーティングに比べスループットが安定していることがわかる.

表 5.3: 不規則なトポロジの SAN におけるスループットとその分散
(32 スイッチ, uniform traffic)

	平均	分散	最小	最大
Up*/Down* (low port)	0.072	0.011	0.057	0.093
Up*/Down* (Sancho's algorithm)	0.078	0.009	0.067	0.093
Up*/Down* (high p-ch first)	0.078	0.009	0.067	0.092
UD* (low port)	0.135	0.011	0.116	0.147
UD* (Sancho's algorithm)	0.212	0.016	0.193	0.233
UD* (high p-ch first)	0.213	0.015	0.195	0.233
Mutual UD-DU (low port)	0.142	0.009	0.125	0.158
Mutual UD-DU (Sancho's algorithm)	0.217	0.015	0.192	0.244
Mutual UD-DU (high p-ch first)	0.217	0.014	0.195	0.242
UD-DU* (low port)	0.135	0.009	0.116	0.147
UD-DU* (Sancho's algorithm)	0.210	0.013	0.192	0.236
UD-DU* (high p-ch first)	0.211	0.015	0.189	0.241

次にパケットの平均ホップ数について表5.4 に示す．表5.4 より，すべての DL ルーティングは，Up*/Down* ルーティングに比べパケットの平均ホップ数を 6%～14%削減することに成功していることが分かる．表 5.4 において UD*, mutual UD-DU,UD-DU* のホップ数に差がないことから，同一ツリーの up 方向と down 方向を基にした禁止ターンの配置は DL ルーティングの平均ホップ数に影響がないことがわかる．また，経路選択アルゴリズムは選択可能な最短経路の中から 1 つ選択するため，平均ホップ数に影響しない．

表 5.4: 不規則なトポロジの SAN における平均ホップ数 (uniform traffic)

	Low port first		Sancho's algorithm		High p-ch first	
	16sw.	32sw.	16sw.	32sw.	16sw.	32sw.
Up*/Down*	2.01	2.85	2.01	2.85	2.02	2.85
UD*	1.88	2.50	1.89	2.52	1.89	2.52
Mutual UD-DU	1.89	2.52	1.88	2.52	1.88	2.52
UD-DU*	1.88	2.50	1.89	2.52	1.88	2.52

また，図 5.4 より，mutual UD-DU は UD*，UD-DU* に比べ若干高いスループットを示してはいるが，DL ルーティングのサブネットワーク内のデッドロック除去アルゴリズムの性能差は小さいことが分かった．

最後に，スループットの次に重要な評価項目であるレイテンシについて図 5.5 に示す．図 5.5 は low port first を用いた Up*/Down* ルーティングに対する mutual UD-DU と high physical channel first を用いた DL ルーティングのスループット向上率が平均的なトポロジにおける受信トラフィックとレイテンシの関係を示している．図 5.5 より，各 DL ルー

ティングは Up*/Down* ルーティングに比べ高スループットのみならず，低レイテンシを達成していることがわかる．

5.2.2.2　経路選択アルゴリズムの比較

次に各経路選択アルゴリズムのスループットをまとめたものを図 5.6 に示す．図 5.6，表 5.2 および表 5.3 より Sancho's algorithm および high physical channel first が low port first に比べ最大 58%スループットが向上したことが分かる．これはルーティングパスの静的な解析を行う Sancho's algorithm, および，high physical channel first がスループットを大きく向上させることを示している．また，図 5.6 より，Up*/Down* ルーティングにおける経路選択アルゴリズムの影響は，DL ルーティングの場合に比べ小さい．これは，Up*/Down* ルーティングにおいて各経路選択アルゴリズムの選択可能な経路数が DL ルーティングの場合に比べ少ないことが原因と考えられる．

また，図 5.4，図 5.6 より DL ルーティング間のスループットの差は経路選択アルゴリズムのトラフィックの分散能力による部分が大きいことがわかる．また，図 5.4，図 5.6, 表 5.2 および表 5.3 より mutual UD-DU および high physical channel first を併用した DL ルーティングは high physical chnnnel first を用いた Up*/Down* ルーティングに比べスループットが 178%向上したことを筆頭に，DL ルーティングの有効性が確認された．

5.2.3　不規則なトポロジの SAN における評価結果 (bit reversal traffic)

次に bit reversal traffic での評価結果を図 5.7 および図 5.8 に示す．

図 5.7 および図 5.8 より，uniform traffic の場合と同様に各 DL ルーティングは同一のスパニングツリーを用いた Up*/Down* ルーティングに比べ大幅な性能向上を達成していることが分かる．

bit reversal traffic の場合，異なる出発地のパケットが目的地の consumption channel においてブロックが生じることがない．このため一部の固定ルーティングは bit reversal traffic においてトラフィックが偏るにも関わらず，uniform traffic の場合とほぼ同様の性能を達成することができたと考えられる．

5.2.4　規則的なトポロジの SAN における評価結果

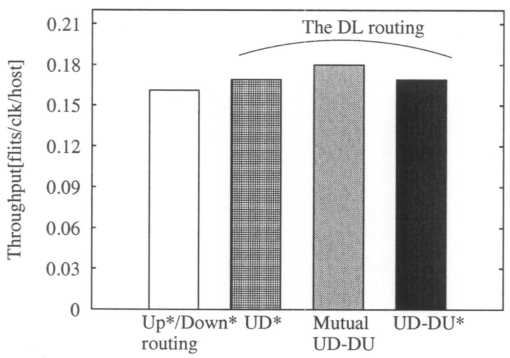
規則的なトポロジとして 8×8 2D トーラスにおける評価結果を図 5.9 および表 5.5 に示す．図 5.9 および表 5.5 において "High p-ch first" は high physical channel first を示す．

実際の SAN では規則性や階層性がある程度見られるため，規則網であるトーラスにおける評価は，不規則なトポロジの場合と同様に重要である．図 5.9 より，実装アルゴリズムによらず DL ルーティングは Up*/Down* ルーティングに比べ常に高スループットであることが確認された．特に，mutual UD-DU と high physical channel first を組み合わせた DL ルーティングは high physical channel first を用いた Up*/Down* ルーティングに比べ最大 266%のスループット向上を達成した．DL ルーティングは規則的なトポロジにおいても効果があるため，規則的なトポロジを用いる大規模並列計算機に適用して高い耐

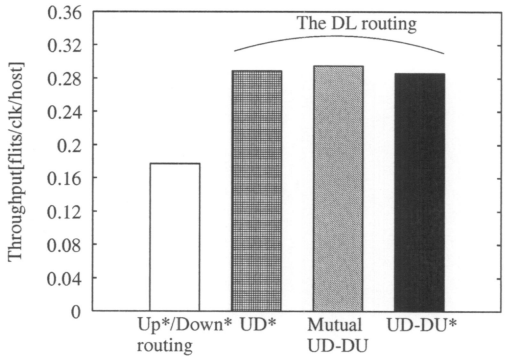
故障性を活用することも可能であると考えられる.

表 5.5: 8×8 2D トーラスの SAN における平均ホップ数

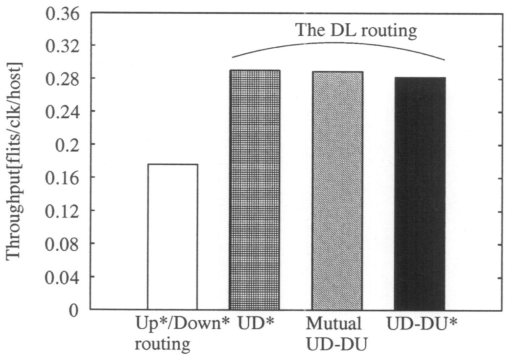
	Uniform traffic	Bit reversal traffic
Up*/Down*(low port first)	4.38	4.02
Up*/Down*(high p-ch first)	4.41	4.23
DL (UD*, low port first)	4.01	3.49
DL (UD*, high p-ch first)	4.01	3.44
DL (mutual UD-DU, low port first)	4.01	3.47
DL (mutual UD-DU, high p-ch first)	4.01	3.44



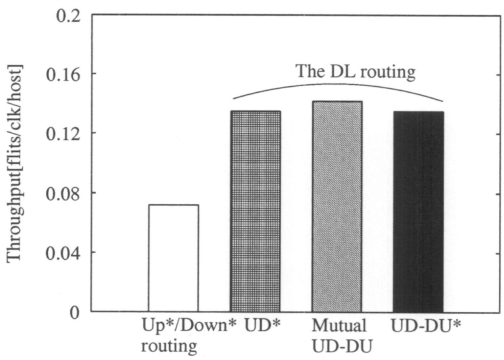
(a) Low port first, 16 スイッチ



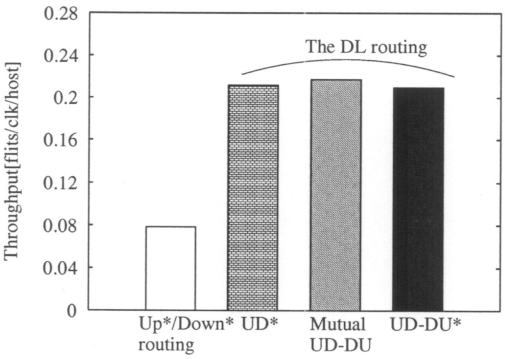
(b) Sancho's algorithm, 16 スイッチ



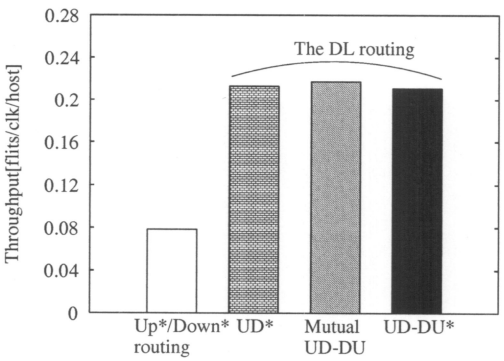
(c) High physical channel first, 16 スイッチ



(d) Low port first, 32 スイッチ

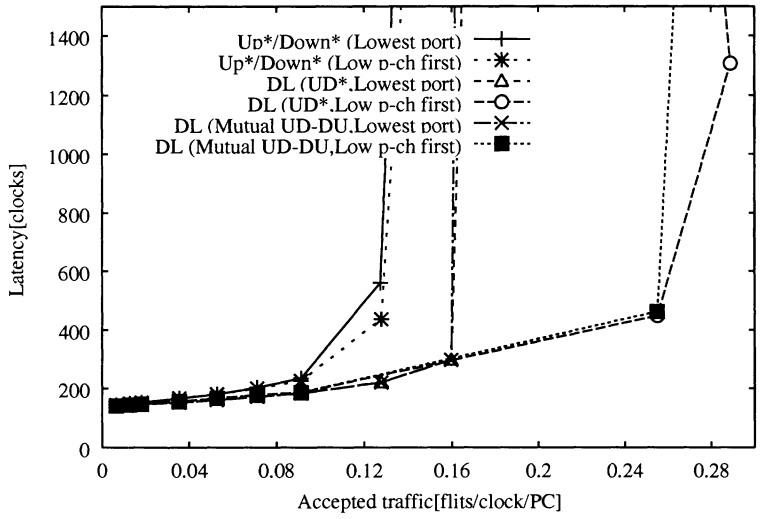


(e) Sancho's algorithm, 32 スイッチ

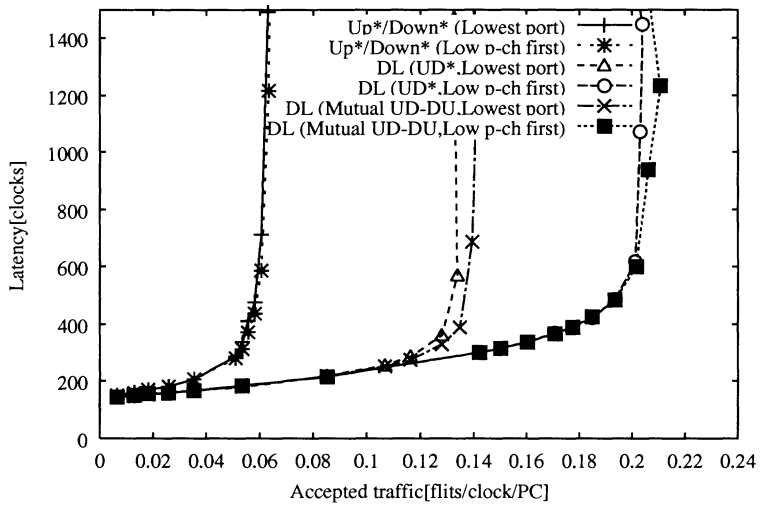


(f) High physical channel first, 32 スイッチ

図 5.4: 不規則なトポロジの SAN におけるデッドロック除去アルゴリズムの
平均スループットの比較 (uniform traffic)

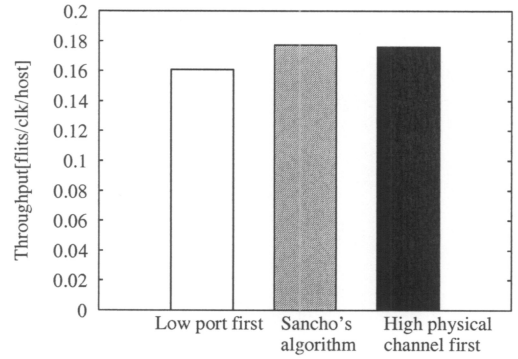


(a) 16 スイッチ

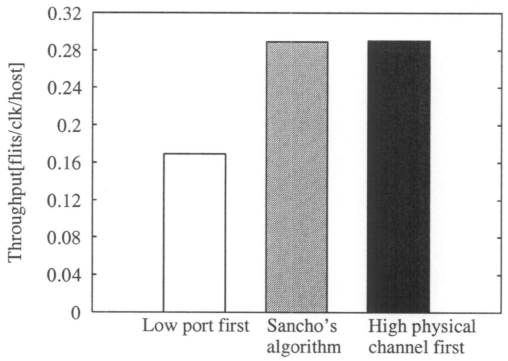


(b) 32 スイッチ

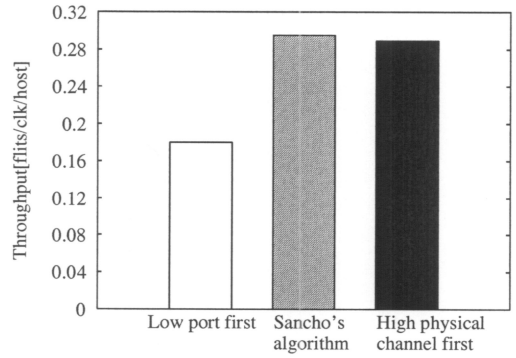
図 5.5: 不規則なトポロジの SAN におけるスループットとレイテンシ (uniform traffic)



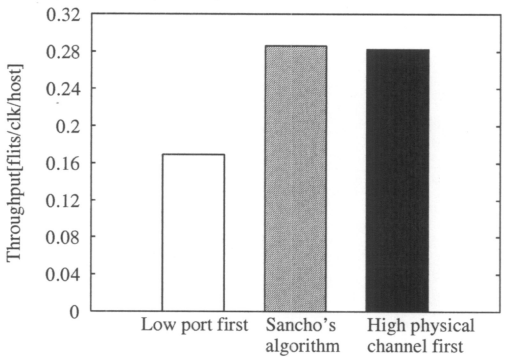
(a) Up*/Down*, 16 スイッチ



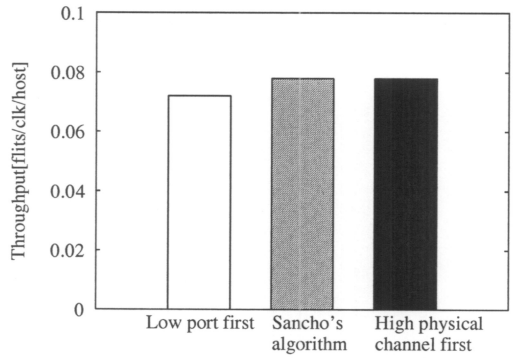
(b) DL (UD*), 16 スイッチ



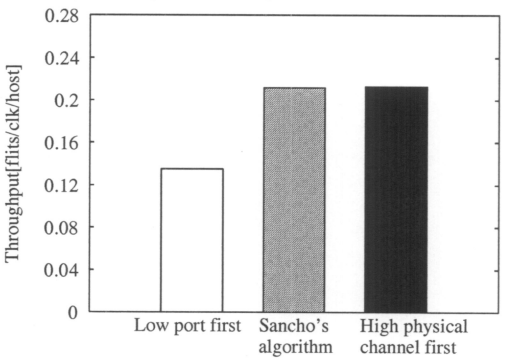
(c) DL (mutual UD-DU), 16 スイッチ



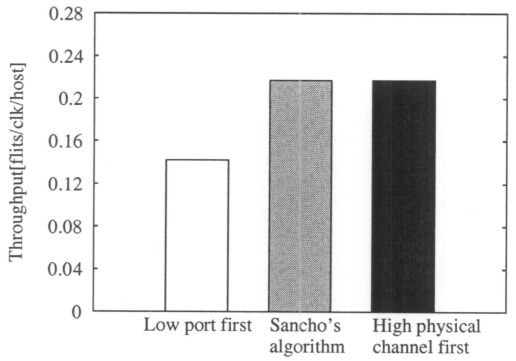
(d) DL (UD-DU*), 16 スイッチ



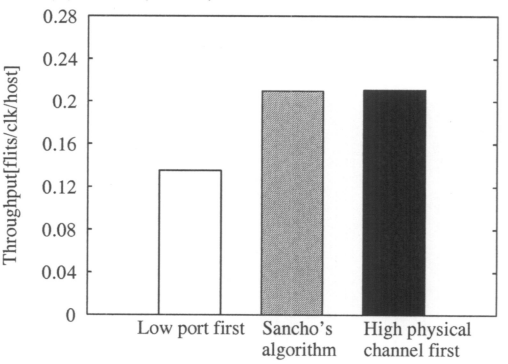
(e) Up*/Down*, 32 スイッチ



(f) DL (UD*), 32 スイッチ

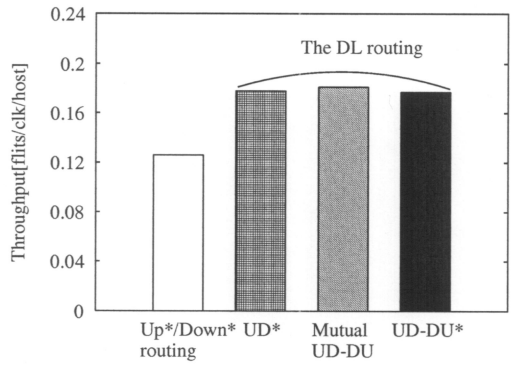


(g) DL (mutual UD-DU), 32 スイッチ

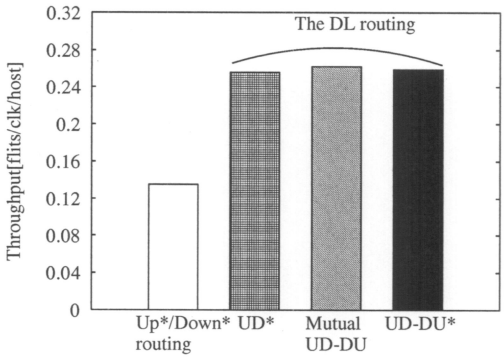


(h) DL (UD-DU*), 32 スイッチ

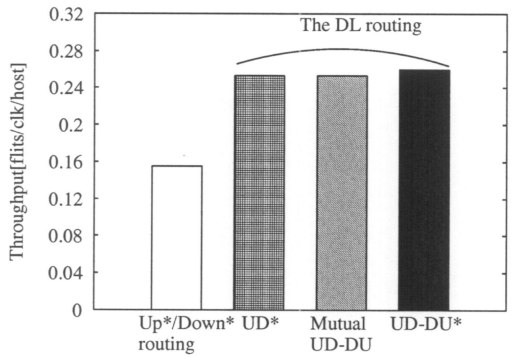
図 5.6: 不規則なトポロジの SAN における経路選択アルゴリズムの
平均スループットの比較 (uniform traffic)



(a) Low port first

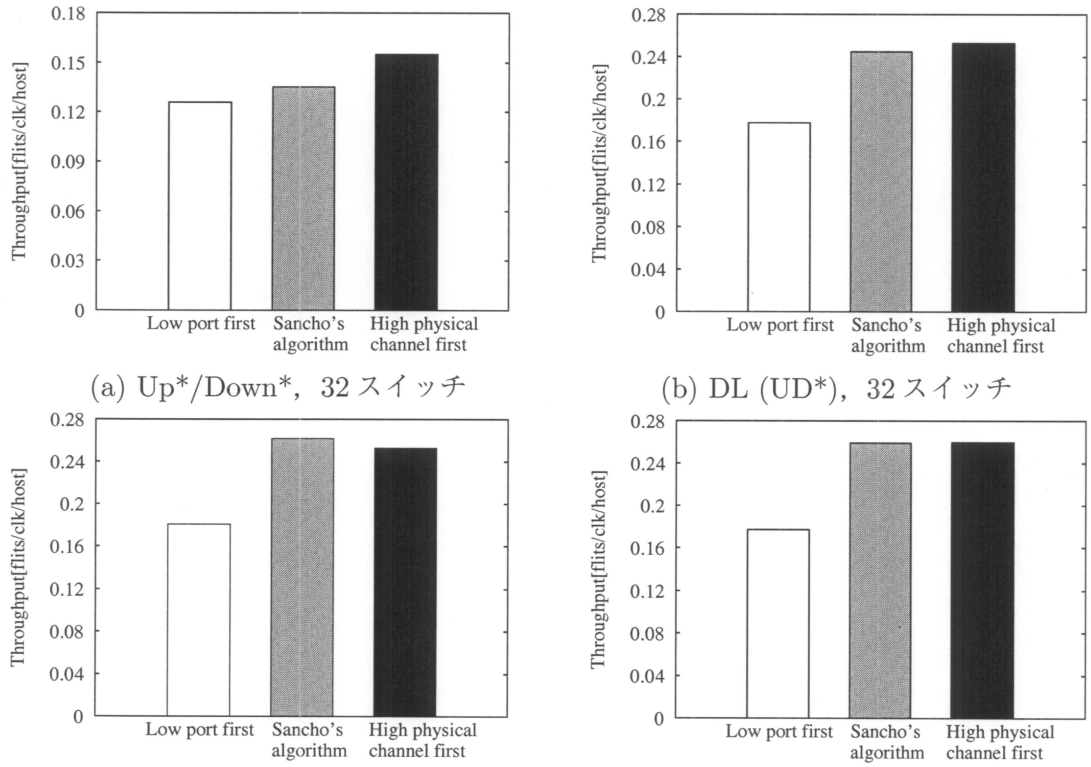


(b) Sancho's algorithm



(c) High phisical channel first の場合

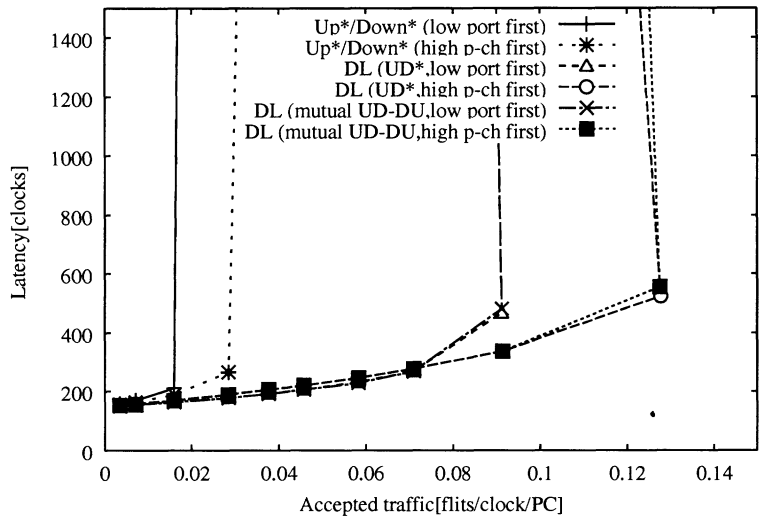
図 5.7: 不規則なトポロジの SAN におけるデッドロック除去アルゴリズムの平均スループットの比較 (bit reversal traffic，32 スイッチ)



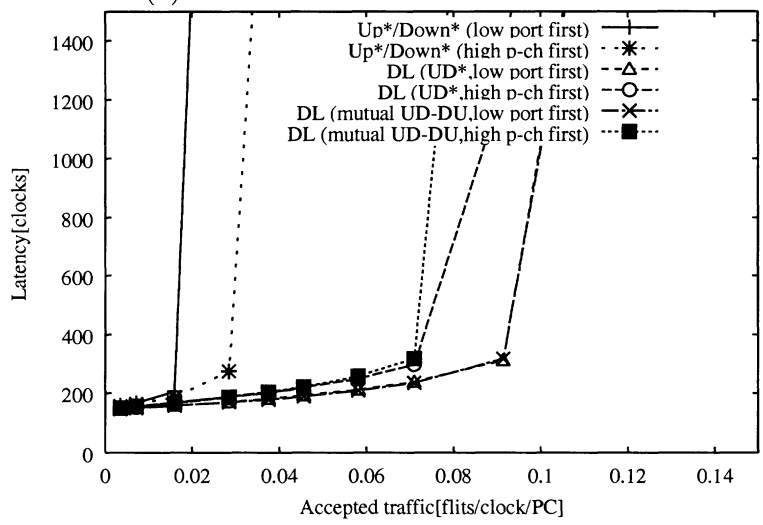
(a) Up*/Down*，32 スイッチ　　(b) DL (UD*)，32 スイッチ

(c) DL (mutual UD-DU)，32 スイッチ　　(d) DL (UD-DU*)，32 スイッチ

図 5.8: 不規則なトポロジの SAN における経路選択アルゴリズムの
平均スループットの比較 (bit reversal)



(a) Uniform traffic



(b) Bit reversal traffic

図 5.9: 8 × 8 2D トーラスの SAN におけるスループットとレイテンシ

5.3　その他の解決策

第 3.3 節で述べた既存の適応型アルゴリズムの多くは経路選択アルゴリズムを適用することで固定ルーティングとして実装可能である．そこで，本節では，これらの同時並行的に進んでいる研究と DL ルーティングの比較を述べる．また，規則的なトポロジにおける適応型アルゴリズムである次元逆転ルーティングは DL ルーティングと同様にサブネットワークを用いる．そこで本節では両者の違いについても明らかにする．

5.3.1　LASH ルーティング, Sancho らの InfiniBand ルーティングおよび in transit バッファ

LASH ルーティング [SLT02] および Sancho らの InfiniBand ルーティング [JAJ+02] は，最短経路を保証する利点がある一方，仮想チャネル数により適用できるネットワークサイズが限定されてしまう．一方，DL ルーティングは仮想チャネル数が少ない場合，非最短経路を生じてしまう可能性があるが，仮想チャネル数によるネットワークサイズの制限がない．

また，in transit バッファ [JPMJ02] は最短経路を保証するためにホストに専用のバッファが必要となる．しかし，DL ルーティングではルーティングテーブルもしくはパケットのヘッダを更新するだけで実装することができる．

5.3.2　Silla らの minimal ルーティング

Silla らの minimal ルーティング [SD00] は固定ルーティングとして実装することができない．これはこの minimal ルーティングが動的な経路選択によりデッドロックフリーを保証しており，静的に経路を固定した場合，デッドロックを防ぐことができないためである．

5.3.3　ヒューリスティックルールを用いた Up*/Down* ルーティング

ヒューリスティックルールを用いた DFS Up*/Down* ルーティング [JA00] は DL ルーティングに活用することができる．これはこの DFS Up*/Down* ルーティングを DL ルーティングにおけるサブネットワーク内のデッドロック除去アルゴリズムとして適用することができるからである．しかし，本質的に DFS Up*/Down* ルーティング単体では仮想チャネルを使うことを想定していないため，DL ルーティングが対象としている SAN には向かない．

5.3.4　次元逆転 (dimension reversal) ルーティング

k-ary n-cube を対象とした適応型アルゴリズムである次元逆転ルーティング [DA93](詳細は付録参照) はサブネットワークの概念を用いている点で DL ルーティングと似ている．

次元逆転ルーティングは e-cube ルーティング [DS87] において，進行方向の出力仮想チャネルが塞がっている場合，使用する仮想チャネル番号を 1 増加させることにより e-cube

ルーティングで禁止されている方向へのパケット転送を許可する．つまり，次元逆転ルーティングはパケットがブロックされる回数を減らすために使用するサブネットワークを切り換える方法である．もちろん，次元逆転ルーティングは最短型の固定ルーティングである e-cube ルーティングを基にしているため，不規則なトポロジに適用することはできない．一方，DL ルーティングは経路の分散，および，最短経路の割合の増加のために，仮想チャネル番号の切り換えを行う．

つまり，次元逆転ルーティングは，e-cube ルーティングの自由度をあげるために仮想チャネル番号を切り換えるため，(1) トポロジが k-ary n-cube に限定される，(2) 固定ルーティングとして用いることができない，という 2 点で根本的に DL ルーティングと異なる．

5.3.5　DL ルーティングとその他の解決策の比較

表 5.6: 固定ルーティングの比較

	DFS Up*/Down*	Sancho らの InfiniBand	LASH	In transit バッファ	DL
トポロジフリー？	yes	yes	yes	yes	yes
サイズ制限？	no	yes	yes	no	no
最短型?	no	yes	yes	yes	no
仮想チャネルが必要？	no	yes	yes	no	yes
ホスト PC の バッファが必要？	no	no	no	yes	no

固定ルーティングは適応型アルゴリズムの場合と同様に最短経路の割合を増加させることがスループット向上につながる．この観点を含めたその他の解決策と DL ルーティングの比較を表 5.6 に示す．表 5.6 より，Sancho らの InfiniBand ルーティング，LASH ルーティング，および in transit バッファは最短経路を保証する．しかし，この 3 つのルーティングはネットワークサイズが仮想チャネル数により制限される，もしくは，付加バッファが必要であるため，限定的な用途に限られる．一方，DL ルーティングは仮想チャネル数による制限がないために最短経路を保証することはできない．しかし，DL ルーティングは限られた仮想チャネル数の SAN においても Up*/Down* ルーティングに比べ最短経路の割合を高めることができ，かつ，経路選択アルゴリズムにより効果的にトラフィックの分散を実現することができる点で効率的にスループット向上を実現することができる方法であるといえる．

また，ヒューリスティックルールを用いた DFS Up*/Down* ルーティングと DL ルーティングはトポロジやそのサイズに制限がない点では共通であるが，DL ルーティングは仮想チャネルを使うことを想定していない．前節で述べた通り，DFS Up*/Down* ルーティングは DL ルーティングを活用することができる点で，共存する技術である．

このように多くのその他の解決策は対象としている SAN が異なるため DL ルーティングとの単純な性能面の比較を行うことが難しい．しかし，各々の方法は異なる特色を持っているということはいえる．

5.4　まとめ

本章では，仮想チャネルをスループット向上に用いる固定ルーティングである DL ルーティングを提案し，評価を行った．DL ルーティングはまず，ネットワークを仮想チャネルを用いて同一トポロジのサブネットワークの層に分割する．次に，サブネットワーク内とサブネットワーク間のデッドロック除去を行った上で，各スイッチ間の経路を経路選択アルゴリズムにより 1 つに決定する．DL ルーティングはサブネットワークを切り換えることにより，パケットの平均ホップ数の削減，及び経路の分散配置を実現する．

DL ルーティングはサブネットワーク内のデッドロック除去アルゴリズムと経路選択アルゴリズムの組み合わせにより複数の実装方法が考えられるが，シミュレーションの結果，いずれも同数の仮想チャネルを用いた Up*/Down* ルーティングに比べ，大幅なスループット向上が確認された．具体的には不規則なトポロジの SAN において mutual UD-DU, high physical channel first を用いた DL ルーティングが high physical channel first を用いた Up*/Down* ルーティングに比べ最大 178%，トーラストポロジの SAN においては最大 266%の各々スループット向上を達成した．

DL ルーティングは仮想チャネル数に関係なく，あらゆるトポロジ，ネットワークサイズの SAN に適用できるが，現状ではこの条件を満たす固定ルーティングは他に Up*/Down* ルーティングしかない．この点で，DL ルーティングは近年主流となりつつある InfiniBand などの SAN において，最も優れた固定ルーティングであるといえる．

第6章　結 論

現在，PC クラスタは大規模科学技術計算システムおよびサーバーシステムの主流になりつつある．PC クラスタはベオウルフ型クラスタと SAN を用いたクラスタの 2 種類にわけられる．しかし，ベオウルフ型クラスタは通信プロトコルに TCP/IP を用いているため，並列処理に必要な低レイテンシ通信を実現することが難しい．そのため，現在では SAN を用いた PC クラスタが主流になりつつある．

SAN においてルーティングアルゴリズムは性能向上の鍵を握る技術の 1 つである．しかし，SAN は並列計算機の相互結合網と異なり，不規則なトポロジをとることが多いため高性能なデッドロックフリールーティングを開発することが難しい．そのため，Myrinet, InfiniBand などの高スループットかつ低レイテンシである SAN が商品化されているにも関わらず，Up*/Down* ルーティングなどの効率の悪いルーティングアルゴリズムが用いられているのが現状である．

そこで，本論文では，SAN におけるデッドロックフリールーティングを開発することを目的とした．しかし，現在，SAN は様々な特徴を持つものが提案されており，包括的に扱うことが難しい．そこで，適応型ルーティングと固定ルーティングに分類し，解決すべき課題を 3 つに細分化した．そして，各課題について集中的に検討を重ねることで様々な形態の SAN においてバンド幅の使用率 (スループット) の向上を実現した．3 つの解決策を順に述べる．

まず，不規則なトポロジの SAN における適応型アルゴリズムである L-turn ルーティングと R-turn ルーティングを提案した．L-turn ルーティングと R-turn ルーティングはデッドロックを除去するためのパケット転送制限をネットワーク全体に分散させることができる．そして，L-turn ルーティングと R-turn ルーティングは仮想チャネルやバッファを追加することなしに，あらゆるトポロジ，ネットワークサイズの SAN に適用することができる．フリットレベルのシミュレーションの結果, L-turn ルーティングは仮想チャネルを持たない不規則なトポロジの SAN において Up*/Down* ルーティングに比べスループット向上が確認できた．また，トーラスなどの規則トポロジの SAN においてそのトポロジに特化したルーティングアルゴリズムを用いた場合，リンク故障に対してシステムを止めて修理し，トポロジを修復する必要がある．一方，L-turn ルーティングと R-turn ルーティングを規則トポロジの SAN に用いた場合，リンク故障が起きたとしてもトポロジの変更を検出し，ルーティングテーブルを更新することにより運用を続けることができる．そのため，L-turn ルーティングと R-turn ルーティングは高い耐故障性を提供することができる点で SAN のみならず，大規模並列計算機の相互結合網への応用も可能である．

次に，適応型ルーティングにおける OSF である LDSF，LRU 選択機構，および MMLRU 選択機構を提案した．この 3 つの OSF は出力物理チャネルの選択を行った後に，その中の出力仮想チャネルの選択を行う点が特徴である．この 3 つの OSF は各物理チャネルに

カウンタを用意する必要がある．LDSF では物理チャネルの通過フリットをカウンタに記録し，その物理チャネルの利用状況を把握する．また，LRU 選択機構はパケット単位で使用されずにいた時間の最も長い物理チャネルをカウンタで把握する．一方，MMLRU 選択機構はこれら2つとは異なり，各物理チャネルにおいて同時に使用される仮想チャネル数を削減することでリンク遅延を抑える．MMLRU 選択機構では同数のパケットを持つ物理チャネル間の選択をパケット単位の LRU ポリシを用いるため，少量のカウンタが必要となる．出力仮想チャネルの選択では，3つの OSF とも適応型アルゴリズムによる制限が最も厳しいものを優先する．これら3つの OSF はトラフィックを分散するためのアプローチは異なるが，適応型アルゴリズム，トラフィックパターンおよびトポロジに依存しない単純な手法である．フリットレベルのシミュレーション結果より，3つの OSF はトラフィックパターン，トポロジによらず，従来の OSF より安定した性能を実現することが確認できた．

最後に，不規則なトポロジの SAN における固定ルーティングである DL ルーティングを提案した．DL ルーティングはネットワークを仮想チャネルを用いて同一トポロジのサブネットワークの層に分割する．そして，サブネットワーク内とサブネットワーク間のデッドロック除去を行った上で，各スイッチ間の経路を経路選択アルゴリズムにより1つに決定する．DL ルーティングはサブネットワークを切り換えることにより，パケットの平均ホップ数の削減，及び経路の分散配置を実現する．また，DL ルーティングはサブネットワーク内のデッドロック除去アルゴリズムと経路選択アルゴリズムについて複数の組み合わせの実装方法が考えられるが，シミュレーションの結果，いずれの DL ルーティングも Up*/Down* ルーティングに対し，スループット向上が確認できた．また，近年，InfiniBand などの仮想チャネルと固定ルーティングを採用した SAN が主流になりつつある．このような SAN におけるルーティングの中で，DL ルーティングは (1) 仮想チャネルをスループット向上に利用し，かつ (2) トポロジ，ネットワークサイズおよび仮想チャネル数に制限がない，唯一のルーティングである．この点で DL ルーティングはこれらの SAN の中で現状では最も優れた固定ルーティングであるといえる．また，DL ルーティングをトーラスなどの規則トポロジの SAN に用いた場合，リンク故障が起きたとしてもトポロジの変更を検出し，ルーティングテーブルを更新することにより運用を続けることができる．そのため，DL ルーティングは L-turn ルーティングと同様に高い耐故障性を提供することができる点で SAN のみならず，大規模並列計算機の相互結合網への応用も可能である．

今後の課題としては，(1) パケット配達の FIFO 性の制限を緩めた固定ルーティング技術の開発，(2) 提案した3つのルーティング技術の実機への移植，および，実機での運用が挙げられる．

(1) は，既存の固定ルーティングの研究が前提としているパケット配達の FIFO 性のモデルを変えることである．通常，PC クラスタでは異なるサービス (プロセス) のパケット間で配送の FIFO 性を保証する必要はない．そこで，(1) の目的は同一サービスにおけるパケット間の FIFO 性のみを保証する，という条件緩和により更なる高スループットを達成するルーティング技術を確立することである．

(2) は，実機での評価を通して本論文で提案した3つのルーティング技術の重要性を示すことが目的である．幸運なことに，当研究室では理論的な研究のみならず，関連施設と

協力して実装を通した研究も盛んに行われている．例えば2000年3月時点で，世界最高速である64 Gbpsの光ネットワーク用スイッチRHiNETの試作，稼働に成功している．また，文部省重点領域研究であった超並列計算機Jump-1の実装および評価(実装は64プロセッサの規模)にも成功している．現在，当研究室ではRHiNETスイッチが稼働しているため，64台のPCを用いたRHiNETによるルーティングの評価を取ることが可能である．したがって，RHiNETによる評価を通して本研究が理論的かつ実際的であることを示すができるであろう．

大規模計算システムを取り巻く環境は，ここ10年で激変した．CC-NUMA (cache coherent non-uniform memory access model: 不均一アクセスモデル)などの大規模並列計算機は注目を集めたが，商業的な成功といえるほど普及しなかった．そのため，新たな大規模計算システムの構築法が切望されている．このような現状の中で，PCクラスタは大規模計算システムおよびサーバーシステムとして成功しつつある．そして，SANにおけるデッドロックフリールーティングはその成功の鍵を握る技術の1つであり，その発展のために本論文が貢献できれば幸いである．

謝辞

本研究の機会を与えてくださり，絶えず御指導頂いた慶應義塾大学理工学部 天野 英晴教授に深く感謝致します．

また，本研究をまとめるにあたり，貴重な御助言を頂いた慶應義塾大学理工学部 笹瀬巌教授, 寺岡 文男教授，山本 喜一助教授に深く感謝致します．

本研究を共に行った慶應義塾大学大学院理工学研究科開放環境科学専攻博士課程 上樂明也氏，現 ERATO 北野共生システムプロジェクト 舟橋 啓博士に深く感謝致します．

校正をしていただいた慶應義塾大学大学院理工学研究科開放環境科学専攻修士課程 河野賢一氏に感謝致します．

また，普段より御助言，御協力頂いた慶應義塾大学理工学部情報工学科天野研究室，山崎研究室，安西今井研究室の皆様に心より感謝致します．

本研究の一部は，文部省科学研究費補助金（特別研究員奨励費）による．

2003 年 3 月 矢上キャンパス創想館にて

鯉渕道紘

参考文献

[Aea90] A.Agarwal and et al. April: A processor architecture for multiprocessing. In *Proceedings of International Symposium on Computer Architecture*, pp. 104–114, June 1990.

[AMAH02] A.Jouraku, M.Koibuchi, A.Jouraku, and H.Amano. Routing Algorithms Based on 2D Turn Model for Irregular Networks. In *Proceedings of the International Symposium on Parallel Architectures, Algorithms, and Networks*, pp. 289–294, June 2002.

[BP89] S. Badr and P. Podar. An Optimal Shortest-Path Routing Policy for Network Computers with Regular Mesh-Connected Topologies. *IEEE Transactions on Computers*, Vol. 38, No. 10, pp. 1362–1371, October 1989.

[C.E85] C.E.Leiserson. "Fat-trees: Universal networks for hardware-efficient supercomputing". *IEEE Transactions on Computers*, Vol. C-34, No. 10, pp. 892–901, October 1985.

[CK92] A. A. Chien and J. J. Kim. Planar-Adaptive Routing: Low-cost Adaptive Networks for Multiprocessors. *Proceedings of International Symposium on Computer Architecture*, pp. 268–277, 1992.

[CW93] C.L.Seitz and W.Su. A family of routing and communication chips based on the mozaic. In *Proceedings of the Washington Symposium on Integrated Systems*, 1993.

[D.A87] D.A.Nichols. Using idle workstations in a shared computing environment. In *Proceedings of the 11th ACM Symposium on Operating Systems Principles*, pp. 5–12, November 1987.

[DA93] W. J. Dally and H. Aoki. Deadlock-Free Adaptive Routing in Multicomputer Networks Using Virtual Channels. *IEEE Transaction on Parallel and Distributed Systems*, Vol. 4, No. 4, pp. 466–475, 1993.

[Dal92] W. J. Dally. Virtual-channel flow control. *IEEE Transaction on Parallel and Distributed Systems*, Vol. 3, No. 2, pp. 194–205, 1992.

[Dea92] D.Lenoski and et al. The Stanford DASH multiprocessor. *IEEE Transactions on Computers*, Vol. 25, No. 3, pp. 63–79, 1992.

[DS87] 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.

[Dua93] J. Duato. A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks. *IEEE Transaction on Parallel and Distributed Systems*, Vol. 4, No. 12, pp. 1320–1331, 1993.

[Dua94] J. Duato. A Necessary And Sufficient Condition For Deadlock-Free Adaptive Routing In Wormhole Networks. *Proceedings of the International Conference on Parallel Processing*, Vol. 1, pp. 142–149, 1994.

[Dua95] J. Duato. A Necessary And Sufficient Condition For Deadlock-Free Adaptive Routing In Wormhole Networks. *IEEE Transaction on Parallel and Distributed Systems*, Vol. 6, No. 10, pp. 1055–1067, 1995.

[FFA+02] F.Petrini, W.C Feng, A.Hoisie, S.Coll, and E.Frachtenberg. The Quadrics network: high-performance clustering technology. *IEEE Micro*, Vol. 22, No. 1, pp. 46–57, 2002.

[FJ00] F.Silla and J.Duato. On the Use of Virtual Channels in Networks of Workstations with Irregular Topology. *IEEE Transactions on parallel and distributed systems*, Vol. 11, No. 8, pp. 813–828, 2000.

[G.C01] G.Ciaccio. Messaging on Gigabit Ethernet: Some experiments with GAMMA and other Systems. In *Proceedings of the International Parallel and Distributed Processinig Symposium*, pp. 1624–1631, 2001.

[GG01] G.Chiola and G.Ciaccio. Gamma: a low-cost network of workstations based on active messages. In *Proceedings of 5th EUROMICRO workshop on Parallel and Distributed Processinig*, January 2001.

[GJ01] G.Bell and J.Gray. Crays, clusters and centers. In *Microsoft Research Technical Report, MSR-TR-2001-76*, 2001.

[GN92] C. J. Glass and L. M. Ni. The Turn Model for Adaptive Routing. *Proceedings of International Symposium on Computer Architecture*, pp. 278–287, 1992.

[Hor96] R. W. Horst. Servernet deadlock avoidance and fractahedral topologies. In *Proceedings of the International Parallel Processing Symposium*, pp. 274–280, April 1996.

[HP02] J. L. Henessy and D. A. Patterson. *Computer Architecture: A Quantitative Approach Third Edition.* Morgan Kaufmann, 2002.

[Int91] Intel. *Paragon XP/S Product Overview.* Beaverton, OR, Supercomputer Systems Division, 1991.

[I.T01] I.T.Association. Infiniband architecture. specification volumen 1,release 1.0.a. *available from the InfiniBand Trade Association, http://www.infinibandta.com*, June 2001.

[JA00] J.C.Sancho and A.Robles. Improving the Up*/Down* Routing Scheme for Networks of Workstations. In *Proceedings of the European Conference on Parallel Computing*, pp. 882–889, August 2000.

[JAJ00] J.C.Sancho, A.Robles, and J.Duato. A New Methodology to Compute Deadlock-Free Routing Tables for Irregular Networks. In *Proceedings of Communication and Architectural Support for Network-Based Parallel Computing*, pp. 45–60, January 2000.

[JAJ01] J.C.Sancho, A.Robles, and J.Duato. Effective Strategy to Compute Forwarding Tables for InfiniBand Networks. In *Proceedings of the International Conference on Paralel Processing*, pp. 48–57, January 2001.

[JAJ+02] J.C.Sancho, A.Robles, J.Flich, , P.Lopez, and J.Duato. Effective methodology for deadlock-free minimal routing in infiniband. In *Proceedings of the International Conference on Parallel Processing*, pp. 409–418, August 2002.

[Jea94] J.Kushkin and et al. The stanford flash multiprocessor. In *Proceedings of International Symposium on Computer Architecture*, pp. 302–313, April 1994.

[JF96] J.Carbonaro and F.Verhoorn. Cavallino: The teraflops router and NIC. In *Proceedings of Hot Interconnects Symposium IV*, pp. 157–150, August 1996.

[JFPJ00] J.C.Martinez, F.Silla, P.Lopez, and J.Duato. On the Influence of the Selection Function on the Performance of Networks of Workstations. In *Proceedings of the International Symposium on High Performance Computing*, pp. 292–300, October 2000.

[JL99] J.Wu and L.Sheng. Deadlock-Free Routing in Irregular Networks Using Prefix Routing. In *Proceedings of Parallel and Distributed Computing Systems*, pp. 424–430, August 1999.

[JMPJ99] J.Flich, M.P.Malumbres, P.Lopez, and J.Duato. Performance evaluation of networks of workstations with hardware shared memory model using execution-driven simulation. In *Proceedings of the International Conference on Parallel Processing*, pp. 146–153, October 1999.

[JMPJ02] J.Flich, M.P.Malumbres, P.Lopez, and J.Duato. Removing the latency overhead of the ITB mechanism in COWs with source routing. In *Proceedings of Euromicro Workshop on Parallel, Distributed and Network-based Processing*, pp. 463–470, 2002.

[JP92] J.M.Hsu and P.Banerjee. Performancde Measurement and Trace Driven Simulatioin of Parallel CAD and Numeric Application on a Hypercube Multicomputer. *IEEE Transaction on Parallel and Distributed Systems*, Vol. 3, No. 4, pp. 451–464, 1992.

[JPJ+02] J.Flich, P.Lopez, J.C.Sancho, A.Robles, and J.Duato. Improving InfiniBand Routing through Multiple Virtual Networks. In *Proceedings of International Symposium on High Performance Computing*, pp. 49–63, May 2002.

[JPMJ02] J.Flich, P.Lopez, M.P.Malumbres, and J.Duato. Boosting the Performance of Myrinet Networks. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, No. 7, pp. 693–709, July 2002.

[JSL02] J.Duato, S.Yalamanchili, and L.Ni. *Interconnection Networks: an engineering approach.* Morgan Kaufmann, 2002.

[JZA94] J.H.Kim, Z.Liu, and A.A.Chien. Compresionless routing: a frame work for adaptive and fault tolerant routing. *Proceedings of International Symposium on Computer Architecture*, pp. 289–300, 1994.

[KK79] P. Kermani and L. Kleinrock. Virtual cut-through: A new computer communication switching techniques. *Computer Networks*, Vol. 3, No. 4, pp. 267–286, 1979.

[KT95a] K.V.Anjan and T.M.Pinkston. An efficient fully adaptive deadlock recovery scheme: DISHA. In *Proceedings of International Symposium on Computer Architecture*, pp. 201–210, June 1995.

[KT95b] K.V.Anjan and T.M.Pinkston. DISHA: A deadlock recovery scheme for fully adaptive routing. In *Proceedings of International Parallel Processing Symposium*, pp. 537–543, April 1995.

[KTJ96] K.V.Anjan, T.M.Pinkston, and J.Duato. Generalized theory for deadlock-free adaptive routing and its application to Disha Concurrent. In *Proceedings of International Parallel Processing Symposium*, pp. 815–821, April 1996.

[LH91] D. H. Linder and J. C. Harden. An adaptive and fault tolerant wormhole routing strategy for k-ary n-cubes. *IEEE Transaction on Computer*, Vol. 40, No. 1, pp. 2–12, 1991.

[L.S00] L.Schwiebert. A Performance Evaluation of Fully Adaptive Wormhole Routing including Selection Function Choice. In *IEEE International Performance, Computing, and Communications Conference*, pp. 117–123, February 2000.

[MAAH01a] M.Koibuchi, A.Funahashi, A.Jouraku, and H.Amano. L-turn routing: An adaptive routing in irregular networks. In *Proceedings of the International Conference on Parallel Processing*, pp. 374–383, September 2001.

[MAAH01b] M.Koibuchi, A.Jouraku, A.Funahashi, and H.Amano. MMLRU selection function: An Output Selection Function on Adaptive Routing. In *Proceedings of ISCA 14th International Conference on Parallel and Distributed Computing Systems*, pp. 1–6, August 2001.

[Mae91] M.D.Schroeder and al et. Autonet: a high-speed, self-configuring local area network using point-to-point links. *IEEE Journal on Selected Areas in Communications*, Vol. 9, pp. 1318–1335, 1991.

[MAH02a] M.Koibuchi, A.Jouraku, and H.Amano. Deterministic routing techniques by dividing into sub-networks in irregular networks. In *the IASTED International Conference on Networks, Parallel and Distributed Processing, and Applications*, pp. 143–148, October 2002.

[MAH02b] M.Koibuchi, A.Jouraku, and H.Amano. The impact of path selection algorithm of adaptive routing for implementing deterministic routing. In *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, pp. 1431–1437, June 2002.

[MDW93] M.Noakes, D.A.Wallach, and W.J.Dally. The j-machine multicomputer: An architectural evaluation. In *Proceedings of International Symposium on Computer Architecture*, pp. 224–235, May 1993.

[Mea96] M.Snir and et al. *MPI: The Complete Reference.* MIT Press, 1996.

[M.G97] M.Galles. Spider: A high speed network interconnect. *IEEE Micro*, Vol. 17, No. 1, pp. 34–39, 1997.

[MJ80] M.P.Merlin and J.P.Schweitzer. Deadlock Avoidance in Store-and-Forward Networks. *IEEE Transactions on Computers*, Vol. COM-28, No. 3, pp. 345–354, 1980.

[MMM88] M.J.Litzkow, M.Livny, and M.W.Mutka. Condor - a hunter of idle workstations . In *Proceedings of the 8th International Conference on Distributed Computing Systems*, pp. 104–111, June 1988.

[N.J95] N.J.Boden and et al. Myrinet: A Gigabit-per-Second Local Area Network. *IEEE Micro*, Vol. 15, No. 1, pp. 29–35, 1995.

[NJH+00a] N.Tanabe, J.Yamamoto, H.Nishi, T.Kudoh Y.Hamada, H.Nakajo, and H.Amano. MEMOnet: network interface plugged into a memory slot. In *Proceedings of IEEE International Conference on Cluster Computing*, pp. 17–26, 2000.

[NJH+00b] N.Tanabe, J.Yamamoto, H.Nishi, T.Kudoh Y.Hamada, H.Nakajo, and H.Amano. On-the-fly sending: a low latency high bandwidth message transfer mechanism. In *Proceedings of the International Symposium on Parallel Architectures, Algorithms, and Networks*, pp. 186–193, 2000.

[NKN+01] S. Nishimura, T. Kudoh, H. Nishi, J. Yamamoto, K. Harasawa, N. Matsudaira, S. Akutsu, K. Tasho, and H. Amano. RHiNET-3/SW: an 80-Gbit/s high-speed network switch for distributed parallel computing. In *Hot Interconnect*, pp. 119–123, 2001.

[Oed93] W. Oed. The Cray Research Massively Parallel Processing System: Cray T3D. *Cray Research*, 1993.

[PFH01] F. Petrini, W.C. Feng, and A. Hoisie. The Quadrics network (QsNet): high-performance clustering technology. In *Proceedings of Hot Interconnects*, pp. 125–130, August 2001.

[PJJ01] P.Lopez, J.Flich, and J.Duato. Deadlock-free Routing in $InfiniBand^{TM}$ through Destination Renaming. In *Proceedings of the International Conference on Parallel Processing*, pp. 427–434, September 2001.

[PPD01] P.Shivam, P.Wyckoff, and D.Panda. Emp: Zero-copy os-byoass nic-driven giga bit ethernet message passing. In *Proceedings of the IEEE Supercomputing Conference*, 2001.

[RADG94] R.Felderman, A.DeSchon, D.Cohen, and G.Finn. ATOMIC: A high-speed local communication architecture. *Journal of High Speed Networks*, Vol. 3, No. 1, pp. 1–28, 1994.

[RS91] T.L. Rodeheffer and M.D. Schroeder. Automatic reconfiguration in Autonet. *Technical Report SRC research report 77,DEC*, September 1991.

[SB97] L. Schwiebert and R. Bell. The Impact of Output Selection Function Choice on the Performance of Adaptive Wormhole Routing. In *Proceedings of International Conference on Parallel and Distributed Computing Systems*, pp. 539–544, October 1997.

[SD93] L. Schwiebert and D.N.Jayasimha. Optimal fully adaptive wormhole routing for meshes. In *Proceedings of Supercomputing Conference*, pp. 782–793, November 1993.

[SD95] L. Schwiebert and D.N.Jayasimha. Optimal fully adaptive minimal wormhole routing for meshes. *Journal of Parallel and Distributed Computing*, Vol. 27, pp. 56–70, 1995.

[SD99] F. Silla and J. Duato. Is it Worth the Flexibility Provided by Irregular Topologies in Networks of Workstations? In *Proceedings of Workshop Comm. and Architectural Support for Network-Based Parallel Computing*, pp. 47–61, January 1999.

[SD00] F. Silla and J. Duato. High-Performance Routing in Networks of Workstations with Irregular Topology. *IEEE Transactions on parallel and distributed systems*, Vol. 11, No. 7, pp. 699–719, 2000.

[Sea99] T. L. Sterling and et al. *How to Build a Beowulf.* MIT Press, 1999.

[Sea01] T. L. Sterling and et al. *Beowulf Cluster Computing with Linux.* MIT Press, 2001.

[SLT02] T. Skeie, O. Lysne, and I. Theiss. Layered Shortest Path (LASH) Routing in Irregular System Area Networks. In *Proceedings of International Parallel and Distributed Processing Symposium*, pp. 162–169, April 2002.

[SRD01] J.C. Sancho, A. Robles, and J. Duato. Improving Minimal Adaptive Routing in Networks with Irregular Topology. In *Proceedings of ISCA 13th International Conference on Parallel and Distributed Computing Systems*, August 2001.

[ST96] S. L. Scott and G. T.Horson. The Cray T3E network: adaptive routing in a high performance 3D torus. In *Proceedings of Hot Interconnects IV*, pp. 147–156, August 1996.

[ST97] S.Warnakulasuriya and T.M.Pinkston. Characterization of deadlocks in interconnection networks. In *Proceedings of IEEE Symposium on Parallel and Distributed Processing*, pp. 80–86, April 1997.

[ST99] S.Warnakulasuriya and T.M.Pinkston. characterization of deadlocks in irregular networks. In *Proceedings of the International Conference on Parallel Processing*, pp. 75–84, October 1999.

[STH+00] S.Nishimura, T.Kudoh, H.Nishi, J.Yamamoto, K.Harasawa, N.Matsudaira, S.Akutsu, and H.Amano. 64-Gbit/s Highly Reliable Network Switch (RHiNET-2/SW) Using Parallel Optical Interconnection. *IEEE Journal of Lightwave Technology*, Vol. 18, No. 12, pp. 1620–1627, 2000.

[T. 95] T. E.Anderson and D. E.Culler and D. A.Patterson. A case for NOW (Networks of Workstations). *IEEE Micro*, Vol. 15, No. 1, pp. 54–64, 1995.

[TSJ+99] T.Kudoh, S.Nishimura, J.Yamamoto, H.Nishi, O.Tatebe, and H.Amano. RHiNET: A network for high performance parallel computing using locally distributed computing. In *Proceedings of IWIA*, pp. 69–73, November 1999.

[Wea94] W.J.Dally and et al. The reliable router: A reliable and high-performance communication substrate for parallel computers. In *Proceedings of the Workshop on Parallel Computer Routing and Communications*, pp. 241–255, May 1994.

[WK97] W.C.Feng and K.G.Shin. Impact of Selection Functions on Routing Algorithm Performance in Multicomputer Networks. *Proceedings of 11th ACM International Conference on Supercomputing*, pp. 132–239, July 1997.

[Wu96] J. Wu. An Optimal Routing Policy for Mesh-Connected Topologies. *Proceedings of International Conference on Parallel Processing*, Vol. 1, pp. 267–270, 1996.

[Wu99] J. Wu. Maximum-shortest-path (MSP): an optimal routing policy for mesh-connected multicomputers. *IEEE Transaction on Reliability*, Vol. 48, No. 3, pp. 247–255, 1999.

[YAA+01] Y.Yang, A.Funahashi, A.Jouraku, H.Nishi, H.Amano, and T.Sueyoshi. Recursive Diagonal Torus: an interconnection network for massively parallel computers. *IEEE Transaction on Parallel and Distributed Systems*, Vol. 12, No. 7, pp. 701–715, 2001.

[鯉渕 99] 鯉渕 道紘. 適応型ルーティングにおける output selection function. 1999 年度慶應義塾大学卒業論文, 1999.

[鯉渕 00] 鯉渕 道紘, 舟橋 啓, 上樂 明也, 天野 英晴. 適応型ルーティングにおける output selection function. 並列処理シンポジウム JSPP'2000 予稿集, pp. 181–188, May 2000.

[鯉渕 01] 鯉渕 道紘, 舟橋 啓, 上樂 明也, 天野 英晴. 適応型ルーティングにおける output selection function に関する研究. 情報処理学会論文誌, Vol. 42, No. 4, pp. 704–713, 2001.

[鯉渕 02a] 鯉渕 道紘, 上樂 明也, 天野 英晴. PC ネットワークにおける仮想チャネルを用いた固定ルーティング手法. 電子情報通信学会技術研究報告コンピュータシステム, pp. 17–22, August 2002.

[鯉渕 02b] 鯉渕 道紘, 上樂 明也, 天野 英晴. イレギュラーネットワークにおける仮想チャネルを用いた固定ルーティング. 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. 43, No. SIG 6(HPS 5), pp. 112–121, September 2002.

[鯉渕 02c] 鯉渕 道紘, 上樂 明也, 天野 英晴. イレギュラーネットワークにおける仮想チャネルを用いた固定ルーティング. 並列処理シンポジウム JSPP 予稿集, pp. 43–50, May 2002.

[住元 00] 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, 2000.

[上樂 99] 上樂 明也. 命令レベルシミュレーションによる 相互結合網の評価. 1999 年度慶應義塾大学大学院修士論文, 1999.

[上樂 00] 上樂 明也, 舟橋 啓, 鯉渕 道紘, 若林 正樹, 天野 英晴. 命令レベルシミュレーション による adaptive routing の評価. 情報処理学会技術研究報告 ARC, pp. 47–52, March 2000.

[西 宏 00] 西 宏章, 西村 信治, 多昌 廣治, 工藤 知宏, 天野 英晴. 効率良い並列処理をサポートするローカルエリア向けネットワークスイッチ. 電子情報通信学会論文誌, Vol. J83D-I, No. 7, 2000.

[天野 96] 天野英晴. 並列コンピュータ. 昭晃堂, 1996.

[田中 97] 田中 良夫, 久保田 和人, 佐藤 三久, 関口 智嗣. Collective 通信を用いたデータ並列プログラムの性能予測. 情報処理学会技術研究報告 HPC, 第 97 巻, pp. 69–74, 1997.

[土屋 02] 土屋 潤一郎. PC/WS クラスタ構築の為のネットワークインタフェース チップ Martini の実装. 2002 年度慶應義塾大学大学院修士論文, 2002.

[堀江 92] 堀江 健志, 石畑 宏明, 池坂 守夫. 並列計算機 AP1000 における相互結合網のルーチング方式. 電子情報通信学会論文誌, Vol. J75-D-1, No. 8, pp. 600–606, 1992.

付 録A　規則網と規則網における適応型アルゴリズムのサーベイ

第2章，第3章，第4章および第5章に関連して，大規模並列計算機およびPCクラスタの規則網とその上で用いられる既存の適応型アルゴリズムについて説明する．

A.1　規則網

大規模並列計算機は様々な形態とその名称があるが，以後，単純に計算システムの構成単位をノードと呼ぶ．また，PCクラスタの場合は1つのスイッチとそのスイッチに接続しているPCをノードと呼ぶ．

A.1.1　n次元メッシュ

ノード間を格子状に接続したトポロジをメッシュと呼ぶ．

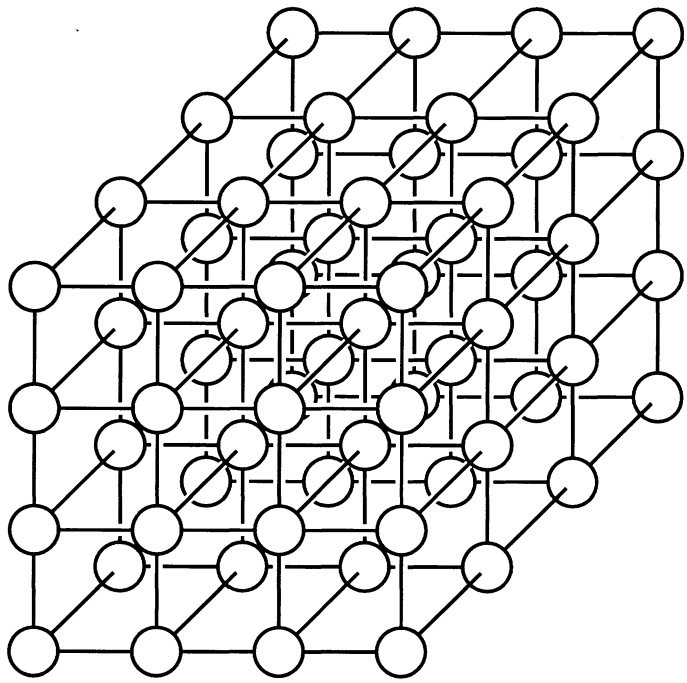


図 A.1: n次元メッシュ(n = 3)

図A.1に，3次元メッシュを示す．図A.1では，各次元毎のノード数は等しいが，等しくなくてもn次元メッシュと呼べる．メッシュはネットワークの端にあるノードとそうでないノードとでは，隣接するノード数が異なるという特徴を持つ．これまで多くのマルチコンピュータで2次元もしくは3次元メッシュが用いられてきた (MIT Alewife[Aea90], Intel Paragon[Int91], Stanford DASH[Dea92], MIT J-Machine[MDW93], Stanford FLASH[Jea94], MIT Reliable Router[Wea94]).

A.1.2 k-ary n-cube

メッシュにおいて全てのノードに対して隣接するノード数が等しくなるように，端のノード間を結んだトポロジをトーラスと呼ぶ．トーラスは各次元のノードをリングで結合した構造を持つ．図A.2は2次元トーラスの例である．

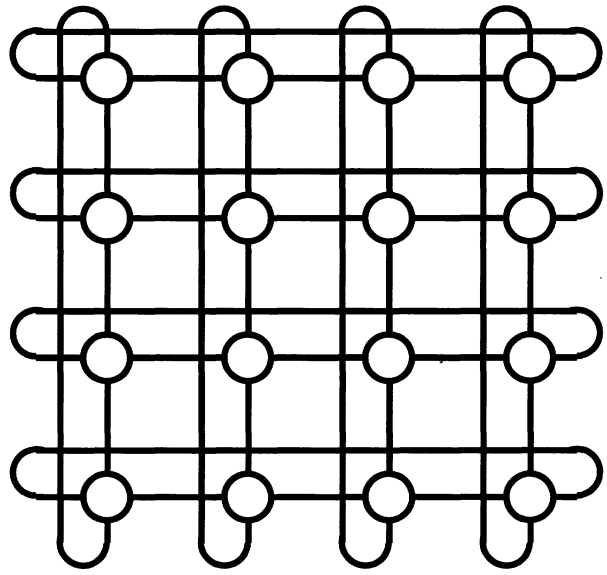


図 A.2: n次元トーラス (n = 2)

k-ary n-cubeは，このトーラスを一般化したもので，各ノードの番号をn桁のk進数で表現して，それぞれの桁 (次元) 方向をリングで結んだトポロジである．したがって，図A.2のトーラスは4-ary 2-cubeとなる．図A.3に4-ary 3-cubeの構成を示す．

これまで商用機としてはCray T3D[Oed93]およびCray T3E[Oed93]がk-ary n-cubeを採用している．

A.1.3 ハイパーキューブ

ハイパーキューブは80年代のNORA(no remote memory access model:無遠隔アクセスモデル) マシンにおいて最もよく用いられた結合網である．図A.4で示すように，ノードをn桁の2進数で表現し，それぞれの桁が1bit異なるもの同士をリンクで結ぶ．図A.4

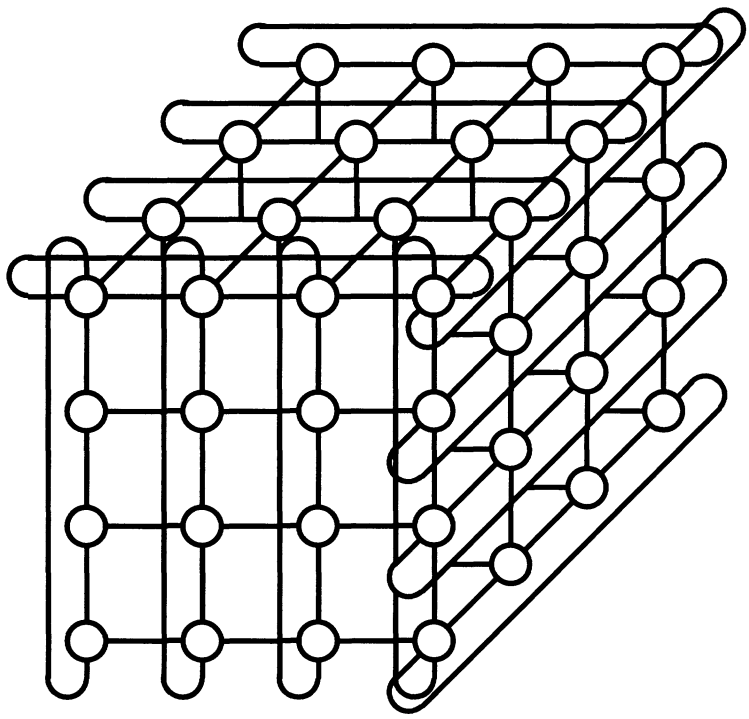


図 A.3: k-ary n-cube(k = 4, n = 3)

は16ノードの例なので，たとえばノード0101は1101, 0001, 0111, 0100の4つのノードとの間にリンクを持つ．ハイパーキューブは 2-ary n-cube と同一の形状になるため，k-ary n-cube の1つとして考えることもできる．

次数は2進数で表した時の桁数に等しいので，全体のノード番号を N とすると $n = \log_2 N$ となる．

A.1.4 Fat ツリー

fat ツリー [C.E85] は図 A.5 に示すように，多重化されたツリーで，ツリーのルート方向へのリンク数 p，ツリーのリーフ方向へのリンク数 q，及び階層数 r の組 (p, q, r) 定義される．図 A.5 は (2,4,2) の網で16プロセッサを結合する．一般的に fat ツリーは $N = q^r$ 個の PC で結合することができる．また，$p = 1$ の場合は単純な q 進木になる．fat ツリーは2002年10月現在，QsNET[PFH01][FFA+02] で用いられている．

A.2 規則網における適応型アルゴリズム

次に規則網における代表的な適応型アルゴリズムについて述べる．

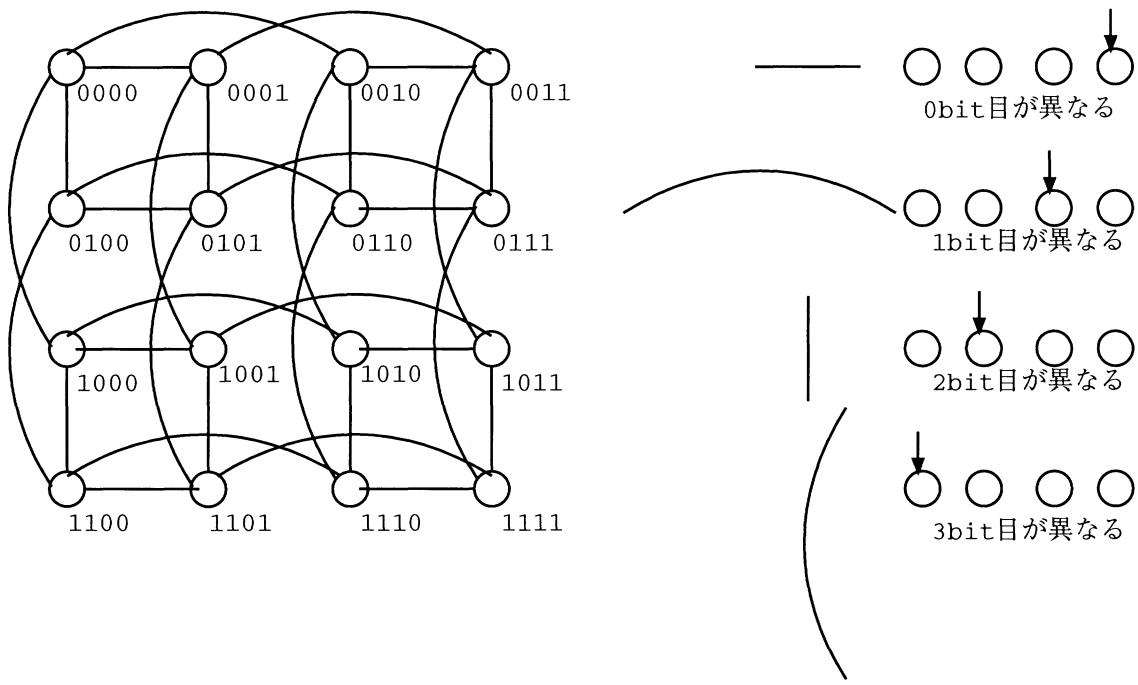


図 A.4: ハイパーキューブ

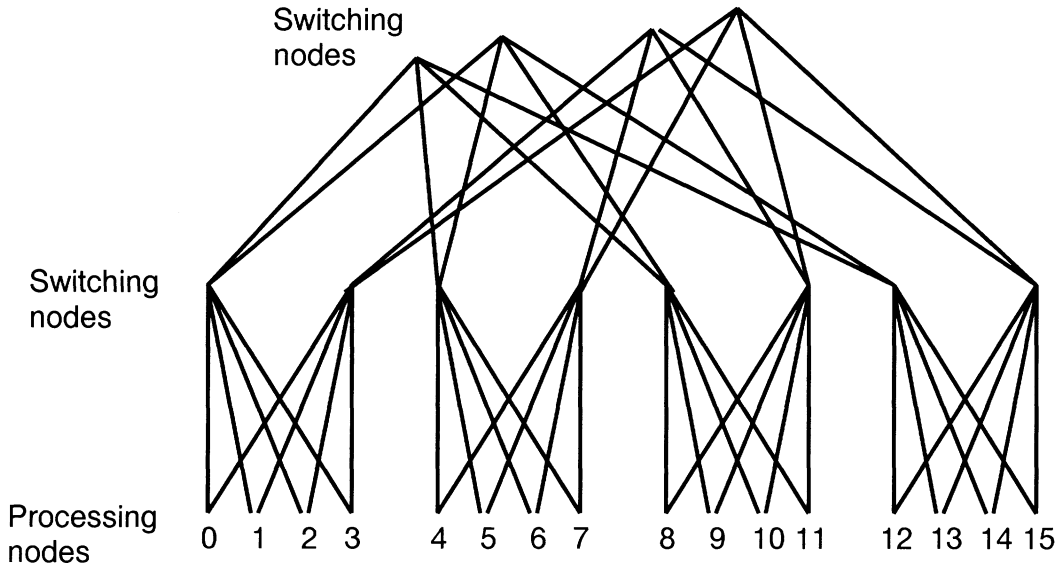


図 A.5: Fat ツリー

A.2.1 Turn モデル

デッドロックが生じるのは, 結合網内のバッファが論理的に循環構造を作ってしまうためである. Glass らによる Turn モデル [GN92] は, 循環を生じないように, パケットがルーティング中に方向を変えるパターンを制限する方法である. このモデルは論理的な循環構造に着目し, トポロジが次元 (方向) をもつものであれば適用することが可能である. Turn モデルは次の方法で適応型アルゴリズムを作る. ここで, チャネルは物理的なものと考えても仮想チャネルと考えてもよい.

(a) 結合網中のチャネルをパケットの転送方向に分ける. 各ノードが 1 つの物理的な方向に対し v 個のチャネルを持つ場合, これらは, v 個の論理的な方向として区別する.

(b) ある方向から別の方向への進路変更 (Turn) を洗い出し, 識別する. 0 度あるいは 180 度の進路変更は無視する.

(c) 進路変更 (Turn) によって構成される循環構造 (Cycle) を識別する. 一般的にはトポロジー上での各平面で行なえばよい.

(d) デッドロックを防ぐように, 各循環構造について 1 つ進路変更に禁止条件を加える. 循環構造はいくつかの循環の複合で生じる場合があるので, 禁止条件は慎重にチェックして決めなければならない.

(e) トーラスの場合はネットワークの端で折り返しを行うラップアラウンドチャネルが存在するが, ラップアラウンドチャネルを使った進路変更も, 禁止条件をやぶらないようにする.

(f) 0 度あるいは 180 度の進路変更を禁止条件をやぶらないように組み込む.

最も単純な 2 次元メッシュでの Turn モデルを考える. この場合, 可能な循環構造 (Step 3) は, 図 A.6(a) に示す二種類になる. ここで, Turn モデルに従い, 各循環の進路変更を 1 つずつ禁止し, デッドロックを防いだ場合を図 A.6(b) に示す. 図中の点線の進路変更が禁止されている. このルーティング方法は, 先に西方向にパケットを送ることから west-first と呼ばれる. デッドロックを防ぐ切り方は 1 つではなく, たとえば図 A.6(c) に示す切り方 (north-last) でも, デッドロックを防ぐことができる.

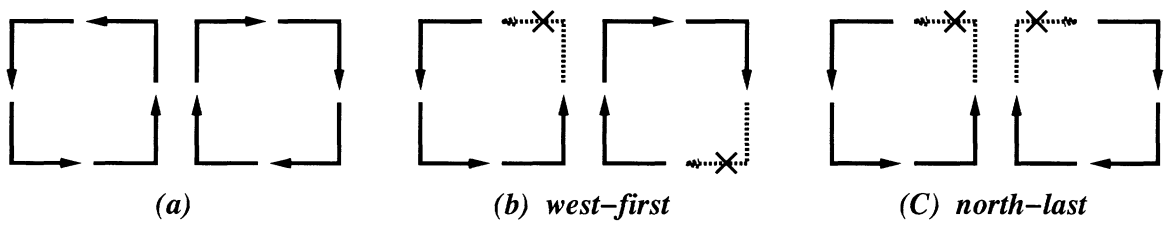


図 A.6: Turn モデル

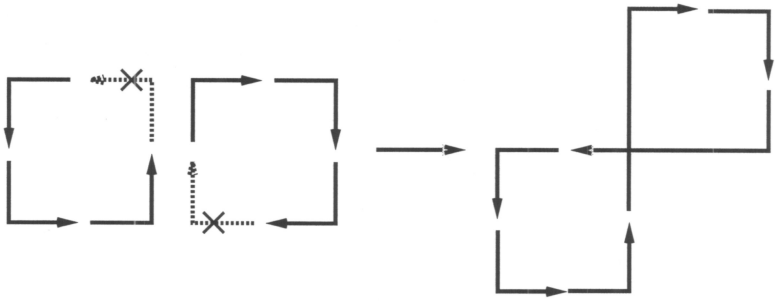


図 A.7: Turn モデルの失敗した切り方

しかし, どこを切ってもよいというわけではない. 図 A.7 に示すように切ると, 切ったはずの循環が, 8 の字型に複合して新たな循環が生じてしまう. このような状況を配慮しつつ禁止条件を与える必要がある.

e-cube ルーティング (x 方向優先) を Turn モデルで考えると, 図 A.8 に示すように, この循環のうち 4 つの進路変更しか許していないことになる. これは確かに循環を切ることができるのでデッドロックを生じないが, 制限が厳しく代替経路を失っていることがわかる. Turn モデルにより生成された west-first や north-last アルゴリズムを使えば, パケットは 6 方向に進むことが許されるため, 図 A.9 に示すように, 故障地点や混雑地点を迂回する適応型アルゴリズムが可能になる.

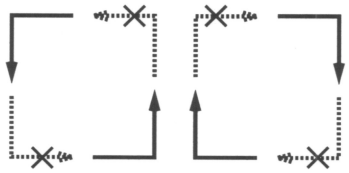


図 A.8: E-cube ルーティングの Turn モデル

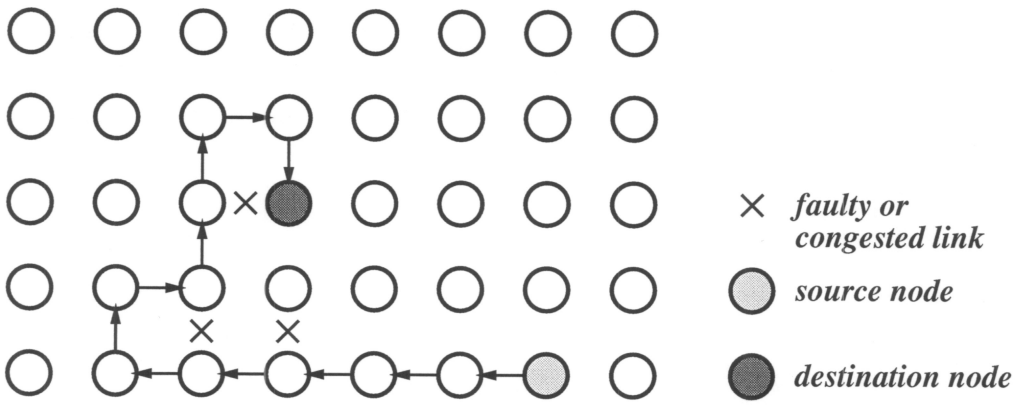


図 A.9: West-first での混雑の回避

A.2.2 Double y/Opt y ルーティング

k-ary n-cube などのトポロジでは，仮想チャネルを必要数設けて独立なサブネットワークを作ることで，デッドロックを起こさずに，複数の経路を自由に選ぶことができる．図 A.10 に Linder らによる最短型適応ルーティングである double y ルーティング [LH91] を示す．ここでは簡単のため 2 次元メッシュを考える．X 方向には通常の双方向のチャネルを持ち，Y 方向にのみ双方向の仮想チャネルを二重に設ける (図 A.10)．ここで，全体のネットワークを x が増加する方向のチャネルと，Y 方向の片方のチャネルを組として $+X$ サブネット，x が減少する方向のチャネルと，Y 方向のもう片方のチャネルを組として $-X$ サブネットの 2 つに分割する．

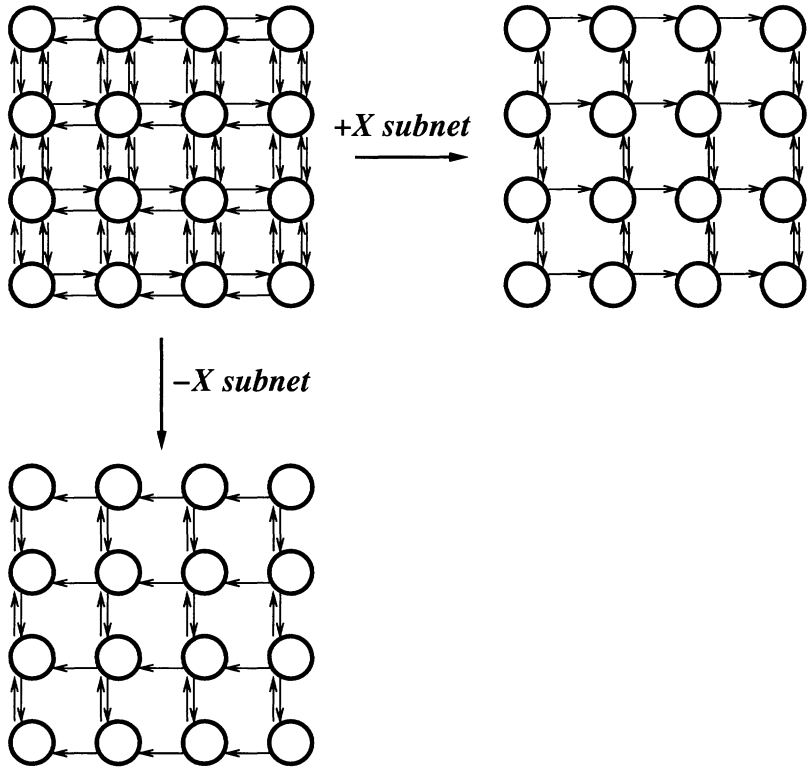


図 A.10: Double y ルーティング

このようにして，x が増加する方向のノードにパケットを送る場合は，$+X$ サブネットを使い，減少する方向のノードにパケットを送る場合は，$-X$ サブネットを使えば，途中の経路はどうあれ，デッドロックなしで目的地に到着することができる．この方法により図 A.11 に示すように混雑を迂回することができる．

この方法では最短経路からの迂回は考えていない．独立なサブネットを用いることにより，チャネルの循環は完全に排除されているので，デッドロックが生じないことは明らかである．この方法は単純かつ強力だが，次元数が増えると，その分独立サブネットワークを作るのに仮想チャネルが必要になる．n 次元のメッシュに対しては 2^{n-1}，n 次元のトーラスに対しては $2^{n-1}(n+1)$ 本必要になる．これは余りにも多過ぎるため，Cheien らは，経

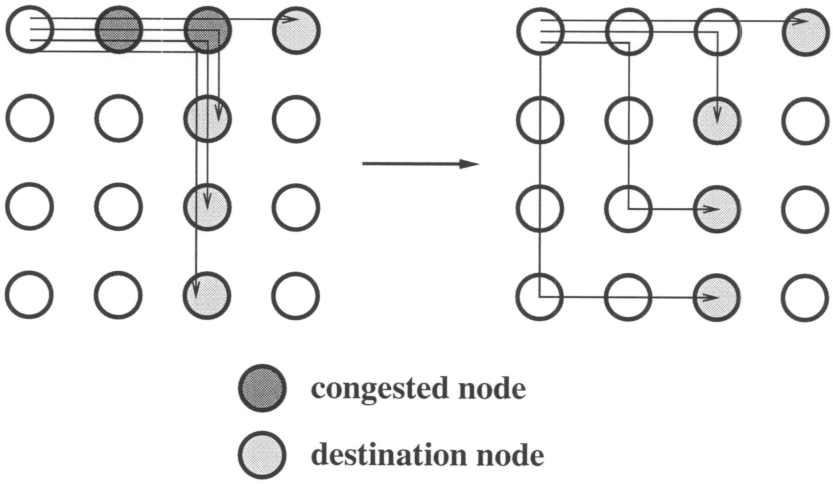


図 A.11: 混雑の迂回

路を変える次元を平面に固定する Planar 適応型アルゴリズムを提案した [CK92]. Planar 適応型アルゴリズムは迂回経路は平面に制限される代わりに, チャネル数は次元に依存せず, メッシュならば 3, トーラスならば 6 となる.

また, double y ルーティングから不必要な制限を除去し, 自由度をあげたものとして opt y ルーティングがある [SD93][SD95]. opt-y ルーティングはトポロジを 2 次元メッシュに限定しているが, 2 次元メッシュにおいて最も自由度の高いデッドロックフリールーティングである.

図 A.12 に opt-y ルーティングの Turn モデル を各々示す.

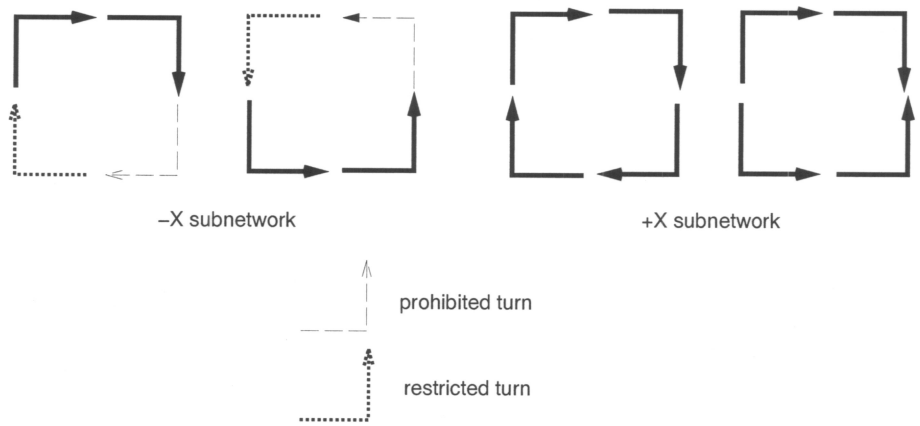


図 A.12: Opt y ルーティングの Turn モデル

図 A.12 において, $-X$ 方向から Y 方向へのターンはパケットが $-X$ 方向への移動が終了している場合にのみ使用することができる. また, opt-y ルーティングでは $-X$ subnetwork の Y 方向と $+X$ subnetwork の Y 方向間の 0 度のターンについても同様の制限がある.

double y ルーティングの Turn モデルを図 A.13 に示すが, opt y ルーティングは double

y ルーティングに比べ自由度が高いことが一目瞭然である.

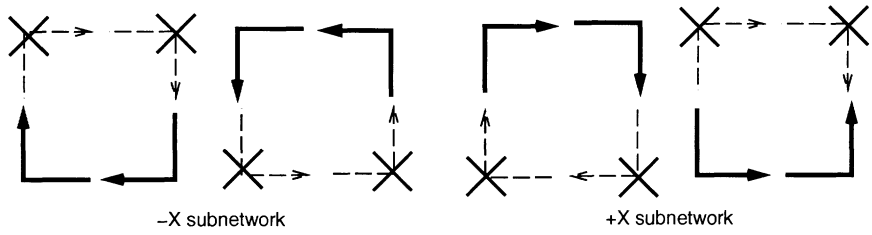


図 A.13: Double y ルーティングの Turn モデル

A.2.3　次元逆転 (dimension reversal) ルーティング

Dally らは, 仮想チャネルの利用による単純で現実的な迂回型ルーティングを提案し, k-ary n-cube に適用した [DA93]. この方法では e-cube ルーティング, すなわち転送する次元の順番を決めておくルーティングを基本とする. 次元逆転ルーティングでは, チャネル i を用いて転送している時に, 次に転送する次元方向のチャネルが空いていない場合, e-cube ルーティングの順番に従うことなく, 任意の次元方向にパケットを送ることをゆるす. ただし, 次元の順番が逆転する場合は, チャネル $(i+1)$ に対して転送しなければならない.

この方法を用いると, 次元の順番に従わない転送を行なうたびに, チャネルの番号はどんどん大きくなっていく. (仮想チャネル数 -1) の番号のチャネルに到達すると, これ以上適応型アルゴリズムはできなくなり, e-cube ルーティングに従って, 次元の順番を守って固定的にルーティングが行なわれる. この方法を静的次元逆転ルーティングと呼ぶ.

静的次元逆転ルーティングでは, 次元の順番を逆転を起こすと常にチャネル番号が増えるため, 次元逆転方向のルーティング回数が制限される. これに対して, 動的次元逆転ルーティングは, 適応型アルゴリズム用チャネルと, 固定ルーティング用チャネルを複数本ずつわけて持つ. 固定ルーティング用のチャネルは, e-cube ルーティングに従う固定ルーティングしかできないのに対し, 適応型アルゴリズム用のチャネルは, どの次元方向にも送ることができる. ただし, 現在使っているチャネル番号が i で, 行き先の 0 から i までがすべて塞がっていた場合, これらを待つことはできず, $(i+1)$ 以上のチャネルを使わなければならない ($(i+1)$ 以上のチャネルなら待つことができる). 適応型アルゴリズム用チャネルの最大番号を使っている時に, 他のすべての適応型用チャネルが塞がっている場合は, パケットは固定ルーティング用チャネルに送られ, 以降固定ルーティングが行なわれる. このように動的次元逆転ルーティングは, 運がよければ何回も次元の逆転を行なうことができる.

A.2.4　Duato の必要十分条件 (Duato's protocol)

double y, Turn モデル, 次元逆転ルーティングなどのルーティング法は, サブネットの分離, 転送方向の制限, 仮想チャネルの使用などの方法で循環を断ち切ることによって, デッドロックを起こさない適応型アルゴリズムを実現した.

これに対して Duato は, 循環を含む経路に対しても, デッドロックを起こさない適応型

アルゴリズムの必要十分条件を示し [Dua93][Dua94] [Dua95], これに基づく適応型アルゴリズムを k-ary n-cube に適用した. Duato の必要十分条件の概要は次の通りである.

図 A.14 に示すリング状の結合網を考える. ここで, 仮想チャネル C_{A0-3} は単方向でリングを一周し, 仮想チャネル C_{H0-2} は, ラップアラウンドループを欠いている. この結合網は, 次の単純な方法でデッドロックを防止することができる.

仮想チャネル C_{A0-3} は, どのノードからどの目的地へも自由にパケットを送ることができる. 仮想チャネル C_{H0-2} は, 現在のノードよりも目的地の番号が大きい時のみ使うことができる. パケットは, この 2 つの条件を満たす限り, C_{A0-3}, C_{H0-2} のどちらか先に空いた方を使って転送する.

この場合, C_{H0-2} は目的地の番号が大きい時にのみ使うことができるので, C_{H2} は塞がり続けることはない. したがって, これに向かう C_{H1}, C_{H0} も塞がり続けることはない. C_{A0-3} は循環構造を作るが, 自分自身のノードにパケットを送らない限り, 途中の C_{H0}-C_{H2} を常に逃げ道として使うことができる. したがってデッドロックは起こらない.

すなわち, この結合網は, デッドロックしない逃げ道 C_{H0}-C_{H2} が用意され, この逃げ道により循環のどこか (C_{A0}) が切られたことにより, デッドロックしない経路 (C_{A1}-C_{A3}) を生じ, これらの経路によりどのノードからどのノードへもパケットを送ることができる.

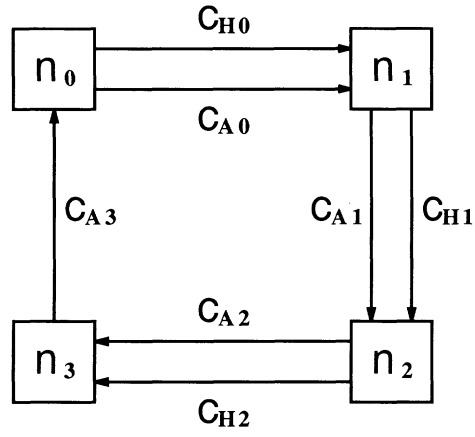


図 A.14: リング状の結合網での Duato's protocol

トーラスなどのラップアラウンドの存在する k-ary n-cube で e-cube ルーティングを用いる場合, 先に示したように, 必ず各方向に仮想チャネルを 2 本用意しなければならない. 固定の e-cube ルーティングでは, 最初仮想チャネル番号を 1 にしておいてラップアラウンドを通過すると, 0 に変更してデッドロックを防ぐが, この方法だと結合網のほとんどの部分では, 仮想チャネルが 1 本しか使われない状態となる. ところが Duato の方法を適用すると, 片方のチャネルについて, 番号の大きい方にのみ送れるようにする簡単な条件を付加するだけで, 結合網の多くの部分で, 両方のチャネルを有効に利用することができる. このため, ハードウェア量の増加を抑え, かつ, 転送容量の大幅な改善が見込まれる.

Duato の条件をまとめると,

(1) 結合網全体に渡る循環のない逃げ道 (escape path) を用意する.

(2) 逃げ道と, 逃げ道により循環が切断されてデッドロックが起きなくなる他の経路によっ

て結合網中のどのノード間でもパケットが送れるようにする.
となり, これらを満足できれば, (1)(2) の経路を適応的に選択することにより, デッドロックしない適応型アルゴリズムが可能になるので, 様々の結合網に応用することができる. 次に,k-ary n-cube での適応型アルゴリズムを示す.

Duato は，次の手順により，k-ary n-cube での適応型アルゴリズムを実現した.

(a) はじめに単方向リング (図 A.14) のネットワークについて考える. A.2.4 節で示したように, パケットが存在しているノードより目的地のノード番号が大きい場合は, チャネル C_H を使い, 低い場合は C_A を使用することで escape path が保証される. escape path が全てのノード間に存在するため, このネットワークはデッドロックフリーである.

(b) 次に, ネットワークを双方向リングに拡張する. しかし, Duato の方法では, ルーティングは常に最短経路を通るため, ルーティングの途中でそれまで使用していたリンクと逆方向のリンクを使用することはありえず, 逆方向のリンクは全く別のネットワークとして分離して考えることができる (図 A.15). よって, 双方向リングのネットワークでもデッドロックフリーである.

(c) 次に, ネットワークを多次元のものに拡張する. e-cube ルーティング [DS87] と同様に, 次元の使用順を固定すると, 結局は双方向 (単方向) リングを決まった順に使用するだけになる (図 A.16). よって, ルーティングのときに使用されるチャネル間の依存関係は各次元で独立であるため, デッドロックフリーである.

この作業により, escape path C_1 が用意された.

(d) ここで, 仮想チャネル $C_F(Fully\ \ adaptive)$ を新たに用意し, そのチャネルでは次元の使用順を無視して移動可能とする. これにより, デッドロックフリー な適応型アルゴリズムが可能となる.

ただしこのルーティングでは, 2. を満足させるために, 常に最短経路を通ることをを守らなければならない. 図 A.17 に k-ary 2-cube での仮想チャネルの使用例を示す.

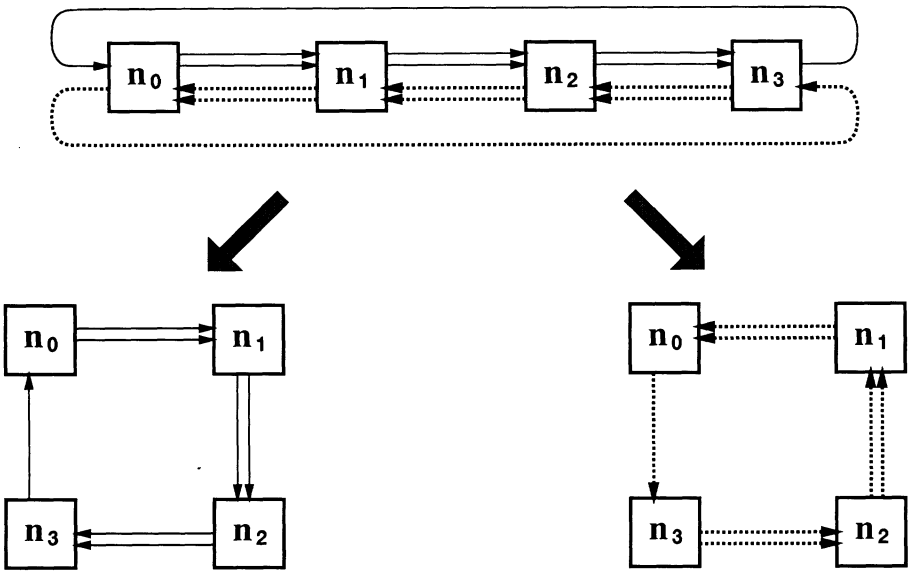


図 A.15: 双方向リングでの Duato's protocol

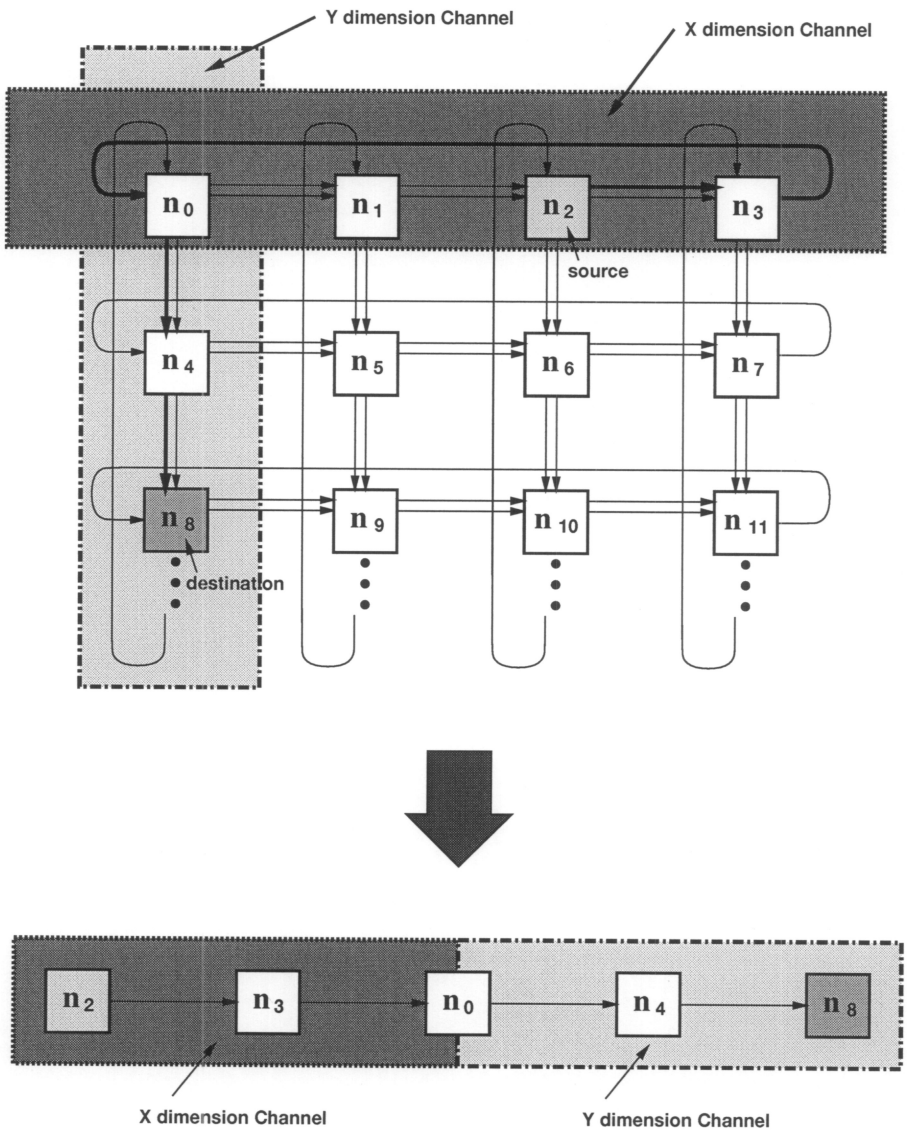


図 A.16: Duato's protocol の多次元への拡張

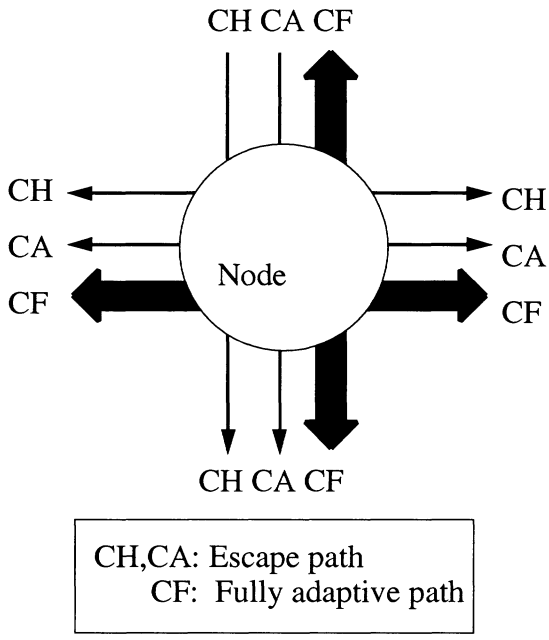


図 A.17: Duato's protocol の仮想チャネルの使用例

付 録B　論文目録

B.1　本研究に関する論文

B.1.1　公刊論文

鯉渕 道紘, 舟橋 啓, 上樂 明也, 天野 英晴
適応型ルーティングにおける output selection function に関する研究
情報処理学会論文誌 42 号 4 巻, pp.704-713, 2001

鯉渕 道紘, 舟橋 啓, 上樂 明也, 天野 英晴
L-turn routing: Irregular Network における Adaptive Routing
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.42 No.SIG9(HPS 3), pp.119-134, 2001

鯉渕 道紘, 上樂 明也, 天野 英晴
イレギュラーネットワークにおける仮想チャネルを用いた固定ルーティング
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.43 No.SIG 6(HPS 5), pp.112-121, 2002.

Akira Funahashi and Michihiro Koibuchi and Akiya Jouraku and Hideharu Amano
"The Impact of Output Selection Function on Adaptive Routing",
ISCA Information: An International Journal, Vol 4, No.4, pp. 541-550, 2001

B.1.2　国際会議，査読付きシンポジウム

鯉渕 道紘, 舟橋 啓, 上樂 明也, 天野 英晴
適応型ルーティングにおける output selection function, 並列処理シンポジウム JSPP 2000 論文集, pp.181-188, June. 2000

Akira Funahashi and Michihiro Koibuchi and Akiya Jouraku and Hideharu Amano
"The Impact of Output Selection Function on Adaptive Routing", ISCA 16th International Conference on Computers And Their Applications (CATA-2001), pp.241-246, Mar. 2001

鯉渕 道紘, 舟橋 啓, 上樂 明也
適応型ルーティングにおける output selection function の提案, 並列処理シンポジウム JSPP 2001 論文集, pp.255-262, June. 2001

舟橋 啓,鯉渕 道紘, 上樂 明也
Irregular Network における Adaptive Routing の提案, 並列処理シンポジウム JSPP'2001 論文集, pp.247-254, June. 2001

Michiihro Koibuchi and Akiya Jouraku and Akira Funahashi and Hideharu Amano "MMLRU selection function: An Output Selection Function on Adaptive Routing", ISCA 14th International Conference on Parallel and Distributed Computing Systems (PDCS-2001), pp.1-6, Aug. 2001

Michiihro Koibuchi and Akira Funahashi and Akiya Jouraku and Hideharu Amano "L-turn routing: An Adaptive Routing in Irregular Networks", the 2001 International Conference on Parallel Processing(ICPP'01), pp.384-393, Sep. 2001

Akiya Jouraku and Michihiro Koibuchi and Akira Funahashi and Hideharu Amano "Routing Algorithms on 2D Turn Model for Irregular Networks", the Sixth International Symposium on Parallel Architectures, Algorithms, and Networks(I-SPAN'02), pp.289-294, May. 2002

鯉渕 道紘, 上樂 明也, 天野 英晴
"イレギュラーネットワークにおける仮想チャネルを用いた固定ルーティング" 並列処理シンポジウム JSPP 2002 論文集, pp.43-50, May. 2002

Michihiro Koibuchi and Akiya Jouraku and Hideharu Amano "The Impact of Path Selection Algorithm of Adaptive Routing for Implementing Deterministic Routing", the 2002 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'02), pp.1431-1437, June. 2002

Michihiro Koibuchi and Akiya Jouraku and Hideharu Amano "Deterministic Routing Techniques by Dividing into Sub-Networks in Irregular Networks", the IASTED International Conference on Networks, Parallel and Distributed Processing, and Applications (NPDPA 2002), pp.143-148, Oct. 2002

B.1.3 研究会

鯉渕 道紘, 舟橋 啓, 上樂 明也, 若林 正樹, 天野 英晴
Irregular Network における Adaptive Routing の提案, 電子情報通信学会技術研究報告 CPSY2000-44, pp.33-40, Aug. 2000

鯉渕 道紘, 上樂 明也, 舟橋 啓, 天野 英晴
MMLRU selection function：適応型ルーティングにおける output selection function, 電子情報通信学会技術研究報告 CPSY2001-13, pp.97-104, Apr. 2001