

# Manhattan Street Network における 代理送受信を用いたルーティング手法

江草俊文<sup>†</sup> 小畑正貴<sup>††</sup>

Manhattan Street Network (MSN) は、ローカルエリア・メトロポリタンエリアの packets 通信を行うために設計されている。我々は、MSN を並列計算機のネットワークへ応用するために、従来のルーティング法を見直し、代理送受信ノードの概念を導入することで、より少ないリソースでデッドロックフリー、メッセージの FIFO 性を満たし、ブロードキャストもできる MSN with Proxy Send/Receive (MSN/P) を提案する。双方向トーラス・MSN などとソフトウェアシミュレーションによって比較し、MSN/P は MSN に対してほとんど性能を低下させずに、並列計算機用ネットワークとしてのさまざまな利点が提供できることを示す。

## Manhattan Street Network with Proxy Send/Receive

TOSHIFUMI EGUSA<sup>†</sup> and MASAKI KOHATA<sup>††</sup>

The Manhattan Street Network (MSN) is a network technology designed for packet communications in a local or metropolitan area. In this paper, we propose a new routing algorithm, MSN/P, based on a "Proxy Send/Receive" model for the MSN. The new routing algorithm is simpler than standard algorithms, and has many features including deadlock free processing, FIFO order message transfer and broadcast abilities. We evaluate the MSN/P network performance by comparison with that of MSN and the uni- or bi-directional torus. Results show that the performance of MSN/P is comparable to MSN, and the MSN/P offers numerous advantages as an inter-connection network of parallel computers.

### 1. はじめに

ローカルエリア・メトロポリタンエリアの packets 通信を行うために設計された Manhattan Street Network (MSN)<sup>1)</sup> は、単方向の通信メディアをトーラス状に接続し、通信可能な方向を互い違いにすることで単方向の欠点をカバーしたネットワークである。

単方向化することで、双方向のトーラスネットワークよりも少ないリソースで構成できるなどの利点がある。また、より単純なハードウェアを用いることでルーターの動作周波数をあげることが容易になり、その結果として複雑なハードウェアを持ったネットワークよりも良い性能が得られる場合もある<sup>2)</sup>。

しかし、従来の MSN のルーティング法<sup>3)</sup> では

- デッドロックへの対応がなされていない
  - メッセージの FIFO 性が保証されない
  - ルーティングアルゴリズムが複雑である
  - 効率的なブロードキャストができない
- など多くの問題点があり、そのまま並列計算機のネットワークとして用いることは難しい。

そこで我々は、MSN を並列計算機のネットワークへ応用するために代理送受信ノードの概念を導入し、

- デッドロックフリーの保証
- メッセージの FIFO 性の保証
- 中継ノードでの容易なルーティング
- 効率的なブロードキャストが可能

などの特徴を持った Manhattan Street Network with Proxy Send/Receive (MSN/P) を提案する。

2 章では従来の MSN を簡単に紹介する。3 章では、我々の提案する MSN/P の概要、ルーティングアルゴリズム、およびデッドロック回避法について詳述し、4 章ではソフトウェアシミュレーションによる性能評価、ハードウェア量の比較について述べる。5 章ではまとめと MSN/P の 3 次元への拡張、単方向ネット

<sup>†</sup> 岡山理科大学大学院工学研究科  
The Graduate School of Engineering, Okayama University of Science

<sup>††</sup> 岡山理科大学工学部情報工学科  
Department of Information and Computer Engineering,  
Faculty of Engineering, Okayama University of Science

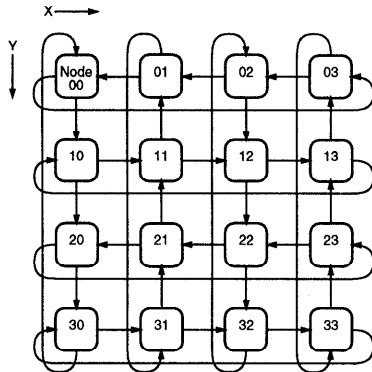


図1 Manhattan Street Network のトポロジ  
Fig.1 Topology of Manhattan Street Network.

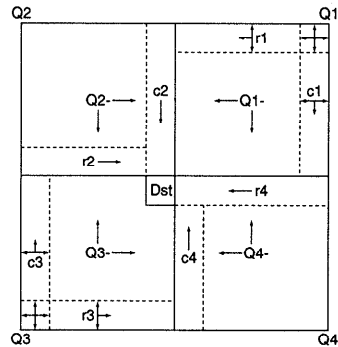


図2 Rule 1での Preferred Path  
Fig.2 Preferred Path in Rule 1.

ワークのメッシュへの応用について簡単に紹介する。

## 2. Manhattan Street Network

MSN のトポロジ (図1) は、

- プロセッサ間は単方向のネットワークで接続
- Y・Xの番号が偶数か奇数かで通信可能な方向が異なる

ようなネットワークである。つまり、隣りあうリンクが相互に逆向きの通信路を提供しあうことで単方向の欠点を補ったネットワークといえる。一般に MSN では row/column, または, avenue/street という表現で横/縦のリンクを表す場合が多いが、本稿では、後述の MSN/P との整合性から X/Y で横/縦のリンクを示すことにする。

ルーティングアルゴリズムとしては、Relative Address (式(1)) に基づいた方法が提案されている<sup>3)</sup>。

$$y = \frac{m}{2} - \left\{ \left( \frac{m}{2} - D_x(y_{fr} - y_{to}) \right) \bmod m \right\},$$

$$x = \frac{n}{2} - \left\{ \left( \frac{n}{2} - D_y(x_{fr} - x_{to}) \right) \bmod n \right\} \quad (1)$$

ただし、 $m \times n$  のネットワークにおいて、 $(y_{fr}, x_{fr})$  は現在のノード、 $(y_{to}, x_{to})$  は受信ノード、 $D_y, D_x$  は受信ノードの Y, X が偶数なら-1, 奇数なら1とする。

実際にルーティングを決定するためには、求められた Relative Address  $(y, x)$  を用いて、複雑な条件判定によって Preferred path を決定し、

- (1) Preferred path が1方向の場合、そちらにルーティングする。
- (2) Preferred path がない、または2方向ある場合は、どちらか一方を任意に選択し、ルーティングする。

Preferred Path の決定法は複数あり、決定論的な方法やランダムルーティングが提案されている。本稿では、文献3)で紹介されている最短経路を選択する

Rule 1 を対象に検討するため簡単に説明する。

Rule 1 では、Relative Address  $(y, x)$  を用いて、図2のように Preferred Path を決定する。ここでは、図によって簡素に示したが、実際に Preferred Path を決定するための条件式は後述する MSN/P のルーティング情報の生成 (図4) と同程度の複雑さになる。

図2は、各ノードのリンクの通信可能な向きと小領域内の矢印の方向が一致した場合、その方向が Preferred Path であることを示す。3方向以上の矢印が示されている領域もあるが、もともと出力は2方向しかないので、2つ以上の Preferred Path はありえない。

同一の Preferred Path を持つセルが入力された場合は、

- 一方のセルをルーティングして、もう一方は再送要求を出す
- 一方のセルをルーティングして、もう一方は Preferred Path ではない方向にルーティングするなどの方法がある。後者では、Preferred Path ではない方向にルーティングされたセルは、次以降のホップで正しい方向にルーティングされ、最終的に受信ノードへと到達する。

以上の方法をそのまま並列計算機のネットワークに持ち込んだ場合の問題は次の点である。

- デッドロックへの対応はまったく考えられていない
- メッセージの FIFO 性がない
- ブロードキャストできない
- ルーティング制御が複雑で、送信ノードから受信ノードに至るすべてのノードでそのつど計算しなくてはならない。

## 3. MSN with Proxy Send/Receive

### 3.1 MSN with Proxy Send/Receive の概要

並列計算機のネットワークでは、デッドロック対策

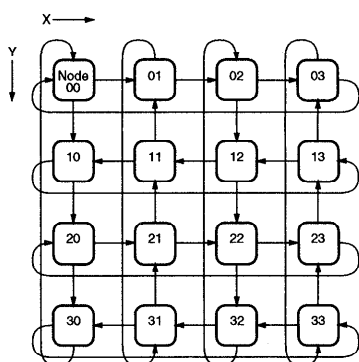


図3 MSN with Proxy Send/Receiveのトポロジ  
Fig.3 Topology of MSN with Proxy Send/Receive.

が必須である。また、メッセージのFIFO性を保証することで、通信プロトコルの単純化が可能となる。また、受信メッセージの再構築が不要となる。

そこで我々は、従来のMSNのルーティングに代理送受信の概念を採り入れることでデッドロックフリー、メッセージのFIFO性の保証、効率的なブロードキャスト、容易な中継ノードでのルーティング制御が実現できるManhattan Street Network with Proxy Send/Receive (MSN/P)を提案する。

MSN/PのトポロジはMSN(図1)とまったく同じでよいが、以降の解説ではXリンクの向きをすべて逆にした図3を用いている。

### 3.2 MSN/Pのルーティング

MSN/Pでのルーティングは、Xリンクを優先して使い、その後Yリンクを用いて受信ノードへと転送する次元順ルーティングを行う。ただし、MSN/Pでは通信可能なリンクの向きが強く制限されているので、単純にXリンクを用いると受信ノードから遠ざかってしまう場合がある。そのため、代理送信ノード・代理受信ノードという概念を導入することで、できるだけ最短に近い経路を選択できるようにする。

- (1) 必要があれば、代理送信ノードへ送信する。
- (2) Xリンクを用いて、受信ノードに近づく。
- (3) Yリンクを用いて、受信ノードに近づく。
- (4) 必要があれば、代理受信ノードから受信ノードへ送信する。

代理送信・代理受信ノードを用いることで、その間のルーティングは、単純なXリンク優先になり、かつ、たいていの場合には最短経路が選択される。

MSN/Pでは、まず、送信開始時にルーティング情報の生成を行う。ルーティング情報は、代理送信・受信のフラグおよびX・Yの相対アドレスで構成される。代理送信・受信のフラグは、真(true)なら代理

送信・受信が必要であることを示す。相対アドレスは、送信ノードと受信ノードの絶対アドレスの差の絶対値である。しかし、代理送信ノードが必要な場合は送信ノードの代わりに代理送信ノードの絶対アドレスを用い、代理受信ノードが必要な場合は受信ノードの代わりに代理受信ノードの絶対アドレスを用いて相対アドレスを求めることとする。

次に実際のルーティングであるが、送信ノードでは代理送信が必要ならばYリンクを用いて代理送信を行い、不要ならば中継ノードと同様の処理を行う。中継ノードでは、Xリンクを優先して送信し、その後Yリンクを使って送信をする。その際、X・Yの相対アドレスをデクリメントする。相対アドレスが(0,0)になれば、受信ノードに達したとして判断されるが、代理受信フラグが真の場合は、Xリンクを用いて1回だけ送信をする。

以上をまとめると、送信ノードを $(y_{src}, x_{src})$ 、受信ノードを $(y_{dst}, x_{dst})$ とすると、相対アドレス $(y', x')$ および、代理送信フラグ $f_{PS}$ 、代理受信フラグ $f_{PR}$ は図4のアルゴリズムで求められる。また、中継ノードでは、図5に示すアルゴリズムで送信するリンクを決定すればよい。

また、MSNの $D_y$ 、 $D_x$ (式(1))は受信ノードで決定されるため、すべてのセルについてそのつど評価する必要があるが、MSN/Pでは送信ノードで決定されるので、実装時には定数として扱うことができる。したがって、図4で頻繁に出てくる $D_y$ 、 $D_x$ との乗算は実質すべて不要となる。

本手法と従来のMSNのルーティングアルゴリズムの違いは、

- デッドロックフリーが保証される
- メッセージのFIFO性が保証される
- 効率良くブロードキャストできる
- 送信ノードでのルーティング情報生成はMSNのルーティングアルゴリズムと同程度に複雑だが、中継ノードでの判定は、MSN/Pが非常に単純でハードウェアでの実装も容易。

図6に代理送受信が必要となる場合の通信パターンの例を示す。MSN/Pでは、送信ノードの通信可能なリンクの向きによって、4通りのマップができる。これは、黒く塗りつぶされたノードを送信ノードとし、グレーで塗りつぶされたノードは代理受信が必要となるノード、代理受信の補正後に矩形で囲まれた領域内にあるノードは、代理送信が必要となるノードである。また、送信ノードは代理送信が必要なノードに含まれる。これは、ブロードキャスト時にメッセージの

```

/* Dy, Dx の初期化. 一度だけ行えばよい. */
if (xsrc is even) then
    Dy := -1;
else
    Dy := 1;
end if;
if (ysrc is even) then
    Dx := -1;
else
    Dx := 1;
end if;
/* 一時的な相対アドレスを求める. */
dy := (ydst - ysrc) mod m;
dx := (xdst - xsrc) mod n;
/* 代理送信が必要かどうかの判定をする. */
if (dy = 0 and Dx × dx ≤ -1) then
    fPR := false;
else if (dy = -Dy and -1 < Dx × dx) then
    fPR := false;
else if (Dy × dy < 0 and dx is odd) then
    fPR := true;
else if (0 ≤ Dy × dy and dx is even) then
    fPR := true;
else
    fPR := false;
end if;

if ((dy = 0 and Dx × dx ≤ -1) or
    (dy = -Dy and -1 < Dx × dx)) then
    fPR := false;
else if ((Dy × dy < 0 and dx is odd) or
    (0 ≤ Dy × dy and dx is even)) then
    fPR := true;
else
    fPR := false;
end if;
/* 代理送信の補正をする. */
if (fPR = true and dy is even) then
    x' := dx + Dx;
else if (fPR = true and dy is odd) then
    x' := dx - Dx;
else
    x' := dx;
end if;
/* 代理受信が必要かどうかの判定をする. */
if (0 ≤ Dx × c') then
    fPS := true;
else
    fPS := false;
end if;
/* 代理受信の補正をする. */
if (fPS = true) then
    y' := dy + Dy;
else
    y' := dy;
end if;

```

図4 MSN/P のルーティング情報生成のアルゴリズム  
 Fig. 4 Algorithm of the routing information generation for MSN/P.

```

if (x' ≠ 0) then
    x' := x' - 1;
    X リンクへ送信;
else if (y' ≠ 0) then
    y' := y' - 1;
    Y リンクへ送信;
else if (fPR is true) then
    X リンクへ代理受信セルとして送信;
end if;

```

図5 中継ノードでのルーティングアルゴリズム  
 Fig. 5 Routing algorithm in relay node.

FIFO 性を保証するために必要である。

図7は実際のルーティングの例である。ノード S から、ノード D0, D1, D2, D3 に送信した場合、それぞれ、

- D0 代理送信ノードも代理受信ノードも不要な場合。X リンク優先でルーティングされる。
- D1 代理受信ノード (D1') が必要な場合。S から D1' は通常のルーティング、D1' から D1 は代理チャネルを用いてルーティングを行う。
- D2 代理送信ノード (S') が必要な場合。S から S' は代理チャネルを用いてルーティングを行い、S'

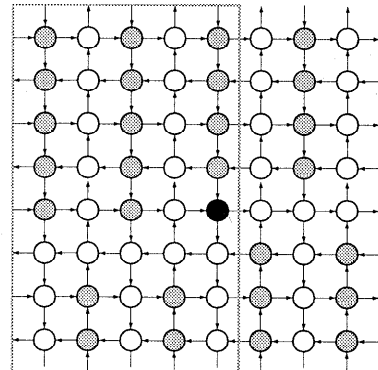


図6 代理送受信フラグのマップ  
 Fig. 6 Map of proxy send/receive flag.

から D2 へは通常のルーティングを行う。

- D3 代理送信ノード (S'), 代理受信ノード (D3') が必要な場合。

### 3.3 デッドロック回避

デッドロックの回避については、ネットワークの物理的な循環構造を仮想チャネルを用いて論理的に断ち切る方法を採用する<sup>4)</sup>。この方法は、仮想チャネルの構成に必要なバッファの量を最小におさえること

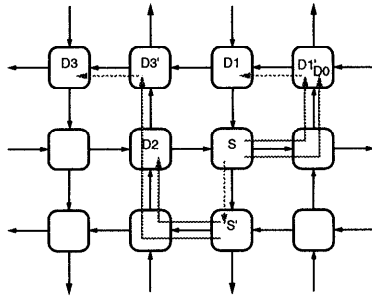


図7 ルーティングの例  
Fig. 7 Examples of routing.

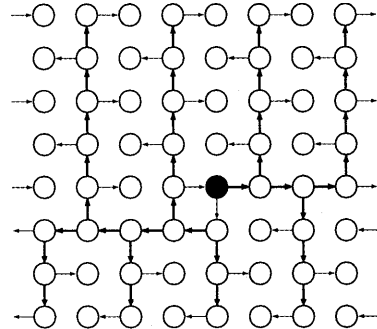


図9 ブロードキャストの一例  
Fig. 9 Example of broadcast.

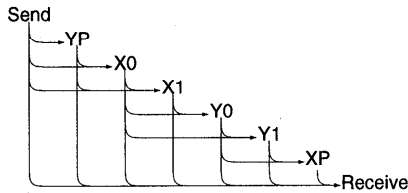


図8 MSN/Pのチャネル依存グラフ  
Fig. 8 Channel dependency graph of MSN/P.

ができる。

具体的には、次に示す優先順位と条件でチャネル変更を行う<sup>5),6)</sup> (図7)。

- (1) 代理受信ノードから受信ノードへの送信は、代理チャネル (X)
- (2) ラップアラウンドを通過する場合にはチャネル 1
- (3) 方向転換をする場合は、チャネル 0
- (4) 代理送信ノードへの送信は、代理チャネル (Y)

したがって、Y リンク、X リンクに対して、それぞれ 3 チャネルの仮想チャネルを設けることで論理的な循環構造がなくなるので、デッドロックしない。図8 は MSN/P のチャネル依存グラフである。この図において、YP は代理チャネル (Y)、X0/1 はそれぞれ X チャネル 0/1、Y0/1 はそれぞれ Y チャネル 0/1、XP は代理チャネル (X) を示す。

### 3.4 ブロードキャスト

MSN/P ではブロードキャストセルに対して、自プロセッサ、X リンク、Y リンクの 3 方向に対して通常のルーティングと同じ規則でセルを送信すればよい。また、 $n \times m$  の MSN/P では、送信元を (0, 0) と見なし相対アドレスが  $(-n/2, -m/2) - (n/2 - 1, m/2 - 1)$  の矩形からはみ出さないようにブロードキャストの終了条件を設定すればよい。図9 は、黒く塗りつぶされたノードを送信ノードとした場合に、ブロードキャストで選択される経路が矢印で示されている。細い矢印は代理送受信での通信を示している。図9 から明ら

かなように  $(-n/2, -m/2) - (n/2 - 1, m/2 - 1)$  の矩形からはみ出すような通信は、すべて代理受信ノードからの送信となるため、相対アドレスは矩形から出ることではない。

また、Point-To-Point の通信の場合と同じ経路を通るので、

- デッドロックフリー
- メッセージの FIFO 性

がつねに保証されていることは明らかである。

## 4. ソフトウェアシミュレーションによる評価

### 4.1 シミュレーションの条件

単方向トラスネットワークと双方向トラスネットワークの通信性能を比較するために、通信シミュレーターを作成し実験を行った。

シミュレーションの対象は、

**Bi-dir** 双方向 2 次元トラスネットワーク. e-cube ルーティングを用いる。半二重で通信を行う。

**MSN** 2 章で述べた Rule 1 を用い、セル衝突時は迂回経路を選択する動的ルーティングを行う。このため他の方法と異なり、メッセージの FIFO 性はない。

**MSN/P** ルーティングや、デッドロック回避については、先に述べた方法を用いる。

**Simple** 通信可能な方向がすべて同じ単方向トラスネットワーク。次元順ルーティングを用いる。

シミュレーションの条件として、

- (1) 各ノードは、異なる方向の送受信を同時に行える。
- (2) 簡単のためセル単位で Store&Forward を用いる。セルとは、シミュレーションに用いるルーター上でルーティングに必要な情報を持ったヘッダと、固定長のデータを持っている。
- (3) 1 セルの通信に必要な時間を 1 セル時間とする。「1 セルの通信に必要な時間」とは、受信パッ

ファにあるセルを次ノードの受信バッファに格納するまでの時間である。したがって、セル中のデータ長はシミュレーションに影響しない。

- (4) ノード構成は、図 10 に示したものを用いた。  
**MSN** X, Y ともに、深さ 4 の FIFO  
**MSN/P** X, Y の仮想チャンネル 0 を深さ 2 の FIFO

**Simple** X, Y のすべての仮想チャンネルを深さ 2 の FIFO

- (5) 通信の形態は、次の 4 種類とした。  
**Random** 不規則で局所性を持たない通信  
**Nearest Neighbor** 隣接ノードの中から、ランダムに選択される 1 つに対して行われる通信

**Reduce** すべてのセルが (0, 0) に集中する通信

**Hot Spot** 発生するセルのうち 50% が (2, j) を受信ノードとする通信

- (6) 時間 0 で複数のセルを同時に発生させ、すべてのセルが受信ノードに到達するまでの時間を“通信時間”とする。ただし、ルーティングの途中でセルを生成しない。
- (7) ネットワーク上につねに同数のセルが存在するようにし、対象とするすべてのネットワークが定常状態になった時間 600 から、時間 1000 の 400 単位時間に到達した平均のセル数を“スループット”とした。

ノード構成を図 10 のように設定した理由は、ハードウェアコストをできるだけ近付けるため、すべてのネットワークに関してノードあたりのバッファ数が 9 になるように、FIFO を追加したためである。また、FIFO を追加する場所については、FIFO がない場合のネットワークをシミュレーションし、通信路の衝突頻度が高かった場所に設定した。シミュレーションは、8 × 8 ノードの場合について行い、その結果を図 13 から図 18 に示す。通信時間は、横軸に時間 0 で発生させるセルの個数、縦軸に通信に要した時間をシミュレーション単位時間ととった。横軸の単位は個、縦軸の単位はセル時間。また、スループットは、横軸にネットワーク上に存在するセルの個数、縦軸にスループットをとった。横軸の単位は個、縦軸の単位は 1 セル時間あたりの平均受信セル数である。

4.2 シミュレーションの結果

4.2.1 “Random” の場合

平均通信時間 (図 11) では、Simple が悪く、Bi-dir が良い。MSN, MSN/P はその中間となる。これは、

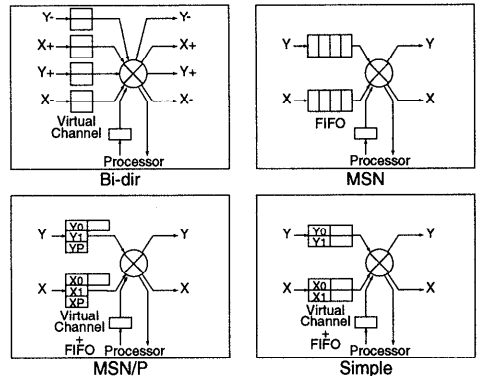


図 10 ノード構成  
 Fig. 10 Node configuration.

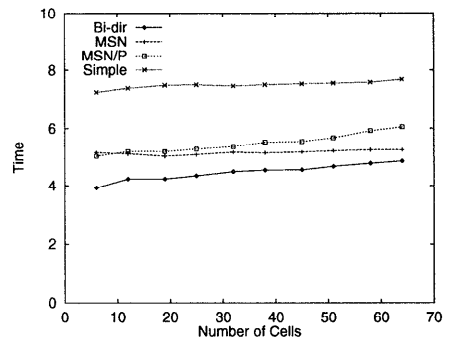


図 11 “Random” の平均通信時間  
 Fig. 11 Average arrival time of “Random”.

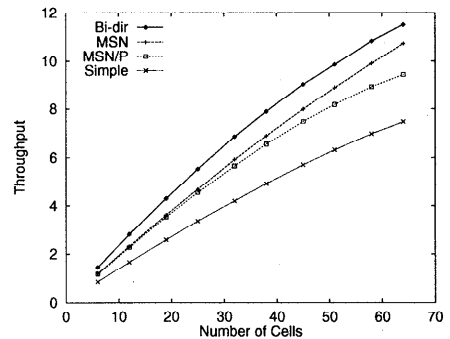


図 12 “Random” のスループット  
 Fig. 12 Throughput of “Random”.

MSN が図 1 のように、通信可能な方向を互い違いに配置したために、Bi-dir と比較して Simple ほどネットワークの直径が大きくなっていないことを示している。また、MSN では動的ルーティングを行っているが、動的ルーティングを行わない MSN/P と大差のない結果が得られた。また、MSN 以外はセル密度があがるにつれて通信時間も伸びているが、MSN はほぼ一定である。

スループット (図 12) も、平均通信時間と同じ傾

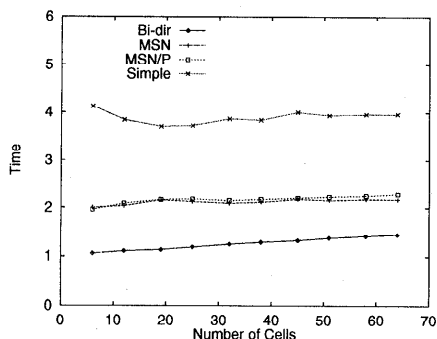


図 13 “Nearest Neighbor” の平均通信時間  
Fig. 13 Average arrival time of “Nearest Neighbor”.

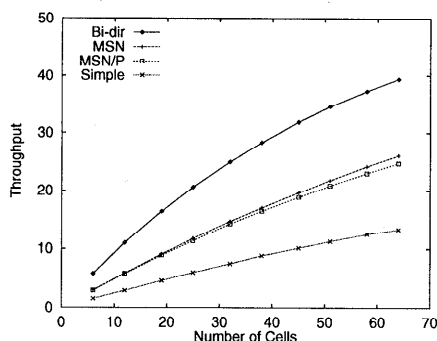


図 14 “Nearest Neighbor” のスループット  
Fig. 14 Throughput of “Nearest Neighbor”.

向を示している。Bi-dirが有利だが、MSN、MSN/Pともに2割程度の悪化にとどまる。MSN、MSN/Pはほぼ同等のスループットが得られる。

また、ここでは結果を示していないが、 $32 \times 32$  ノードでのシミュレーションも行った。その結果から、ノード数がさらに多くなった場合、平均通信時間、スループットともにSimpleは極端に性能が悪化するのに対して、MSN、MSN/PはよりBi-dirに近い性能を示すようになることが分かった。

#### 4.2.2 “Nearest Neighbor” の場合

平均通信時間(図13)では、Simpleが悪く、Bi-dirが良い。MSN、MSN/Pはその中間でほぼ同等の結果である。また、Bi-dirでは平均通信時間が若干増加傾向なのに対して、MSN、MSN/Pではセル密度にかかわらずほぼ一定になっている。これは、Bi-dirでは、隣接ノードから同時に送信されたセルとの衝突により通信待ちが起こるが、MSN、MSN/Pでは起こらないためである。

スループット(図14)でも、同様の傾向を示している。Simpleでは、Bi-dirの3割程度まで性能が落ち込むが、MSN、MSN/Pでは、Bi-dirの6割程度の性能を保っている。

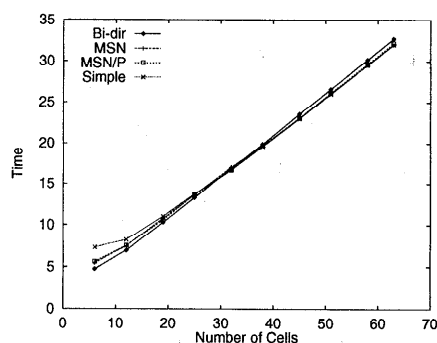


図 15 “Reduce” の平均通信時間  
Fig. 15 Average arrival time of “Reduce”.

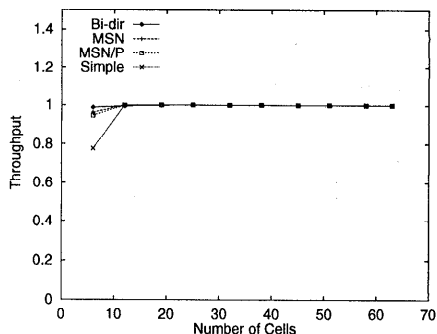


図 16 “Reduce” のスループット  
Fig. 16 Throughput of “Reduce”.

ノード数が多くなった場合、平均通信時間、スループットともにSimpleは極端に性能が悪化するのに対して、MSN、MSN/Pはノード数による影響は少なくBi-dirに近い性能を示すようになる。これは、ノード数によらずMSN、MSN/Pは隣接ノードのみの平均距離は2、Bi-dirは1であるのに対して、Simpleはノード数の増加によって平均距離が長くなるためである。

#### 4.2.3 “Reduce” の場合

平均通信時間(図15)、スループット(図16)ともにはほぼ同じ結果になった。これは、通信バッファプロセッサ間も1単位時間で1セルしか受けとらないためである。

#### 4.2.4 “Hot Spot” の場合

平均通信時間(図17)では、他の通信パターン同様Simpleが悪く、Bi-dirが良い。MSN、MSN/Pはその中間となる。セル密度が高くなると、すべてのネットワークについて性能が悪くなっていくが、MSNは比較的悪化の度合いが少ない。MSN/PはBi-dirとほぼ並行して悪化していく。

スループット(図18)でも、他の通信パターン同様Simpleが悪く、Bi-dirが良い。MSN、MSN/Pはその中間となる。MSN、MSN/Pともにセルの密度が

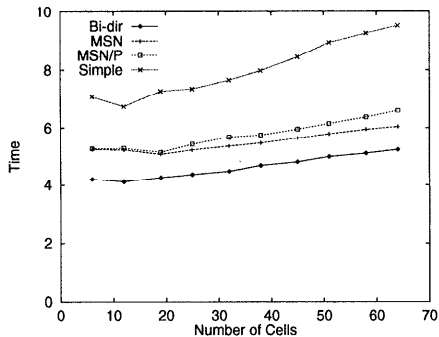


図 17 “Hot Spot” の平均通信時間  
Fig. 17 Average arrival time of “Hot Spot”.

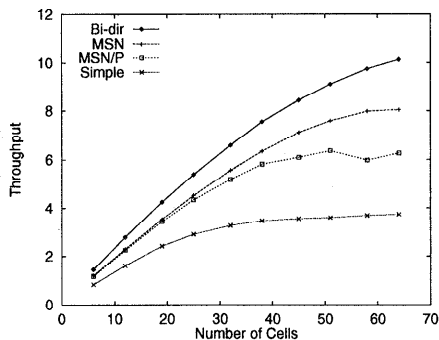


図 18 “Hot Spot” のスループット  
Fig. 18 Throughput of “Hot Spot”.

上がると性能低下が目立ち、Bi-dirと比較して、それぞれ8割、6割程度まで落ちている。Bi-dirより経路選択の制限が強いため、セルの衝突が頻繁に起こっているためである。また、MSNよりもさらに経路選択の制限が強いMSN/Pではいっそう目立っている。

ノード数が多くなった場合、MSNはデッドロックを起こし正しい計測ができなくなってしまった。Bi-dir、MSN/P、Simpleではデッドロックはもちろん起きないが、Simpleは極端な性能低下が見られる。また、MSN/Pもスループットの低下が激しくなる。

4.3 ハードウェア量の比較

ルーターのハードウェア量・コストに影響する要素として、スイッチ、チャネル数、バッファ量、ドライバ/レシーバ数などが考えられる。

表1に、シミュレーションで用いたノード構成のBi-dir、MSN/P、MSNについて、スイッチの入出力数、ノードあたりの仮想チャネル数、ノードあたりのバッファ数、ノードあたりのドライバ/レシーバ数を示す。

MSN/Pと双方向のトラスとを比較すると、リンクあたりのチャネル数は多くなっているが、リンク数自体が少ないためノードあたりの仮想チャネルの総数

表1 ハードウェア量の比較  
Table 1 Quantity of hardware components.

	スイッチ In×Out	仮想チャ ネル数	バッファ数	ドライバ/ レシーバ
Bi-dir	5×5	8	9	4/4
MSN/P	3×3	6	9	2/2
MSN	3×3	3	9	2/2

(バッファ数)は、MSN/Pの方が少ない。スイッチの実装では、入出力の組合せが増えると多くのチップ面積が必要となる。また、ドライバ/レシーバは1/2ですむため、それらが高価になりがちな高速通信デバイスではより有利であると考えられる。

5. まとめ

MSN/Pの概要、ルーティングアルゴリズム、デッドロック回避法について述べた。MSN/Pは、MSNのトポロジを用い、代理送信・受信ノードの概念を導入した新たなルーティングアルゴリズムを用いることでMSNの欠点を解消し、

- デッドロックフリーの保証
- メッセージのFIFO性の保証
- ブロードキャストへの対応
- 中継ノードでのルーティングアルゴリズムの簡素化

という特徴を持たせることができる。

また、ソフトウェアシミュレーションによる通信性能の比較によって、MSN/PはほぼMSNと同様の性能であることを示し、特にノード数が多い場合のランダム通信においては、双方向トラスとほぼ同様の性能であることを示した。

シミュレーションに要した時間は、Bi-dirやMSNよりもMSN/Pが短い。今回用いたソフトでは、ルーティングの判定にかかる時間がシミュレーションの大半を占めており、32×32ノードのランダム通信のスループットのシミュレーション時間は、MSNの249秒に対して、MSN/Pでは205秒であった。客観的とはいえないが、このことからMSN/Pのルーティングは単純な割に効率的であるといえるだろう。

シミュレーション結果および、ハードウェア量の比較から、MSN/Pは双方向トラスより低性能であるが、総合的なコストパフォーマンスの点では同等と考えられる。また、本稿のシミュレーションでは考慮しなかったが、MSN/Pはルーティングアルゴリズムが単純であることから、より高い周波数でルーターを動作させることができると考えられる。これは、ネットワーク全体のスループットの向上、レイテンシの低減



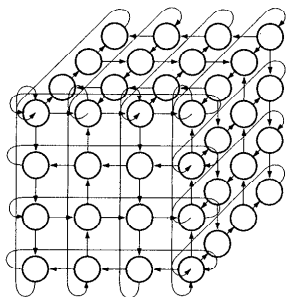


図 19 3次元に拡張された MSN/P  
Fig. 19 MSN/P-3D.

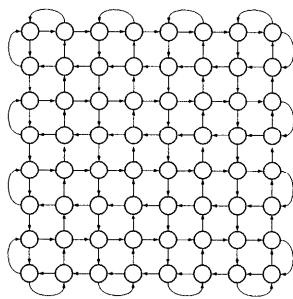


図 20 単方向メッシュ  
Fig. 20 MSN/P-Mesh.

につながる。これらを考慮すれば、MSN/Pはトラス・メッシュ状のインターコネクタを構成する場合の新たな選択肢の1つとなりうる。

MSN/Pで用いた代理送受信を用いることで、MSNを3次元に拡張することができる(図19)。このネットワークでは、3入力3出力のリンクが必要になり、X/Y/Zのリンクに対してそれぞれ3チャンネルの仮想チャンネルを持たせ、各リンクに対して代理ノードを設定することで、デッドロックフリーなどMSN/Pの特徴を保ったまま3次元に拡張することができる。

従来のMSNのルーティングでは、ラップアラウンドの配線が不可欠であるが、MSN/Pでは代理受信ノードを2つにすることで、ラップアラウンドを取り除いた単方向メッシュ(図20)のルーティングも行えるようになる。この単方向メッシュでは、代理送受信が必要かどうかの判定も複雑になるが、すべての配線が平面上で行えるという利点がある。One-Chip Multi Processorのように3次元的な配線を行にくいものや、計算機実習室などでWS・PCクラスタを構成す

る場合のように計算機が空間的に広く配置されラップアラウンドの配線が困難な場合などへの応用が考えられる。

謝辞 この研究の一部の費用は、並列・分散処理研究推進機構により援助されたものです。

### 参考文献

- 1) Maxemchuk, N.F.: The Manhattan street network, *Proc. GLOBECOM '85*, pp.255-261 (1986).
- 2) 林 匡哉, 堀田真貴, 大津金光, 吉永 努, 馬場敬信: HDL 設計に基づく並列計算機ルータの評価, 情報処理学会研究報告, 98-ARC-130, Vol.98, No.70, pp.79-84 (1998).
- 3) Maxemchuk, N.F.: Routing in the Manhattan Street Network, *IEEE Trans. Communications*, Vol.COM-35, No.5, pp.503-512 (1987).
- 4) 天野英晴: 並列コンピュータ, 昭晃堂 (1996).
- 5) 江草俊文, 小畑正貴: 単方向2次元トラスネットワークの構成と, シミュレーションによる評価, 情報処理学会研究報告, 96-ARC-120, Vol.96, No.106, pp.13-18 (1996).
- 6) 可児純一, 江草俊文, 小畑正貴: PC クラスタのための単方向2次元トラス網用ルータ, 電気・情報関連学会中国支部第48回連合大会講演論文集, p.449 (1997).

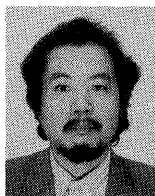
(平成10年8月31日受付)

(平成11年1月8日採録)



江草 俊文 (学生会員)

昭和46年生。平成6年岡山理科大学工学部電子工学科卒業。平成8年同大学院修士課程修了。現在、同大学院博士課程在学中。計算機アーキテクチャ、並列処理等に興味を持つ。



小畑 正貴 (正会員)

昭和32年生。昭和55年神戸大学工学部電子工学科卒業。昭和60年同大学院自然科学研究科システム科学専攻博士課程修了。学術博士。昭和59年岡山理科大学理学部助手。平成8年岡山理科大学工学部教授。現在に至る。計算機アーキテクチャ、並列処理の研究に従事。電子情報通信学会, ACM, IEEE-CS 各会員。