

中規模コモディティクラスタ向け相互結合網 Three Quads の提案

吉村 知 晋[†] 三輪 忍[†] 嶋田 創[†]
中島 康彦^{††} 森 眞一郎[†] 富田 眞治[†]

現在我々の研究室では、人間の視覚・触覚を通じた実時間インタラクションを考慮にいれた超高速体感型シミュレーションシステム Scube の開発を行っている。Scube のノード間ネットワークは、実時間数値シミュレーションとシミュレーション結果の可視化処理の両方で共有する形態をとる。Scube を構築するにあたり、比較的小規模なギガビットイーサネット・スイッチの組合せで、高いコストパフォーマンスを実現する中規模コモディティクラスタ向け相互結合網 Three Quads を提案した。Three Quads は、ベース- m 3-キューブの拡張と考えると理解しやすい。ベース- m 3-キューブが 3 次元方向の同一座標軸上のみリンクをもつものに対し、Three Quads は 3 つの同一平面上の全ノードと結合している。多様な網の埋め込みが可能であり、ランダムな通信要求が発生した場合など、柔軟なネットワークリソースの提供が可能である。3 次元の隣接間通信と共に、数値計算で多用される行列の転置処理のための全対全通信、並列画像合成処理における 3 次元 Reduction 型の通信等に対して優れた特性をもつ相互結合網である。

Three Quads : A Versatile Interconnection Network for Medium Scale Commodity Cluster

TOMOYUKI YOSHIMURA,[†] SHINOBU MIWA,[†] HAJIME SHIMADA,[†]
YASUHIKO NAKASHIMA,^{††} SHIN-ICHIRO MORI[†] and SHINJI TOMITA[†]

In this paper, we have propose an interconnection network for Medium Scale Commodity Cluster. This network has originally designed the Visualization Subsystem for the Sensable Simulation System(Scube) which the authors have been developing. Scube is a 64-nodes PC-based cluster system in which a commodity GPU as the visualization accelerator is configured with each node. There is no dedicated special purpose network for the numerical simulation and visualization, however, the high cost-performance inter-connection network is originally designed for Scube. All the hardware components for this network is essentially the small-scale and commodity hardware designed for Giga-bit Ethernet. The network configuration and its characteristics are discussed in this paper.

1. はじめに

近年、汎用 PC の性能向上はめざましいものがあり、かつてはスーパーコンピュータを必要としたような演算も安価な PC による高速処理が可能になってきている。従来のスーパーコンピュータは、演算処理に関しては非常に高速で実行可能であるが、シミュレーション結果のインタラクティブな可視化には適していない。シミュレーション結果を、人間に理解しやすいよう画像にして提示する可視化技術は、数値シミュレーション結果の意味を理解し、また次のシミュレーションを方向づけるための一助となる役割を持つという点で、非常に重要である^{1)~4)}。

我々は人間の視覚・触覚を通じた実時間インタラクションを考慮にいれた、超高速体感型シミュレーションシステムの研究を行っており⁵⁾、そのプロトタイプとして Sensable Simulation System(以下 Scube) と呼ぶ中規模 PC クラスタ⁶⁾を開発している。

我々と同様に、数値シミュレーションと実時間可視化の同時処理を目指したシステムとして、緒方らは VGCluster¹⁾を提案している。VGCluster では可視化処理を高速化する画像合成処理専用のハードウェアを設けることで、画像合成処理のための通信が数値シミュレーションのための通信を妨げない手法を提案している。これに対して Scube の提案は、システム全体をコモディティ部品のみで構成し、複数系統設けたネットワークの冗長性を利用した通信のスケジューリングにより、シミュレーションと可視化処理の同時実行を目指すとともに、シミュレーション自体で必要となる通信の高速化も目指している。

この目的を達成するため、比較的小規模なギガビツ

[†] 京都大学

Kyoto University

^{††} 京都大学 / 科学技術振興事業団さきがけ研究 21

Kyoto University / PRESTO, JST

トイラーネット・スイッチ (以下 GbE SW) の組合せで構成する, 中規模コモディティクラス向け相互結合網 Three Quads を開発した⁶⁾. 本稿では, 相互結合網 Three Quads の説明と定性的な評価を行う.

2. Scube の概要

Scube は, 64 ノード構成の PC クラスタで, 各ノードに可視化処理のための GPU を搭載する. 計算負荷の高いシミュレーションと多次元データの可視化までを PC クラスタベースの計算サーバで行い, オペレータの操作や結果の呈示を行うユーザインタフェース端末としてのクライアントを用いたクライアントサーバモデルである. Scube のノード間ネットワークは, 実時間数値シミュレーションとシミュレーション結果の可視化処理の両方で共有する形態をとる. Scube を構成するクラスタノードおよびネットワークの仕様を表 1 に示す.

表 1 Scube の仕様

CPU	Pentium4 3.4GHz
OS	Linux 2.6.9
Memory	1.0GB
Motherboard	ASUSTeK P5AD2-E Premium
Gfx Card	GeForce6800GT PCI-Express
Gfx Memory	256MB
Network	Gigabit Ethernet×3

Scube の各ノードは, 論理的には図 1 に示すように, 3 次元に配置されているものと考え, 各ノードの 3 次元座標 (x, y, z) に対応した 3 次元のノード番号を付与する. 各座標系のネットワーク構成を図 2~4 に示す. 3 次元ノード番号において x の値が一定のノード群 $(x, *, *)$ に対して図 2 のように, 4 つの GbE SW を用いて独立した 4 組のネットワークを構成する. y, z の値が一定のノード群についても x と同様に, それぞれ図 3, 4 のようにネットワークを構成する. 基本的には, それぞれの座標系に対して 4 つのノード群が存在し, 各群ごとに 16 ポートの GbE SW を割当て, ノード数は常に 16 台である. なお, Z 座標系のネットワークでは, システム全体の管理を目的として, 独立した 4 つの Z 座標系ネットワーク間を接続する上位の GbE SW を設け, Host 計算機やファイルサーバと全ノード間での通信路を確保している. しかし, 以下の説明では, 議論の混乱を避けるため, これらはないものとして考える.

3. 相互結合網 Three Quads

Three Quads は, 数値シミュレーションと実時間可視化の同時並列実行を可能にすることを目的として設計した相互結合網であり, 3 次元空間をプロセッサ空間に直接写像する数値シミュレーションにおける 3 次元隣接間通信^{7),8)}, 3 次元画像合成処理における 3 次元 Reduction 型の通信¹⁾ を 1 つのネットワーク上

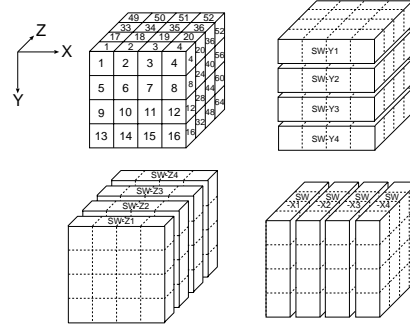


図 1 ノード配置および各スイッチへのノード割当て

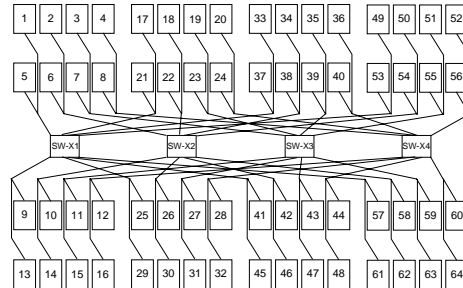


図 2 X 座標系のネットワーク構成

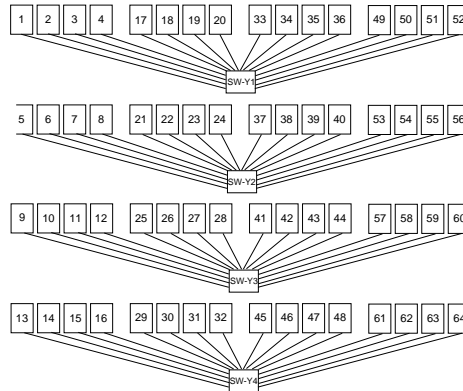


図 3 Y 座標系のネットワーク構成

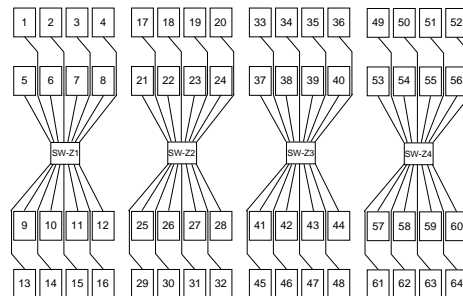


図 4 Z 座標系のネットワーク構成

で効率よく実現することを目的とした相互結合網である。またその実装に際しては中規模コモディティクラスタにターゲットを絞り、比較的小規模なギガビットスイッチのみを用いて構成するとともに、複数系統設けたネットワークの冗長性を利用した通信のスケジューリングにより、シミュレーションと可視化処理の同時実行を目指している。

以下では、Three Quads の性質を説明した後、各種相互結合網の埋め込みについて述べる。

3.1 トポロジ

以下簡単のため、Three Quads の各次元に対して各々 2^n 台のノードが配置されているものとする。任意のノード番号を N_i とし、この時、ノード番号は 2 進 $3n$ 桁表記でそれぞれ $(a_{3n}, a_{3n-1}, \dots, a_2, a_1)$ 、 2^n 進 3 桁表記でそれぞれ (A_3, A_2, A_1) と表記できるものとする。

Three Quads は、X 系統、Y 系統、Z 系統の 3 系等のネットワーク群で構成する。ノード数が X, Y, Z 方向それぞれ l, m, n 台のノードで構成されるシステムでは、X 系統のスイッチ群は $m \times n$ ポートのスイッチ l 台、Y 系統のスイッチ群は $l \times n$ ポートのスイッチ m 台、Z 系統のスイッチ群は $l \times m$ ポートのスイッチ n 台で構成し、各系統のスイッチは 3 次元座標系における当該系統の座標値に対応する。例えば、Z 系統の i 番目のスイッチは $Z = i$ 平面に対応しており、同一平面上のノードは 1 ホップで直接通信が可能となる。

各ノードが持つ 3 本のリンクは、X 系統、Y 系統、Z 系統のネットワークに 1 本ずつ接続する。この際、ノード番号が (A_3, A_2, A_1) の場合、X 系統の A_1 番目のスイッチ、Y 系統の A_2 番目のスイッチ、Z 系統の A_3 番目のスイッチに接続する。

このように Three は、ネットワーク全体において、3 系統 (X, Y, Z 座標系統) のネットワークが存在するということを表している。ベース- $m \times n$ -キューブ⁹⁾ は、その定義に従うと各次元のスイッチのサイズを一定に定めているが、PACS-CS⁷⁾ 等において採用されているハイパークロスバにはこの制限がない¹⁰⁾。Three Quads も同様、3 系統のネットワークの各スイッチのポート数を等しく設定する必要はない。

一方、Quads は、各スイッチに結合しているノード群が、X-Y, Y-Z, Z-X 平面を構成しているということを表す。

今、各次元方向のノード数が m であるとする。ベース- $m \times n$ -キューブは 1 次元方向 (直線上) の m 台のノードがスイッチにより結合しているのに対し、Three Quads は 2 次元方向 (平面上) の m^2 台のノードがスイッチにより結合している。この時、Three Quads はベース- $m \times n$ -キューブの拡張と考えると非常に理解しやすく、ベース- $m^2 \times n$ -キューブと呼ぶことができる。

3.2 ノード間通信

任意の 2 ノード A, B 間の 1 対 1 通信が最大 2 ホップで実現可能であることを示す。簡単のため、Three Quads の各次元に対して各々 2^n

台のノードが配置されているものとする。この時、ノード A および B のノード番号は 2 進 $3n$ 桁表記でそれぞれ $(a_{3n}, a_{3n-1}, \dots, a_2, a_1)$ および $(b_{3n}, b_{3n-1}, \dots, b_4, b_3, b_2, b_1)$ 、 2^n 進 3 桁表記でそれぞれ (A_3, A_2, A_1) および (B_3, B_2, B_1) と表記できるものとする。

Three Quads では、上記 2^n 進 3 桁表記において最大 2 桁まで異なるノード間の直接通信が可能である。今、 2^n 進 3 桁表記においてノード A と 1 桁が等しく、それ以外の桁がノード B と一致するノード C (C_3, C_2, C_1) を仮定する。この時、Three Quads 上では、ノード A は 1 ホップでノード C と通信が可能であると同時に、ノード C はノード B と 1 ホップで通信が可能である。したがって、ノード C を経由することで、ノード A とノード B の通信が可能であり、任意の 2 ノード間が 2 ホップで通信可能であることが示せる。

この仮定ではノード C の選び方、すなわち、ルーティング経路が 3 通り存在することが分かる。さらに、より細かく 2 進 $3n$ 桁表記で考えると、一致させるビット数の選択、また、一致させるビット位置の選び方にも複数の自由度があり、2 ホップという制約を課しても非常に多くのルーティング経路を設定することができる。

3.3 耐故障性 (fault-tolerance)

任意の 2 ノード間通信は、特定の 2 系統のスイッチのみを用いて実現が可能であり、ある特定の 1 系統のスイッチが故障した場合でも直径 2 を維持できる。冗長リンク (リンク) が多数存在するため、ある隣接ノードのリンクが切断された場合であっても、最小 1 ホップで通信が可能となる。例えば、 N_1 と N_{64} が通信を行う場合ルーティングにおいて、Z 系統のネットワークを用い、SWZ1 内でノード N_4 を中継し、X 系統のネットワークの SWX4 内で N_4 と N_{64} が通信を行うと設定されていたとする。この場合、 N_1 と N_{64} は 2 ホップで通信を行うことになる。ここで、 N_1 の Z リンクが故障した場合、Y リンクを用い、SWY1 内で N_4 を中継し、X 系統のネットワークの SWX4 内で N_4 と N_{64} が通信を行うことで、2 ホップで N_1 と N_{64} が通信が完了する (図 5)。

3.4 スケーラビリティ

m ポートのスイッチを使用してシステムを構築する場合、ベース- $m \times n$ -キューブでは m^3 のノード構成が可能であるのに対して、Three-Quads では、1 つのスイッチで平面状のノードを収容する必要があるため、最大ノード数は $m^{3/2}$ となる。これが中規模クラスタ向けと呼んでいる所以である。なお、 $m=64$ とすれば、512 ノードのクラスタまでは構築可能である。

例えば $n=2$ の場合、ノード C として $(a_6, a_5, a_4, b_3, b_2, b_1)$ や $(a_6, a_5, a_4, a_3, b_2, b_1)$ 、さらには $(a_6, a_5, c_4, c_3, b_2, b_1)$ 等も選択可能である。

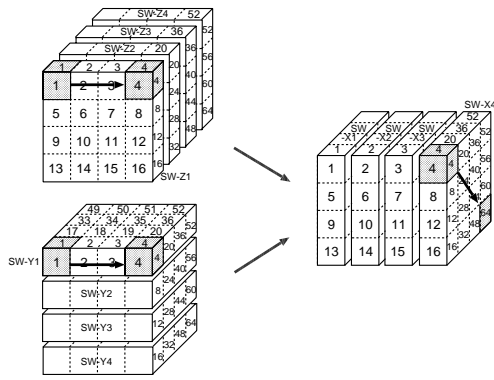


図5 ノード N_1 と N_{64} 間の代替経路の例

3.5 埋め込み性 (Embedability)

多数の他の網の埋め込みが可能である必要がある¹¹⁾。64台のノードが構成する Three Quads への主なネットワークの埋め込み能力について、トポロジーの観点から述べる。

- 1/2/3 次元トラス網

3次元トラスの埋め込みはほぼ自明であるため、2次元トラスの埋め込みについて説明する。Z系統のネットワークを図6のように2次元に平面展開し、2次元トラスを考える。このとき、SWZ1とSWZ2ならびにSWZ3とSWZ4を接続する横方向のリンクは、Y系統のネットワークを用いることによって接続可能である。一方、SWZ1とSWZ3ならびにSWZ2とSWZ4を接続する縦方向のリンクは、X系統のネットワークを用いることによって接続可能である。これにより2次元トラスの埋め込みが完了する。

1次元トラス(リング)に関しても、Z系統のネットワークを1次元展開することで同様に考えることが可能である。

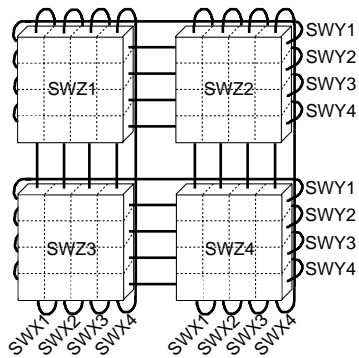


図6 2次元トラスの埋め込み

- 単純トリー網

Z系統のネットワークで、16ノードが直接結合している各SWにおいて段数が4段のトリーが成立

する。各スイッチ内におけるトリー網の親ノードのノードを仮に $N_{16}, N_{32}, N_{48}, N_{64}$ とする。これら親ノードはX系統またはY系統のネットワーク内の1つのスイッチにより結合しており、このスイッチにおいて2段のトリーが成立する。先の16ノードによる4段のトリーと親ノードによる2段のトリーによって、最終的に6段の単純トリーが完成する。

- Fat Tree

ScubeへのFat Treeの埋め込みを示す。Fat Treeの要素(上向きリンク数, 下向きリンク数, 階層数)を(2,4,3)とし、多重化されたツリー網を考える(図7)。任意のノード $(a_6, a_5, a_4, a_3, a_2, a_1)$ は、まずZ系統のネットワークにおいて4ノード $(a_6, a_5, a_4, a_3, *, *)$ と結合し、4分木が成立する。次にX系統のネットワークにおいて16ノード $(a_6, a_5, *, *, *, *)$ と結合し、16分木が成立する。最後にY系統のネットワークにおいて64ノードと結合し、64ノードのFat Treeが完成する。但し、各階層におけるScubeの物理的なリンク数は1なので、Fat Tree(図7)のような2本の上向きリンクを同時に使用した通信をエミュレートすることはできない。

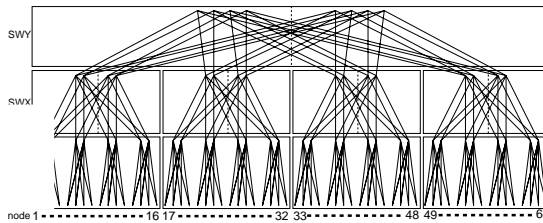


図7 ファットトリーの埋め込み

- ハイパーキューブ網(2進6-キューブ)

任意の隣接ノード(ハミング距離が1となるノード)間が結合していることを示せばよい。任意のノード $(a_6, a_5, a_4, a_3, a_2, a_1)$ はZ系統のネットワークにおいて、ノード $(a_6, a_5, a_4, a_3, a_2, \bar{a}_1), (a_6, a_5, a_4, a_3, \bar{a}_2, a_1)$ と結合している。Y系統のネットワークにおいて、ノード $(a_6, a_5, a_4, \bar{a}_3, a_2, a_1), (a_6, a_5, \bar{a}_4, a_3, a_2, a_1)$ と結合している。X系統のネットワークにおいて、ノード $(a_6, \bar{a}_5, a_4, a_3, a_2, a_1), (\bar{a}_6, a_5, a_4, a_3, a_2, a_1)$ と結合している。よってX,Y,Z3系統のネットワークを用いることで、2進6-キューブの埋め込みが可能である。但し、Fat Treeと同様、Scubeでは物理的なリンク数は3なので、2進6-キューブの6本のリンクを同時に使用した通信をエミュレートすることはできない。

1本の物理リンクを時分割で2本のリンクとしてシミュレートすることで同時通信をシミュレートすることは可能である。

Fat Treeと同様、1本の物理リンクを時分割で2本のリンクとしてシミュレートすることで同時通信をシミュレートするこ

- ベース- m n -キューブ (ベース-4 3-キューブ)
 ベース-4 3-キューブの埋め込みは Three Quads
 において各スイッチを独立した 4 個のスイッチと
 みなすことで、埋め込みが可能である (図 8 参照) .

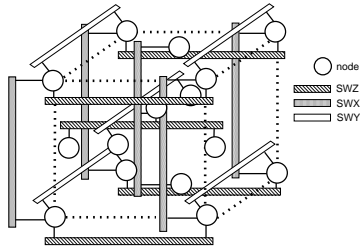


図 8 ベース- m 3-キューブ網

3.6 性能比

ハイパーキューブ網, ベース-4 3-キューブ, ハイパー
 クロス網, ベース-4² 3-キューブについての主な性能
 について比較した. ノード数が 64 台の場合の性能比較
 を表 2 に示す. 隣接ノード数は 3 次元シミュレーション
 空間上の隣接ノード数を表す. また, スイッチ数は
 物理的なクロスバースイッチの数であり, ベース-4 3-
 キューブ, ベース-4² 3-キューブのスイッチ数におい
 て $(n \times n)$ は n ポートのクロスバースイッチを表す.

ベース-4² 3-キューブは他網と比較すると, スイ
 ッチ数は多くなるが, 平均ホップ数が少なく, 隣接ノ
 ード間も含めた任意のノード間の通信経路が多く存在
 するため, 対故障性に優れている. 特に隣接ノード数
 が多く存在し, 3 次元シミュレーションモデルにおいて
 非常に優位性を有する. 1 ホップ通信が可能なノ
 ード数は, ベース- m 3-キューブの場合は 9 台 (内, 隣接
 ノード数は 6 台) であるのに対し, Three Quads の場
 合は 36 台 (内, 隣接ノード数は 18 台) である. その
 ため, Three Quads は, ベース- m 3-キューブと比較
 するとランダムな通信要求が発生した場合など, 柔軟
 なネットワークリソースの提供が可能である.

ハイパークロス網¹²⁾ は Three Quads と同様, 直径
 が 2 であり, 数値シミュレーションにおける行列転置
 の重要性に注目して, 転置を効率良く実現するネット
 ワークとして提案されたネットワークであるが, 次章
 で示すように, Three Quads は高い行列転置能力を
 有している.

4. Three Quads を用いた転置行列演算

科学技術計算において多用される転置行列演算
 が^{13)~15)}, Three Quads を用いることで, 効率良く
 実現できることを示す.

いま, 各ノードに $N \times N$ の配列を割り当て, シス
 テム全体 (64 ノード) で $8N \times 8N$ の行列 M を転置
 する場合を考える. この時, Scube のノードを図 9 の

ような 2 次元平面展開した時の i 行 j 列のノードがも
 つ $N \times N$ の配列を M_{ij} と表すこととする.

第 1 ステップでは, M をサイズ $4N \times 4N$ の 4 つの
 部分行列 A, B, C, D を用いて, $M = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$ によ
 って表した時の A^t, B^t, C^t, D^t を求める. なお, A, B, C, D
 の各要素を $a_{ij}, b_{ij}, c_{ij}, d_{ij} (0 \leq i, j \leq 3)$ と表現するもの
 とする. このステップでは各部分行列の転置に必要な
 通信がそれぞれ, $SWZ1, SWZ2, SWZ3, SWZ4$ 内に閉
 じた 1 対 1 通信であるため, ネットワーク上でのリ
 ソース競合が発生せず, 全ノードが並列に同時処理す
 ることが可能である. なお, 通信に先立ち, 各ノ
 ード内では M'_{ij} を求めておくものとする. これにより,
 $M' = \begin{pmatrix} A^t & B^t \\ C^t & D^t \end{pmatrix}$ が得られる.

第 2 ステップでは, M' における C^t, B^t を入れ換
 えることで, $M = \begin{pmatrix} A^t & C^t \\ B^t & D^t \end{pmatrix}$ を求める. M' と M の関
 係をみると A^t, D^t は既に転置は完了しており, 通信
 を行う必要はない. いま, c_{ij}, b_{ij} が置かれているノ
 ードの位置関係を考えると, これらは X 系統あるいは
 Y 系統のスイッチにおいて直接通信可能な位置に
 あり, すべての i, j に対して同時に通信を行っても, 一
 切のネットワークの競合は発生しない. よって, この
 ステップも全ノードが並列に同時実行可能である. こ
 れにより, 2 ステップで行列の転置が可能であることを
 示せる.

さらに第 1 ステップと第 2 ステップで使用するネッ
 トワークが全く独立であることに注目すると, 通信の
 タイリングなどの技術を用いることで, これらの 2 つ
 のステップが融合可能であり, 実質的には 1 ステップ
 で転置が可能であることが分かる.

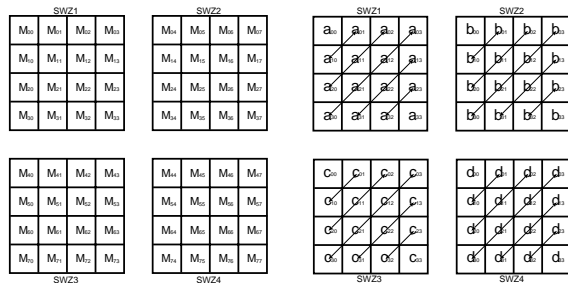


図 9 初期設定

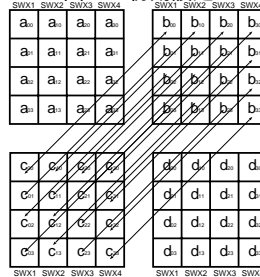


図 11 第 2 ステップ

図 10 第 1 ステップ

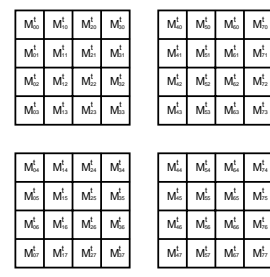


図 12 最終状態

とは可能である.

これは最後のステップに行ってもよい.

表 2 各種ネットワークの性能比較 (ノード数 64 台の場合)

	Hypercube	base-4 3-cube	Hypercross	base-4 ² 3-cube
次数	6	3	2	3
直径	6	3	2	2
隣接ノード数	6	6	-	18
平均ホップ数	3	2.25	1.75	1.41
スイッチ数	0	(4×4)×64	8×8×8	(16×16)×12

5. ま と め

本稿では、コモデティ技術を用いた中規模 PC クラスタ向けの相互結合網 Three Quads の提案を行った。

Three Quads は耐故障性、多様な網の埋め込み能力に優れていることを示した。科学技術計算に広く使用されている行列の転置演算処理の全対全通信において、ネットワークの競合が生じることなく、高速に実現が可能である。

今回は、ネットワークの定性評価についてのみの報告となったが、今後は定量評価を行っていく。また、現在、並列画像合成アルゴリズムを開発しており、Three Quads への実装評価を行う予定である。

参 考 文 献

- 1) Ogata, M., Kajihara, K., Kukukawa, T. and Terada, T.: An Evaluation of Communications Cost for Simultaneous Processing with Simulation and Visualization using an Image-Composition Device, ハイパフォーマンスコンピューティングと計算科学シンポジウム (HPCS), pp.63-70 (2006).
- 2) 助村俊一ほか: パーチャルマイクロスコープの制御手法の開発, 情報処理学会研究報告 99-HPC-66(HOKKE'97), Vol.97, No.37, pp.25-30 (1997).
- 3) S., D. et al.: RVSLIB:A Library for Concurrent Network Visualization of Large Scale Unsteady Network Visualization of Large Scale Unsteady Simulation, *SPEEDUP journal 11*, pp.55-65 (1997).
- 4) Groner, J., M.Lee, Martin, J., Moorhed, R. and Newman, J.: A Concurrent Visualization System for High-Performance Computational Simulations, *IEEE Visualization Poster Abstracts*, pp.139-140 (2005).
- 5) 富田眞治: 超高速体感型シミュレーションシステムの研究, 科学研究費補助金平成 16 年度基盤研究 (S) 新規採択課題の概要について, 日本学術振興会, pp.10-12 (2004).
- 6) 吉村知晋, 吉良祐司, 森真一郎, 中島康彦, 富田眞治: 汎用 GPU を用いた大規模可視化クラスタの構築, 第 33 回可視化情報シンポジウム講演

- 文集, Vol.25-Suppl, No.1, pp.277-280 (2005).
- 7) 朴 泰祐, 佐藤三久, 宇川 彰: 計算科学のための超並列クラスタ PACS-CS の概要, 情報処理学会研究報告, Vol.2005-HPC-103, pp.133-138 (2005-08).
 - 8) 住元真司, 久門耕一, 朴 泰祐, 佐藤三久, 宇川 彰: PACS-CS のための Ethernet を用いた高性能通信機構の設計, 情報処理学会研究報告, Vol.2005-HPC-103, pp.139-144 (2005-08).
 - 9) 鈴岡 節, 田邊 昇, 中村定雄, 藤田純一, 小柳 滋: 並列 AI マシン Prodigy の相互結合網の評価, 信学論, Vol.J71-D, No.8, pp.1496-1501 (1988).
 - 10) 天野英晴: 並列コンピュータ, 昭晃堂 (1996).
 - 11) 富田眞治: 並列コンピュータ工学, 昭晃堂 (1996).
 - 12) Kadota, H. et al.: Parallel Computer ADEN-ART-Its Architecture and Application-, Vol. ICS-91, pp.1-8 (1991).
 - 13) 小国 力, 村田健郎, 三吉俊郎, J.J. ドンガラ, 長谷川秀彦: 行列計算ソフトウェア, 丸善 (1991).
 - 14) Agarwal, R.C., Gustavson, F.G. and Zubair, M.: A High Performance Parallel Algorithm for 1-D FFT, *Proc. Supercomputing '94*, pp.34-40 (1994).
 - 15) 高橋大介, 朴 泰祐, 佐藤三久: PC クラスタにおける並列一次元 FFT のブロックアルゴリズム, 情報処理学会論文誌, Vol.43, No.SIG 6, pp.134-142 (2002).

謝 辞

日頃より御討議いただく京都大学大学院情報学研究科富田研究室の諸氏に感謝致します。また、Scube の実装においてご協力頂いた、住商情報システム (株) の山本英明氏, 宮澤剛氏に感謝致します。なお、本研究の一部は、日本学術振興会科学研究費補助金基盤研究 S (課題番号 16100001), 21 世紀 COE プログラム (課題番号 14213201) による。