# 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

鯉渕道紘[†] 上樂明也[†] 天野英晴[†]

PC クラスタ，Infinibandで用いられているイレギュラーネットワークでは，パケットの管理の容易さ，FIFO 性の保証の点から固定ルーティングを用いている場合が多いが，最近多数装備することが可能になった仮想チャネルを効果的に用いていない場合が多い．そこで，本稿では既存の固定ルーティングを用いて connectivity を保証し，仮想チャネルを活用することで最短経路をとるパケットの割合の増加，トラフィックの分散を実現する固定ルーティングの設計方法 MEDR（Methodology for design of Efficient Deterministic Routing）を提案する．そして，この方法をイレギュラーネットワークの代表的なルーティング法である up*/down* routing に適用した UDWM（up*/down* routing with multi-channels）を提案する．シミュレーションの結果，固定ルーティングを前提とする場合，UDWM は同数の仮想チャネルを用いた up*/down* routing に比べて，スループットが最大 273%向上した．

# A Deterministic Routing Using Virtual Channels in Irregular Networks

Michihiro Koibuchi,[†] Akiya Jouraku[†] and Hideharu Amano[†]

In PC clusters or high performance I/O networks including InfiniBand, network topologies often become irregular. In order to cope with the complicated network topology, and in-order packet transfer property, most of such networks use a deterministic routing technique. However, most deterministic routing methods cannot make the best use of recent increasing virtual channels. Here, we propose a methodology for the design of efficient deterministic routing called the “MEDR (Methodology for design of Efficient Deterministic Routing)” which increases minimal paths and distributes traffic by using virtual channels. As a simple example of the MEDR application, the UDWM (up*/down* routing with multi-channels) is shown based on up*/down* routing which is a typical routing for irregular networks. Results of simulations show that the UDWM achieves 273% improvement in throughput compared with deterministic up*/down* routing with the same number of virtual channels per physical link.

## 1. はじめに

スイッチベースのイレギュラーネットワークは PC（personal computer）/WS（workstation）による高性能分散コンピューティングシステム[1)~4)]や，ストレージ，サーバネットワークとしての利用がさかんであり，これらのネットワークの標準化を目指す InfiniBand[5)] でもサポートされている．

イレギュラーネットワーク向けの適応型ルーティング手法は数多く提案されており，高い性能が明らかになっている[6),7)]．しかし，実際には，パケット配送エラー時の検出の容易さ，FIFO 性の保証を重視して，固定ルーティングが用いられているのが現実である．InfiniBand では目的地のみをインデックスにする方法と，Destination Renaming を用いることにより入力チャネルと目的地をインデックスにして出力チャネルを得る分散ルーティング方式の双方を使用することが可能であるが，いずれも経路の決定については固定的な方法が主体である[8),☆]．

一方で，実装技術の向上により，仮想チャネルを数多く実装することが可能となっている．分散コンピューティング用ネットワークである RHiNET-2/SW[3)]では 16，RHiNET-3/SW[4)]では 64 にのぼる仮想チャネルが利用可能であり，InfiniBand でも仮想レーン（VL）

† 慶應義塾大学理工学部
Faculty of Science and Technology, Keio University

☆ InfiniBand はサブネット内の CA（Channel Adapter）間において複数経路が選択可能であり，各経路は LID（local identifier）により識別される．しかし，出発地の CA が目的地のポートに割り当てられた LID の 1 つを選択するため経路が一意に定まる[5)]．

の利用が可能である[5]．

しかし，イレギュラーネットワークにおいて，適応型ルーティングについては仮想チャネルを効果的に利用する方法が提案されている[6]にもかかわらず，固定ルーティングの研究の多くは仮想チャネルを想定すらしていないため，効果的に利用することが困難である[9]．このため，実際のイレギュラーネットワークにおいて，仮想チャネルは，Myrinet で用いられている構造化チャネル法[10]あるいは RHiNET で用いられている縮約構造化チャネル法[2]など，純粋にデッドロックを避ける目的でのみ用いられている．しかし，この方法では仮想チャネル数により，使用できるネットワークのサイズが限定される問題点がある．

そこで，本稿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に分割する固定ルーティングの設計方法である MEDR（Methodology for design of Efficient Deterministic Routing）を提案する．MEDR は既存の固定ルーティングを用いて connectivity を保証する．そして，MEDR はサブネットワークを切り換えることにより最短経路をとるパケットの割合の増加，トラフィックの分散を実現する．

次に，up*/down* routing を用いて MEDR に基づいて設計したシンプルな固定ルーティングである UDWM（up*/down* routing with multi-channels）を提案する．UDWM では up*/down* routing で禁止されている down chanel から up channel へのパケット転送を仮想チャネル数回未満許す，という制約下で目的地までの経路を探索し，固定することができる．最後にフリットレベルのシミュレータにより，既存の方法と比べた UDWM の性能向上について評価する．

## 2. MEDR

### 2.1 固定ルーティング

以下，本稿の議論はすべて固定ルーティングを基本とする．固定ルーティングは，出発地から目的地まで決定的な経路と利用仮想チャネルでパケットを送る方法である．この方法では，各スイッチは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持っていない．固定ルーティングは，実現の方法により，(1) それぞれのスイッチでテーブルを参照することで，入力物理リンク，仮想チャネルと目的地から出力物理リンクと利用仮想チャネルを得る分散的な手法，(2) 出発地で目的地までの経路と利用仮想チャネルを生成してパケット内に持たせるソースルーティングの 2 種類に分けることができる．しかし，経路と利用仮想チャネルの決定手法は同一のものが利用可能である．本稿で，ルーティングアルゴリズムとは，出発地から目的地までの経路と利用仮想チャネルの決定法を意味する．固定ルーティングは，研究がさかんに行われている適応型ルーティングに比べて，性能面では劣るが，FIFO 性を保証することができ，パケット配送エラー時の検出が容易であるなどの利点を持つ．このため，現状では，ほとんどすべてのイレギュラーネットワークで実際に用いられている現実的な方法である．

### 2.2 MEDR の提案

MEDR（Methodology for design of Efficient Deterministic Routing）は，まず，ネットワークを $r$ 個のサブネットワーク（サブネットワーク 0，サブネットワーク 1 ... サブネットワーク $(r-1)$）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バーチャルネットワークを指す☆．ただし，$r$ は仮想チャネル数以下の数であり，各サブネットワークに含まれる物理リンクあたりの仮想チャネル数は 1 本以上とする．そして，以下のように経路および利用する仮想チャネルを決定する．

a. 各サブネットワーク内におけるルーティングでは，デッドロックフリーを保証したうえで，可能な限り目的地までの最短経路をとるようにする．ただし，サブネットワーク 0 内では既存の固定ルーティングにより定まる経路でパケットを転送する．
b. しかし，番号が 1 以上のサブネットワークにおいて，そのサブネットワーク内のデッドロックフリーの条件を破らなければより短い経路をとることができない場合，もしくは目的地に到達できない場合，より小さい番号のサブネットワークに切り換え，その経路にパケットを転送する．

MEDR において，a. の条件より，サブネットワーク 0 内の connectivity は保証されている．このため，必ずしも番号が 1 以上のサブネットワークから成る経路のみでネットワークの connectivity を保証する必要はない．また，b. の条件より各サブネットワーク内のデッドロックフリーの条件に最大 $(r-1)$ 回反する経路を設定することが可能である．

MEDR 上で利用する固定ルーティングを決める手順は以下のようになる．

1. ネットワークを $r$ 個のサブネットワークに分割する．
2. サブネットワーク 0 内の経路を決定する固定ルー

☆ サブネットワーク間は独立であり，仮想チャネルを共有しない．

ティングを定める．

3. 番号が 1 以上の各サブネットワーク内のルーティングを定め，b. の条件に従い番号が 1 以上のサブネットワークを用いた経路を定める．

目的地までの経路の探索については，たとえば Autonet で行われているように，まず始めに対象ネットワークに対し spanning tree を構築し，その後ルートノードに集められたトポロジ情報（コネクション情報）を基に各ノードがルーティングアルゴリズムの制約を満たす各目的地までの最短経路を調べる，など，通常のイレギュラーネットワークでのルーティング手法に基づいて行うことが可能である[11]．

MEDR により設計された固定ルーティングは，用いた既存の固定ルーティングがデッドロックフリーであれば定理 1 よりデッドロックフリーである．

**定理 1** MEDR により設計された固定ルーティングはデッドロックフリーである．

*Proof*：MEDR において以下の 2 つが成立する．

(1) MEDR において使用されるサブネットワーク番号の切換えは降順のみで行われる．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．

(2) サブネットワーク 0 内においては既存の固定ルーティングを用いているため，デッドロックは発生しない．また，番号が 1 以上のサブネットワーク内ではデッドロックフリーな経路を設定し，パケットを転送する．したが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．

(1)，(2) より，MEDR により設計された固定ルーティングはデッドロックフリーである． ∎

MEDR により設計された固定ルーティングは以下のような特長を持つ．

**最短経路をとるパケットの増加：**

spanning tree の構築を基にする多くの固定ルーティングの場合[9),12)]，非最短経路をとる物理リンクへパケットを転送するノードが発生する．しかし，MEDR の場合，使用するサブネットワークを切り換えることにより，パケットの転送先を最短経路をとる物理リンクに設定することができる．したがってパケットの平均ホップ数を減らすことができる．

**トラフィックの分散：**

spanning tree の構築を基にする多くの固定ルーティングの場合[9),12)]，ツリー構造が本質的に持つ一次元的な up/down の概念をそのまま利用しているため，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，MEDR により設計された固定ルーティングはこれら既存の固定ルーティングで禁止されている物理リンクへの転送を設定することができるため，トラフィックの分散を図ることができる．

## 3. UDWM

本章ではイレギュラーネットワークにおける代表的なルーティング手法である up*/down* routing に対して MEDR を適用した固定ルーティング UDWM の提案を行う．

### 3.1 up*/down* routing

up*/down* routing はイレギュラーネットワークにおける代表的なルーティングアルゴリズムであり，Autonet [11)]や Myrinet [1)]などのネットワークで利用されている．

up*/down* routing は，トポロジを構成するすべてのチャネルに *up*または *down*の方向が割り当てられた有向グラフを必要とするため，まず最初に有向グラフの基礎となる spanning tree を構築する必要がある．spanning tree の構築は一般的に BFS（Breadth-First Search）または DFS（Depth-First Search）のいずれかの方針に基づいて行われる．BFS に基づいた手法としては，Autonet で利用されている MDST（Minimum Depth Spanning Tree）[11)]および POST（Propagation Order Spanning Tree）[13)]がある．これらはともに spanning tree における全ノードのルートノードからの距離が最小となることを目標としている．ここでは POST の概念による spanning tree 構築アルゴリズムを簡単に示す．

1. ネットワークの中から任意に spanning tree のルートノードを選択する．
2. ルートノードは，すべての隣接ノードに join 要求メッセージを送信し，要求を受諾したノードをルートノードの子ノードとして spanning tree に付け加える．
3. あるノードの子ノードとなったノードは，同様にしてすべての隣接ノードに join 要求メッセージを送信し，要求を受諾したノード（すでに spanning tree に含まれているノードは要求を拒否する）を自身の子ノードとして spanning tree に付け加える．
4. 全ノードが spanning tree に含まれるまで 3 の作業を繰り返す．

spanning tree の構築が完了した後，ネットワーク上のすべてのチャネルに対して以下に示す規則に基づいて up または down の方向を割り当て，有向グラフ

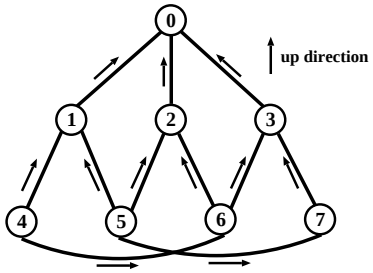


**図 1** BFS spanning tree に基づいた有向グラフ
Fig. 1 A directed-graph based on the BFS spanning tree.

を構築する．

まず，up 方向を次の 2 つの条件のいずれかを満たすチャネルに対して割り当てる．

1. 移動先のノードが移動元のノードよりもルートノードに近い．
2. 移動先のノードと移動元のノードのルートノードからの深さが同一であり，移動先のノード ID の方が移動元のノード ID よりも小さい．

次に，残りのすべてのチャネルに対して down 方向を割り当てる．

以上の作業により，例として**図 1** のような有向グラフが構築される．

デッドロックフリーと任意のノード間の経路を保証するために，すべての経路は必ず 0 回以上 up 方向に必要なだけ移動した後に 0 回以上 down 方向に移動して目的ノードまで到達する，という条件を守る必要がある．この条件により，down 方向から up 方向への移動を行うことはできなくなるので，チャネル間の cyclic dependency が除去されデッドロックフリーが保証される．条件を守る限りは経路を自由に選択できるが，つねに最短経路を選択することはできないので，up*/down* routing は非最短型の適応型ルーティングとなる．たとえば，図 1 においてノード **5** からノード **0** へパケットを転送する場合には，ノード **1** またはノード **2** を経由してノード **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．一方，ノード **1** からノード **6** 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ノード **4** を経由する最短経路は選択することができず，ノード **0** とノード **2** またはノード **3** を経由する非最短経路しか選択することができない．

up*/down* routing は本来適応型ルーティングであるが，各ノード間の経路を 1 つに選択すること[9]により up*/down* routing を固定ルーティングとして実装することができる．

### 3.2 MEDR による固定ルーティングの設計

UDWM（up*/down* routing with multi-channels）は，up*/down* routing を基本として MEDR を適用することにより設計した固定ルーティング手法である．まず，UDWM ではネットワークを仮想チャネル数個のサブネットワークに分ける．そして，サブネットワーク 0 内は固定ルーティングとして実装した up*/down* routing を用いる．

次に，番号が 1 以上のサブネットワーク内のルーティングについて述べる．

UDWM では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，シンプルに各仮想チャネルに割り当てられた up もしくは down の方向を基にしてデッドロック除去を行う．方向を基にしたデッドロックの除去には，(1) down 方向から up 方向へのターンを禁止する，(2) up 方向から down 方向へのターンを禁止する方法の 2 つが考えられるが，UDWM では前者を採用し，デッドロックフリーを実現する．

これは spanning tree の構成リンク以外のリンクが少ない場合，down 方向から up 方向へのターン数が up 方向から down 方向へのターン数に比べ大幅に少なくなることに配慮している．後者を採用した場合，このために，番号が 1 以上のサブネットワークの仮想チャネルを効果的に使えず，性能が安定しない可能性がある．

上記の制限を与えた結果，UDWM では，同一サブネットワーク内のパケット転送において down channel から up channel へのターンは設定できない．しかし，UDWM では，より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 channel から up channel へのターンを行うパケット転送を実現している．

まとめると，UDWM は以下の制限を基に経路を一意に決定する固定ルーティングである．

- パケットを down channel から up channel に転送する場合，使用する仮想チャネルのサブネットワーク番号を減少させる．

UDWM は，up*/down* routing と最短経路探索条件は異なるが，同様の手順により最短経路を検索することができる．つまり，UDWM は最短経路の探索時に up*/down* routing の禁止ターンが必要な場合，使用するサブネットワーク番号を減らすことでその最短経路の探索を続けることができる．UDWM のルーティングテーブル作成コストは，最短経路を対象にしているためサブネットワーク数の増加にともない大幅

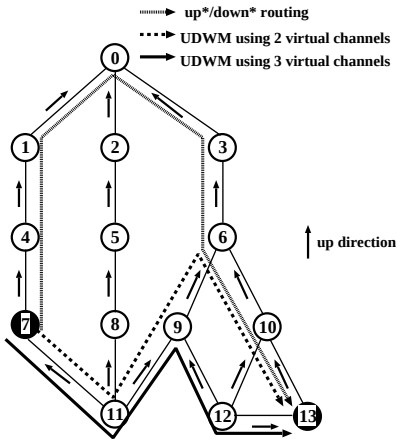


**図 2** UDWM と up*/down* routing のルーティング例
Fig. 2 A routing example of the UDWM and up*/down* routing.

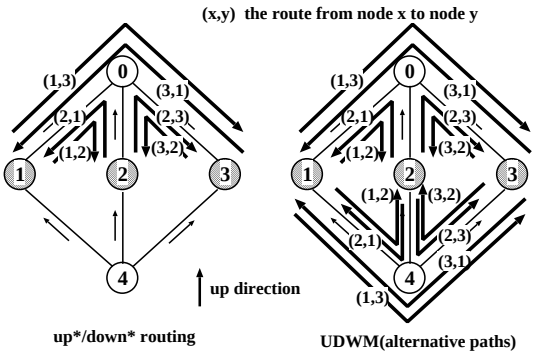


**図 3** 5 ノードのネットワークにおけるトラフィックの分散の比較
Fig. 3 An example of path distribution in the UDWM and up*/down* routing.

に増すことはない．

UDWM においてトポロジによっては上記の制限だけでは一部のノード間に複数の最短経路が存在する可能性があるが，これは実装する際に自由に 1 つの経路に選択（たとえばランダムに選択，もしくは，最も小さい番号のポートを使用する経路を選択）すればよい．

たとえば**図 2** においてノード **7** からノード **13** へパケットを転送する場合，up*/down* routing では，仮想チャネル数に関係なく，7 hop(**7** → **4** → **1** → **0** → **3** → **6** → **10** → **13**) 必要になる．しかし UDWM では仮想チャネルが 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，**7** → ($sn.2$) → **11** → ($sn.1$) → **9** → ($sn.1$) → **12** → ($sn.0$) → **13** とサブネットワーク番号を 2 回減少させることにより 4 hop で到達することができる．また，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 hop(**7** → ($sn.1$) → **11** → ($sn.0$) → **9** → ($sn.0$) → **6** → ($sn.0$) → **10** → ($sn.0$) → **13**) となる．

up*/down* routing は connectivity と acyclicity を満たすために，本質的にこの 2 つの性質を持つツリー構造をそのまま利用したため，トラフィックに偏りが生じやすい．一方，UDWM は仮想チャネルを効果的に使うことにより，この問題を大幅に緩和することができる．たとえば 5 ノードで構成される単純なネットワークである**図 3** において，up*/down* routing の場合，仮想チャネル数に関係なく，1 ホップで目的地に到達できる経路，ルートノード，リーフノードを出発地とする経路を除く経路はすべてルートノードを通過するためトラフィックに偏りが生じる可能性が高い．一方，UDWM の場合，設定可能な経路が図 3 のように広く分散している．したがってルーティングテーブルに登録する固定経路をランダムに選択する，などにより，各物理リンクにかかるトラフィック負担を分散させることができる．

## 4. 評　　価

UDWM，up*/down* routing の各ルーティングアルゴリズムを C++で記述したフリットレベルシミュレータを用いて評価を行った．各ノードは 8 ポートを持つ 1 つのスイッチと，スイッチに直結している 4 つのプロセッシングエレメントにより構成される．スイッチの残りの 4 ポートは他のノード（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同一ノード間に物理リンクを 2 本以上接続しない，という制約を課したうえでランダムに生成した．

spannning tree の構築アルゴリズムとしては，親子関係を結ぶノード対を決定する付加リンクの選択アルゴリズムに，(1) MDST を基にした BFS（Breadth First Search）と，(2) Sancho らが提案したヒューリスティックルールによる DFS（Depth First Search）を用いた[9]．(2) のヒューリスティックルールはすでに spanning tree に組み込まれているノードとの接続数の最も多いノードへのリンクを選択するものである．ルートノードは BFS spanning tree の場合 Autonet と同様に ID 0 のノードとし，DFS spanning tree 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[9]により選択した．また，UDWM，up*/down* routing ともに目的地までの経路が複数存在する場合，ルーティングテーブル作成時にランダムに片方の経路を削除することで経路を単一化した．この際，up*/down* routing はネットワーク

**表 1** シミュレーションパラメータ
Table 1 Simulation parameters.

| | |
|---|---|
| Simulation time | 1,000,000 clocks (ignore the first 50,000 clocks) |
| Topology | irregular network |
| Network size | 16 or 64 nodes |
| The number of virtual channels | 1,2,3 (up*/down*), 2,3 (UDWM) |
| Packet length | 128 flits |
| Spanning tree policy | BFS or heuristic DFS |
| Switching tech. | virtual cut-through |
| Traffic pattern | uniform |

へパケットを注入する際にランダムに設定（固定）した番号の仮想チャネルを使用するが，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．

また，UDWM，up*/down* routing ともに，ネットワークへのパケットの注入時に，同一目的地に対して，各々同一のサブネットワーク，仮想チャネルを用いることにより FIFO 性を保証するよう実装した．なお，UDWM においてサブネットワーク数は仮想チャネル数と同数である．したがって，以後，up*/down* routing との比較を容易に行うために $X$ 個のサブネットワークを使った UDWM を仮想チャネル $X$ 本を使った UDWM として説明する．

トラフィックは uniform traffic を用いた．uniform traffic では各パケットの目的地ノードはランダムで決定されており，等確率に分散されている．

シミュレーションに用いた条件を**表 1** に示す．

シミュレーション開始後の 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，評価には加えないこととした．

### 4.1 16 ノード イレギュラーネットワーク

#### 4.1.1 BFS spanning tree を基にした場合

16 ノードのイレギュラーネットワークにおいて同一の BFS spanning tree を基にした up*/down* routing と UDWM のシミュレーション結果を**図 4** に示す．横軸はトポロジ番号，縦軸はスループットを表している．スループットは，全プロセッシングエレメントが毎クロックに 1 flit 受信する場合を 1.00 としており，accepted traffic の最大値を示している．

図 4 において，ud,vch=X は仮想チャネル X 本を用いた up*/down* routing を示し，UDWM,vch=Y は仮想チャネル Y 本を用いた UDWM を示す．

図 4 の 10 個のトポロジにおける各ルーティングアルゴリズムの平均スループット，パケットの平均ホップ数，全ノード対の経路の中でトポロジ的に最短経路をとることができた経路の割合，および，channel

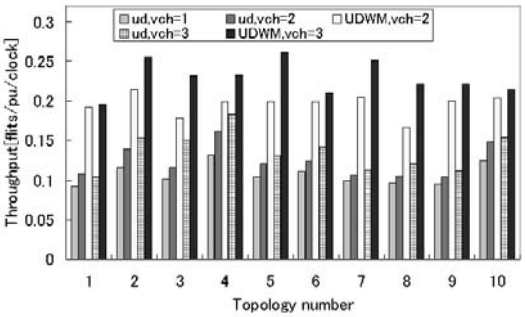


**図 4** BFS，16 ノードにおけるスループット
Fig. 4 Throughput on 16 nodes with the BFS.

**表 2** BFS，16 ノードにおける諸性能
Table 2 Routing metrics on 16 nodes with the BFS.

| | 平均スループット | 平均ホップ数 | 最短経路の割合 | SD of ch. cross. path |
|---|---|---|---|---|
| ud,vch=1 | 0.108 | 2.02 | 95.58 | 5.63 |
| ud,vch=2 | 0.124 | 2.02 | 95.58 | 5.63 |
| UDWM,vch=2 | 0.195 | 1.89 | 100.00 | 3.01 |
| ud,vch=3 | 0.136 | 2.03 | 95.58 | 5.62 |
| UDWM,vch=3 | 0.230 | 1.89 | 100.00 | 2.95 |

crossing path の分散を**表 2** に示す．channel crossing path は 1 つの物理リンクの単方向を通過するノード対の経路数を表す．したがって channel crossing path の最大値は仮想チャネルのないネットワークにおける crossing path である[9]．channel crossing path の分散は物理リンク間の経路の分散状況を示す．channel crossing path の分散が小さければ各物理リンクにかかる負担がより均等であることを示す．

図 4，表 2 より，UDWM は up*/down* routing と比べると仮想チャネルが 2 本の場合，平均スループットが約 57%，仮想チャネルが 3 本の場合，約 69%性能が向上した．

また，表 2 より，16 ノードにおいて UDWM はすべてのパケットが最短経路をとることができたことが分かる．また，16 ノードでは up*/down* routing，UDWM とも仮想チャネル数が 3 本の場合が 2 本の場合に比べ高性能である．これは両ルーティングアルゴリズムともパケットのホップ数が仮想チャネル数によらず一定であることから，チャネル数の増加による仮想チャネルフローコントロールの性能向上によるものと考えられる．

次に UDWM が同数の仮想チャネルを用いた up*/down* routing に比べ，大幅な性能向上を果たした要因として 2 つあげる．

- パケットの経路の分散

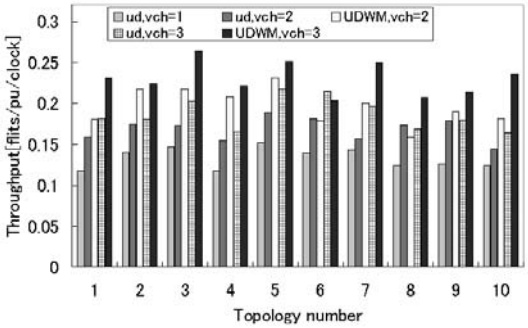


**図 5** DFS，16 ノードにおけるスループット
Fig. 5 Throughput on 16 nodes with the DFS.

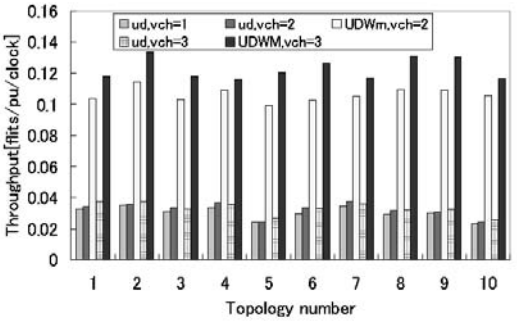


**図 6** BFS，64 ノードにおけるスループット
Fig. 6 Throughput on 64 nodes with the BFS.

**表 3** DFS，16 ノードにおける諸性能
Table 3 Routing metrics on 16 nodes with the BFS.

| | 平均スループット | 平均ホップ数 | 最短経路の割合 | SD of ch. cross. path |
|---|---|---|---|---|
| ud,vch=1 | 0.133 | 1.95 | 96.00 | 3.43 |
| ud,vch=2 | 0.168 | 1.95 | 96.00 | 3.44 |
| UDWM,vch=2 | 0.196 | 1.89 | 100.00 | 2.91 |
| ud,vch=3 | 0.187 | 1.95 | 96.00 | 3.44 |
| UDWM,vch=3 | 0.230 | 1.89 | 100.00 | 2.86 |

**表 4** BFS，64 ノードにおける諸性能
Table 4 Routing metrics on 64 nodes with the BFS.

| | 平均スループット | 平均ホップ数 | 最短経路の割合 | SD of ch. cross. path |
|---|---|---|---|---|
| ud,vch=1 | 0.030 | 3.84 | 78.12 | 59.94 |
| ud,vch=2 | 0.032 | 3.84 | 78.12 | 59.95 |
| UDWM,vch=2 | 0.106 | 3.14 | 99.97 | 14.28 |
| ud,vch=3 | 0.033 | 3.84 | 78.12 | 59.94 |
| UDWM,vch=3 | 0.123 | 3.14 | 100.00 | 13.68 |

- パケットの平均ホップ数の削減

表 2 より，UDWM は up*/down* routing に比べ channel crossing path の分散を約半分に抑えていることから，パケットの経路が up*/down* routing に比べ広く分散されていることが分かる．また，表 2 より UDWM の場合全パケットが最短経路をとっている．しかし，up*/down* routing の場合一部のパケットが非最短経路をとっており，UDWM が up*/down* routing に比べ効率的にネットワーク資源を利用したことが分かる．

#### 4.1.2 DFS spanning tree を基にした場合

図 4 で用いたものと同一のイレギュラーネットワークにおいて DFS spanning tree を基にした up*/down* routing と UDWM の評価を**図 5** に示す．横軸はトポロジ番号，縦軸はスループットを表している．また，10 個のトポロジにおける各ルーティングアルゴリズムの平均スループット，パケットの平均ホップ数，最短経路をとることができた経路の割合，および，channel crossing path の分散を**表 3** に示す．

図 4，表 2，図 5，表 3 より，up*/down* routing は DFS spanning tree を構築する場合が BFS spanning tree を構築する場合に比べ，大幅な性能向上を実現したことが分かる．これに対し，UDWM の性能は spanning tree の構築法によらず安定して高性能である．これは，DFS spanning tree は BFS spanning tree に比べ，up*/down* routing の禁止ターンを削減し，パケットの平均ホップ数の削減を実現するためである．したがって 16 ノードにおいてすでに全経路で最短経路をとっている UDWM の性能には影響しなかったといえる．それにもかかわらず，UDWM は同数の仮想チャネルを用いた DFS up*/down* routing に比べ約 17〜23%平均スループットが向上した．

また，表 3 より，UDWM は up*/down* routing に比べ channel crossing path の分散を削減しており，UDWM は DFS spanning tree を基にした場合においても up*/down* routing に比べ経路の分散を実現していることが分かる．

### 4.2 64 ノード イレギュラーネットワーク

#### 4.2.1 BFS spanning tree を基にした場合

64 ノードのイレギュラーネットワークにおいて同一の BFS spanning tree を基にした up*/down* routing と UDWM の評価を**図 6** に示す．横軸はトポロジ番号，縦軸はスループットを表している．また，10 個のトポロジにおける各ルーティングアルゴリズムの平均スループット，パケットの平均ホップ数，最短経路をとることができた経路の割合，および，channel crossing path の分散を**表 4** に示す．

図 6，表 4 より up*/down* routing では最短経路

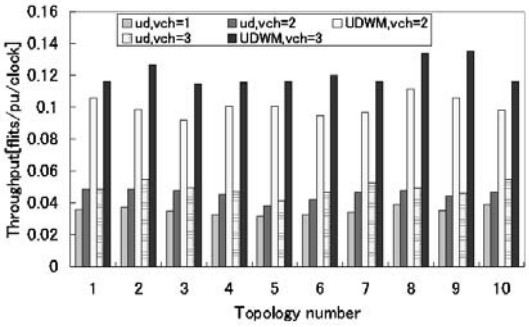


**図 7** DFS，64 ノードにおけるスループット
Fig. 7 Throughput on 64 nodes with the DFS.

**表 5** DFS，64 ノードにおける諸性能
Table 5 Routing metrics on 64 nodes with the DFS.

| | 平均スループット | 平均ホップ数 | 最短経路の割合 | SD of ch. cross. path |
|---|---|---|---|---|
| ud,vch=1 | 0.035 | 3.54 | 90.13 | 26.36 |
| ud,vch=2 | 0.046 | 3.53 | 90.13 | 26.40 |
| UDWM,vch=2 | 0.101 | 3.13 | 100.00 | 13.66 |
| ud,vch=3 | 0.049 | 3.53 | 90.13 | 26.38 |
| UDWM,vch=3 | 0.121 | 3.14 | 100.00 | 13.58 |

をとることができるパケットの割合が著しく低下し，channel crossing path の分散が大きいことからトラフィックに偏りが発生しやすい状況になっており，大変効率の悪い事態を引き起こしていることが分かる．一方，UDWM ではほぼすべてのノード間において最短経路をとることができ，かつ，channel crossing path の分散を抑えている．その結果，UDWM は同数の仮想チャネルを用いた up*/down* routingi に比べ平均スループットが約 231〜272%向上した．

#### 4.2.2 DFS spanning tree を基にした場合

図 6 で用いたものと同一の 64 ノードのイレギュラーネットワークにおいて DFS spanning tree を基にした up*/down* routing と UDWM の評価を**図 7** に示す．横軸はトポロジ番号，縦軸はスループットを表している．また，10 個のトポロジにおける各ルーティングアルゴリズムの平均スループット，パケットの平均ホップ数，最短経路をとることができた経路の割合，および，channel crossing path の分散を**表 5** に示す．

図 7，表 5 より，UDWM は同数の仮想チャネルを用いた up*/down* routing に比べ約 120〜147%平均スループットが向上し，安定して高い性能を示していることが分かる．また，表 4，表 5 より up*/down* routing において BFS spanning tree を基にした場合が BFS spanning tree を基にした場合に比べ，channel crossing path の分散を約半減させ，パケットの経路の分散，最短経路の割合の増加を実現したことが分かる．さらに表 4，表 5 より UDWM では高性能な DFS spanning tree を基にした up*/down* routing に比べスループット，パケットの平均ホップ数，最短経路の割合，channel crossing path の分散，すべてにおいて大幅な向上を実現したことが分かる．

## 5. 関連研究

MEDR は，基本となるアイディアでは他の研究と類似した点が多い．まず，デッドロックフリーの経路を 1 つ用意しておき，最終的にそれを利用することにより，全体の connectivity とデッドロックフリーを保証する点では，Duato の手法[14]と同じである．また，デッドロックフリーの条件に違反するたびに仮想チャネルを切り換える点では Dally らによる Dimension reversal Routing [15]と似ている．さらに，これを up*/down* routing に対して適用する手法については，Z-routing [16]と類似している．

しかし，上記の手法と MEDR は，MEDR が固定ルーティングである点で大きく異なっている．MEDR は，動的な経路選択ができない代わりに，実際に運用されている多くのイレギュラーネットワークに適用可能であり，単純な up*/down*ルーティングに比べて，確実に性能を改善することができる点で非常に現実的な方法である．

すなわち，MEDR は，適用型ルーティングがダイナミックに用いている経路選択の技法を固定ルーティングに適用し，イレギュラーネットワーク上で仮想チャネルが有効に生かせるようにまとめあげた方法と考えることができる．MEDR では，混雑に対する動的な回避は実現できない代わりに，ネットワークの形状と利用法がある程度予想可能な場合には，物理リンクを共有する経路を静的に解析して，分散することができる．今回は uniform traffic を前提に，ランダムに経路を分散したが，物理リンク負荷の静的分散手法[9]を適用することが可能であり，従来のイレギュラーネットワークに対する固定ルーティングの枠組みを広げており，この点で類似の研究はあまり行われていない．

## 6. まとめ

イレギュラー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に分割する固定ルーティングの設計方法である MEDR（Methodology for design of Efficient Deterministic Routing）を提案した．MEDR は既存の固定ルーティングを用いて connec-

tivityを保証し，サブネットワークを切り換えることにより最短経路をとるパケットの割合の増加，トラフィックの分散を実現する．次に，up*/down* routingを用いてMEDRにより設計された固定ルーティングであるUDWMの提案を行った．シミュレーションの結果，UDWMは同数の仮想チャネルを用いたup*/down* routingに比べ，最大273%性能向上した．

今後は，適応型ルーティングを固定ルーティングに実装するため，各ノード間の経路を1つに選択する負荷分散アルゴリズム[9]を仮想チャネルがあるネットワークに対して検討する．そしてUDWMの物理リンク間，および，物理リンク内の仮想チャネル間のトラフィック分散能力を向上させ，さらなる性能向上を実現する予定である．また，MEDRにより様々な固定ルーティングを設計し，様々な特徴を持つイレギュラートポロジに対し，その効果を確認していく予定である．さらに，稼働を開始したRHiNET上に実装し，実システム上での性能評価を行う予定である．

## 参 考 文 献


1) Boden, N.J., et al.: Myrinet: A Gigabit-per-Second Local Area Network, *IEEE Micro*, Vol.15, No.1, pp.29–35 (1995).
2) 西　宏章，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.2, pp.245–254 (2000).
3) Nishimura, S., Kudoh, T., Nishi, H., Yamamoto, J., Harasawa, K., Matsudaira, N. and Amano, H.: 64-Gb.s Highly Reliable Network Switch Using Parallel Optical Interconnection, *IEEE Journal of Lightwave Technology*, Vol.8, No.12, pp.1620–1627 (2000).
4) Nishimura, S., Kudoh, T., Nishi, H., Yamamoto, J., Harasawa, K., Matsudaira, N., Akutsu, S., Tasho, K. and Amano, H.: RHiNET-3/SW: an 80-Gbit/s high-speed network switch for distributed parallel computing, *Hot Interconnect 9*, pp.119–123 (2001).
5) I.T. Association: InfiniBand architecture, Specification Volumen 1, Release 1.0.a, available from the InfiniBand Trade Association, http://www.infinibandta.com (2001).
6) Silla, F. and Duato, J.: High-Performance Routing in Networks of Workstations with Irregular Toporogy, *IEEE Trans. parallel and distributed systems*, Vol.11, No.7, pp.699–719 (2000).
7) Koibuchi, M., Funahashi, A., Jouraku, A. and Amano, H.: L-turn Routing: An Adaptive Routing in Irregular Networks, *Proc. International Conference on Parallel Processing* (*ICPP*), pp.374–383 (2001).
8) Lopez, P., Flich, J. and Duato, J.: Deadlock-free Routing in $InfiniBand^{TM}$ through Destination Renaming, *Proc. International Conference on Parallel Processing* (*ICPP*), pp.427–434 (2001).
9) Sancho, J.C. and Robles, A.: Improving the Up*/Down* Routing Scheme for Networks of Workstations, *Proc. European Conference on Parallel Computing* (*EURO-PAR*), pp.882–889 (2000).
10) 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).
11) Schroeder, M.D., et al.: Autonet: A high-speed, selfconfiguring local area network using point-to-point links, Technical Report SRC research report 59, DEC (1990).
12) Wu, J. and Sheng, L.: Deadlock-Free Routing in Irregular Networks Using Prefix Routing, DIMACS Technical Report 99-19 (1999).
13) Rodeheffer, T. and Schroeder, M.: Automatic reconfiguration in Autonet, Technical Report SRC research report 77, DEC (1991).
14) Duato, J.: A Necessary and Sufficient Condition for Deadlock-Free Adaptive Routing in Wormhole Networks, *IEEE Trans. Parallel and Distributed Systems*, Vol.6, No.10, pp.1055–1067 (1995).
15) Dally, W.J. and Aoki, H.: Deadlock-Free Adaptive Routing in Multicomputer Networks Using Virtual Channels, *IEEE Trans. Parallel and Distributed Systems*, Vol.4, No.4, pp.466–475 (1993).
16) 井川郁哉，舟橋　啓：PCクラスタにおける適応型ルーティング，信学技報，CPSY2001-1, pp.1–8 (2001).


(平成14年1月29日受付)
(平成14年5月17日採録)

**鯉渕　道紘**

平成12年慶應義塾大学理工学部情報工学科卒業．現在，同大学大学院理工学研究科開放環境科学専攻博士課程に在学中．相互結合網に関する研究に従事．

**上樂 明也**

平成 10 年慶應義塾大学理工学部電気工学科卒業．現在，同大学大学院理工学研究科開放環境科学専攻博士課程に在学中．相互結合網に関する研究に従事．

**天野 英晴**（正会員）

昭和 56 年慶應義塾大学工学部電気工学科卒業．昭和 61 年同大学大学院工学研究科電気工学専攻博士課程修了．現在，同大学理工学部情報工学科教授．工学博士．計算機アーキテクチャの研究に従事．