

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

江草俊文[†] 小畑正貴[†]

本稿では、性能を維持しながらコストを下げるために、単方向のネットワークの通信可能な方向を交互に配置した単方向2次元トラスネットワークを提案し、そのルーティングアルゴリズム、デッドロック回避の手法、シミュレーションによる性能評価について述べる。また、結果としてノード数が多く、パケット密度が高い場合のランダム通信性能は、双方向のものに匹敵する性能であることがわかった。

Design of One-way 2D Torus Network and it's evaluation by software simulation

TOSHIFUMI EGUSA[†] and MASAKI KOHATA[†]

This paper proposes the One-way 2D torus network. Each link of the network is one-directional and link directions are same on each row and column and each direction of rows and columns is reverse to the nearest neighbor rows and columns. Routing algorithm, escape method of deadlock and software simulation results are also described. As the results, it is showed that same performance of random communication of One-way 2D torus network as bidirectional torus network, when the number of nodes and packets are large.

1. はじめに

近年、既存のワークステーション (WS) やパーソナルコンピュータ (PC) にプロセッサ間のネットワークを付加し、並列計算機を構成することは珍しくなくなってきた。

商用機では IBM の SP/2²⁾、Thinking Machines の GlobalWorks³⁾ など、アカデミックなものとしては、CUB の NOW Project⁴⁾ などがある。

これは、

- WS や PC などの量産機が並列計算機の要素プロセッサとして十分な性能を持つようになったこと
- 量産品であるため低価格で、品質が安定していること
- 基本的な回路が含まれているため開発期間の短縮なども期待できること

などから、WS や PC を要素プロセッサとして用いることで高いコストパフォーマンスが得られるためである。このような構成の並列計算機の場合には、要素プ

ロセッサの部分に関しては、量産効果により高いコスト・パフォーマンスが実現されているため、ネットワークのコストと性能が、システム全体のそれを決定する大きな要素となる。

そこで、さらなるコストパフォーマンスの向上をめざした場合、ネットワーク部分のコストパフォーマンスが向上することは、重要である。

我々は、性能を維持しながらコストを下げるために、徹底的にネットワークのハードウェアを簡略化するというアプローチをとった。従来、並列計算機は双方向の通信路で接続される場合が多かった。それを単方向にすることでハードウェア量の削減が期待できる。また、通信可能な方向を適当に配置することで通信性能を維持することができる。その具体的な例として、2次元トラスネットワークをベースとした単方向の2次元トラスネットワークが考えられる。

本稿では、単方向2次元トラスネットワークの構成について述べ、そのルーティングアルゴリズム、デッドロック回避の手法、シミュレーションによる性能評価を行なう。

[†] 岡山理科大学工学部

Faculty of Engineering, Okayama University of Science

2. 単方向トラスネットワークとは

従来、並列計算機ではプロセッサ間の通信に双方向のネットワークを用いていた。双方向の通信路を用いた方が、単方向のものよりさまざまな面で柔軟な対応ができそうである。しかし、構成さえ十分に検討されていれば単方向のネットワークの場合でも、ルーティング自体は行なうことができる。

しかし、単方向化することで経路長が伸びてしまうことが予想される。そこで、単方向のネットワークをうまく組み合わせることで経路長が長くなることを避けられ、パケットの密度が高い場合について、双方向のネットワークを用いた場合と比較して大差のない性能を得られる可能性がある。

また、双方向のネットワークでは図1に示すように、1組の通信路を送受信で共有し調停を行なうことで双方向の通信を実現する方法と、2組の通信路を設けることで双方向の通信を実現する方法がある。

前者の場合では、通信路は1組で済むが、その切替えのための回路と、調停のための回路が必要になる。また、調停のためのバスフェーズが必要となり、それらを機能の一部として実装しなければならない。

後者では、調停回路は不用であるが、通信路を2組保持しなければならない。

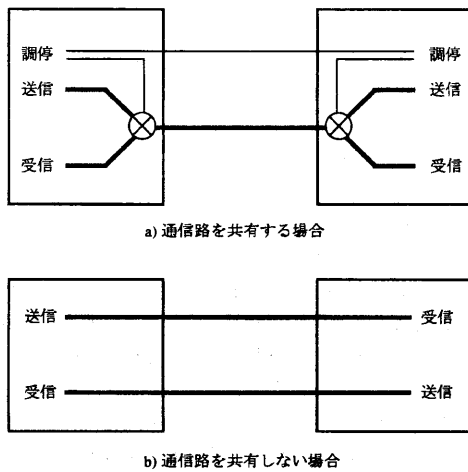


図1 双方向通信の実装

これに対して、単方向の通信路だけでネットワークが構成できれば調停回路は不用になり、通信路も1組で済む。したがって、ネットワークを構成するための配線量や、ルーターの規模の縮小が期待できる。

単方向の通信しか行なえないメディアをもちいてネットワークを構成する場合を考えてみる。例えば、光ファイバーを用いた場合、双方向バスを構成するためには、2本1組で用いるほかないが、単方向ならば1組で済むため、高速なネットワークを低コストで実現できる可能性がある。

そこで、

- (1) 単方向化によってハードウェアを簡略化する。
 - (2) ネットワークの直径を大きくしない。
 - (3) ルーティングや、デッドロックの回避が容易。
- の3点を最重要の条件として、次に示すような単方向2次元トラスネットワーク(図2)を考えた。

- ネットワークの外観は2次元トラスと同様である。
- プロセッサ間は単方向のバスで接続。
- 列・行の番号が偶数が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。

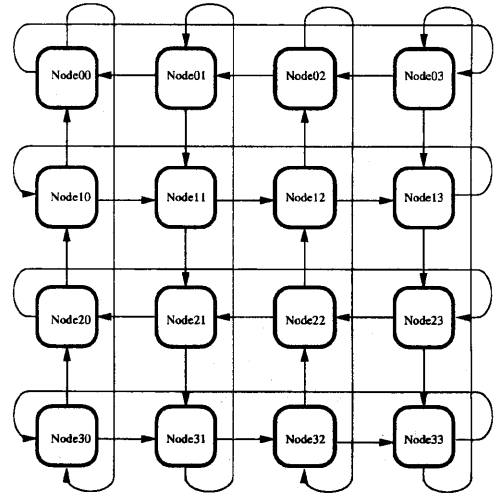


図2 単方向2次元トラスネットワーク

このネットワークは、単方向のネットワークを採用することでハードウェア量を減らし、通信可能な方向を互い違いに配置することで、直径が極力大きくならないように配慮した。

$N \times N$ ノードで考えた場合、双方向ネットワークでは、ネットワークの直径は N である。次に、単方向で南北、東西の全てのネットワークの通信可能な方向を同じ向きにすると、ネットワークの直径は $2N$ となる。それに対して、互い違いに配置することで $N+2$ となり、双方向の場合とほぼ同じになる。また、平均

距離を考えた場合でも、双方向の場合 $N/2$ に対して、単方向は $N/2 + 1$ となる。

3. ルーティング

ルーティングアルゴリズムを決定するに当たって、次の点について考慮した。

- (1) デッドロックの回避も考慮して全体のハードウェア量が少ない。
- (2) ルーティングのアルゴリズムがハードウェアで実現しやすい。
- (3) 最短距離を通るようにする。

この条件を満たすために、送信ノードを起点に、受信ノードを通過して送信ノードに至る最小の矩形経路をネットワークの通信可能な方向に逆らわないように決定し、その経路に沿ってルーティングを行えばよい。

そのような経路は、次に示す単純なルールで決定できる。

- (1) パケットが目的ノードに近付ける方に進む。南北方向、東西方向のどちらかに送信しても近付ける場合は方向転換しない方向に進む。
- (2) 近付ける方向がなければ目的ノードと現在のノードのノード番号の差が南北方向、東西方向ともに奇数もしくは、偶数なら方向転換をしない方向に進む。
- (3) 南北方向、東西方向のノード番号の差が、奇数になるように移動する。

また、パケットが発生したノードにおいては、「方向転換しない方向」が存在しないが、その場合はどちらか任意の方向を選択すれば良い。

このようにルーティングすることによって、ネットワーク上でのパケットの通信方向の変化が最小になり、次に述べるデッドロックの回避法と合わせて、仮想チャネル用のバッファ量を削減できる。また、常に経路長が最短になるような経路が選択される。

図3は実際のルーティングの例である。単方向にしたことによって、もっとも経路が長くなってしまうケースの1つである。この例では、距離が4伸びているが、平均的には1伸びるだけである。

また、上記のルールを実現するためには、常に宛先ノードとの相対座標を知っておく必要があるが、それ以外の判定は、相対座標の最下位ビットのみで判定できるため、ハードウェアでの実現は困難ではない。

4. デッドロックの回避

デッドロックの回避については、ネットワークの物

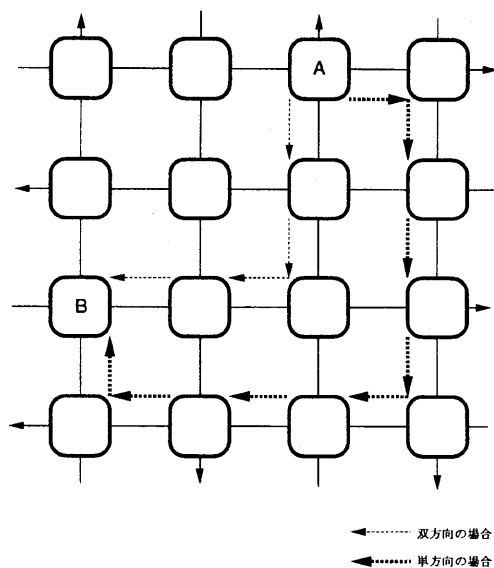


図3 ルーティングの例

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¹⁾。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押えることができる。

具体的には、次に示すパターンの方向転換(図4の矢印)が起こった場合について仮想チャネルを変更する。

- ラウンドトリップループを用いる場合。
- 東北の角での方向転換をする場合。

の2つの場合について仮想チャネルを変更すれば論理的な循環構造がなくなるので、デッドロックしない。

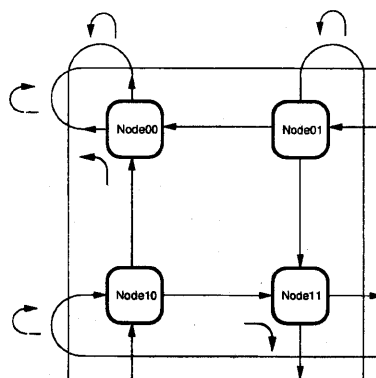


図4 チャネル変更によるデッドロックの回避

上記のルーティング法にしたがった場合に必要になる仮想チャネル数は4である。全ての経路を考えると、

ラウンドトリップループは、最大3回通過する可能性があり、東北の角は最大1回通過する可能性がある。この場合、東北の角での方向変換は必ず同時にラップアラウンドループの通過を伴う。このように、チャンネル変更の条件に同時に複数当てはまる場合でも、チャンネルの変更は1回だけ行なえば良い。

したがって、仮想チャンネルの変更は最大3回で済む。よって、4つの仮想チャンネルを準備すれば論理的な循環構造を断ち切ることが可能であり、デッドロックの回避ができる。

5. シミュレーションによる評価

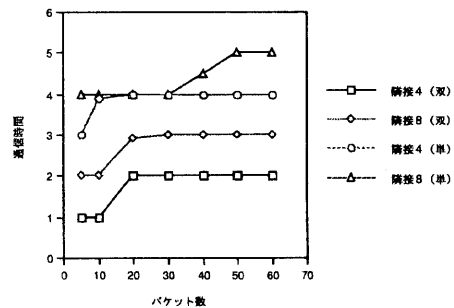
単方向トラスネットワークと双方向トラスネットワークの通信性能を比較するために、簡単な通信シミュレーターを作成し、実験を行なった。

シミュレーションの条件として、

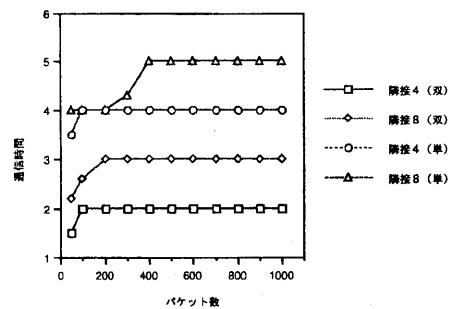
- (1) それぞれのノードは、2方向から同時に受信・送信でき、1つの通信を行なうのに必要な時間を1単位時間とする。そのためパケットの送信法として、ストア&フォワード的な扱いをしている。
- (2) 通信の形態は、
 - (a) 不規則で局所性を持たない場合
 - (b) 隣接ノード(4つ、8つ)の中から、ランダムに選択される1つに対して通信が行なわれる場合の2種類について。
- (3) 単方向トラスネットワークでは、
 - (a) ルーティングや、デッドロック回避については、先に述べた方法を用いる。
 - (b) 通信経路が使用中であるパケットは、ブロックされる。
- (4) 双方向トラスネットワークでは、
 - (a) ルーティング、デッドロック回避については、単方向トラスネットワークのルーティング、デッドロック回避法と同様の手法を、双方向通信の場合に拡張した方法を用いる。
 - (b) 通信経路が使用中であるパケットは、他に最適な経路(最短の経路)があれば、その経路を迂回路として用いる。
- (5) 通信時間は、時間0でパケットが発生し、それが目的ノードに到達するまでの時間とし、途中でパケットを生成しない。
- (6) スループットは、ネットワーク上に常に同数のパケットが存在するようにし、時間100から、

時間200の100単位時間に到達した平均のパケット数とした。

シミュレーションは、 8×8 、 16×16 、 20×20 、 32×32 ノードの単方向2次元トラス及び双方向2次元トラスで行なった。また、ランダム通信の通信時間に関しては、その平均及び、平均の待ち時間についても求めた。



8 x 8 ノードの場合



32 x 32 ノードの場合

図5 隣接通信の通信時間

スループットと、通信時間については、顕著な傾向が現れた 8×8 ノードの場合と 32×32 ノードの場合について示し、平均の待ち時間と通信時間の関係については、異なるノード数の場合でもほぼ同様の結果となったので、 32×32 ノードの場合のみを示す。

隣接ノードに対する通信の通信時間(図5)を見ると、双方向と単方向では常に2単位時間ずつの遅れが生じることが特徴的である。これは、隣接4ノードの場合でも隣接8ノードの場合でも、経路長の差の最大が2であるため、双方向と単方向では常に2単位時間ずつの遅れが生じるためである。また、パケット発生の密度が高くなると必ず1回は通信待ち時間が起こるので、経路長+1の時間がかかっている。ノード数に

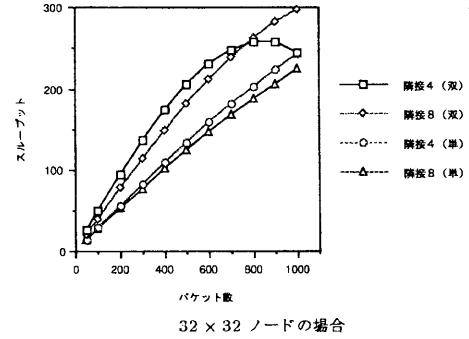
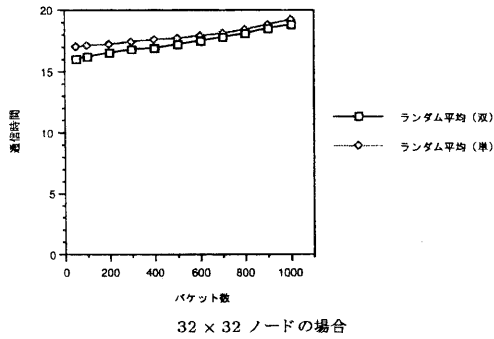
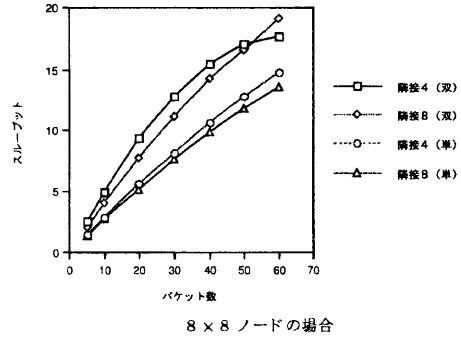
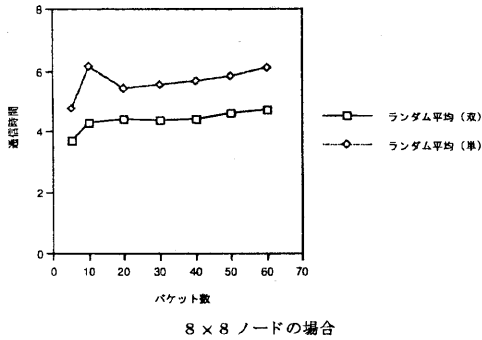


図 6 ランダム通信の通信時間

図 8 隣接通信のスループット

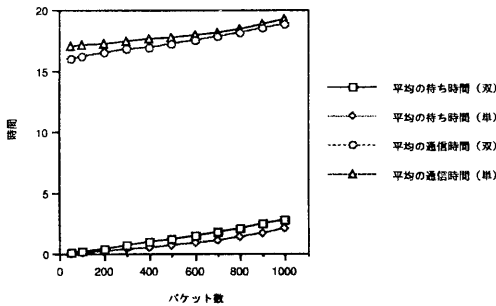


図 7 32 × 32 ノードの場合の平均の待ち時間と通信時間

関わらず、同じ結果になっている。

ランダムノードに対する通信の通信時間(図6)から、ノード数が多い場合では、双方向の場合とほぼ同じ結果となっている。これは、平均経路長はノード数に合わせて増えていくにもかかわらず、平均経路長の差は、ノード数に依存せず1であるため、割合として目立たなくなっているためである。

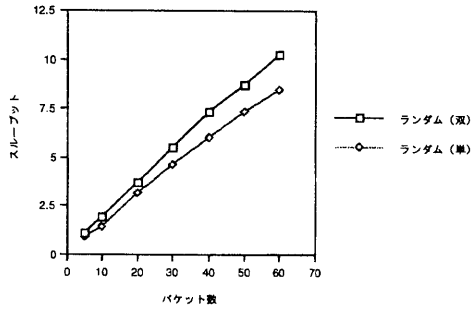
図7は、図6の32×32ノードの場合の内訳を示している。通信待ち時間だけを見ると、当初の予想通

り、単方向の場合の方が少ない。しかし、平均距離が1大きいと、全体で見ると若干双方向の方が良い結果が得られている。

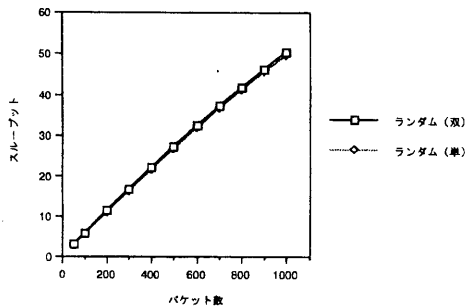
スループット(図8、図9)については、常に単方向のスループットの方が低い。また、隣接4ノードの双方向の場合、総ノード数の7~8割を越えるパケット数がネットワーク上に存在する時、スループットの低下が起こる。これは、パケットの密度が高い場合は、迂回路も通れなくなるため、単方向の場合と同じような状態になるためと考えられる。

また、ランダム通信をした場合、隣接通信の場合よりもスループットの低下は少ない。これは、ネットワーク上のパケットの密度が高い場合、隣接した二つのノード間でお互いにパケットを送信し合おうとするケースが増える。その場合、どちらか一方が必ず待つことになる。それに対して、単方向のネットワークの場合には、そのような状況は起こり得ないことから、パケットの密度が高い場合では、単方向ネットワークの方が通信待ち時間が少なくなる可能性があるからである。

これらの結果から、隣接通信が頻繁に起こると考え



8 × 8 ノードの場合



32 × 32 ノードの場合

図 9 ランダム通信のスループット

られる数値シミュレーションには向かないが、ランダム通信が起こるアプリケーションに対しては、双方向の2次元トーラスと同等の通信性能が期待できると考えられる。

6. まとめ

並列計算機のネットワーク部のコストを削減するために、単方向のネットワークを用いた単方向2次元トーラスネットワークを提案し、そのルーティングアルゴリズム、およびデッドロック回避の手法、シミュレーションによる通信性能について述べた。

単方向2次元トーラスネットワークでは、近接ノードに対する通信性能は双方向のものと比較して劣っているものの、ランダム通信性能に関しては近接通信の場合と比較して、双方向2次元トーラスネットワークに近い通信性能が期待できる。特に、パケットの密度が高くなる場合は、ほぼ同等となることが確かめられた。

今回行ったシミュレーションは、現実の問題に応用した場合のものではなく純粋なネットワーク性能のみのシミュレーションである。したがって、それをそ

のまま応用問題に当てはめて考えることはできない。今後は、応用問題を考慮した通信パターンでの通信性能の評価や、どのような応用範囲で有効であるか、また、実装のための詳細などを検討していく予定である。

参考文献

- 1) 天野 英晴: 並列コンピュータ, 昭晃堂, 1996.
- 2) <http://www.yamato.ibm.co.jp/rs6000sp/0030.html>
- 3) <http://www.think.com/Prod.web/index.html#gw>
- 4) <http://now.cs.berkeley.edu/>