

2次元 Turn モデルに基づくイレギュラーネットワーク向けルーティングアルゴリズムの設計と評価

上 樂 明 也[†] 鯉 淵 道 紘[†] 天 野 英 晴[†]

大規模 PC クラスタにおける PC 間の相互結合には、通常、トポロジに制限のないイレギュラーネットワークが用いられる。しかし、イレギュラーネットワークにおける代表的なデッドロックフリールーティングアルゴリズムである up*/down*ルーティングは、1次元有向グラフを基に Turn モデルを適用するため、パケットの転送禁止ターンが偏り、効率的にネットワークバンド幅を利用することが難しい。本稿では、up*/down*ルーティングにおける問題点を改善するため、既存の有向グラフを2次元に拡張する。そして、2次元有向グラフに対して Turn モデルを適用することにより、パケット転送禁止ターンを分散する効率的なデッドロックフリールーティングアルゴリズムを設計する手法を示す。シミュレーション結果より、提案したルーティングアルゴリズムは up*/down*ルーティングに比べ、最大 96%のスループット向上を実現することが分かった。

The Design and Evaluation of Routing Algorithms Based on 2D Turn Model for Irregular Networks

AKIYA JOURAKU,[†] MICHIIHIRO KOIBUCHI[†] and HIDEHARU AMANO[†]

Irregular networks are usually used to connect personal computers in massively PC clusters. Up*/down* routing algorithm, which is a typical deadlock-free routing algorithm in irregular networks, tends to make traffic unbalancing because it is based on a one-dimensional directed-graph. In order to solve the traffic unbalancing caused by up*/down* routing algorithm for irregular networks, two-dimensional directed graph is introduced, and novel routing algorithms based on two-dimensional turn model are proposed. The proposed routing algorithms improve traffic balancing by distributing prohibited turns of packet transfer over the network. Simulation results demonstrate that proposed algorithms improve throughput up to 96%.

1. はじめに

PC(パーソナルコンピュータ)およびWS(ワークステーション)などのコモディティコンポーネントを Myrinet¹⁾に代表される高速なネットワーク——SAN(システムエリアネットワーク)——で接続することにより構築される高性能クラスタシステムは、コストパフォーマンスに優れた大規模並列/分散コンピューティング環境として注目されている^{1)~3)}。

SANは、point-to-point リンクにより相互結合される高速なスイッチ群から構成され、分散配置されているPC/WSをユーザが容易に接続できることを重要視している。そのため、結合方式としてはトポロジに制限がないイレギュラーネットワークが用いられることが多い。

しかし、イレギュラーネットワークでは、並列計算機で用いられるトーラスなどのレギュラーネットワークに比べて経路保証とデッドロックフリーの実現が難しいため、ルーティングアルゴリズムの設計がより複雑になる。このため、既存のルーティングアルゴリズムのほとんどは、この問題を解決するためにトポロジ上へスパニングツリーのマッピングを行い、ツリー構造を持つ結合性および非循環性の特性を利用している。これらの中で、最も代表的なルーティングアルゴリズムである up*/down*ルーティング⁴⁾は、ネットワークを1次元の方向(up/down)を持つ有向グラフに見立て、単純な1次元のTurnモデル⁵⁾を適用することにより、デッドロックフリーを実現している。しかし、1次元のTurnモデルでは禁止ターンの偏りが大きくなるため、トラフィックの分散の実現が困難になり、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くなってしまう。

我々は、既存の up*/down*ルーティングにおける

[†] 慶應義塾大学理工学部

Faculty of Science and Technology, Keio University

上記の問題点を改善し、デッドロックフリーと効率的なバンド幅の利用を可能とするために、2次元の方向 (up/down と left/right) を導入したルーティングアルゴリズム⁶⁾を提案した。しかし、この提案では循環構造除去に必要な禁止ターン集合の選択法が確立されていなかったため、ルーティングアルゴリズムの一例を示すにとどまった。

そこで、本稿では、Turn モデルの手法⁵⁾に基づき、循環構造除去のための経路の分散を考慮した禁止ターン集合の決定手法を提案する。これにより、形成可能なデッドロックフリールーティングアルゴリズムが全部で4つ存在することを明らかにした。さらに、任意のトポロジを選択可能なフリットレベルシミュレータを実装することにより、提案したルーティングアルゴリズムを様々なパターンのイレギュラートポロジについて評価した。

以降、2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アルゴリズムである up*/down* ルーティングについて述べる。そして、3章で2次元 Turn モデルベースルーティングアルゴリズムの提案を行い、5章にてシミュレーション結果を提示、考察を行う。最後に、6章で結論を述べる。

2. up*/down*ルーティング

up*/down*ルーティングは、イレギュラーネットワーク向けの適応型ルーティングであり、Autonet⁴⁾や Myrinet¹⁾などで実用化されている。

up*/down*ルーティングは、トポロジ上のすべてのチャンネルに up または down の方向を割り当てた有向グラフを必要とする。そのため、最初に、スイッチをノードとしたスパニングツリーを構築する必要がある。代表的なスパニングツリーの構築方法は、Autonet で用いられている BFS (Breadth-First Search) に基づく方法⁴⁾である。

有向グラフの構築は、BFS スパニングツリーを構築した後、ネットワークの全チャンネルに対して、次のように方向を割り当てることによって行われる。

1. up 方向を次の 2 つの条件のいずれかを満たすチャンネルに対して割り当てる。
 - a. 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
 - b. 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 が移動元のノード ID よりも小さい
2. 残りのすべてのチャンネルに対して down 方向を割り当てる。

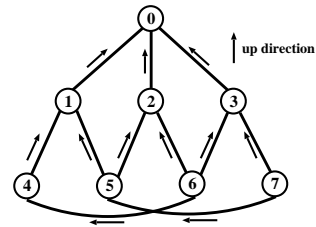


図1 BFS スパニングツリーに基づいた有向グラフ
Fig.1 BFS spanning tree based directed-graph.

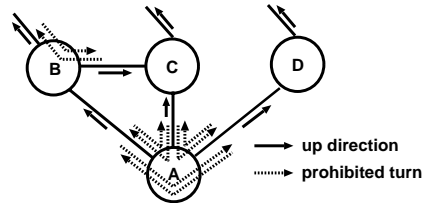


図2 up*/down*ルーティングにおける禁止ターンのペア
Fig.2 Pairs of prohibited turns in up*/down* routing.

これにより、図1のような有向グラフが構築される。up*/down*ルーティングは、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、次のような1次元の Turn モデルを適用している。

まず、構築した有向グラフには up, down の2つの方向のみが存在するので、パケット転送時に発生するターンは、up→down および down→up の2パターンとなり、これらのターンの連鎖により1パターンの循環構造が形成される。したがって、すべてのパケットは必ず0回以上 up 方向に必要なだけ移動した後0回以上 down 方向に移動して目的ノードまで到達する、という単純な制限により、down 方向から up 方向へのターンを行うことができなくなるため、チャンネル間の循環依存が除去されデッドロックフリーが保証される。

up*/down*ルーティングでは、上記の単純な制限によりデッドロックフリーを保証できるが、up 方向から down 方向へのターンを禁止すると任意のノード間の経路が保証されないため、禁止ターンの選択に関しては自由度がない。このため、図2のように、禁止ターンが形成される2つのリンク間において、互いに反対の方向に向かう禁止ターンのペアが必ず形成されてしまうという欠点を持つ。図2において、ノード B に1つ、ノード A には3つの禁止ターンのペアがそれぞれ形成されている。このような禁止ターンの偏りにより、ネットワーク内のトラフィックに偏りが生じ、ネットワークバンド幅の有効利用が困難になるという問題が発生する。

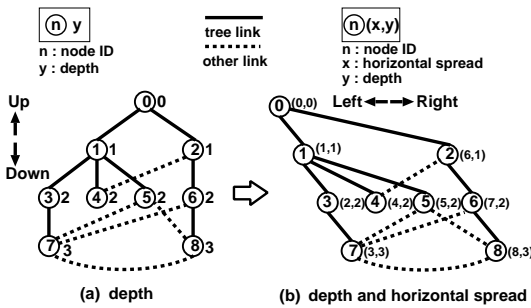


図3 depth と horizontal spread の割当て

Fig. 3 Assignment depth and horizontal spread.

3. 2次元 Turn モデルベースルーティングアルゴリズム

up*/down*ルーティングでは、パケット転送時に発生するターンが2パターンのみであるため、禁止ターンに偏りが生じやすい。そこで、我々はこの問題点を改善するために、up*/down*ルーティングの1次元有向グラフを拡張して2次元化することで、禁止ターンの偏りを減らす方式を提案した⁶⁾。しかし、この方法は、2次元有向グラフ上でデッドロックフリールーティングが可能な禁止ターンの選択方法のうち1つを示したにすぎず、ほかにも禁止ターンの選択方法とルーティングアルゴリズムの可能性が残されていた。

そこで、本稿ではH/Vグラフと呼ばれる2次元有向グラフを導入し、2次元のTurnモデルを適用してデッドロックフリーを実現するための禁止ターンの選択をシステムティックに行い、可能なルーティングアルゴリズムをすべて導出する手法を示す。

3.1 H/Vグラフの構築

3.1.1 各ノードへのdepthの割当て

最初に、up*/down*ルーティングと同様に、BFSスパニングツリーを構築し、各ノードに対してdepthを割り当てる。depthは、各ノードのルートノードからの垂直方向における最短距離を示し、各チャネルのvertical direction(upおよびdown)の決定に用いられる。

例として、9ノードのイレギュラーネットワークに対するdepthの割当てを図3(a)に示す。図3(a)において、各リンクは互いに反対の方向を持つ2つの単方向チャネルから構成され、実線と破線はそれぞれスパニングツリーを構成するリンク(tree link)とそれ以外のリンク(other link)を示している。

3.1.2 各ノードへのhorizontal spreadおよび2次元座標の割当て

次に、2次元有向グラフを構築するために、depth

に加えて各ノードに対しhorizontal spreadを割り当て、horizontal direction(leftおよびright)の概念を導入する。

horizontal spreadは、構築したスパニングツリー上でルートノードを起点とした前順走査を行ったときの訪問順序であり、走査における訪問順にしたがって0から始まる昇順の値が各ノードに割り当てられる。

horizontal spreadを前順走査により割り当てている理由は、スパニングツリーにおける経路保証と各ノードに対する一意のdepthおよびhorizontal spreadの割当てを実現するために、次の2つの条件を満たす必要があるためである。

- (1) 子ノードのhorizontal spreadがつねに親ノードよりも大きい。
- (2) 同じdepthを持つ各ノードのhorizontal spreadは互いに異なる。

前者の条件により、depthと同様にして、horizontal spreadが小さくなる方向に進むことにより任意のノードからルートノードに到達可能となり、また、大きくなる方向に進むことによりルートノードから任意のノードへ到達可能であることが保証される。一方、後者の条件により、depthとhorizontal spreadの組合せが同一となるノードが存在しないことが保証される。

horizontal spreadは、図3(b)に示すように、直観的には、スパニングツリー上の水平方向における座標を表すものであり、各チャネルのhorizontal directionおよび、同じdepthを持つノード間のvertical directionの決定に用いられる。

これにより、各ノードに対してhorizontal spread(h)とdepth(d)からなる一意の2次元座標(h, d)を割り当てることが可能となる。

例として、図3(a)のネットワークに対するhorizontal spreadおよび2次元座標の割当ては、図3(b)のようになる。

3.1.3 各チャネルへの方向の割当て

最後に、各ノードに割り当てられた2次元座標を基に、各チャネルに対するhorizontal directionとvertical directionの割当てを行い、H/Vグラフの構築に必要なH/V directionの割当てを行う。

まず、次のようにしてhorizontal directionを各チャネルに割り当てる。座標 (x_s, y_s) から座標 (x_d, y_d) に向かうチャネルにおいて、

- (1) $x_s > x_d$ ならばleft方向を割り当て、
- (2) $x_s < x_d$ ならばright方向を割り当てる。

次に、同様にしてvertical directionを各チャネルに割り当てる。

表 1 H/V 方向の定義

Table 1 Definition of the H/V direction.

	$x_s > x_d$	$x_s < x_d$
$y_s > y_d$	left-up(LU)	right-up(RU)
$y_s = y_d$	left-down(LD)	right-up(RU)
$y_s < y_d$	left-down(LD)	right-down(RD)

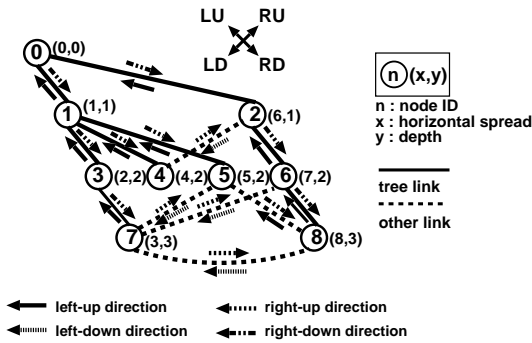


図 4 H/V グラフ
Fig. 4 H/V graph.

- (1) $(y_s > y_d) \vee ((y_s = y_d) \wedge (x_s < x_d))$, ならば up 方向を割り当て,
- (2) $(y_s < y_d) \vee ((y_s = y_d) \wedge (x_s > x_d))$, ならば down 方向を割り当てる.

そして、各チャネルに対して 4 つの方向からなる H/V direction を割り当てる。各チャネルの H/V direction は、horizontal direction (h) と vertical direction (v) の組合せ $HV(h, v)$ により次のように定める。

- (1) $HV(left, up)$ に対し left-up(LU)方向を割り当て,
- (2) $HV(left, down)$ に対し left-down(LD)方向を割り当て,
- (3) $HV(right, up)$ に対し right-up(RU)方向を割り当て,
- (4) $HV(right, down)$ に対し right-down(RD)方向を割り当てる。

以上をまとめたものを表 1 に示す。

本稿では、以降 H/V direction dir を持つチャネルを dir チャネルと呼ぶ。

各チャネルに対して H/V direction が割り当てられることにより、2次元有向グラフである H/V グラフが構築される。例として、図 3 におけるネットワークの H/V グラフは、図 4 に示すものになる。

H/V グラフにおいて、スパンニングツリーを構成するチャネルのみからなる部分グラフを、特に H/V ツリーと呼ぶ。

3.2 2次元 Turn モデルベースルーティングアルゴリズムの設計と定義

H/V グラフに対して、Turn モデルを適用してデッドロックフリールーティングアルゴリズムの設計を行う。

Turn モデルの適用手順は、通常、次のようになる。

- (1) パケット転送時に形成可能なターンを列挙する。
- (2) 列挙されたターンの連鎖により形成される循環構造の全パターンを列挙する。
- (3) 列挙された循環構造の除去に最低限必要となる禁止ターンを選択する。

我々は、これに次の 2 つの手順を追加することにより、H/V グラフにおけるトラフィックの分散を実現する。

- (1) 循環構造除去において、分散を考慮した禁止ターンの選択を行う。
 - (2) 特定の循環構造を探索により検出するアルゴリズムを適用して、不要な禁止ターンを除去する。
- 以下では、この手順に従って、デッドロックフリールーティングアルゴリズムを設計する方法を示す。

3.2.1 準備

最初に準備として、本項で用いられる表記を次に示す。

まず、ノード到着時のパケット転送方向 p_dir とノード通過後のパケット転送方向 n_dir により形成されるターンを T_{p_dir, n_dir} と表す。次に、ターン T_i をともなうパケット転送後にターン T_j をともなう転送が可能である場合のターンの連鎖を $C(T_i, T_j)$ と表す。そして、H/V グラフにおいて、 $\{C(T_i, T_j) \mid j = (i+1) \bmod n, i = 0, 1, \dots, n-1\}$ を形成する n 個のターンの集合 $\{T_0, T_1, \dots, T_{n-1}\}$ 、により循環構造が形成される場合、その循環構造を $L(T_0, T_1, \dots, T_{n-1})$ と表す。例として、up*/down*ルーティングにおける 1次元有向グラフでは、 $C(T_{up, down}, T_{down, up})$ となるターンの連鎖と $L(T_{up, down}, T_{down, up})$ の循環構造が形成される。

3.2.2 ターンの列挙

H/V グラフにおいて、ある H/V direction へ移動した後、その他の H/V direction へ移動した際に形成可能なすべてのターンを図 5 に示す。図 5 より、H/V グラフでは 4 つの H/V direction が存在するため、形成可能なターンは全部で 12 パターンとなる。

3.2.3 循環構造の列挙と禁止ターンの選択

図 5 に示したターンの連鎖により形成されるすべての循環構造の列挙とその除去に必要な禁止ターンの選択を、次の手順で行う。

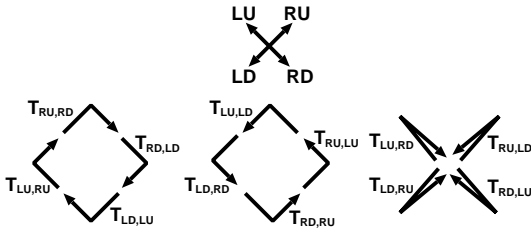


図5 H/V グラフにおいて形成可能なターン
Fig. 5 All possible turns in H/V graph.

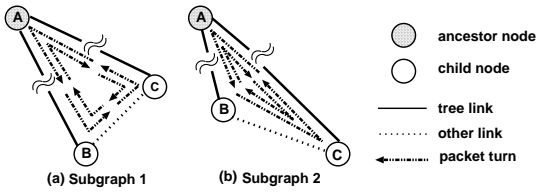


図6 H/V グラフにおける4つの基本的な循環構造
Fig. 6 Four possible cycles in H/V graph.

まず、H/V グラフ内の任意の2つのノードが、ツリーに属さないリンク (other link) により直接接続されている場合を考える。

図6は、H/V グラフにおける2パターンの部分グラフを表しており、図6(a), (b)において、ノードBとノードCはother linkにより直接接続され、ノードAがそれらの祖先ノードとなっている。2つの部分グラフの違いは、ノードBとノードCの垂直方向における相対位置の違いである。このような2つの子ノードは互いに、共通の祖先ノードを経由して2つ以上のtree linkをたどることにより到達可能であるので、このようなother linkが存在する場合には、これらのtree linkとother linkを通る2つの循環構造がたねに形成される。図6(a), (b)において、各部分グラフには、そのような2つの循環構造がそれぞれ形成されている。図6(a)における循環構造は、左回りの循環 $L_1(T_{LU,RD}, T_{RD,RU}, T_{RU,LU})$ 、および右回りの循環 $L_2(T_{LU,RD}, T_{RD,LD}, T_{LD,LU})$ であり、図6(b)における循環構造は、左回りの循環 $L_3(T_{LU,RD}, T_{RD,LU})$ 、および右回りの循環 $L'_3(T_{LU,RD}, T_{RD,LU})$ となる。ただし、循環 L_3 と L'_3 は論理的に同一である。

これらの循環構造をすべて破るために、各循環構造内の1つのターンを次のポリシーに基づいて禁止する。

- (1) ターン $T_{LU,RD}$ を禁止しない。
- (2) 可能な限り、選択した禁止ターンの組合せにより図2のような禁止ターンの偏りが発生しないようにする。

H/V グラフにおいて、任意のノード間の経路を保証するためには、任意のノードからLU方向のtree

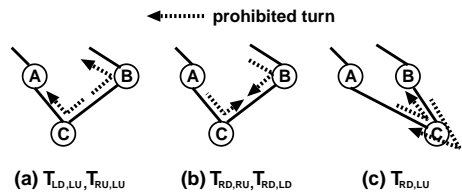


図7 H/V グラフにおける禁止ターン
Fig. 7 Prohibited turns in H/V graph.

channel を0回以上用いて任意の目的地ノードの祖先ノードに到達可能であり、かつ、祖先ノードに到達後にRD方向のtree channelを0回以上用いて任意の目的地ノードに到達可能である、という条件を満たす必要があるため、ターン $T_{LU,RD}$ を禁止することはできない。

これらのポリシーを考慮すると、循環 L_1 および L_2 を破るために禁止するターンの集合は、 $\{T_{RU,LU}, T_{LD,LU}\}$ または $\{T_{RD,RU}, T_{RD,LD}\}$ となり、循環 L_3 を破るための禁止ターンは $T_{RD,LU}$ となる。図7(a), (b)に、これらの選択により禁止ターンの分散が実現されていることを示す。図7(c)に示すように、ターン $T_{RD,LU}$ を禁止すると偏りが発生してしまうが、循環 L_3 を破るためにはこれ以外に選択肢がないので、この場合はやむをえない。

以上より、図6に示したすべての循環構造を破るために禁止するターン集合は2通り存在し、次のように定められる。

$$P_1 = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}\}$$

$$P_2 = \{T_{RD,RU}, T_{RD,LD}, T_{RD,LU}\}$$

次に、禁止ターン集合 P_1 または P_2 に属さないターンの連鎖により形成される循環構造の列挙を行う。ここでは禁止ターン集合として P_1 を選んだ場合の手順を示すが、 P_2 を選択した場合についても同様にして考えることができる。

まず、禁止ターン集合 P_1 に属するターンを含まない循環を検出するために、残りの9パターンのターン集合を $Q_1 = \{T_{LU,n_dir} \mid n_dir \in \{LD, RU, RD\}\}$ および $Q'_1 = \{T_{p_dir,n_dir} \mid p_dir, n_dir \in \{LD, RU, RD\}, p_dir \neq n_dir\}$ の2種類のターン集合に分類する。

定理1 ターン集合 Q_1 に属するターンを含む循環構造には、禁止ターン集合 P_1 に属するターンが必ず含まれる。 □

証明 ターン集合 Q_1 に属するターン T_x を含み、かつ禁止ターン集合 P_1 に属するターンを含まない循環構造が形成可能であると仮定する。 T_x はLU方向からその他のH/V directionへのターンであるので、このと

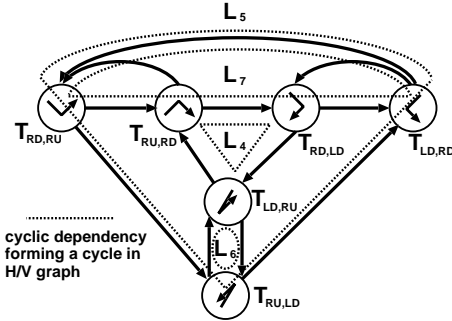


図8 ターン集合 Q'_1 における Turn Dependency Graph
Fig.8 Turn Dependency Graph for the turn set Q'_1 .

き、ターン T_x の直前に連鎖して循環を形成するターンは、ターン集合 $\{T_{p-dir,LU} \mid p-dir \in \{LD, RU, RD\}\}$ に属するものでなければならない。しかし、このターン集合は、禁止ターン集合 P_1 と同一であるため先の仮定に矛盾する。ゆえに、ターン集合 Q_1 に属するターンを含む循環構造には、禁止ターン集合 P_1 に属するターンが必ず含まれる。 □

定理 1 より、禁止ターン集合 P_1 に属する禁止ターンにより循環構造が破れるため、ターン集合 Q_1 に属するターンを禁止する必要がなくなる。これにより、 LU 方向をとまなうターンを含むすべての循環構造が除去される。このため、禁止ターン集合 P_1 を選択した場合に形成可能な循環構造は、ターン集合 Q'_1 に属する LU 方向をとまなわないターンのみで構成されるものに絞られる。そこで、そのような循環構造を識別するために、ターン集合におけるターン間の依存関係を示す TDG (Turn Dependency Graph) を導入する。TDG D は、 $D = G(V, E)$ で表され、 V は形成可能なターン集合を表し、 E は V に属する 2 つのターン間で形成可能なターンの連鎖の集合を表す。

図 8 に、ターン集合 Q'_1 における TDG を示す。

図 8 において、各頂点は、ターン集合 Q'_1 に属するターンの 1 つを表し、各頂点間を結ぶ矢印は 2 つのターン間の連鎖を表している。図 8 より、 Q'_1 に属するターンによって形成されるすべての循環構造は、破線で表される 4 つの循環構造のいずれか 1 つを必ず含むことが分かる。これら 4 つの循環構造は以下のとおりとなる。

- (1) $L_4(T_{RU,RD}, T_{RD,LD}, T_{LD,RU})$
- (2) $L_5(T_{RD,RU}, T_{RU,LD}, T_{LD,RD})$
- (3) $L_6(T_{LD,RU}, T_{RU,LD})$
- (4) $L_7(T_{RD,RU}, T_{RU,RD}, T_{RD,LD}, T_{LD,RD})$

先の選択ポリシーに基づいて 4 つの循環構造を破る

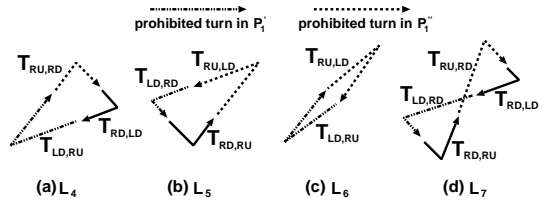


図 9 ターン集合 Q'_1 により形成可能な 4 つの循環構造
Fig.9 Four possible cycles formed by the turn set Q'_1 .

ための禁止ターン集合を選択すると、次の 2 通りの禁止ターン集合が定められる。

$$P'_1 = \{T_{LD,RU}, T_{LD,RD}\}$$

$$P''_1 = \{T_{RU,LD}, T_{RU,RD}\}$$

図 9 に、上記の 4 つの循環構造とそれらを破るための禁止ターン集合 P'_1 および P''_1 に属するターンをそれぞれを示す。

先に選択した禁止ターン集合 P_1 と合わせて、最終的に次の 2 通りの禁止ターン集合が定められる。

$$P_1 + P'_1 = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}, T_{LD,RU}, T_{LD,RD}\}$$

$$P_1 + P''_1 = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}, T_{RU,LD}, T_{RU,RD}\}$$

禁止ターン集合 P_1 により LU 方向をとまなうターンを含む循環構造が破れ、禁止ターン集合 P'_1 または P''_1 によりその他のターンを含む循環構造が破れるので、H/V グラフにおいて形成可能なすべての循環構造が破れ、デッドロックフリーであることが保証される。

先の選択において、禁止ターン集合として P_1 の代わりに P_2 を選択した場合には、同様の手順によって次の 2 通りの禁止ターン集合が定められる。

$$P'_2 = \{T_{LD,RU}, T_{LU,RU}\}$$

$$P''_2 = \{T_{RU,LD}, T_{LU,LD}\}$$

同様に、すべての循環構造を破るための禁止ターン集合として、次の 2 通りの禁止ターン集合が定められる。

$$P_2 + P'_2 = \{T_{RD,RU}, T_{RD,LD}, T_{RD,LU}, T_{LD,RU}, T_{LU,RU}\}$$

$$P_2 + P''_2 = \{T_{RD,RU}, T_{RD,LD}, T_{RD,LU}, T_{RU,LD}, T_{LU,LD}\}$$

3.2.4 循環構造検出アルゴリズムによる冗長禁止ターンの削減

ここでは禁止ターン集合として $P_1 = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}\}$ と $P'_1 = \{T_{LD,RU}, T_{LD,RD}\}$ を選択した場合について述べるが、その他のターン集合を選択した場合も同様にして考えることができる。

禁止ターン集合 P'_1 に属する 2 つの禁止ターンは図 9 の 4 つの循環構造を破るために必要である。しか

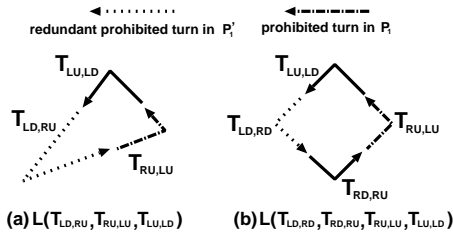


図 10 冗長な禁止ターン
Fig. 10 Redundant prohibited turns.

し、これら 2 つのターンを含む循環構造には、図 10 のように、禁止ターン集合 P_1 に属するターンを一緒に含むものも存在する。

図 10 における 2 つの循環構造には、禁止ターン集合 P_1 および禁止ターン集合 P_1' に属する禁止ターンがそれぞれ 1 つずつ含まれている、このような場合、 P_1' に属するターンを禁止せずとも P_1 に属する禁止ターンにより循環構造は破れる。そのため、禁止ターン集合 P_1' に属するターンをすべて禁止してしまうと冗長な禁止ターンが発生し、ルーティングの自由度が低下してしまう。

そこで、この問題を解決するために、H/V グラフにおける図 9 の 4 つの循環構造を探索により検出して、循環の除去に必要なターンのみを禁止するアルゴリズムを提案する。このアルゴリズムにより、ターン集合 P_1' に属するターンは、検出された循環構造に含まれる場合にのみ禁止されるようになるので、冗長な禁止ターンの削減が実現される。

以下、循環構造検出アルゴリズムの説明を行う。

まず、循環構造の検出は、H/V グラフにおいて、次の 2 つの条件のいずれか、または両方を満たす各ノードをそれぞれ起点として、深さ優先探索を行うことにより行われる。

- (a). ターン集合 P_1' に属するターン $T_{LD,RD}$ が形成可能である (1 つ以上の RU チャネルおよび RD チャネルが接続されている)。
- (b). ターン集合 P_1' に属するターン $T_{LD,RU}$ が形成可能である (2 つ以上の RU チャネルが接続されている)。

探索において、隣接ノードの訪問に利用される出力チャネルは、次の条件を満たす場合に選択可能であるとし、利用後に利用済のマークをつける。

1. P_1 に含まれる禁止ターンを形成しない (LU チャネルではない)。
2. 利用済マークがついてない。
3. 以前の探索において禁止された、ターン集合 P_1' に属するいずれかのターンを形成しない

探索の手順を次に示す。探索は 2 つの手順からなり、起点となるノードが条件 (a) を満たす場合に手順 1 を、条件 (b) を満たす場合に手順 2 をそれぞれ実行する。

手順 1: 条件 (a) に該当するノードを起点とした探索

まず、ターン $T_{LD,RD}$ を形成するチャネルのうち、起点ノードから出る方向となる RD チャネルのいずれか 1 つを選び、到達可能な隣接ノードを訪問する。以降、到達先のノードにて選択可能な出力チャネルがある限り深さ優先探索により訪問を続け、訪問先のノードにて選択可能な出力チャネルが存在しなければ直前のノードに戻って探索を続ける。探索により、 LD チャネルを通して起点ノードに戻ってきたならば循環構造が検出されたことになり、その LD チャネルと出発時に利用した RD チャネルの間に形成されるターン $T_{LD,RD}$ を禁止する。検出される循環構造は、ターン $T_{LD,RD}$ を含み禁止ターン集合 P_1 に属するターンを含まないので、循環 L_5 または L_7 のいずれかとなる。

探索は選択可能な出力チャネルがなくなるまで続ける。この作業は、起点ノードに存在する各 RD チャネルに対して順に行われる。

手順 2: 条件 (b) に該当するノードを起点とした探索

この探索は、次の点を除いて条件 (a) における探索とほぼ同様に行われる。

1. 起点ノードからの最初の訪問には、ターン $T_{LD,RU}$ を形成するチャネルのうち、起点ノードから出る方向となる RU チャネルを用いる。
2. 循環構造検出時には、ターン $T_{LD,RU}$ が禁止される。

この探索により検出される循環構造は、ターン $T_{LD,RU}$ を含み禁止ターン集合 P_1 に属するターンを含まないので、循環 L_4 または L_6 のいずれかとなる。

上記のアルゴリズムは、禁止ターン集合として $P_1 + P_1'$ 、 $P_2 + P_2'$ および $P_2 + P_2''$ のいずれかを選択した場合についても、検出の対象となる 2 つのターン P_1' を、 $P_1'' = \{T_{RU,LD}, T_{RU,RD}\}$ 、 $P_2' = \{T_{LD,RU}, T_{LU,RU}\}$ および $P_2'' = \{T_{RU,LD}, T_{LU,LD}\}$ にそれぞれ置き換えることにより同様にして適用することが可能である。ただし、禁止ターン集合として P_2 を選択した場合には、探索における禁止ターン集合 P_1 を P_2 に置き換える必要がある。

図 11 に、禁止ターン集合として $P_1 + P_1'$ を選択した場合に、探索アルゴリズムにより検出される循環構造の例を示す。

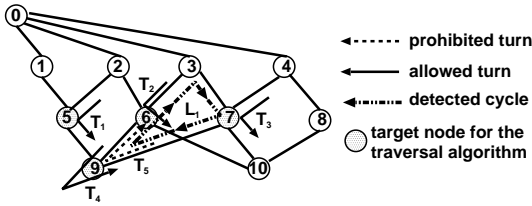


図 11 探索アルゴリズムにより検出された循環構造

Fig. 11 A detected cycle by the traversal algorithm.

図 11 において、5 つのターン T_1, \dots, T_5 は、ターン集合 P'_1 に属するターンであり、ノード 5, 6, 7, 9 が探索の起点となっている。図 11 より、探索アルゴリズムにより検出される循環構造は、循環 L_1 だけであるため、5 つのターンのうち、循環 L_1 に含まれるターン T_5 のみを禁止すればよいことが分かる。

ノード数を n 、ノードあたりのリンク数を l とすると、探索アルゴリズムの計算量は $O(n^2 * l)$ となる。

3.2.5 ルーティングアルゴリズムの定義

3.2.3 項で定めた 4 通りの禁止ターン集合と 3.2.4 項で定めた循環構造検出アルゴリズムに基づいて、4 つのルーティングアルゴリズムを次のように定義する。

ここで、循環構造検出アルゴリズムの適用により禁止される全禁止ターン集合 DP を次のように表す。

$$DP = DA(H, P_{stat}, P_{cond})$$

DA は、対象とする H/V グラフ H において、つねに禁止されるターン集合と検出対象とする 2 つの禁止ターンからなるターン集合をそれぞれ P_{stat}, P_{cond} として、循環構造検出アルゴリズムを適用した結果禁止されるすべてのターン集合を返す関数とする。

まず、ターン集合 $P_1 = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}\}$ を禁止ターンに含む 2 つのルーティングアルゴリズムを定義する。

ターン集合 P_1 を禁止することにより、 LU 方向に向かうすべてのターンが禁止されるため、 LU 方向への移動は最初に行う必要がある。そこで、このようなルーティングアルゴリズムをまとめて L-turn (Left-up first turn) ルーティングと呼ぶ。

1. L-turn/ α ルーティング

H/V グラフにおいて、ターン集合 $P_1 = \{T_{LD,LU}, T_{RU,LU}, T_{RD,LU}\}$ に属するターンを禁止し、ターン集合 $P'_1 = \{T_{LD,RU}, T_{LD,RD}\}$ を検出対象とする循環構造検出アルゴリズムの適用により、ターン集合 $DP_1 = DA(H, P_1, P'_1)$ を禁止するルーティングアルゴリズムを L-turn/ α ルーティングと呼ぶ。

2. L-turn/ β ルーティング

H/V グラフにおいて、ターン集合 $P_1 = \{T_{LD,LU},$

$T_{RU,LU}, T_{RD,LU}\}$ に属するターンを禁止し、ターン集合 $P''_1 = \{T_{RU,LD}, T_{RU,RD}\}$ を検出対象とする循環構造検出アルゴリズムの適用により、ターン集合 $DP_2 = DA(H, P_1, P''_1)$ を禁止するルーティングアルゴリズムを L-turn/ β ルーティングと呼ぶ。

次に、ターン集合 $P_2 = \{T_{RD,RU}, T_{RD,LD}, T_{RD,LU}\}$ を禁止ターンに含む 2 つのルーティングアルゴリズムを定義する。

ターン集合 P_2 を禁止することにより、 RD 方向からその他の方向に向かうすべてのターンが禁止されるため、 RD 方向への移動は最後に行う必要がある。そこで、このようなルーティングアルゴリズムをまとめて R-turn (Right-down last turn) ルーティングと呼ぶ。

3. R-turn/ α ルーティング

H/V グラフにおいて、ターン集合 $P_2 = \{T_{RD,RU}, T_{RD,LD}, T_{RD,LU}\}$ に属するターンを禁止し、ターン集合 $P'_2 = \{T_{LD,RU}, T_{LU,RU}\}$ を検出対象とする循環構造検出アルゴリズムの適用により、ターン集合 $DP_3 = DA(H, P_2, P'_2)$ を禁止するルーティングアルゴリズムを R-turn/ α ルーティングと呼ぶ。

4. R-turn/ β ルーティング

H/V グラフにおいて、ターン集合 $P_2 = \{T_{RD,RU}, T_{RD,LD}, T_{RD,LU}\}$ に属するターンを禁止し、ターン集合 $P''_2 = \{T_{RU,LD}, T_{LU,LD}\}$ を検出対象とする循環構造検出アルゴリズムの適用により、ターン集合 $DP_4 = DA(H, P_2, P''_2)$ を禁止するルーティングアルゴリズムを R-turn/ β ルーティングと呼ぶ。

定理 2 L-turn/ α , L-turn/ β および R-turn/ α , R-turn/ β はデッドロックフリーである。 □

証明 H/V グラフにて形成可能なすべての循環構造は、各ルーティングアルゴリズムにおいて選択している禁止ターン集合により破られる。ゆえに、L-turn/ α , L-turn/ β および R-turn/ α , R-turn/ β はデッドロックフリーである。 □

定理 3 L-turn/ α , L-turn/ β および R-turn/ α , R-turn/ β では任意のノード間の経路が保証される。 □

証明 H/V ツリーに属するチャンネルの方向は、 LU 方向または RD 方向のみであり、 RD 方向から LU 方向へのターンは禁止されているので、H/V ツリー内で形成可能なターンは $T_{LU,RD}$ のみとなる。 $T_{LU,RD}$ は禁止されていないので、H/V ツリーにおいては任意のノード間でのパケット転送が保証される。ゆえに、L-turn/ α , L-turn/ β および R-turn/ α , R-turn/ β では任意のノード間の経路が保証される。 □

L-turn および R-turn ルーティングでは、禁止ター

ンを行わない限り任意の非最短経路を選択することが可能である。しかし、ライブロックフリーの保証や任意の非最短経路を許した場合に発生しやすくなるホットスポット形成の防止のため、各ルーティングアルゴリズムでは、任意のノード間における選択可能な経路のうち最短となるものだけを選択するものとする。

各ルーティングアルゴリズムにおけるノード間の経路計算は、 $up^*/down^*$ ルーティングと同様にして次のように行われる。ここでは、L-turn/ α の場合について述べるが、提案した他のルーティングアルゴリズムについても同様にして行われる。

- (1) H/V グラフの各ノードにおいて、ターン集合 P_1 に属するターンを形成するチャンネル間の移動を禁止する。
- (2) H/V グラフにおいて、3.2.4 項で示した条件 (a), (b) のいずれか、または両方を満たす全ノードにおいて循環構造検出アルゴリズムを順に適用し、循環構造の検出を行う。検出された循環構造において、ターン集合 P_1' に属するターンを形成するチャンネル間の移動を禁止する。
- (3) H/V グラフの全ノードにおいて、ダイクストラのアルゴリズム⁷⁾を適用し、全ノード間の最短経路を算出する。探索においては、次の条件を守るものとする。
 - (a) 禁止されたチャンネル間の移動はできない。
 - (b) 各辺の重みはすべて等しい。

最短経路が複数存在する場合には理論的にはすべての経路が選択可能であるが、(1) 複数経路選択の可否、(2) 経路選択が行われるタイミング、(3) 経路選択ポリシー、などは、 $up^*/down^*$ ルーティングと同様に対象とするスイッチの実装に依存する。

複数経路が選択可能な場合に、どのタイミングで経路を選択できるかは、分散ルーティング方式とソースルーティング方式のどちらを選択するかで異なる。前者では、スイッチ到達時にパケットがそのつどルーティングテーブルを参照して経路を決定するため、途中経路で動的に経路を選択できるが、後者では、出発ノードにおいて目的ノードまでの利用経路が決定されるため、出発ノードでのみ経路の選択が可能となる。例として、Autonet⁴⁾は、分散ルーティング方式を採用し、経路選択ポリシーは利用可能な出力ポートの中でポート番号が最小のものを選択するものとなっている。一方、Myrinet¹⁾は、ソースルーティング方式を採用し、経路選択ポリシーは、制御プログラム(GM)における設定により、ランダム、ラウンドロビン、負荷分散を考慮したもの、などを指定することができる⁸⁾。

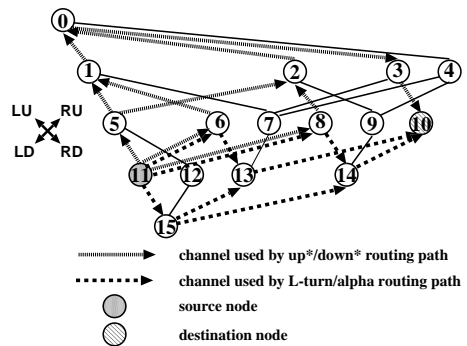


図 12 $up^*/down^*$ および L-turn/ α ルーティングの経路例
Fig. 12 An expmale of paths of $up^*/down^*$ and L-turn routing.

$up^*/down^*$ ルーティングが適用可能なこれらのネットワークに対して、理論上は、提案した各ルーティングアルゴリズムを適用することが可能である。

図 12 に、L-turn/ α および $up^*/down^*$ ルーティングの経路例を示す。図 12 は、16 ノード構成の H/V グラフにおけるノード 11 からノード 10 へのパケット転送において、各ルーティングアルゴリズムにより選択可能なすべての経路を示している。図 12 において、各ルーティングアルゴリズムは、ともに 4 通りの経路を持つことが分かる。しかし、 $up^*/down^*$ ルーティングの経路はすべて 5 ホップを要し、ルートノードを必ず通らなくてはならないのに対し、L-turn/ α の経路はすべて 3 ホップを要するだけであり、また、ルートノードを通る必要がなく、経路も分散されている。

4. 関連研究

本稿で提案した L-turn, R-turn ルーティングは、2 次元有向グラフに対する Turn モデルの適用により、仮想チャンネルを付加せずともトラフィックの分散を実現し、スループットを向上させている。

これに対し、複数の仮想チャンネルを用いることにより上記の問題解決を図るルーティングアルゴリズムに関する研究^{9)~11)}も行われている。これらの手法では、仮想チャンネルを用いて複数の仮想ネットワークを必要数用意し、(1) $up^*/down^*$ ルーティングもしくは L-turn, R-turn ルーティングなどの仮想チャンネルを必要としないルーティングを併用し、必要に応じてパケット転送中に仮想ネットワークを切り換える^{9),10)}、(2) ノード対ごとに利用する仮想ネットワークを振り分ける¹¹⁾、などにより最短経路を実現している。これにより、パケットの衝突やネットワークバンド幅の占有時間が減少し、トラフィックの分散が実現される。

現在、InfiniBand³⁾などの仮想チャンネル(レーン)

の利用が可能な SAN と Myrinet¹⁾のように仮想チャネルの切替えができない SAN との両方が存在しており、仮想チャネルの有無に依存しない L-turn, R-turn ルーティングは、両者のルーティングに適用可能な数少ない研究であるといえる。

5. 評価

シミュレーションによる各 L-turn, R-turn ルーティング、および比較のための up*/down*ルーティングの評価を以下に示す。

5.1 評価用シミュレータ

我々は、評価用のフリットレベルシミュレータを、C++ 言語を用いて実装した。このシミュレータは、任意のトポロジにより結合されたスイッチ、およびそれに接続されるホスト PC 間のパケット転送動作をフリットレベルでシミュレートする。シミュレータ上では、スイッチは、チャネルバッファ、クロスバ、リンクコントローラ、制御回路により構成され、パケットスイッチング方式、ルーティングアルゴリズム、ポート数などを指定できる。また、ホスト PC は、パケットの生成および送受信を行い、パケットサイズ、トラフィックパターンなどを指定できる。

5.2 評価条件

評価は、まず 64 ノードからなるイレギュラーネットワークを対象として行った。このイレギュラーネットワークは、異なる 10 パターンのトポロジを、同一ノード間にリンクを 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ものである。一方、実際の SAN では規則性や階層性がある程度みられるため、イレギュラーネットワークと同様にレギュラーネットワークにおける評価も重要となる。最近では Fat-tree^{12),13)}や Myrinet-Clos⁸⁾などの多数のリンクと中間スイッチを用いる階層接続網が用いられる傾向にあるが、これらのコストの大きい結合網では、up*/down*ルーティングと L-turn, R-turn ルーティングの双方でルーティングは最適となり、差は生じない。そこで、より低コストの実装時に用いられる 8×8 の 2 次元トーラスを、レギュラーネットワークの代表として評価対象とした。

各トポロジにおけるルートノードは、crossing path が最小であるものが選択され、crossing path が等しい場合には、各ノード間の最短距離の平均が最小であるものを選択した。

各ノードは、8 ポートを持つ 1 つのスイッチとそれ

表 2 10 パターンの 64 ノードイレギュラーネットワークにおける各性能値の平均値

Table 2 The average value of the performance metrics on 10 irregular networks with 64 nodes.

	PT(SDPT)	PPT	スループット	
			Uni.	Bit.
up*/down*	3.019(3.669)	96.6	0.0452	0.0477
L-turn/ α	2.875(2.225)	18.9	0.0576	0.0707
L-turn/ β	2.990(2.269)	20.6	0.0572	0.0711
R-turn/ α	2.882(2.238)	19.4	0.0468	0.0485
R-turn/ β	2.900(2.272)	20.8	0.0471	0.0472

に接続される 4 つのホスト PC からなり、スイッチの残りの 4 ポートは他のスイッチとの接続に用いられる。

各ルーティングアルゴリズムは、分散ルーティング方式によるものとし、選択可能な経路のうちホップ数が最小となる経路のみを選択可能であるとしている。また、目的ノードまでに複数の経路が存在する場合には、複数の出力ポートが選択可能となる経路上の各スイッチにおいて、選択可能な出力ポートのうちポート番号が最小となるものを選択するようにしている⁴⁾。出力ポートの選択ポリシー (output selection function) には、このほかにランダムや LRU など存在し、評価対象のトポロジにおいては、選択ポリシーの違いが 2 本から 4 本の仮想チャネルを用いる適応型ルーティングのスループットおよびレイテンシに若干の影響を与えることが確認されている¹⁴⁾。しかし、本稿で対象としている各ルーティングアルゴリズムは仮想チャネルを利用しないため、出力数が少なくなる分、皆無とはいえないものの、選択ポリシーが評価結果に与える影響はより小さなものになると考えられる。また、本評価の主な目的がルーティングアルゴリズムどうしの比較にあるため、今回は経路選択ポリシーについては特に考慮しないことにする。

各シミュレーションは 1,000,000 クロック実行され、最初の 50,000 クロックを評価の対象外とした。パケット長は 128 フリットで、トラフィックパターンには uniform と非一様である bit-reversal をそれぞれ用いた。スイッチング方式はパーチャルカットスルーを用い、スイッチ間のフリット転送には、3 クロックを要するものとした。

5.3 イレギュラーネットワークの評価

表 2 は、10 パターンの 64 ノードイレギュラーネットワークにおける各ルーティングアルゴリズムの、(1) ノードあたりの禁止ターン数 (PT) およびその標準偏差 (SDPT)、(2) ベア禁止ターン数 (PPT)、(3) uniform および bit-reversal トラフィックにおけるスループット、の平均値をそれぞれ表している。SDPT と

1 つのチャネルを通るノード対の経路数の最大値を示す

表3 8×8 次元トラスにおける各性能値Table 3 The performance metrics on 8×8 2D Torus.

	$PT(SDPT)$	PPT	スループット	
			Uni.	Bit.
up*/down*	2.500(2.264)	80	0.0455	0.0414
L-turn/ α	2.640(1.789)	17	0.0771	0.0812
L-turn/ β	2.625(2.012)	20	0.0691	0.0800
R-turn/ α	2.641(1.789)	17	0.0507	0.0513
R-turn/ β	2.625(2.012)	20	0.0491	0.0513

PPT は、ネットワークにおける禁止ターンの分散の度合いであり、数値が小さいほどより均等に分散していることを示す。スループットは、受信トラフィック（フリット/クロック/ホスト PC ）の最大値であり、全ホスト PC が毎クロックに1フリット受信する場合を1.00としている。

まず、表2より、L-turn および R-turn ルーティングは、up*/down*ルーティングよりも小さな PT 、 $SDPT$ および PPT を示していることが分かる。特に PPT は、up/down*ルーティングの約20%にまで減少している。これより、L-turn および R-turn ルーティングでは、より均等な禁止ターンの分散と禁止ターンの削減を実現していることが分かる。

次に、2つのトラフィックにおけるスループットを見てみると、up*/down*ルーティングに比べて2つのL-turn ルーティングの性能向上が大きく、uniformトラフィックにおいて、約28%、bit-reversalトラフィックにおいて約49%の性能向上を達成していることが分かる。一方、禁止ターンの分散の度合いはほぼ同じであるにもかかわらず、2つのR-turn ルーティングのスループットはL-turn ルーティングに比べて劣り、up*/down*ルーティングと同程度となっている。この理由として、R-turn ルーティングにおける次の2つの要因が考えられる。(1) ルートノードに近づく LU 方向へのターンが、ターン $T_{RD,LU}$ を除いてすべて許可されている、(2) ルートノードから離れる RD 方向から他の方向へのターンがすべて禁止されている。この2点より、R-turn ルーティングではルートノード方向へトラフィックが集中しやすくなり、スループットが低くなったものと考えられる。

5.4 2次元トラスの評価

表3は、 8×8 次元トラス上の uniform および bit-reversal トラフィックにおける各ルーティングアルゴリズムの、(1) PT および $SDPT$ 、(2) PPT 、(3) uniform および bit-reversal トラフィックにおけるスループット、をそれぞれ表している。

表3より、L-turn および R-turn ルーティングの PT は、up*/down*ルーティングに比べて大きいもの

の、 $SDPT$ および PPT は、イレギュラーネットワークの場合と同様に小さくなっている。これより、L-turn および R-turn ルーティングは、2次元トラスにおいても、より均等な禁止ターンの分散を実現していることが分かる。

次に、スループットに関しては、2つのL-turn ルーティングがその他のルーティングアルゴリズムよりも大きなスループットを達成していることが分かる。特に、L-turn/ α は、up*/down*ルーティングに対し、uniformトラフィックにおいて、約70%、bit-reversalトラフィックにおいて約96%の性能向上を達成している。一方、2つのR-turn ルーティングのスループットは、up*/down*ルーティングと比べるとbit-reversalにおいては約24%の性能向上を達成しているが、イレギュラーネットワークの場合と同様に、L-turn ルーティングに比べると劣っている。

以上の結果より、L-turn ルーティングにおいて、禁止ターンの分散を実現することにより高スループットが達成されることが確認された。

6. ま と め

本稿では、up*/down*ルーティングにおいて禁止ターンが偏る問題を改善するために、2次元有向グラフである H/V グラフを構築し、Turn モデルを適用して禁止ターンの分散を実現するデッドロックフリールーティングアルゴリズムを設計する方法を示した。

提案した方法により、形成可能なデッドロックフリールーティングアルゴリズムが全部で4つ存在することが明らかになった。4つのルーティングアルゴリズムは、それぞれ2つのL-turn ルーティングとR-turn ルーティングに分類され、各ルーティングアルゴリズムは、分散を考慮した禁止ターンの選択と循環構造検出アルゴリズムの適用により、禁止ターンの分散と不要な禁止ターンの除去を実現した。

シミュレーションの結果、L-turn ルーティングおよびR-turn ルーティングは、禁止ターンの分散を実現し、up*/down*ルーティングよりも高いスループットを示すことが分かった。特に、L-turn ルーティングは、最大で96%のスループット向上を示すことが分かった。

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) Kudoh, T., et al.: RHiNET: A network for

- high performance parallel computing using locally distributed computing, *Proc. Innovative Architecture for Future Generation High-Performance Processors and Systems*, pp.69–73 (1999).
- 3) I.T. Association: InfiniBand architecture, Specification, available from the InfiniBand Trade Association (2001).
 - 4) 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).
 - 5) Glass, C.J. and Ni, L.M.: Maximally Fully Adaptive Routing in 2D Meshes, *Proc. International Symposium on Computer Architecture*, pp.278–287 (1992).
 - 6) 鯉淵道紘, 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.43, No.SIG 9(HPS 3), pp.119–134 (2001).
 - 7) Dijkstra, E.W.: A Note on Two Problems in Connexion with Graphs, *Numerische Mathematik*, Vol.1, pp.269–271 (1959).
 - 8) Seitz, C.L.: *Recent Advances in Cluster Networks*, available from the Myricom, Inc. (2001).
 - 9) Silla, F. and Duato, J.: 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).
 - 10) Koibuchi, M., Jouraku, A. and Amano, H.: Deterministic Routing Techniques by Dividing into Sub-Networks in Irregular Networks, *IASTED International Conference on Networks, Parallel and Distributed Processing, and Applications*, pp.143–148 (2002).
 - 11) Skeie, T., Lysne, O. and Theiss, I.: Layered Shortest Path (LASH) Routing in Irregular System Area Networks, *Proc. International Parallel and Distributed Processing Symposium*, pp.162–169 (2002).
 - 12) Petrini, F., Feng, W.C. and Hoisie, A.: The Quadrics network (QsNet): High-performance clustering technology, *Proc. Hot Interconnects*, pp.125–130 (2001).
 - 13) Leiserson, C.E.: Fat-trees: Universal networks for hardware-efficient supercomputing, *IEEE Trans. Comput.*, Vol.34, No.10, pp.892–901 (1985).
 - 14) Martinez, J.C., Silla, F., Lopez, P. and Duato, J.: On the Influence of the Selection Function on the Performance of Networks of Workstations, *Proc. 2000 International Symposium on High Performance Computing*, pp.292–300 (2000).

(平成 15 年 2 月 3 日受付)

(平成 15 年 5 月 19 日採録)

上樂 明也

平成 12 年慶應義塾大学大学院修士課程修了。現在、同大学大学院理工学研究科博士課程に在学中。相互結合網に関する研究に従事。



鯉淵 道紘

平成 15 年慶應義塾大学大学院修士課程修了。現在、同大学理工学部情報工学科訪問研究員、日本学術振興会特別研究員。相互結合網に関する研究に従事。



天野 英晴 (正会員)

昭和 61 年慶應義塾大学大学院工学研究科博士課程修了。工学博士。現在、慶應義塾大学理工学部情報工学科教授。計算機アーキテクチャの研究に従事。

