Title	ウェーハスタック型超並列システムのネットワーク構造
Author(s)	大木，孝之
Citation	
Issue Date	1997-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1049
Rights	
Description	Supervisor:堀口 進，情報科学研究科，修士

修 士 論 文

ウェーハスタック型超並列システムのネットワーク構造に関する研究

指導教官　堀口 進 教授

北陸先端科学技術大学院大学
情報科学研究科情報システム学専攻

大木孝之

1997 年 3 月 18 日

目 次

図目次

表 目 次

第 1 章

はじめに

1.1 研究の背景と目的

従来の単一のノイマン型プロセッサによる処理能力向上は限界に近く、大型化と高価格化が進んでいる。そのため新たな処理能力向上の手段として、複数の安価なプロセッサを並列化して処理を行なう並列処理が注目されている。特にシミュレーションや動画像処理のような膨大な処理能力を必要とする分野では、10^4を越えるプロセッサを用いる超並列処理が求められている。

超並列計算機システムを従来のチップを用いた方法で実現することは、膨大なチップ面積や信頼性の低下などの問題が生じるため困難である。しかし、近年の VLSI 技術の進歩によって、超並列計算機システムを 1 つのチップのように 1 枚のウェーハ上に実装する Wafer Scale Integration(WSI) が実現可能となってきた。WSI[1] を用いることにより、コンパクトでより高い信頼性を得ることが可能である。しかし、技術的制約からウェーハの規模は数 inch が限界であるため、1 枚のウェーハ上に実装できる PE(Processing Element) 数は限られる。このため従来の WSI では、十分な PE を搭載できないために超並列計算機システムとしては制限されたシステムとなる。またウェーハの規模が大きくなるにつれて、ウェーハ上の回路の欠損による故障の増加、工業コストの増加、電源消費量の増加など種々の問題が発生する。これらの問題を解決する手法として、HRL(Hughes Reserch Laboratories) によってウェーハを縦に積み重ねたウェーハスタック構造が提案されている [2]。

ウェーハスタック構造では、個々のウェーハの規模を抑制しながら、ウェーハ枚数を増加

させることによって、より大規模な超並列処理システムが実装可能となる。ウェーハ間結線にはマイクロブリッジを用いる手法が提案され実装されている。マイクロブリッジを用いることで短い間隔でウェーハ間を結合できる。このため超並列ネットワークを 1 枚のウェーハ上にマッピングする場合と比較して PE 間の物理的距離も短縮可能である。Etchells と Grinberg は、各機能をモジュール化する手法で 16 枚のウェーハ上に $128PE \times 128PE$の超並列計算機を実現している [2]。Campbell と Toborg は、実際にウェーハスタック構造を用いたニューロコンピュータを実現している [3]。

しかし、ウェーハスタック構造ではウェーハ間結線に問題が生じる。マイクロブリッジは、短い間隔で信頼性の高いウェーハ間結合が可能であるが、大きさが数百μm に達する。このためウェーハ間結線数はウェーハ上のチップ面積に大きく影響する。超並列システムの実装性能に大きな影響を及ぼすウェーハ間結線や処理能力に大きな影響をおよぼす通信遅延時間は、相互結合網によって決定される。

本研究で扱う直接網は、結合手法から非階層型と階層型に分類できる。超並列計算機システムに用いる非階層型ネットワークとして、2 次元トーラス、3 次元トーラス及びハイパーキューブ (binary-n-cube) 等がある。2 次元トーラスは直径が大きいため、PE 間の通信遅延が大きくなるという問題がある。3 次元トーラスは 2 次元トーラスと比較して、直径の増加を抑えることができるが、ウェーハ間の結線数がやや多いという問題がある。またハイパーキューブは、PE 数が大きくなると次数が増加して総結線数が膨大になり、実装が困難になるという問題がある。他方、ハイパネット [4] などの階層型ネットワークは、直径の増加を抑えながら、全体の結線数を減少させることができる。また、ウェーハスタック構造へ階層化ネットワークを実装する場合、一つの階層を一つのウェーハにのせ、階層間の結線数を少なく設計することにより、ウェーハ間の結線が少ないネットワークを構成できる。このようなウェーハスタック構造を考慮した階層化ネットワークには、TESH(Tri connected mESHes) がある [5]。

本研究では、ウェーハスタック構造を用いて超並列ネットワークを実現する手法について提案し、ウェーハ間結線数やネットワーク特性について詳しく検討する。次にウェーハスタック構造に実装した従来のネットワーク特性について議論した結果に基づいて、ウェーハスタック構造に適したネットワークとして 3 次元階層型ネットワークの提案を行なう。また、ウェーハ間結線数、チップ面積、直径などのネットワーク特性について詳しく議論する。最後にアプリケーションのマッピングを行ない、実行ステップ数を用いて比較し、

その有効性を確かめる。

1.2 本論文の構成

本論文の構成は次のとおりである。第 2章では、まずウェーハスタック構造の構成と問題点について述べる。更に既存のネットワークを用いウェーハスタック構造におけるネットワーク特性を検討する。その結果からウェーハスタック構造に適したネットワークの検討を行なう。第 3 章では、ウェーハスタック構造に適したネットワークとして 3 次元階層型ネットワークを提案する。更にネットワーク特性を調べ、既存のネットワークとの比較を行ない、その有効性を検討する。第 4章は、アプリケーションのマッピングを行ない、その性能を評価する。第 5 章は結論である。

第 2 章

ウェーハスタック構造の特性

2.1 はじめに

本章ではウェーハスタック構造の問題点を明らかにし、ウェーハスタック構造に適したネットワークを検討する。最初にウェーハスタック構造の構成とその技術的問題点を述べる。つぎに既存のネットワークを用いてネットワーク特性を調べ、ウェーハスタック構造の問題点をより具体的に示す。最後にウェーハスタック構造に適したネットワークの検討を行なう。

プロセッサ間相互結合網は、プロセッサ間結合が静的に固定されている直接網とスイッチにより、結合網を動的に変化させることができる間接網に分けられる。本研究では、直接網の中から以下のネットワークを選択し、ネットワーク特性の検討に用いる。

- 2 次元トーラス
- ハイパーキューブ
- 3 次元トーラス
- TESH(Tori connected mESHes)

2 次元トーラスは、メッシュの上下の端、左右の端をそれぞれ結合したネットワークで、PE から出るの結線数 (次数) が 4 本と低次数で一定している。3 次元トーラスは、3 次元メッシュの対になっている面同士を結合したネットワークで 3 次元構造を持つ。ハイパーキューブは、超並列向きネットワークとして注目されているネットワークで、直径が小さ

い反面、PE の増加にともない次数が大きく増加する。TESH は、唯一ウェーハスタック構造を考慮したネットワークで階層型構造を用いている。

検討に用いるネットワーク特性には、一般的なネットワーク特性とウェーハスタック構造に特有のネットワーク特性から以下のものを用いる。

- 直径
- ウェーハ間最大結線数
- ウェーハ上の最大レイアウト面積

ウェーハ間最大結線数は、ウェーハ間にあるマイクロブリッジの最大数である。ウェーハ上の最大レイアウト面積はウェーハ間結線数が最大となる位置におけるレイアウト面積である。

2.2 ウェーハスタック構造

ウェーハスタック構造は、HRL で構成方法に関する技術的検討が行なわれている新しい WSI の手法で、複数のウェーハを用いるのがその特徴である。ウェーハスタック構造を用いることによって、超並列計算機システムを複数のウェーハに分割して構成できるため、1 枚のウェーハを比較的小型にしながら、大規模な超並列計算機システムを構成できる。また各々のウェーハの規模を小型化できるため、ウェーハ上の回路の信頼性向上、消費電力の低減、工業コストの低減を行なうことができる。そのためより大規模な超並列計算機システムには必要不可欠な手法である。

ウェーハスタック構造は、図 2.1に示すようにウェーハを縦に積み重ねた構造を持つ。図中のウェーハ上の長方形は各種ネットワークを表し、ウェーハ間の円柱はウェーハ間結線を表す。ウェーハ間の通信は、ウェーハ内を貫通している結線を用いて行う。ウェーハ間結線は、ウェーハ内を貫通したフィードスルー (FeedThrough) とウェーハ間の結合を行なうマイクロブリッジ (MicroBridge) に分けることができる。フィードスルーは、結線を通す穴の生成とウェーハ内を貫通した結線の構成を同時に行なう必要がある。ウェーハは、先に穴を開けた場合、微小な力でも穴の周囲に応力集中がおき、小さな亀裂が発生するため、取扱いが非常に困難となるためである。そのため、レーザーを用いて穴を開ける手法の適用は難しい。そこで、図 2.2に示すように、小さなアルミニウムの断片を加

熱する手法が用いられる。この手法ではウェーハの融解と同時にアルミニウムが導線を構成するため、応力集中による亀裂の発生を避けることができる。マイクロブリッジは、図 2.3に示すように中央を凸にした薄い金属板を、ウェーハの上面 (回路を構成する面) と下面に互いに直交するように取り付け、金属板の中央を接続することで結線を構成する。マイクロブリッジの利点は高い信頼性でウェーハ間の間隔を短く結線できることである。HRL で作成されたウェーハスタック構造を用いた超並列計算機システムでは、ウェーハの厚さ 5mm、ウェーハ間隔 0.5mm のようにウェーハ間の結線距離が 5.5mm で実現されている [2]。また、ウェーハ間結線はウェーハ間の通信に加え、熱の伝搬によって熱を分散させる機能も持つ。

その反面、マイクロブリッジの大きさがウェーハスタック構造の問題となる。図 2.3に示してあるように、マイクロブリッジの長さは数百μm に達するため、数μm の幅で実現できるウェーハ上の結線と比較して非常に大きなウェーハ面積を必要とする。このためウェーハ間結線数の増加はチップ面積を大きく増加させ、超並列計算機システムのウェーハスタック構造への実装を困難にする。したがって、ネットワークをウェーハスタック構造にマッピングする場合、ウェーハ間結線数をできる限り少なくする必要がある。ただし、マイクロブリッジの長さよりも小さい回路は、金属版中央の凸部分の下に配置することで、チップ面積の増加をある程度減らすことが可能である。

以降の節では、ウェーハスタック構造に適したネットワークを検討するため、各種ネットワークをウェーハスタック構造にマッピングした場合のネットワーク特性について、特にウェーハ間結線数に着目しながら議論する。

2.3 ネットワーク構造の表記法

本節では、各ネットワークのネットワーク特性を数式で表す際に用いる表記の記号について定義を行なう。総 PE 数を N、ウェーハ枚数を h、ウェーハ 1 枚当りの PE 数を m($=\frac{N}{h}$) とする (図 ??)。また、階層型ネットワークの基本モジュールの PE 数を n、レベル i を構成するレベル i-1 サブネット数を g、ウェーハ内最高階層レベルを $L_d(=1+\lfloor\log_g\frac{m}{n}\rfloor)$、ウェーハ 1 枚あたりの L_dの階層数を $s(=m\ mod\ ng^{L_d-1})$ とする。ここで定義した値を用いて、直径 D、ウェーハ間最大結線数 C_{max}、最大レイアウト面積 A などのネットワーク特性を表現する。

各ネットワークで定義したウェーハ間最大結線数を再び用いる場合には、ネットワー

ク、PE 数、ウェーハ 1 枚あたりの PE 数を用いて以下のように表現する。

$$C_{max}(network, N, m)$$

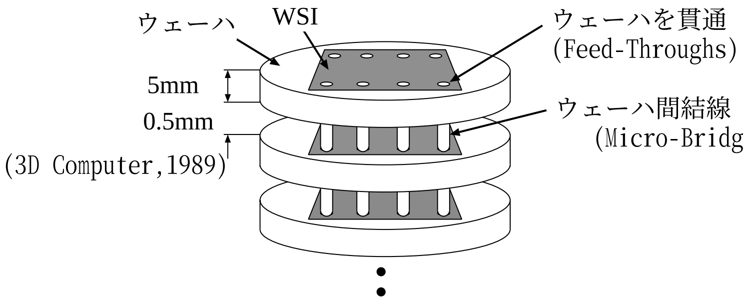


図 2.1: ウェーハスタック構造

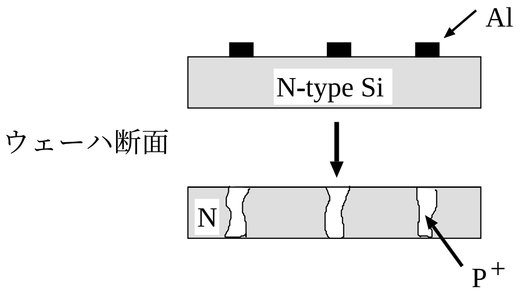


図 2.2: FeedThrough

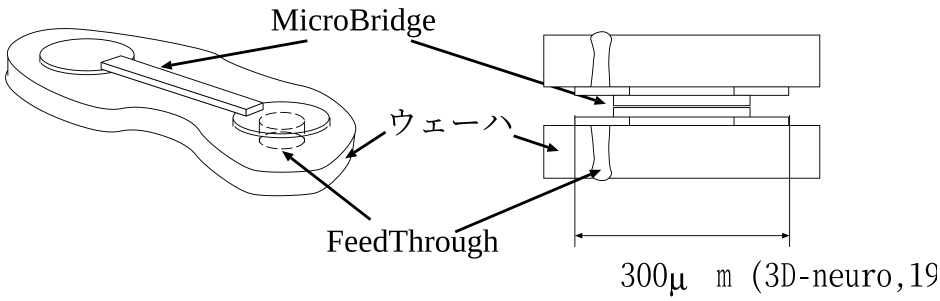


図 2.3: MicroBridge

2.4　2 次元メッシュ・トーラス

2.4.1　ネットワーク構造

2 次元メッシュネットワーク (2D メッシュ) は、PE を格子状に並べ、上下左右 4 方向の隣合う PE 同士を結合したネットワークである。2 次元トーラスネットワーク (2D トーラス) は、2D メッシュの行 (column) の両端、列 (row) の両端を結合したネットワークで、このトーラスリンクにより直径を 2D メッシュの 1/2 に抑えることができる。図 2.4と図 2.5に 4 × 4 の 2D メッシュと 4 × 4 の 2D トーラスを示す。

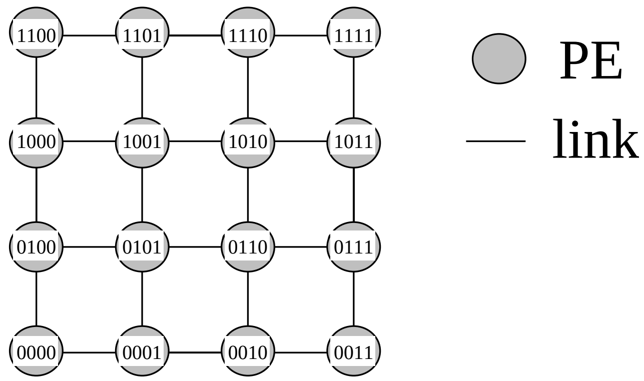


図 2.4: 2 次元メッシュ(N=16)

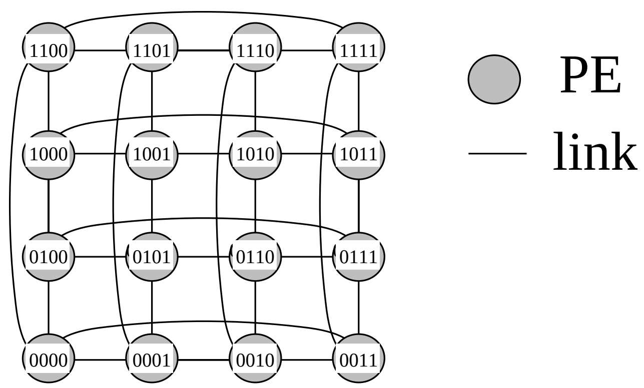


図 2.5: 2 次元トーラス (N=16)

2.4.2　ネットワーク特性

各 PE のアドレスの添字付け (indexing) には、行主対添字付け (row-major indexing) を用いる。行主対添字付けは、左から右、下から上の順序でアドレス付けを行なう方法である。$u \times u$ の 2D メッシュと 2D トーラスのアドレスは以下のようになる。

$$A = (a_y)(a_x) \quad (a_x, a_y = 0, 1, \ldots, u-1)$$

a　2D メッシュのルーティングと直径

2D メッシュのルーティングは、まず行方向のアドレスを一致させ、次に列方向のアドレスを一致させることで実行する。ここで、パケット送信元のアドレスを $(s_y)(s_x)$、送信先のアドレスを $(d_y)(d_x)$ とする。この場合、2D メッシュのパケットの移動は以下のように行なわれる。

- $d - s > 0$　アドレスが増加するノードへパケットを送る
- $d - s < 0$　アドレスが減少するノードへパケットを送る。

例えば 4×4 の 2D メッシュの P_{00} から P_{12} へのルーティングは、以下のように行なわれる。

$$\begin{array}{llll} (00) \longrightarrow (10) & \text{行方向の移動終了} & & \\ \qquad\qquad\ \downarrow & & & \\ \qquad\qquad (11) \longrightarrow (12) & \text{列方向の移動終了} & & \end{array}$$

直径は、最大移動距離なので列と行の最大移動距離の和となる。したがって $u \times u$ の 2D メッシュ(N=u^2) の直径は以下のようになる。

$$\begin{aligned} D_{2D-mesh} &= 2(u-1) \\ &= 2(\sqrt{N}-1) \end{aligned} \tag{2.1}$$

b　2D トーラスのルーティングと直径

2D トーラスのルーティングも、まず行方向のアドレスを一致させ、次に列方向のアドレスを一致させることで実行する。ここで、パケット送信元のアドレスを $(s_y)(s_x)$、送信先のアドレスを $(d_y)(d_x)$ とする。$u \times u$ の 2D トーラスのパケットの移動は、トーラスリ

ンクによる移動を考慮して、以下のように行なわれる。ここではアドレスが増加する方向を時計回り、アドレスが減少する方向を反時計回りと定義する。

- $0 < d - s \leq u - (d - s)$ 時計回りでパケットを送る
- $d - s > u - (d - s)$ 反時計回りでパケットを送る
- $0 < s - d \leq u - (s - d)$ 反時計回りでパケットを送る
- $s - d > u - (s - d)$ 時計回りでパケットを送る

例えば 4×4 の 2D トーラスの P_{00} から P_{32} へのルーティングは、以下のように行なわれる。

$$\begin{array}{llll} (00) \longrightarrow (30) & \text{行方向の移動終了} & & \\ \qquad\qquad\ \downarrow & & & \\ \qquad\quad\ (31) \longrightarrow (32) & \text{列方向の移動終了} & & \end{array}$$

2D トーラスでは、行と列がそれぞれ環結合になっている。要素数 u の環結合の直径は u/2 なので、$u \times u$ の 2D トーラス ($N = u^2$) の直径は以下のようになる。

$$\begin{aligned} D_{2D-torus} &= 2\frac{u}{2} \\ &= u \\ &= \sqrt{N} \end{aligned} \tag{2.2}$$

2.4.3 ウェーハ間最大結線数

本節ではウェーハスタック構造の問題であるウェーハ間結線数についての検討を行なうため、2D メッシュと 2D トーラスのウェーハ間最大結線数 (C_{max}) の定式化を行ない、ウェーハ間最大結線数に必要な面積を含めたレイアウト面積の定式化を行なう。

ウェーハ間最大結線数は、ネットワークの分割方法や分割したネットワークのウェーハスタック構造への配置順序によって大きく異なる。そのためどの様な配置順序の時にウェーハ間最大結線数が最小になるかを調べる必要がある。最初に $u \times u$ の 2D トーラスを 1(PE/wafer) でウェーハスタック構造にマッピングした場合について、配置順序によるウェーハ間結線数の増減を調べ、最適な配置を検討する。次にこの結果を用いて、2D

表 2.1: 配置済み隣接 PE 数によるウェーハ間結線の増減

	入力ウェーハ間結線数	出力ウェーハ間結線数	ウェーハ間結線の増減
4-0PE	4	0	-4
3-1PE	3	1	-2
2-2PE	2	2	0
1-3PE	1	3	+2
0-4PE	0	4	+4

トーラスを h 枚のウェーハ上へのマッピングする方法を示し、ウェーハ間最大結線数を求める。最後に 2D メッシュを h 枚のウェーハ上にマッピングした場合のウェーハ間最大結線数を求める。

2 次元トーラスの i 行 j 列 ($0 \leq i \leq m-1$, $0 \leq j \leq m-1$) にある PE を PE(i,j) で表す。また P(i,j) は P(i-1,j),P(i+1,j),P(i,j-1),P(i,j+1) の 4 つの隣接 PE と結合されている。ただしネットワークの外周に配置された PE はトーラスリンクによって反対側の PE と接続されているので、PE(0,*) は PE(u-1,*) と互いに結合し、PE(*,0) は PE(*,u-1) と互いに結合している。ここで、* は全ての PE を表す。ウェーハスタック構造に配置を行なう PE は、配置済み PE との隣接状態によって 5 種類に分けることができる。例えば、配置する PE が 4 つの配置済み PE と隣接している場合、この PE をウェーハ上に配置することにより、ウェーハ間結線は 4 本減少する。これは隣接している配置済み PE 数に等しいウェーハ間結線が配置する PE に入力されるからである。同様に 3 つの配置済み PE と隣接している PE をウェーハ上に配置する場合、ウェーハ間結線数が 2 本減少する。以降では、表 4.1に示すように、配置する PE に (入力ウェーハ間結線数)-(出力ウェーハ間結線数) を付けて表現する。この表現を用いた場合、最初の例は 4-0PE、次の例は 3-1PE となる。今回は場合分けの表現を簡単にするためウェーハに配置する最初の PE は PE(0,0) とする。2 次元トーラスでは完全な対称構造を持つため、PE の位置による差異はない。したがって、この条件によって一般性が失われることはない。$u \times u$ の 2D トーラスをウェーハスタック構造へ 1(PE/wafer) で配置を行なった場合について、マッピング方法による場合分けを行ないながらウェーハ間最大結線数の変化を以下に示す。**a. 配置済み PE との**

隣接数が 0 の PE(0-4PE) から優先して配置する場合 (完全独立配置法)

図 2.6(a) のように、各行 (列) の PE を 1 つおきにウェーハ上に配置し、更に偶数行 (列) と奇数行 (列) の PE をずらした配置を行なうと 0-4PE の配置数が最大となる。この状態で PE を配置すると 4-0PE となり、ウェーハ間結線数は減少する。したがってこの状態でウェーハ間最大結合線数をとる。0-4PE は$\frac{N}{2}$個だけ配置できるので、ウェーハ間最大結線数 C_{max}は以下のようになる。

$$\begin{aligned} C_{max} &= \frac{N}{2} \times 4 \\ &= 2N \end{aligned} \tag{2.3}$$

b. 配置済み PE と隣接している PE(4-0PE,3-1PE,2-2PE,1-3PE) から優先して配置する場合

b.1 1-3PE を優先して配置する場合 (1 隣接配置法)

図 2.6(b) に 1-3PE を最大配置した場合を示す。以降の配置は 2-2PE,3-1PE,4-0PE なのでウェーハ間結線の増加はない。したがってこの状態でウェーハ間最大結合線数をとり、以下のようになる。

$$\begin{aligned} C_{max} &= 4 + 2\{u - 2 + \frac{u}{2}(u-2)\} \\ &= 4 + u^2 - 4 \\ &= N \end{aligned} \tag{2.4}$$

b.2 2-2PE を優先して配置する場合 (2 隣接配置法)

2-2PE を優先してウェーハ上に配置することは、配置する PE をネットワーク上から長方形 (正方形) に選択することを意味する。例えば図 2.6(c) において 1-3PE(3,1) を配置した場合、PE(3,0),PE(3,2),PE(3,3) の順序で配置する場合は全て 2-2PE であり、ウェーハ間結線数を増加させることなく配置可能である。さらに配置済み PE のネットワーク上の位置は再び長方形となる。新しい行 (列) の PE を配置する場合は 1-3PE となり、ウェーハ間結線数の増加が起きる。以降長方形の行 (列) を埋めるまでは 2-2PE が配置可能であり、ウェーハ間結線の増加はない。したがって 2 隣接配置法では新しい行 (列) の PE を配置する時のみ、1-3PE となりウェーハ間結線が増加する。すなわち、2 隣接配置法のウェーハ間結線数は行と列の数に比例する。図 2.6(d) にウェーハ間最大結線数をとる場合の配置を示す。これ以降に PE の配置を行なう場合は、トーラスリンクのため 2-2PE,3-1PE,4-0PE

となり、ウェーハ間結線数は増加しない。したがってこの状態でウェーハ間最大結合線数をとり、以下のようになる。

$$\begin{aligned} C_{max} &= 2 \times 2(u-1) \\ &= 4(\sqrt{N}-1) \end{aligned} \tag{2.5}$$

b.3 行 (列) 単位で PE を配置する場合 (行主体配置法)

2 隣接配置法と同じ配置方法だが、行と列の PE をウェーハ上に配置するのではなく、行 (列) の PE のみを配置する方法がある。図 2.6(e) のように 0 行の最後の PE を配置する場合、2 隣接配置法でも述べたようにトーラスリンクにより 2-2PE となりウェーハ間結線の増加は抑えられる。したがって 0 行の全 PE を配置した時点のウェーハ間結線数は以下のようになる。

$$\begin{aligned} C &= 4 + 2(u-2) \\ &= 2u \end{aligned}$$

新たな隣接行 (列) の PE を配置する場合、2 隣接配置法と同様、最初に 1-3PE を配置する以外、2-2PE を満たした配置が可能であり、ウェーハ間結線数の増加は起きない。更に図 2.6(f) のように 2 行 (列) 目以降最後の PE を配置する場合は 3-1PE となり、ウェーハ間結線が減少する。2 隣接配置法では最後の行や列の PE を配置するまでウェーハ間結線数が増加するのに対して、この点が大きく異なる。新たな行で最初に配置する 1-3PE と行の最後の 3-1PE が、ウェーハ間結線数の増加を打ち消し合うため、以降ウェーハ間結線数の増加がなくなる。したがって図 2.6(g) の状態でウェーハ間最大結線数をとり、以下のようになる。

$$\begin{aligned} C_{max} &= 2u + 2 \\ &= 2\sqrt{N} + 2 \quad (N \neq 4) \end{aligned} \tag{2.6}$$

ただし PE 数が 4 の場合の配置順序は 2 隣接配置法と同じであるため、ウェーハ間最大結線数は (2.5) 式となる。したがってまとめるとウェーハ間最大結線数は以下のようになる。

$$C_{max} = \begin{cases} 2\sqrt{N} + 2 & N \neq 4, \\ 4 & N = 4 \end{cases} \tag{2.7}$$

したがってウェーハ間最大結線数は行 (列) 主体配置法が最小になる。したがって本研究では 2D トーラスのウェーハスタック構造への PE 配置は行 (列) 主体配置法で行う。行 (列) 主体配置法をまとめると以下のようになる。

1　PE(0,0) を配置する。

2　0 行 (0 列) にある 1-3PE を配置する。以降この行 (列) 中の全ての PE を順番に配置する。ただし配置する場合は 1-3PE であることが必要である。また行 (列) の最後の PE は 2-2PE である。

3　隣接する行 (列) に 1-3PE を配置する。以降この行 (列) の PE を配置する。ただし、配置する PE は 2-2PE であることが条件となる。また行 (列) の最後の PE は 3-1PE である。

4　以降最後の行 (列) まで 3 を繰り返す。

5　最後の行 (列) は最初 2-2PE で、以降 1-3PE、最後は 0-4PE となる。

この条件を満たす限りどの様な並べ方でもウェーハ間最大結線数は最小となる。しかし、現実にウェーハスタック構造に配置する場合は結線長の問題が生じる。上記の条件を満たした配置方法でも、各行ごとに配置順序が異なると結線長に差が生じるため、配置方法としては現実的ではない。結線長を均一にするには、各行 (列) 内で結線長を平均化して、各行 (列) の PE の配置順序を等しくすることが必要である。以降、2D メッシュはアドレス順序での配置を行なう。しかし、2D トーラスの場合トーラスリンクがあるためアドレス順序の配置では、図 2.9(a) のように結線長が非常に長くなる。したがって、2D トーラスはインライン順序でウェーハ上へ配置を行なう。インラインはトーラスに用いられる配置方法で、ネットワークを折り畳むように配置することで最大結線長を低く抑えることができる。$u \times u$ の 2D トーラスのインラインによる配置は以下のようになる。

$$I(N,i) = \begin{cases} \dfrac{i}{2} & i : \text{even}, \\ N - \lceil \dfrac{i}{2} \rceil & i : \text{odd} \end{cases} \tag{2.8}$$

$$PE(k,j) = PE(I(u,k), I(u,j))$$

このインラインを用いることによって、図 2.9 (b) に示すように配線長の平均化が行なわれ、最大結線長も減少している。また、インラインによる配置は行主体配置法の配置条件を満たしているので、ウェーハ間最大結線数は最適である。

次に $u \times u$ の 2D トーラスを格子状に分割して h 枚のウェーハに配置を行なう場合のウェーハ間最大結線数を求める (図 2.8)。1 枚のウェーハには m 個の PE が載っていると仮定する。2D トーラスを 1(PE/wafer) で配置する場合と比較して、m(PE/wafer) の場合はウェーハ間の結線数が$\sqrt{m}$倍になる。したがって (2.7) 式よりウェーハ間最大結線数は以下のようになる。

$$\begin{aligned} C_{max} &= (2\sqrt{h}+2)\sqrt{m} \\ &= 2\sqrt{mh}+2\sqrt{m} \\ &= 2\sqrt{N}+2\sqrt{m} \quad (\frac{N}{m} \neq 4) \end{aligned} \tag{2.9}$$

したがってまとめるとウェーハ間最大結線数は以下のようになる。

2D トーラスのウェーハ間最大結線数：

$$C_{max} = \begin{cases} 2\sqrt{N}+2\sqrt{m} & \frac{N}{m} \neq 4, \\ 4\sqrt{m} & \frac{N}{m} = 4 \end{cases} \tag{2.10}$$

同様に 2D メッシュを正方格子に分割し、ウェーハスタック構造にマッピングした場合のウェーハ間最大結線数を求める (図 2.7)。分割した 2D メッシュから出るリンク数は 2D トーラスの半分である。このため、2D トーラスのウェーハ間最大結線数は 2D トーラスの半分である。2D トーラスのウェーハ間最大結線数を以下に示す。

2D メッシュのウェーハ間最大結線数：

$$C_{max} = \begin{cases} \sqrt{N}+\sqrt{m} & \frac{N}{m} \neq 4, \\ 2\sqrt{m} & \frac{N}{m} = 4 \end{cases} \tag{2.11}$$

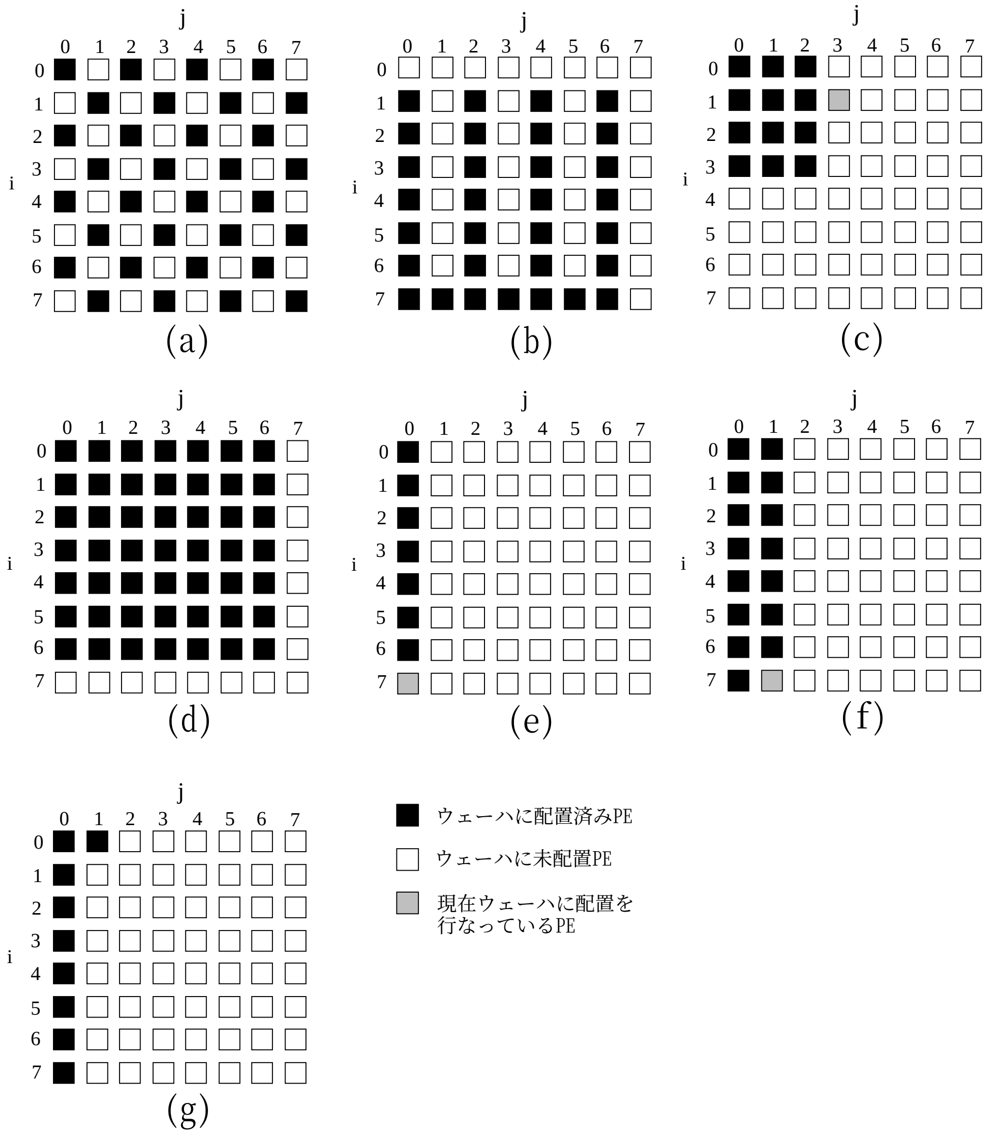


図 2.6: 2 次元トーラスのウェーハスタック構造への配置順序

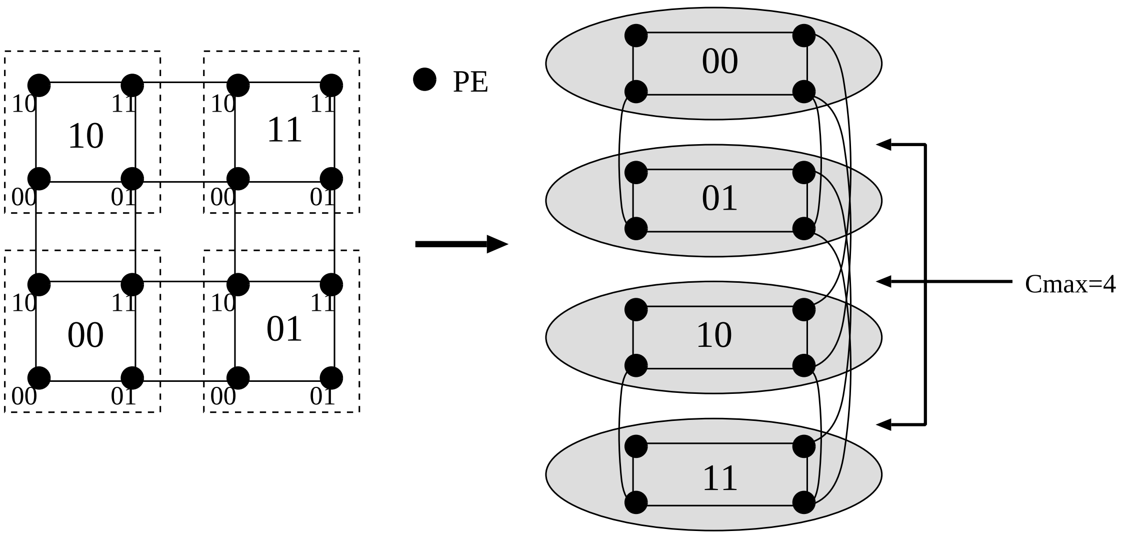


図 2.7: 2 次元メッシュの C_{max}(N=16,m=4)

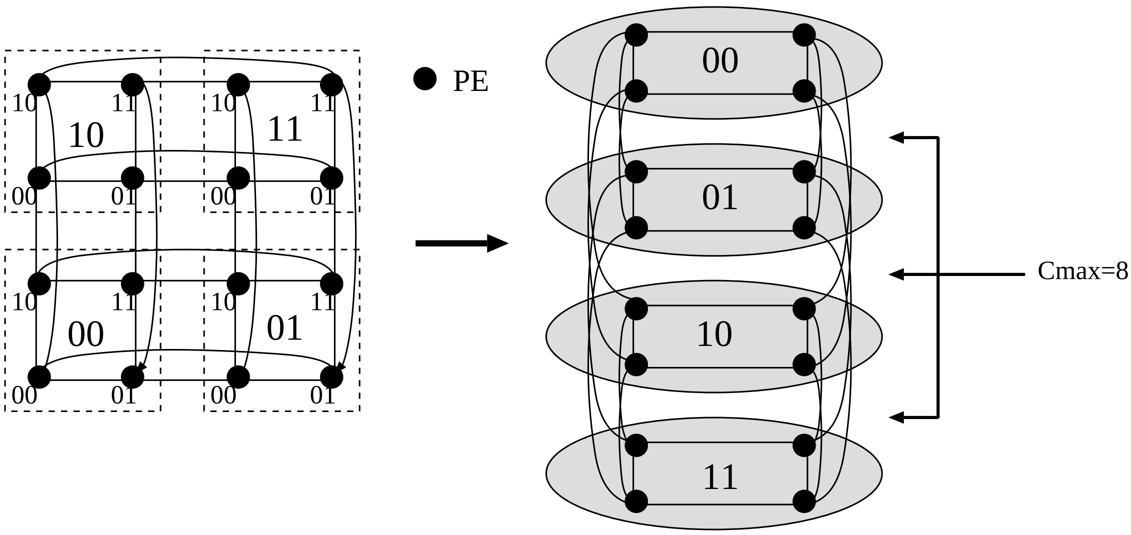


図 2.8: 2 次元トーラスの C_{max}(N=16,m=4)

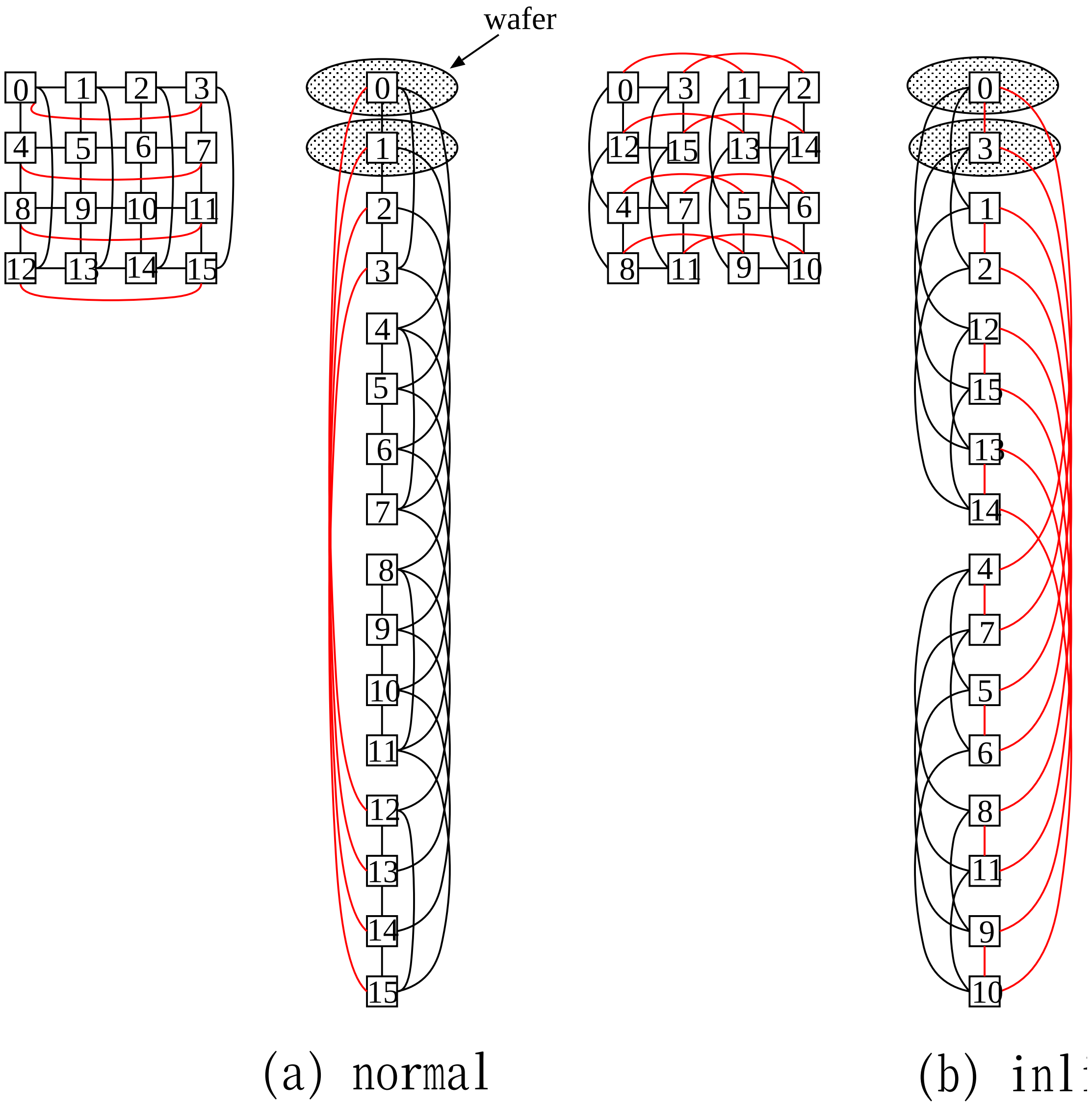


図 2.9: 2 次元トーラスにおけるインライン化と結線長

2.4.4 レイアウト面積

本節では、ウェーハ間最大結線数がレイアウト面積にどの様な影響を及ぼすかを調べる。まず各ネットワークのレイアウト面積を求める一般式の定義を行なう。その後で 2D メッシュや 2D トーラスのレイアウト面積に必要な記号の定義を行なう。

レイアウト面積とはPE、リンク、全空白領域を含めた全面積を表す。ここでは第 2.3節と以下のパラメータを用いてレイアウト面積を表す。先に求めたウェーハ間最大結線数とトーラスリンクの本数が各ネットワーク固有のパラメータとなる。各行 (列) あたりのトーラスリンク本数に差があると、行 (列) の幅が不均一になる。これを回避するため、トーラスリンク本数は最大本数を用いる。すなわち列方向のトーラスリンクの最大本数を全ての列に当てはめる。同様に行方向のトーラスリンクの最本数を全ての行に当てはめる。

$$
\left[\begin{array}{lll}
PE\text{の幅} & : & W_{PE} \\
\text{ウェーハ間結線 (マイクロブリッジ) の幅} & : & W_{MB} \\
\text{結線 1 本当りのデータ線数} & : & p \\
\text{データ線の幅} & : & W_{link} \\
\text{ブロック数} & : & L \\
\text{トーラスリンクの最大本数} & :\text{縦} & t_y \\
 & :\text{横} & t_x
\end{array}\right]
$$

ウェーハスタック構造ではウェーハ間結線は、熱の伝搬によってウェーハの熱分散の役割を担う。そのため、ウェーハ間結線はウェーハ上に均等に配置することが望ましい。本研究では 1 枚のウェーハ上に載っているネットワークを正方格子状に L 分割したブロック単位でウェーハ間結線の均等配置を行なう。ただし 1 本のウェーハ間結線 (p 本のデータ線を持つ) を複数ブロックへの分割はしない。またウェーハ間結線の均等配置に必要な移動回数として 1 本のウェーハ間結線 (p ビット幅) に、縦横 1 本のリンク (p ビット幅) が必要であると仮定する。また最悪の条件を仮定し、全ての均等配置に用いるリンクは同一線上に配置できないとした。つまり、均等配置に必要なリンク幅はリンク本数の和になると仮定している。したがってレイアウト面積は PE 幅、トーラスリンク幅、ウェーハ間結線幅、ウェーハ間結線の均等配置のためのリンク幅を考慮して以下のようになる。

$$
\left\{
\begin{array}{lll}
\text{レイアウト面積} & & A = W_y W_x \\
 & & \\
\text{ウェーハ上の全}PE\text{幅} & \text{：縦} & S_{PE,y} = \sqrt{m}\ W_{PE} \\
 & \text{：横} & S_{PE,x} = \sqrt{m}\ W_{PE} \\
\text{全トーラスリンク幅} & \text{：縦} & S_{TL,y} = \sqrt{m}\ t_y\ p\ W_{link} \\
 & \text{：横} & S_{TL,x} = \sqrt{m}\ t_x\ p\ W_{link} \\
\text{ウェーハ間結線分散のためのリンク幅} & \text{：縦} & S_{DMB,y} = C_{max}\ p\ W_{link} \\
 & \text{：横} & S_{DMB,x} = C_{max}\ p\ W_{link} \\
\text{ウェーハ間結線の幅を除いた全幅} & \text{：縦} & W'_y = S_{PE,y} + S_{TL,y} + S_{DMB,y} \\
 & \text{：横} & W'_x = S_{PE,x} + S_{TL,x} + S_{DMB,x} \\
\text{ウェーハ間結線の幅を除いたブロック幅} & \text{：縦} & W'_{B,y} = \frac{W'_y}{\sqrt{L}} \\
 & \text{：横} & W'_{B,x} = \frac{W'_x}{\sqrt{L}} \\
\text{ブロック当たりのウェーハ間結線数} & \text{：} & c = \lceil \frac{C_{max}}{L} \rceil \\
\text{ブロック幅に並べたウェーハ間結線の厚さ} & \text{：} & w_y = \lceil \frac{c\ p\ W_{MB}}{W_{B,y}} \rceil W_{MB} \\
\text{ウェーハ間結線の幅を含めたブロック幅} & \text{：縦} & W_{B,y} = W'_{B,y} + w_y \\
 & \text{：横} & W_{B,x} = W'_{B,x} \\
\text{全チップ幅} & \text{：縦} & W_y = W_{B,y}\ \sqrt{L} \\
 & \text{：横} & W_x = W_{B,x}\ \sqrt{L}
\end{array}
\right\}
$$

図 2.10に L=4,p=1 におけるハイパーキューブのレイアウト面積を求める際に用いる各記号を示す。

レイアウト面積を求めるには、各ネットワーク固有のトーラスリンク本数が必要である。ここでは、2D メッシュと 2D トーラスのトーラスリンク本数をもとめる。2 次元メッシュはトーラスリンクが一切ない。また 2 次元トーラスは各行と列に 1 本づつトーラスリンクがある。したがって 2 次元メッシュと 2 次元トーラスの t_x, t_yは以下のようになる。

2 次元メッシュの最大トーラスリンク本数：

$$t_x = t_y = 0 \tag{2.12}$$

2 次元トーラスの最大トーラスリンク本数：

$$t_x = t_y = 1 \tag{2.13}$$

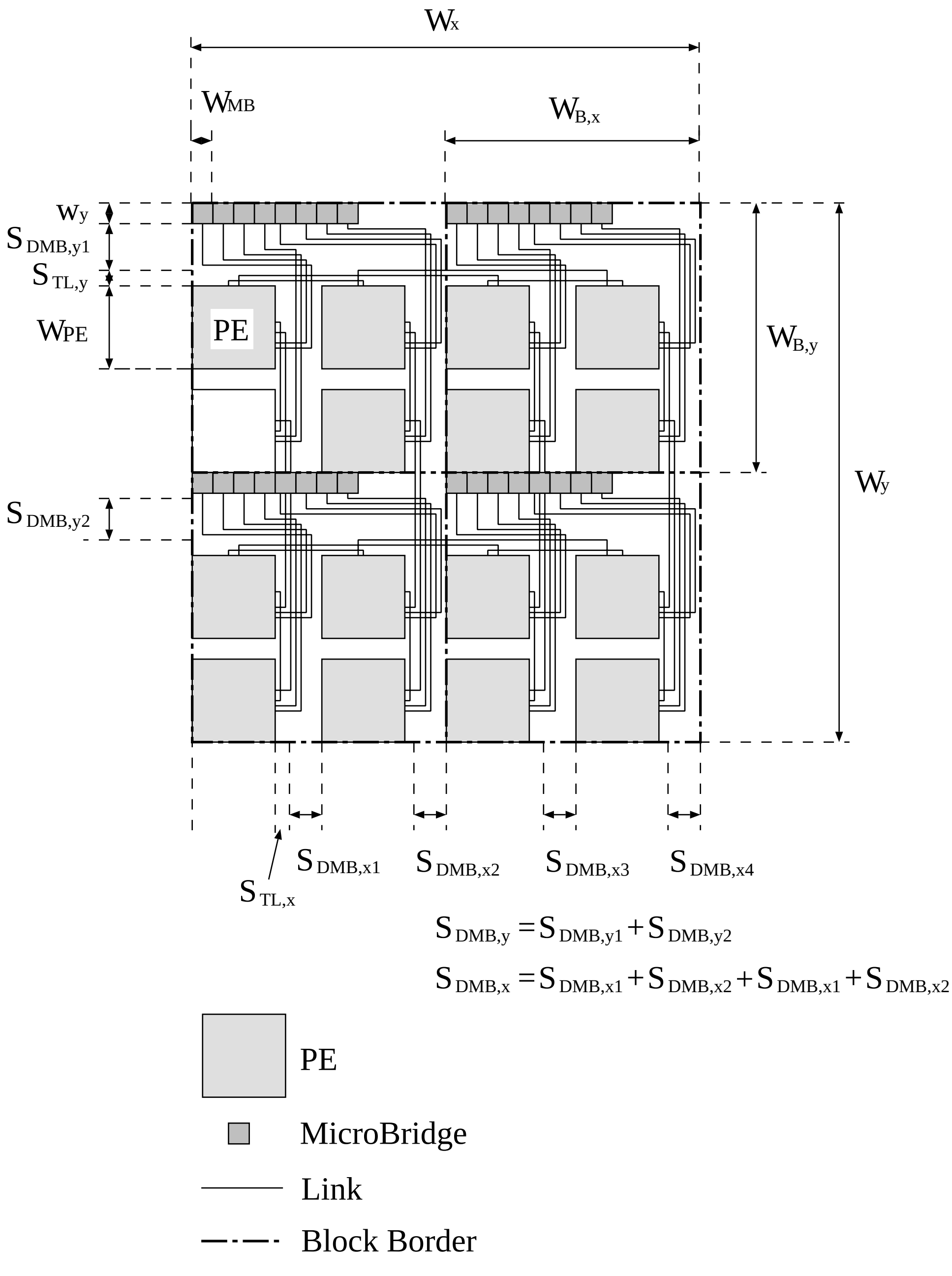


図 2.10: ハイパーキューブのレイアウト面積

2.5　ハイパーキューブ

2.5.1　ネットワーク構造

ハイパーキューブはさまざまなネットワークを内包する強力なネットワークである。超並列計算機システムのネットワークとしても、注目されており、幾つかの並列計算機システムに採用されている [6]。

k-次元のハイパキューブは 2^K の PE から構成され、アドレスを k bit の 2 進数で表した場合、k bit の内 1bit だけ異なる PE 同士が結合される。4 次元のハイパーキューブを図 2.11に示す。

2.5.2　ネットワーク特性

ハイパーキューブのルーティングは、パケット送信元のアドレスとパケット送信先のアドレスの内、異なっているビットを下位ビットから 1 ビットずつ一致させるような経路を選択することで行なわれる。例えば 4 次元のハイパーキューブにおいて、P_{0000}から P_{1111} へののパケットの移動経路は、以下のように行なわれる (図 2.12)。

$(0000) \longrightarrow (0001) \longrightarrow (0011) \longrightarrow (0111) \longrightarrow (1111)$

直径は、全てのビットが異なっている場合なので以下のようになる。

$$\begin{aligned} D &= k \\ &= \log_2 N \end{aligned} \tag{2.14}$$

2.5.3　ウェーハ間最大結線数

ハイパーキューブのウェーハスタック構造へのマッピングは以下のように行なう。アドレスは A=$a_{k-1}a_{k-2}\ldots a_1a_0$とする。まず、アドレス順に並んだネットワークを格子状に j 分割する。上位 $\log j$ bit のアドレスを参照し、この順序でウェーハに配置を行なう。この配置ではウェーハに垂直方向の PE 列を考えると、各列がそれぞれハイパーキューブを構成している。このためハイパーキューブのウェーハ間最大結線数は、(ウェーハ枚数 h 個

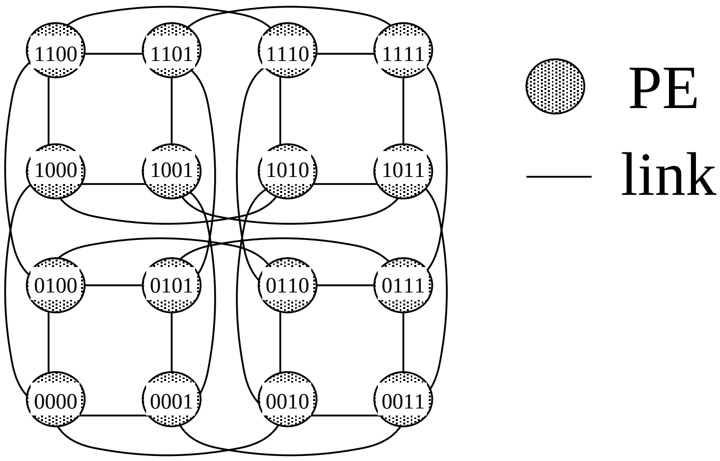


図 2.11: ハイパーキューブ (N=16)

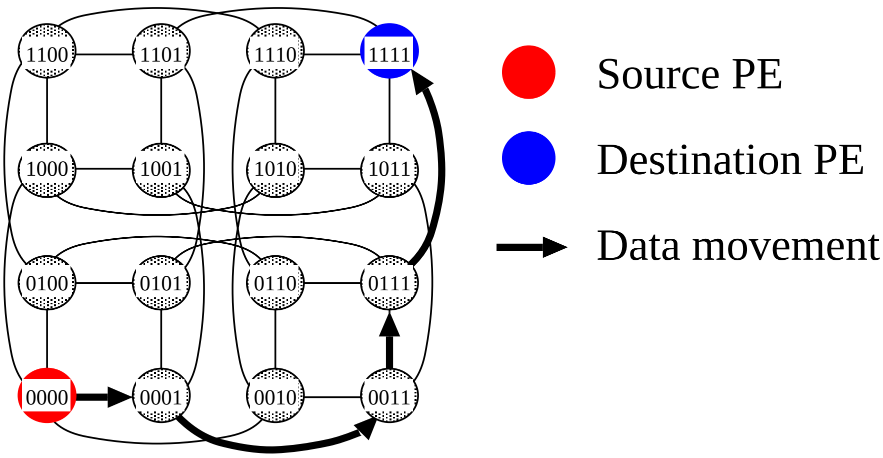


図 2.12: ハイパーキューブのルーティング (N=16)

に等しい PE 数を 1(PE/Wafer) で配置した場合のウェーハ間最大結線数)×(ウェーハ 1 枚当りの PE 数 m) となる。したがって、まず 1(PE/Wafer) の場合についてハイパーキューブのウェーハ間最大結線数を求める。

この配置におけるハイパーキューブのウェーハ間最大結線数を求めるためにハイパーキューブの PE をアドレス順に 1 列に並べた場合の特性を以下に示す。またアドレス j と j+1 の間のリンク数を n_jとする。

1　ハイパーキューブはアドレスが 1 ビット異なるノード間の結合を行なう。つまり $2^0, 2^1, \ldots, 2^{k-1}$離れたノード間の結合が行なわれる。

2　ハイパーキューブは完全対称である。

3　k 次元のハイパーキューブは 2 つの k-1 次元のハイパーキューブを並べ、2^{k-1} 離れたノード間の結合を行なったものである。つまり n_jで最大リンク数をとる場合、鏡面位置 $(2^k - 1) - n_j$でも同じリンク数を持つ。また k 次元のハイパーキューブのリンク数は $n_0, n_1, \ldots, n_{2^{k-1}-1}$の間は、k-1 次元のハイパーキューブのリンク数にアドレスを足したものになる。

したがって、まず最初に最大リンクをとるノード間の鏡面位置、すなわち最後に最大リンク数をとる位置を求める。

k 次元のハイパーキューブにおいて鏡面の中心位置は以下のようになる。

$$
\begin{aligned}
n_{ck} &= \frac{N-2}{2} + 1 \\
&= \frac{N}{2} \\
&= 2^{k-1}
\end{aligned}
\tag{2.15}
$$

したがって k 次元のハイパーキューブにおいて最大リンク数をとるノード間の位置を n_k とすると、k+1 次元のハイパーキューブでは、この鏡面位置で最大リンク数をとる。したがって k+1 次元のハイパーキューブの最大リンク数をとる位置 n_{k+1}は以下の漸化式によって求まる。

$$
\begin{aligned}
n_{k+1} &= 2n_{ck} - n_k \\
\Leftrightarrow n_{k+1} + n_k &= 2^k \\
n_1 &= 1
\end{aligned}
\tag{2.16}
$$

この (2.17) 式から以下の 3 項漸化式が得られる。

$$n_{k+2} - n_{k+1} - 2n_k = 0 \tag{2.17}$$

(2.17) 式を解くと以下のようになる。

$$n_k = \frac{2^k + (-1)^{k-1}}{3} \tag{2.18}$$

特性の 3 で述べたように k 次元のハイパーキューブのリンク数は $c_0, c_1, \ldots, c_{k-1} - 1$ の間は、k-1 次元のハイパーキューブのリンク数にアドレスを足したものになる。したがって、k 次元のハイパーキューブの最大リンク数をとる位置を C_k とすると k+1 次元のハイパーキューブの最大リンク数は以下の漸化式で求まる。

$$C_{k+1} = C_k + \frac{2^k + (-1)^{k-1}}{3} \tag{2.19}$$

この (2.19) 式を k が偶数と奇数に場合分けを行なって解くと以下のようになる。

$$C_k = \begin{cases} \dfrac{2(2^k - 1)}{3} & k\text{: even,} \\ \dfrac{2^{k+1} - 1}{3} & k\text{: odd} \end{cases} \tag{2.20}$$

したがって k 次元のハイパーキューブを m(PE/wafer) でマッピングした場合のウェーハ間最大結線数は以下のようになる。

$$C_k = \begin{cases} \dfrac{2m(N - m)}{3} & k\text{: even,} \\ \dfrac{2N - m}{3} & k\text{: odd} \end{cases} \tag{2.21}$$

2.5.4 レイアウト面積

ここではレイアウト面積を求めるために必要な最大トーラスリンク本数を求める。ハイパーキューブは HC 結合リンクを用いたウェーハ上への実装を考える [7]。k 次元ハイパーキューブを構成する場合、HC リンクには $2^k - 1 = N - 1$ 本の幅が必要である。k=2 の場合の HC リンクを図 2.14に示す。したがって m 個の PE から構成されるハイパーキューブの行、列あたりの PE 数は$\sqrt{m}$本なので、t_x, t_yは以下のようになる。

$$t_x = t_y = \sqrt{m} - 1 \tag{2.22}$$

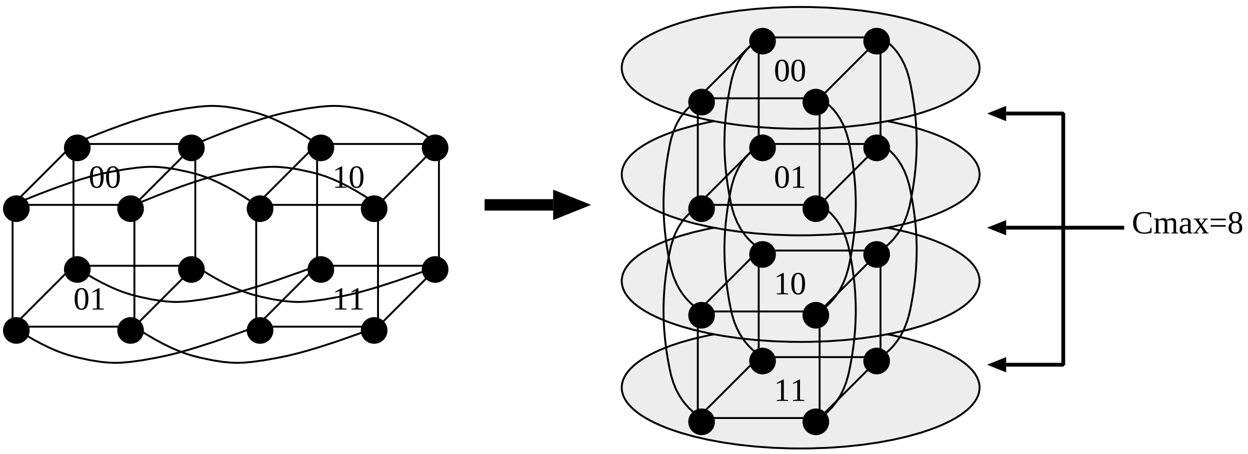


図 2.13: ハイパーキューブのウェーハ間最大結線数 C_{max}(N=16,m=4)

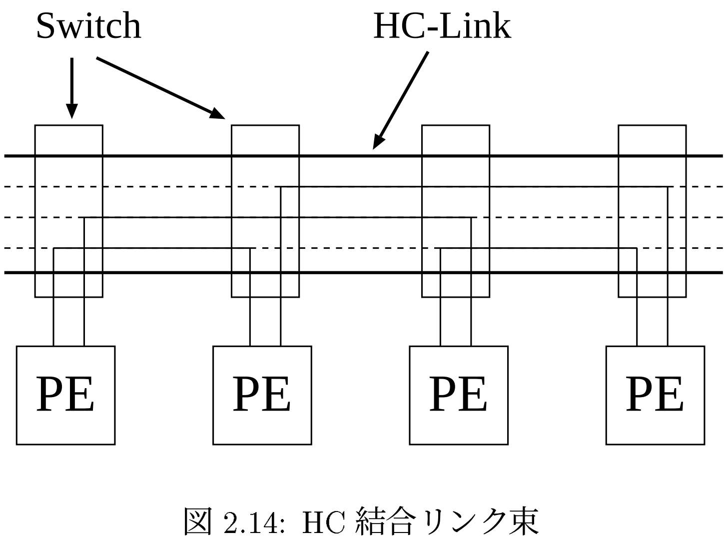


図 2.14: HC 結合リンク束

2.6　3 次元メッシュ・トーラス

2.6.1　ネットワーク構造

3D メッシュは 2D メッシュを立体的な構造にしたもので、PE を立方格子状に並べ、隣合う PE 同士を結合したネットワークである。3D トーラスは 2D トーラスを立体的な構造にしたもので、3D メッシュの対になっている面をトーラスリンクで結合したネットワークである。このトーラスリンクによって直径が 3D メッシュの半分になる。PE 数 64 の 3D メッシュと 3D トーラスを図 2.15と図 2.16に示す。

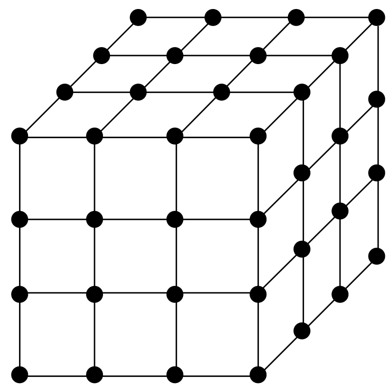


図 2.15: 3D メッシュ(N=64)

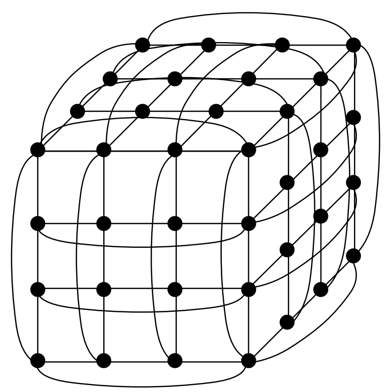


図 2.16: 3D トーラス (N=64)

2.6.2 ネットワーク特性

3D メッシュのルーティングは、2D メッシュのルーティングほとんど同じである。2D メッシュではパケットの送信方向が行と列の 2 方向であるのに対して、3D メッシュの送信方向が x,y,z の 3 方向になっていること以外に相違点はない。3D トーラスのルーティングについても 2D トーラスのルーティングとの違いは、送信方向の数のみである。

3D メッシュの直径とルーティング

3D メッシュのルーティングは z 軸、y 軸、x 軸の順序で行なう。ここで、パケット送信元のアドレスを $(s_z)(s_y)(s_x)$、送信先のアドレスを $(d_z)(d_y)(d_x)$ とすと、3D メッシュのパケットの移動は以下のように行なわれる。

- $d - s > 0$ アドレス増加
- $d - s < 0$ アドレス減少

例えば $4 \times 4 \times 4$ の 3D メッシュにおいて、P_{000}から P_{112}へののパケットの移動経路は、以下のように行なわれる。

$$\begin{array}{lll} (000) \longrightarrow (100) & \text{z 方向の移動終了} & \\ \qquad\qquad\;\downarrow & & \\ \qquad\qquad (110) & \text{y 方向の移動終了} & \\ \qquad\qquad\;\downarrow & & \\ \qquad\qquad (111) \longrightarrow (112) & \text{x 方向の移動終了} & \end{array}$$

$u \times u \times u$ の 3D メッシュ$(N = u^3)$ の直径は、以下のようになる。

$$D_{3D-mesh} = 3(u-1) \tag{2.23}$$

$$= 3(N^{\frac{1}{3}} - 1) \tag{2.24}$$

3D トーラスの直径とルーティング

3D トーラスのルーティングも z 軸、y 軸、x 軸の順序で行なう。ここで、パケット送信元のアドレスを $(s_z)(s_y)(s_x)$、送信先のアドレスを $(d_z)(d_y)(d_x)$ とすと、3D トーラスのパケット移動は以下のように行なわれる。

- $0 < d - s \leq u - (d - s)$ 時計回りでパケットを送る
- $d - s > u - (d - s)$ 反時計回りでパケットを送る
- $0 < s - d \leq u - (s - d)$ 反時計回りでパケットを送る
- $s - d > u - (s - d)$ 時計回りでパケットを送る

例えば 4 × 4 × 4 の 3D トーラスにおいて、P_{000}から P_{332}へののパケットのルーティングは、以下のように行なわれる。

$$
\begin{array}{lll}
(000) \longrightarrow (300) & \text{z 方向の移動終了} & \\
\qquad\qquad\quad\downarrow & & \\
\qquad\qquad (330) & \text{y 方向の移動終了} & \\
\qquad\qquad\quad\downarrow & & \\
\qquad\qquad (331) \longrightarrow (332) & \text{x 方向の移動終了} &
\end{array}
$$

$u \times u \times u$ の 3D トーラス ($N = u^3$) の直径は、以下のようになる。

$$
\begin{aligned}
D_{3D-torus} &= 3\frac{u}{2} && (2.25)\\
&= \frac{3}{2}N^{\frac{1}{3}} && (2.26)
\end{aligned}
$$

2.6.3 ウェーハ間最大結線数

$u \times u \times u$ の 3D メッシュと 3D トーラスのウェーハスタック構造へのマッピングは以下のように行なう。アドレスは A=$(a_x)(a_y)(a_z)$ ($0 \leq a_x, a_y, a_z \leq u - 1$) とする。

1 a_z=0 の x-y 断面をウェーハに配置する。

2 x-y 断面がウェーハに配置できない場合 ($m < u^2$) は、x-y 断面を更に正方格子に分割して 2D メッシュ(2D トーラス) と同じ方法で配置を行なう。

3 x-y 断面がウェーハに複数配置可能な場合は、以下のように行なう。3D メッシュでは x-y 断面を行主体添字付けのアドレス順でウェーハ上へ配置する。3D トーラスではインライン順序でウェーハ上へ配置する。

4　次の x-y 断面を配置して、2 から繰り返す。これを $a_z = u - 1$ の x-y 断面をウェーハに配置するまで行なう。

3D メッシュのウェーハ間最大結線数は次の 2 通りに分けることができる。1 枚以上の x-y 断面をウェーハに配置する場合は x-y 断面の PE 数に等しい。この場合、図 2.17(a) のようにウェーハ間結線数はすべてのウェーハ間で等しい。x-y 断面を分割してウェーハに配置を行なう場合は (x-y 断面の PE 数)+(x-y 断面を分割した場合のウェーハ間最大結線数) となる。これは図 2.17(b) に示すように $a_z = 1, 2, \ldots, u-1, u-2$ の x-y 断面では z 軸方向のリンク数が減少しないためである。したがって 3D メッシュのウェーハ間最大結線数は以下のようになる。

3D メッシュのウェーハ間最大結線数：

$$C_{max} = \begin{cases} m < N^{\frac{2}{3}} & N^{\frac{1}{3}} + N^{\frac{2}{3}} + \sqrt{m} & \frac{N^{\frac{2}{3}}}{m} \geq 4, \\ & N^{\frac{1}{3}} + 2\sqrt{m} & \frac{N^{\frac{2}{3}}}{m} = 4 \\ m \geq N^{\frac{2}{3}} & N^{\frac{2}{3}} & \end{cases} \tag{2.27}$$

3D トーラスはトーラスリンクの配線長を一定に保つため (2.8) 式のインラインで順序でウェーハに配置する。3D トーラスは 3D メッシュと比べ、x-y 断面間、x-y 断面内ともに 2 倍のリンク数を持つので、ウェーハ間最大結線数は 3D メッシュの 2 倍になる。したがって、3D トーラスのウェーハ間最大結線数は以下のようになる。

3D トーラスのウェーハ間最大結線数：

$$C_{max} = \begin{cases} m < N^{\frac{2}{3}} & 2(N^{\frac{1}{3}} + N^{\frac{2}{3}} + \sqrt{m}) & \frac{N^{\frac{2}{3}}}{m} \geq 4, \\ & 2(N^{\frac{2}{3}} + 2\sqrt{m}) & \frac{N^{\frac{2}{3}}}{m} = 4 \\ m \geq N^{\frac{2}{3}} & 2N^{\frac{2}{3}} & \end{cases} \tag{2.28}$$

2.6.4　レイアウト面積

3D メッシュや 3D トーラスはウェーハ 1 枚当りの PE 数によってトーラスリンクの最大本数が異なる。ウェーハ 1 枚当りの PE 数が $N^{\frac{2}{3}}$ 以下の場合は、ウェーハには 3D メッシュと 3D トーラスの xy 断面である 2D メッシュや 2D トーラスが配置される。この場合

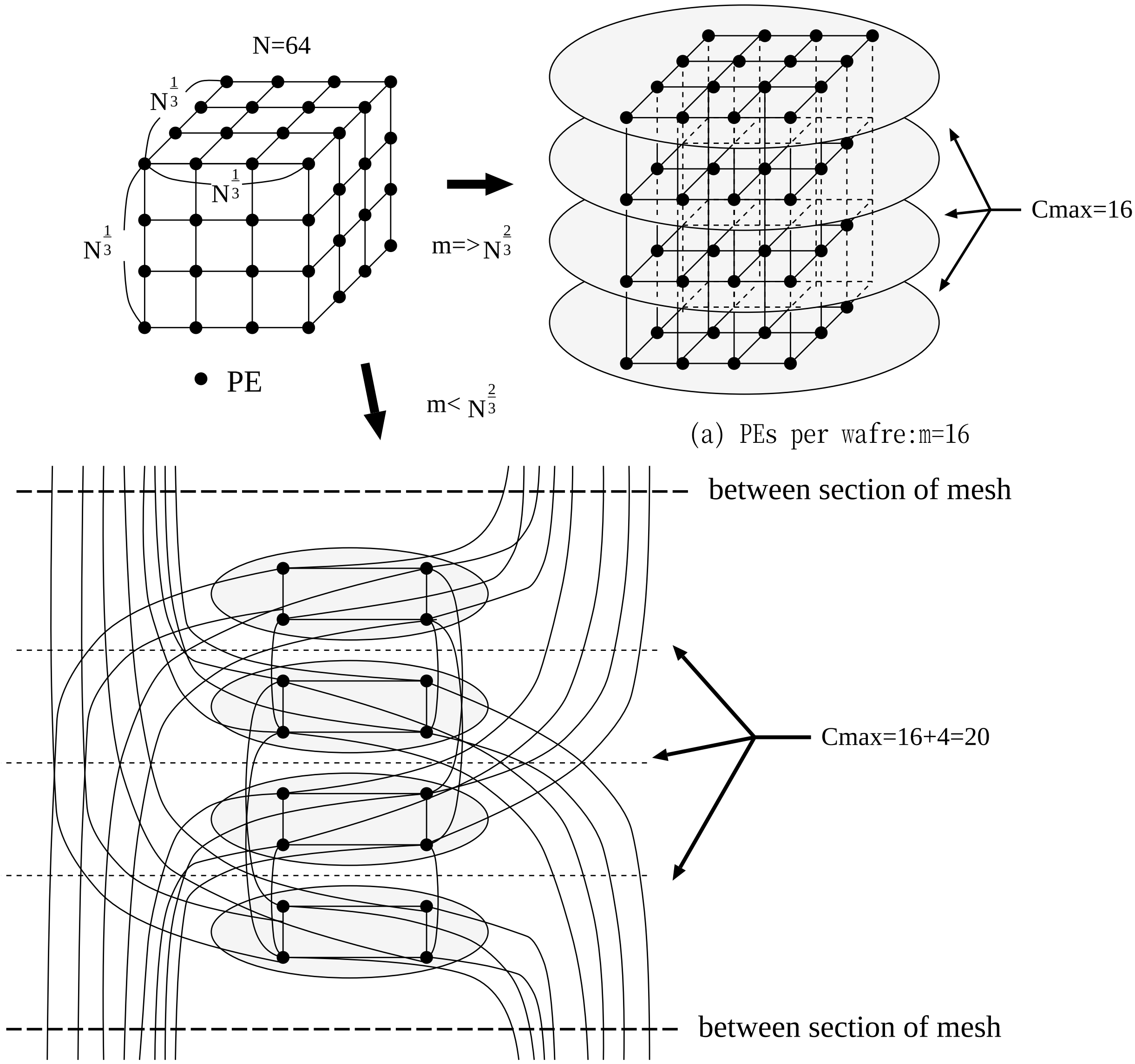
のトーラスリンクの最大本数は 2D メッシュや 2D トーラスと同じである。ウェーハ当りの PE 数が $N^{\frac{2}{3}}$ 以上の場合は、ウェーハには x-y 断面の 2D メッシュまたは 2D トーラスが複数配置される。この場合 3D メッシュでは、1 辺の PE 数に等しい本数で断面間の結合が行なわれる。3D トーラスではトーラスリンクのため、行長の 2 倍に等しい本数で断面間の結合が行なわれる。したがって 3D メッシュと 3D トーラスの t_x, t_y は以下のようになる。

3 次元メッシュのトーラスリンクの最大本数：

$$t_x = t_y = \begin{cases} 0 & m \geq N^{\frac{2}{3}}, \\ N^{\frac{1}{3}} & m < N^{\frac{2}{3}} \end{cases} \tag{2.29}$$

3 次元トーラスのトーラスリンクの最大本数：

$$t_x = t_y = \begin{cases} 1 & m \geq N^{\frac{2}{3}}, \\ 2N^{\frac{1}{3}} + 1 & m < N^{\frac{2}{3}} \end{cases} \tag{2.30}$$



(a) PEs per wafre:m=16

(b) PEs per wafer:m=4

図 2.17: 3D メッシュのウェーハ間最大結線数 C_{max}(N=64,h=16)

2.7 TESH

2.7.1 ネットワーク構造

TESH(Tori connectd mESHes)[5] は、ウェーハスタック構造を考慮した階層型ネットワークで、基本モジュール (BM) と呼ばれる 4×4 の 2D メッシュの間を 4×4 の 2D トーラスで結合して階層化した構造をもつ。

レベルは階層の最下層にあたる基本モジュールをレベル 1 として、階層化を行なう毎に 1 ずつ増加するものとする。基本モジュールは、全レベルへのリンクの入出力位置があらかじめ決定されている (図 2.18)。各レベルに対して行方向のリンク 2 本、列方向のリンク 2 本、計リンク 4 本が割り当てられている。この 4 本のリンクを用いて、基本モジュール間を 4×4 の 2D トーラスで結合し、階層化を行なうことができる。レベル 2 階層は、16 個の基本モジュールの間を 4×4 の 2D トーラスで結合して階層化を行なう (図 2.19)。レベル 3 階層も 16 個のレベル 2 を 4×4 の 2D トーラスで結合して階層化を行なう。しかし、レベル 3 階層はサブネットに多数の基本モジュールを含むため、2D トーラス結合する基本モジュールの選択が必要となる。TESH では、各レベルで位置が等しい基本モジュール間を 2D トーラス結合をする。つまりレベル 2 階層の中で左下の基本モジュールは、他のレベル 2 階層の左下の基本モジュールと結合を行なう。そのため、レベル 3 階層ではレベル 2 階層に含まれる BM 数 (16 個) の 2D トーラスで並列に階層間の結合を行なう。同様にレベル 3 以降もサブネット内のアドレスが等しい基本モジュール同士を結合する。

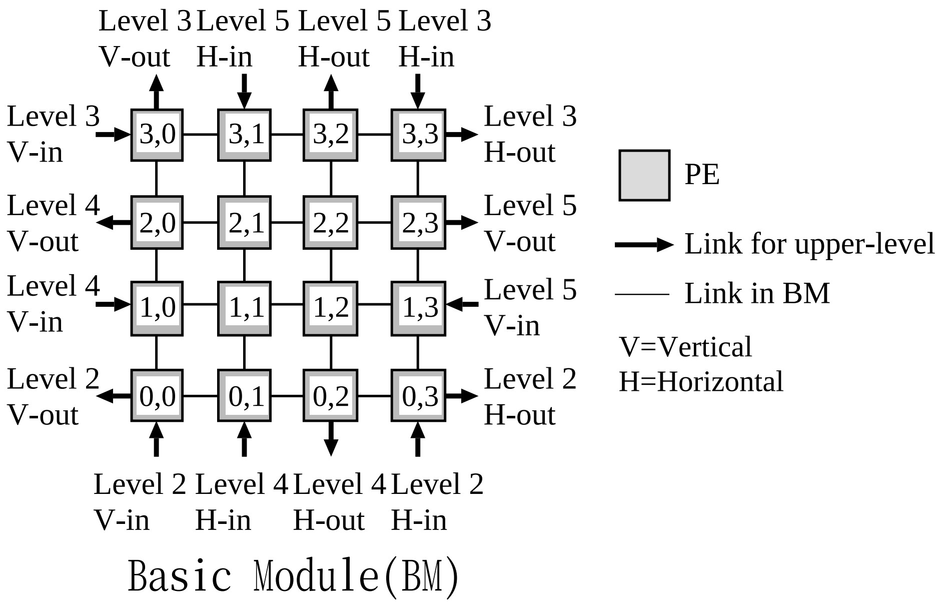


図 2.18: TESH(基本モジュール)

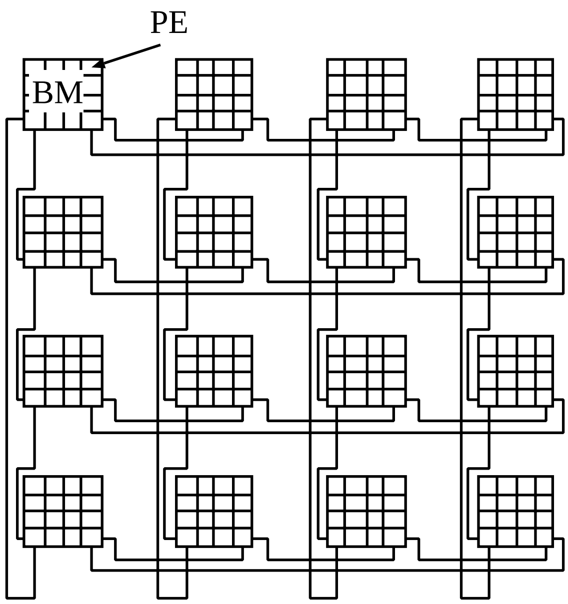


図 2.19: TESH(Level-2)

2.7.2 ネットワーク特性

TESH は各階層を、4×4 の 2D トーラス (BM は 2D メッシュ) で結合していく。そのため、4 進数を用いたアドレス付けを用いている。基本モジュールのアドレスは 0〜3 の 2 つの整数で表す。最初の数字は行アドレス、次の数字は列アドレスを示す。レベル L 階層のアドレスは以下のようになる。ただし a は 0 から 3 の整数を取るものとする。

$$
\begin{aligned}
A &= a_{n-1}a_{n-2}a_{n-3}a_{n-4}\ldots a_3a_2a_1a_0 \\
&= a_{2L-1}a_{2L-2}a_{2L-3}a_{2L-4}\ldots a_3a_2a_1a_0 \\
&= (a_{2L-1}a_{2L-2})(a_{2L-3}a_{2L-4})\ldots(a_3a_2)(a_1a_0) \qquad (2.31)
\end{aligned}
$$

最初のアドレスの組はレベル 1(BM) 内の PE のアドレスを示し、L 番目の組はレベル L を構成するレベル L-1 のサブネットワークのアドレスを表す。

TESH のルーティングは上位レベルから下位レベルの順序で行なう。TESH では、階層間の結合に 2D トーラスを用いていることを利用して、階層間はパケットの移動を単方向に制限している。しかし 2D メッシュを用いている BM 内は双方向のパケット移動を行なう。

例えばレベル 2 の TESH(N=256) において、$P_{(0,0)(1,2)}$から $P_{(0,1)(0,2)}$へののメッセージの移動経路は、以下のように行なわれる。

$(0,0)(1,2) \longrightarrow (0,0)(0,2) \longrightarrow (0,0)(0,3)$　ゲート PE へ移動終了

$\downarrow$　基本モジュール間移動

$(0,1)(0,3) \longrightarrow (0,1)(0,2)$　移動終了

ただし、TESH の直径に関しては他の双方向通信を行なうネットワークと比較をするために、TESH の階層間の結線を双方向通信と仮定して求めた。

表 2.2: TESH の直径

16PE	256PE	4096PE	65536PE	1048576PE
6	19	32	43	53

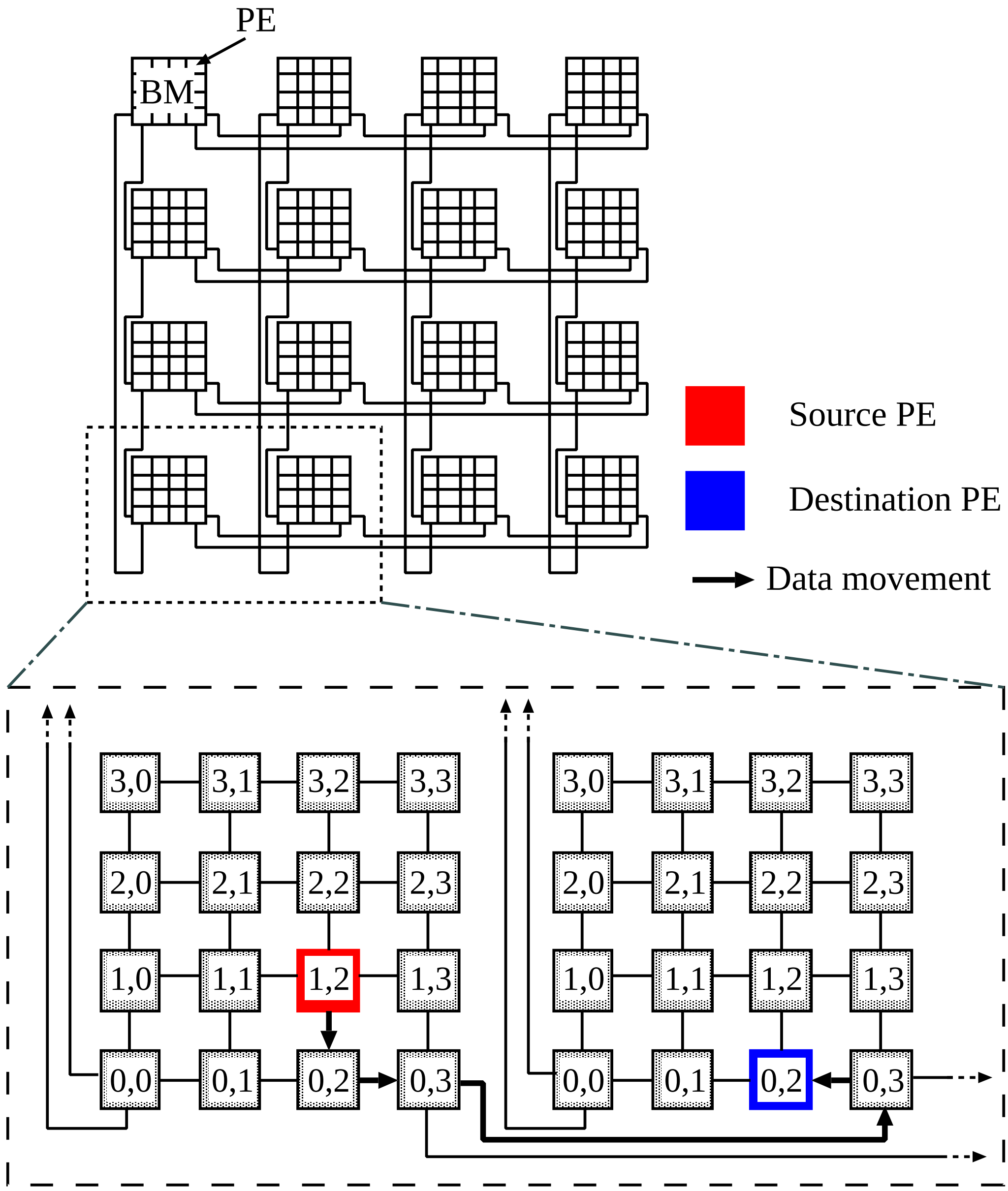


図 2.20: TESH のルーティング (N=256)

2.7.3　ウェーハ間最大結線数

TESH のウェーハスタック構造へのマッピングは以下のように行なう。ただし、基本モジュールの PE 数は 16 と少ないため複数ウェーハへの分割は考慮しない。

1　レベル L_dをウェーハに s 個載せる。TESH は階層間を 2D トーラスで結合しているため、配置は各レベル毎に (2.8) 式のインラインで行なう。

2　レベル L の配置が終るまで 1 を繰り返す。

TESH はサブネットを越えて上位レベルへのリンクが行なわれるため、ウェーハ間最大結線数は各レベルのウェーハ間最大結線数の和となる。TESH はサブネット内の基本モジュール数に等しい 2D トーラスで階層間の結合を行なうため、レベル L_iのみを考慮したウェーハ間最大結線数は以下のようになる。レベル L_i階層のウェーハ 1 枚あたりのサブネット数を s_{L_i}で表す。ただし、$s_{L_i} < 1$ の場合は $s_{L_i} = 1$ とする。

$$
\begin{aligned}
C_{max}^{L_i} &= C_{max}(2Dtorus, 16, s_{L_i})16^{L_i-2} \\
&= \begin{cases} 10(16^{L_i-2}) & s_{L_i} = 1, \\ 8(16^{L_i-2}) & s_{L_i} = 4 \end{cases}
\end{aligned}
\tag{2.32}
$$

したがって TESH のウェーハ間最大結線数は以下のようになる。

$$
\begin{aligned}
C_{max} &= C_{max}(2Dtorus, 16, s)16^{L_d-2} + \sum_{L_i=L_d+1}^{L} C_{max}(2Dtorus, 16, 1)16^{L_i-2} \\
&= C_{max}(2Dtorus, 16, s)16^{L_d-2} + \sum_{L_i=L_d+1}^{L} 10(16^{L_i-2})
\end{aligned}
\tag{2.33}
$$

2.7.4　レイアウト面積

ここではレイアウト面積を求めるために必要なトーラスリンクの最大本数を求める。TESH のトーラスリンクの最大本数は、1 枚のウェーハに載る階層レベルによって決まる。1 つの基本モジュールのみがウェーハに載っている場合は、$t_x = t_y = 0$ である。4 つの基本モジュールがウェーハに載っている場合は、レベル 2 階層を構成するリンクがウェーハ内に含まれるので $t_x = t_y = 2$ となる。レベル 3 以降ではサブネット内の基本モジュール数に等しい 2 次元トーラスでサブネット間の結合が行なわれるため、トーラスリンク本数

は 1 階層毎にサブネットの行 (列) あたりの基本モジュール数だけ増加する。したがって t_x, t_yは以下のようになる。ここで L_eは 2 つ以上のサブネットが載っている最高階層レベルである。

$$L_e = \lceil \log_{16} m \rceil \tag{2.34}$$

$$t_x = t_y = 2 \sum_{i=Le}^{L} (4^{i-2}) \tag{2.35}$$

2.8　ウェーハスタック構造に適したネットワークの検討

ウェーハスタック構造に適したネットワークの特性を検討するため、低次数の 2D トーラスネットワーク (2Dtorus)、3 次元構造を持つ 3D トーラスネットワーク (3Dtorus)、高次数のハイパーキューブ (Hyper-Cube)、ウェーハスタック構造を考慮した階層型ネットワーク TESH の特性の比較を行なう。

図 2.21にはネットワーク直径、図 2.22にはウェーハ間最大結線数、そして図 2.8にはトーラスを 1.0 とした場合の各ネットワークのレイアウト面積比を示す。ウェーハ間最大結線数は、ウェーハ枚数を 16 枚に固定して評価を行なった。面積比は、総 PE 数:4096PE、1 ウェーハに載せるネットワーク規模:$16PE \times 16PE (= 256PE)$、ウェーハ枚数:16 枚、PE の幅:$1mm$(medium grain)、ウェーハ上の結線:$1\mu m$ CMOS technology、マイクロブリッジの幅:$500\mu m$ という条件を用いた。

ハイパーキューブは、直径が小さいが各 PE の次数が高いためウェーハ間結線数が非常に多く、ネットワーク面積も非常に大きくなるため、ウェーハスタック構造への実装は困難である。一方、2D トーラスはウェーハ間最大結線、ネットワーク面積の増加は小さいが、ネットワーク直径の増加が大きく、PE 数が増加した場合、通信遅延が非常に大きくなる。TESH および 3D トーラスはネットワーク直径、ウェーハ間の最大結線数を比較的少なくできる。特に TESH のような階層構造を有するネットワークは、少ない階層間のリンクを用いてウェーハ間結線数を減らしつつ、直径の増加を抑えることができるため、ウェーハスタック構造に最も適している。

以上の結果から、ウェーハスタック構造には低次元ネットワーク、3D トーラスのような 3 次元ネットワークおよび TESH のような階層型ネットワークが適していることが分かった。

2.9　まとめ

本章ではウェーハスタック構造に適したネットワークの検討を行なうため既存のネットワークを用いてネットワーク特性を調べた。ネットワーク特性として、直径の他に、ウェーハ間最大結線数、レイアウト面積を用いた。ウェーハ間最大結線数を求めるため、ウェーハスタック構造へのネットワーク配置方法の考察を行なった。またウェーハ間結線を用いたレイアウト面積の評価手法の検討を行なった。以上のネットワーク特性を用いてウェーハスタック構造に適したネットワークの検討を行なった。

次章ではこの結果を用いて、ウェーハスタック構造に適したネットワークの提案を行なう。

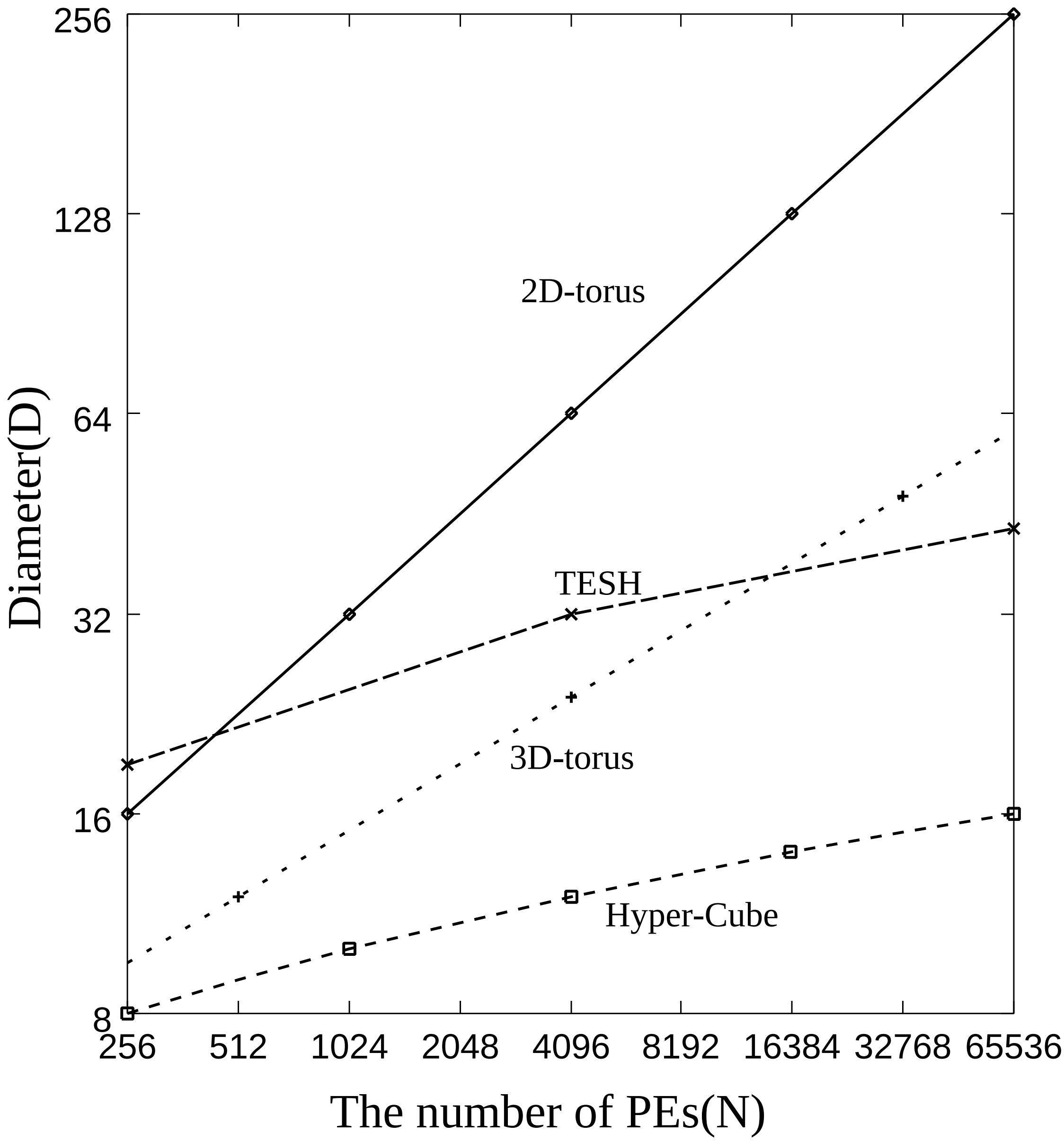

256
128
64
32
16
8
256
512
1024
2048
4096
8192
16384
32768
65536
Diameter(D)
The number of PEs(N)
2D-torus
TESH
3D-torus
Hyper-Cube

図 2.21: 直径 (D)

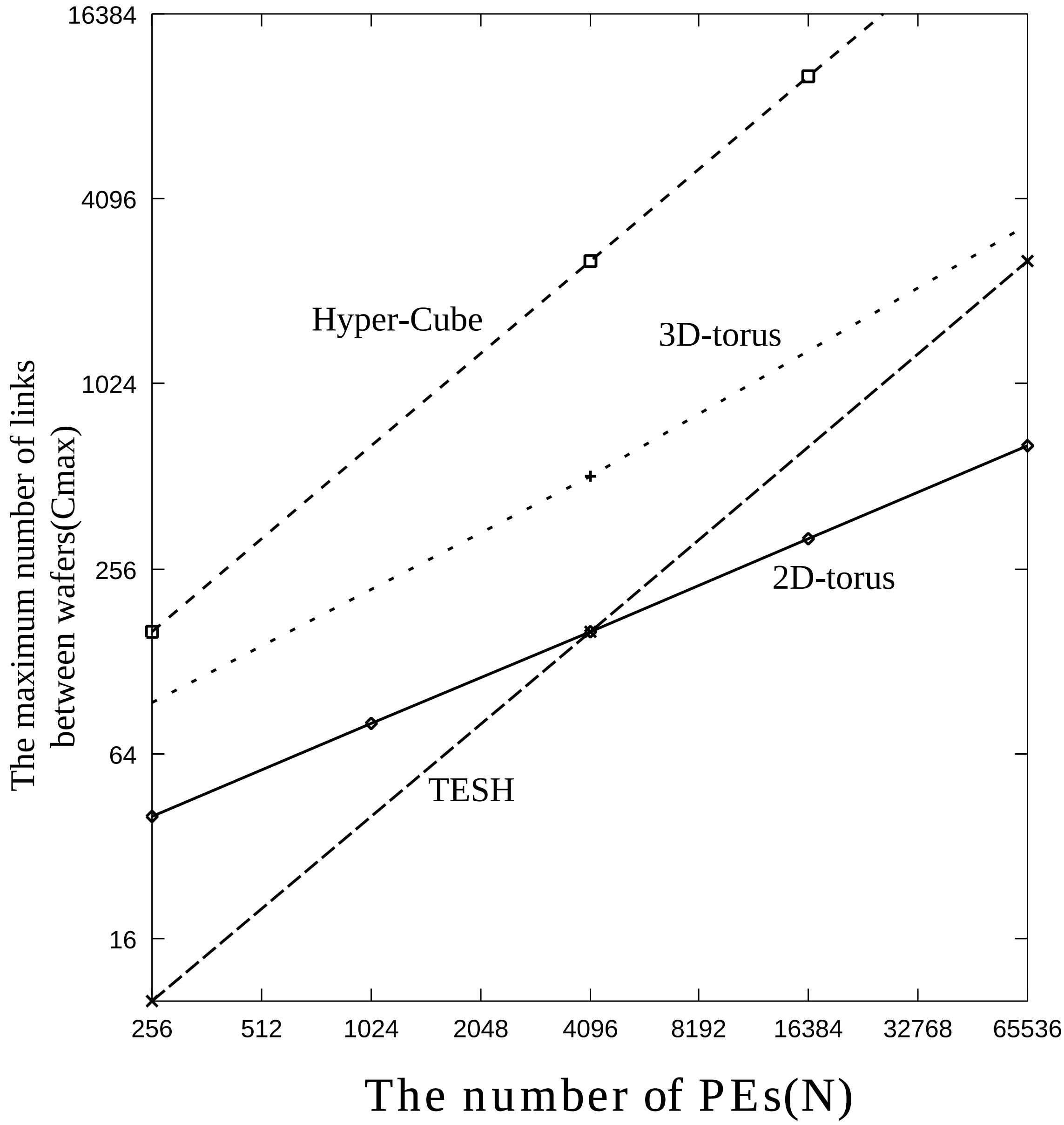


図 2.22: ウェーハ間最大結線数 (C_{max})

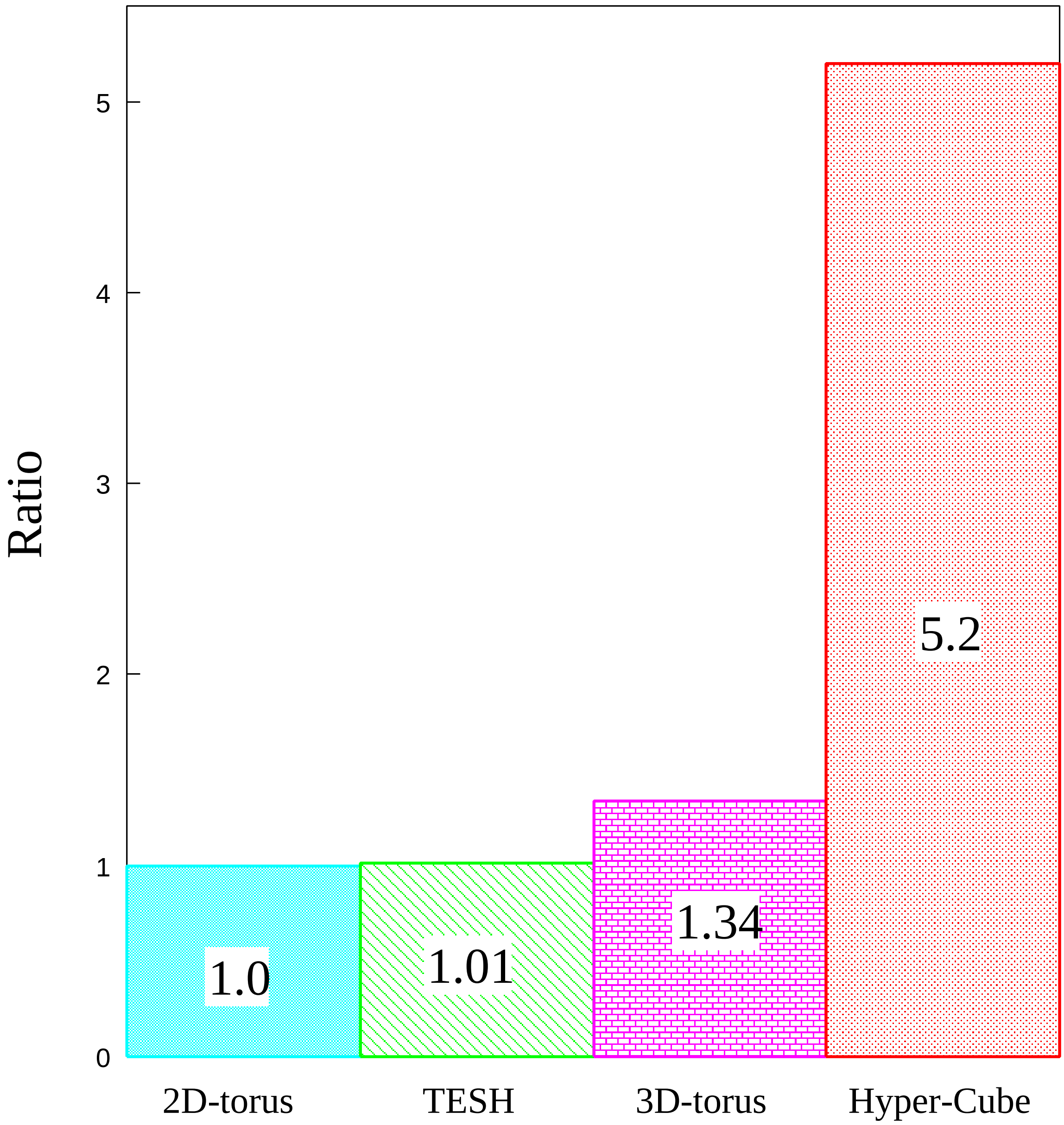


図 2.23: レイアウト面積 A(N=4096)

第 3 章

3 次元階層型ネットワーク

3.1 はじめに

ウェーハスタック構造は、大規模超並列計算機システムの構築を可能にする WSI の手法である。しかし、ウェーハ間の結合に用いるマイクロブリッジには数百μm の幅が必要である。このため、ウェーハ間結線は数μm で実現可能なウェーハ内のリンクのと比較して、非常に大きな面積を必要とする。そのため、ウェーハ間結線を減少させる必要がある。しかしウェーハ間結線を減少させることによって直径が増加すると、通信遅延の増大によって超並列計算機システムの性能が大きく制限される。そのためウェーハスタック構造では、直径の増加を抑えつつ、ウェーハ間結線を減少させる必要がある。本章では、これらの要求を満たすネットワークとして 3 次元階層型ネットワークの提案を行なう。

3 次元階層型ネットワークは、基本モジュールを用いて階層結合したネットワークで、直径の増加を抑えつつ、ウェーハ間結線数を減少させることが可能である。本論文では、3 次元階層型ネットワークとして、3 次元階層型トーラスと 3 次元階層型メッシュの 2 つのネットワークを提案する。この 2 つのネットワークは、基本モジュールを用いた結合を行なうという点は等しいが基本モジュールと階層間の結合に用いるネットワークが異なる。3 次元階層型トーラスは、特に階層間の結線を抑えたネットワークで、階層化の規模も大きく、大規模な超並列ネットワークを目的としている。基本モジュールには $4 \times 4 \times 4$ の 3D メッシュを用い、階層間の結合には 3D トーラスを用いている。3 次元階層型メッシュは、PE 間通信の際に基本モジュール間のリンクに起こる通信の集中を低減させるため、基本モジュール間のリンク数を増加させたネットワークである。基本モジュールには

$4 \times 4 \times 4$ の 3D トーラスを用い、階層間の結合には 4×4 の 2D メッシュを用いているため、階層化の規模は小さい。

本章では 3 次元階層型ネットワークの提案とネットワーク特性の検討を行なう。更に、第 2章で検討したネットワークとの比較を行ない、その有効性を議論する。

3.2 3 次元階層型トーラス結合

3.2.1 ネットワーク構造

3 次元階層型トーラスは基本モジュール (3D メッシュ) 間を 3D トーラスで結合して階層化を行なったネットワークである (図 3.2.1)。TESH と同じように基本モジュールをレベル 1 として、以降、階層化を行なう毎にレベルを 1 ずつ増加させる。

基本モジュール

基本モジュールは、図 3.2.1に示すように $4 \times 4 \times 4$ の 3 次元メッシュである。基本モジュールのアドレスは 4 進数を用いて以下のように表す。

$$A = (a_z)(a_y)(a_x) \quad (a_z, a_y, a_x = 0, 1, 2, 3) \tag{3.1}$$

また、アドレス $(a_z)(a_y)(a_x)$ の PE を PE(a_z, a_y, a_x) で表す。基本モジュールは全階層レベルへのリンクを持つ。階層間のリンクを行なうゲート PE には $a_y = a_x = 0, 3$ を用いる。この 4 つの PE を各レベルに反時計回りに割り当てる。レベル 2 階層へのゲート PE を PE(a_z,0,0) とすると、レベル 3 階層へのゲート PE は PE(a_z,0,3)、レベル 4 階層へのゲート PE は PE(a_z,3,3)、レベル 5 階層へのゲート PE は PE(a_z,3,0) となる。また階層レベル毎に $a_z = 0, 1, 2$ の 3 つのゲート PE を割り当て、$a_z = 0$ を z 軸方向のリンク、$a_z = 1$ を y 軸方向のリンク、$a_z = 2$ を x 軸方向のリンクに用いる。各ゲート PE から出るリンクは 2 本とする。これによって、各レベルには 3 つのゲート PE から z,y,x 軸にそれぞれ 2 本のリンクが割り当てられる。これによって、階層間を 3 次元トーラスで結合できる。この結合方式の利点は階層間の移動の際に、同階層の同軸上ならば基本モジュール内部での移動量は 0 になることである。また軸の移動の際にも基本モジュール内での移動量は 1 で良い。

レベル 2〜5 階層

3 次元階層型トーラスは階層間を $4 \times 4 \times 4$ の 3 次元トーラスで結合する。レベル L 階層

における総 PE 数は以下のようになる。

$$N = 64^L \tag{3.2}$$

3 次元階層型トーラスのレベル L_i階層のアドレスは基本モジュールのように 4 進数を用いて以下のように表す。

$$A^{L_i} = (a_z^{L_i})(a_y^{L_i})(a_x^{L_i}) \quad (a_z^{L_i}, a_y^{L_i}, a_x^{L_i} = 0, 1, 2, 3) \tag{3.3}$$

レベル 2 階層は基本モジュール間を $4 \times 4 \times 4$ の 3 次元トーラスで結合したネットワークである。同様にレベル 3 階層は、レベル 2 階層間を $4 \times 4 \times 4$ の 3 次元トーラスで結合したネットワークである。ただしレベル 3 階層のサブネットワークであるレベル 2 階層には、複数の基本モジュールが含まれるため、結合する基本モジュールを選択す必要がある。3 次元階層型トーラスでは TESH と同じようにサブネットワーク内のアドレスが等しい基本モジュールの結合を行なう。つまりレベル 3 ではレベル 2 の基本モジュールの数に等しい 64 個の 3D トーラスで階層間の結合が行なわれる。同様にレベル 3 以降もサブネットワーク内のアドレスが等しい基本モジュール同士を 3 次元トーラスで結ぶ。

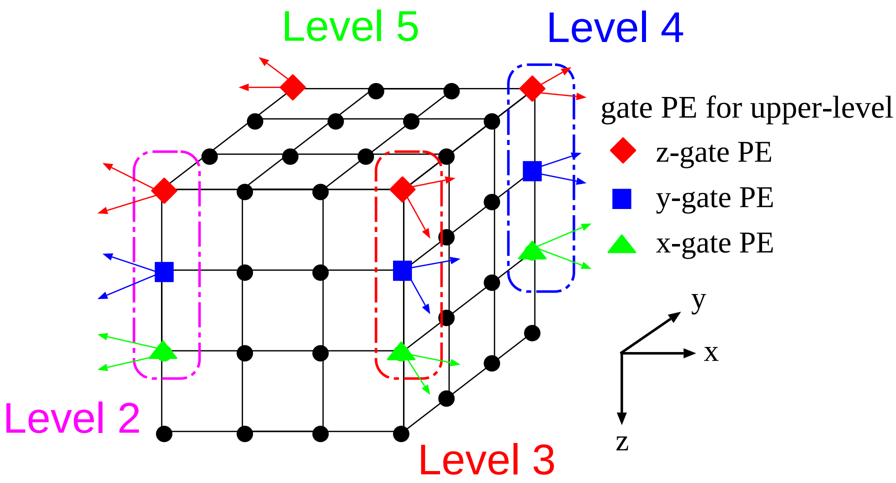


図 3.1: 3D 階層型トーラスの基本モジュール

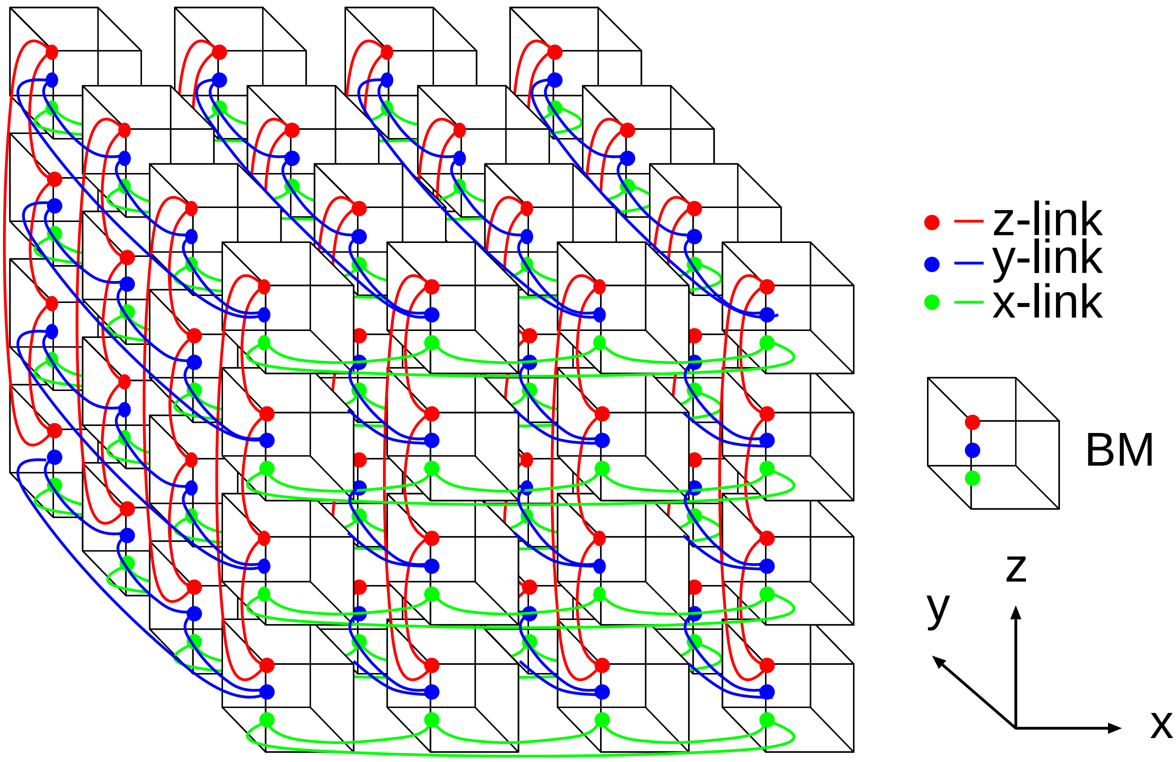


図 3.2: 3D 階層型トーラスのレベル 2 階層結合

3.2.2 ネットワーク特性

ここでは3次元階層型トーラスのルーティングを示し、直径を求める。3次元階層型トーラスのルーティングは上位レベルから下位レベルの順序で行なう。各レベルのルーティングはz軸、y軸、x軸の順序で行なう。3次元階層型トーラスのルーティングはソースアドレスと目的アドレスのみで一意に決定できる。3次元階層型トーラスのルーティングを以下に示す。ここでソースアドレスを $s_{n-1}s_{n-2}\dots s_1s_0$、目的アドレスを $d_{n-1}d_{n-2}\dots d_1d_0$ とする。またルーティングに用いるタグを以下のよう設定する。

$$t_i = d_i - s_i \tag{3.4}$$

Routing Algorithm for Level-L 3D-htorus Network：

Routing();

source node address：$s_{n-1}, s_{n-2}, \dots, s_0$　destination node address：$d_{n-1}, d_{n-2}, \dots, d_0$

tag：$t_{n-1}, t_{n-2}, \dots, t_0$

for i=n-1:3;

if ((t_i> 0 and 2 $\geq t_i$) or (t_i < 0 and $t_i = -3$)),movedir=positive;endif;

if ((t_i> 0 and $t_i = 3$) or (t_i < 0 and $t_i = -1$)),movedir=negative;endif;

if (movedir=positive and t_i> 0),distance=t_i;endif;

if (movedir=positive and t_i< 0),distance=4+t_i;endif;

if movedir=negative,distance=1;endif;

j=i mod 3;

while($t_i \neq 0$ *or distance* $\neq 0$)do

if j=2,gate_node=z-axis gate_node of Level-⌈i/3⌉ ; endif;

if j=1,gate_node=y-axis gate_node of Level-⌈i/3⌉ ; endif;

if j=0,gate_node=x-axis gate_node of Level- i/3+1; endif;

send packet to next BM;destance=distance−1;

endwhile;

endfor;

BM_tag{t_2, t_1, t_0}=receiveing node address−destinaton(d_2, d_1, d_0)

while($t_2 \neq 0$) do

if t_2 > 0, move packet to +z node;$t_2 = t_2 - 1$;endif;

if $t_2 < 0$, move packet to $-z$ node;$t_2 = t_2 + 1$;endif;

endwhile;

while($t_1 \neq 0$) do

if $t_1 > 0$, move packet to +y node;$t_1 = t_1 - 1$;endif;

if $t_1 < 0$, move packet to $-y$ node;$t_1 = t_1 + 1$;endif;

endwhile;

while($t_0 \neq 0$) do

if $t_0 > 0$, move packet to +x node;$t_0 = t_0 - 1$;endif;

if $t_0 < 0$, move packet to $-x$ node;$t_0 = t_0 + 1$;endif;

endwhile;

end;

例えば $P_{(123)(211)}$から $P_{(333)(111)}$へのルーティングは以下のように行なわれる。まず、レベル 2 の z 軸方向のゲート PE である (123000) に移動する。次に z 軸のアドレスが一致するまで移動する。(323000) に達したら、y 軸方向のルーティングを行なうためレベル 2 の y 軸方向のゲート PE である (323100) に移動する。(333100) に達したら基本モジュール内のルーティングを行なうことでルーティングが終了する。この 3 次元階層型トーラスのレベル L における直径は以下のようになる。

$$\begin{aligned} D &= D_{BM}^{to\ z\text{-}gate\text{-}L} + (L-1)(D_{3D\text{-}torus} + D_{BM}^{axis\text{-}move}) \\ &\quad +(L-2)D_{BM}^{level\text{-}move} + D_{BM}^{to\ x\text{-}gate\text{-}2} \end{aligned} \tag{3.5}$$

表 3.1: 3 次元階層型トーラスの各階層での直径

$D_{BM}^{to\ z\text{-}gate\text{-}L}$	9	基本モジュール内でレベル L の z 軸方向ゲート PE までの直径
$D_{3D\text{-}torus}$	6	$4 \times 4 \times 4$ の 3D トーラスの直径
$D_{BM}^{axis\text{-}move}$	2	レベル L_iの z 軸方向ゲート PE から L_iの x 軸方向ゲート PE までの直径
$D_{BM}^{level\text{-}move}$	5	レベル L_iの x 軸方向ゲート PE から $L_i - 1$ の z 軸方向ゲート PE までの直径
$D_{BM}^{to\ x\text{-}gate\text{-}2}$	8	基本モジュール内でレベル 2 の x 軸方向ゲート PE までの直径

表 3.2: 3 次元階層型トーラスの直径

レベル L	2	3	4	5
総 PE 数 N	4096	262144	16777216	10^9
直径 D	25	38	51	64

3.2.3　ウェーハ間最大結線数

3 次元階層型トーラスのウェーハスタック構造へのマッピングは以下のように行なう。この際、基本モジュールの PE 数は 64 と少ないため複数ウェーハへの分割は考慮しない。また表記に用いる記号は第 2.3節で定義したものを用いる。

1　レベル L_dをウェーハに s 個載せる。3 次元階層型トーラスは階層間を 3D トーラスで結合しているため、配置は各レベル毎に (2.8) 式のインラインで行なう。

2　L≥3 の場合は、各レベルをインラインに並べ換えた後、以降に示すようにアドレスをレベルの逆順でウェーハに配置する。

$$A = (a_z^1 a_y^1 a_x^1)(a_z^2 a_y^2 a_x^2)(a_z^3 a_y^3 a_x^3)(a_z^4 a_y^4 a_x^4)(a_z^5 a_y^5 a_x^5)$$

3　レベル L の配置が終るまで 1 を繰り返す。

3 次元階層型トーラスはサブネットにまたがるように上位レベルへのリンクが行なわれるため、レベル L のウェーハ間最大結線数はレベル L 以下の各レベルのウェーハ間最大結線数の和となる。つまりレベル L_iを 1 ウェーハに 1 サブネットを載せた場合のウェーハ間最大結線数の和となる。3 次元階層型トーラスはサブネット内の基本モジュール数に等しい 3D トーラスで階層間の結合が行なわれるため、サブネットに含む BM 数が最も多い最高レベルのウェーハ間結線数が最も多い。3 次元階層型トーラスでは、各階層へのリンクの入出力が同じ PE から行なう。このためアドレスをレベルの逆順にすることによって、上位階層と下位階層を入れ換えることができる。この配置を用いることによって基本モジュールが上位階層の結合を構成し、レベル 5 が最下位を構成ようになる。3 次元階層型トーラスでは、基本モジュールは 3 次元メッシュを用いているため、ウェーハ間結線数が 3

次元トーラスの半数となる。この配置方法ではレベル 3 階層以降に用いることで、ウェーハ間結線数を減らすことができる。したがって、レベル $L_i(L_i \neq L\ and\ L \neq 2\ or\ L = 2)$ のみを考慮したウェーハ間最大結線数は以下のようになる。

$$\begin{aligned} C_{max}^{L_i} &= C_{max}(3Dtorus, 64, s_{L_i})64^{L_i-2} \\ &= \begin{cases} 42(64^{L_i-2}) & s_{L_i} = 1, \\ 40(64^{L_i-2}) & s_{L_i} = 4, \\ 32(64^{L_i-2}) & s_{L_i} = 16 \end{cases} \end{aligned} \tag{3.6}$$

またレベル $L(L \geq 3\ and\ L \neq 2)$ のみを考慮したウェーハ間最大結線数は以下のようになる。

$$\begin{aligned} C_{max}^{L_i} &= C_{max}(3Dmesh, 64, s_{L_i})64^{L_i-2} \\ &= \begin{cases} 21(64^{L_i-2}) & s_{L_i} = 1, \\ 20(64^{L_i-2}) & s_{L_i} = 4, \\ 16(64^{L_i-2}) & s_{L_i} = 16 \end{cases} \end{aligned} \tag{3.7}$$

またレベル $L(L \geq 3\ and\ L \neq 2)$ のみを考慮したウェーハ間最大結線数は以下のようになる。ここでレベル L_i階層のウェーハ 1 枚あたりのサブネット数を s_{L_i} で表す。ただし s_{L_i} < 0 の場合は $s_{L_i} = 1$ とする。

L=2 の 3 次元階層型トーラスのウェーハ間最大結線数は以下のようになる。

$$C_{max} = C_{max}(3Dtorus, 64, s) \tag{3.8}$$

$L \geq 3$ の 3 次元階層型トーラスのウェーハ間最大結線数は以下のようになる。

$$\begin{aligned} C_{max} &= C_{max}(3Dmesh, 64, s)64^{L_d-2} + \sum_{L_i=L_d+1}^{L-1} C_{max}(3Dtorus, 64, 1)64^{L_i-2} \\ &\quad + C_{max}(3Dmesh, 64, 1)64^{L-2} \\ &= C_{max}(3Dmesh, 64, s)64^{L_d-2} + \sum_{L_i=L_d+1}^{L-1} 42(64^{L_i-2}) + 21(64^{L-2}) \end{aligned} \tag{3.9}$$

3.2.4 レイアウト面積

本研究では、レイアウト面積をウェーハ間最大結線数と行 (列) 当りのトーラスリンクの最大本数を用いて求める。ここでは、1 枚のウェーハに載るネットワークの最大トーラスリンク本数を求める。

3 次元階層型トーラスの最大トーラスリンク本数 t_x, t_yは、1 枚のウェーハに載る階層レベルによって決まる。1 つの基本モジュールがウェーハに載っている場合は、$t_x = t_y = 4$ である。4 つの基本モジュールがウェーハに載っている場合は、レベル 2 階層を構成するトーラスリンクがウェーハ内に含まれるので $t_x = t_y = 4 + 2 = 6$ となる。レベル 3 以降ではサブネット内の基本モジュール数に等しい 2 次元トーラスでサブネット間の結合が行なわれるため、トーラスリンク本数は 1 階層毎にサブネットの行 (列) あたりの基本モジュール数だけ増加する。したがって t_x, t_yは以下のようになる。

$$L_e = \lceil \log_{64} m \rceil \quad (3.10)$$

$$t_x = t_y = 4 + 2 \sum_{i=L_e}^{L} 4^{i-2} \quad (3.11)$$

ここで L_eは 2 つ以上のサブネットが載っている最高階層レベルである。

3.3 3 次元階層型メッシュ結合

3.3.1 ネットワーク構造

3 次元階層型メッシュは基本モジュール (3D トーラス) 間を 2D メッシュで結合して階層化を行なったネットワークである (図 3.3.1)。TESH と同じように基本モジュールをレベル 1 として、以降、階層化を行なう毎にレベルを 1 ずつ増加させる。

基本モジュール

基本モジュールは、図 3.3.1に示すように $4 \times 4 \times 4$ の 3 次元トーラスである。基本モジュールのアドレスは 4 進数を用いて以下のように表す。

$$A = (a_z)(a_y)(a_x) \quad (a_z, a_y, a_x = 0, 1, 2, 3) \quad (3.12)$$

また、アドレス $(a_z)(a_y)(a_x)$ の PE を PE(a_z, a_y, a_x) で表す。基本モジュールは全階層レベルへのリンクを持つ。3 次元階層型メッシュでは 1 つの xy 断面の外周の PE を 1 つの階層に割り当て、ゲート PE に用いる。$a_z = 0$ の断面のゲート PE をレベル 2、$a_z = 1$ の断面のゲート PE をレベル 3、$a_z = 2$ の断面のゲート PE をレベル 4、$a_z = 3$ の断面のゲート PE をレベル 5 に割り当てる。各断面のゲート PE は図 3.3.2のように N,S,W,E の 4 方向に各 4 本のリンクを持つ。

$$G = \{00_S, 01_S, 02_S, 03_S; 00_W, 10_W, 20_W, 30_W; 30_N, 31_N, 32_N, 33_N; 03_E, 13_E, 23_E, 33_E\}$$

階層間の結合は、N 方向と S 方向のリンクを用いて y 軸方向の結合を行ない、W 方向と E 方向のリンクを用いて x 軸方向の結合を行なう。これによって、階層間のを 2D メッシュで結合できる。

レベル 2〜5 階層

3 次元階層型メッシュは階層間を 4 × 4 の 2D メッシュで結合する。レベル L 階層における総 PE 数は以下のようになる。

$$N = 64(16^{L-1}) \tag{3.13}$$

3 次元階層型メッシュのレベル L_i階層のアドレスは基本モジュールのように 4 進数を用いて以下のように表す。

$$A^{L_i} = (a_y^{L_i})(a_x^{L_i}) \;\; (a_y^{L_i}, a_x^{L_i} = 0, 1, 2, 3) \tag{3.14}$$

レベル 2 階層は基本モジュール間を 4×4 の 2D メッシュで結合したネットワークである。結合は$\{00_S, 01_S, , 02_S, 03_S\}$ と $\{30_N, 31_N, 32_N, 33_N\}$、$\{00_W, 10_W, 20_W, 30_W\}$ と $\{03_E, 13_E, 23_E, 33_E\}$ の間で行なう。同様にレベル 3 階層は、レベル 2 階層間を 4 × 4 の 2D メッシュで結合したネットワークである。ただしレベル 3 階層のサブネットワークであるレベル 2 階層には、複数の基本モジュールが含まれるため、結合する基本モジュールを選択す必要がある。3 次元階層型メッシュでも TESH や 3 次元階層型トーラスと同じようにサブネットワーク内のアドレスが等しい基本モジュールの結合を行なう。つまりレベル 3 ではレベル 2 の基本モジュールの数に等しい 16 個の 2D メッシュで階層間の結合が行なわれる。同様にレベル 3 以降もサブネットワーク内のアドレスが等しい基本モジュール同士を 2D メッシュで結合する。

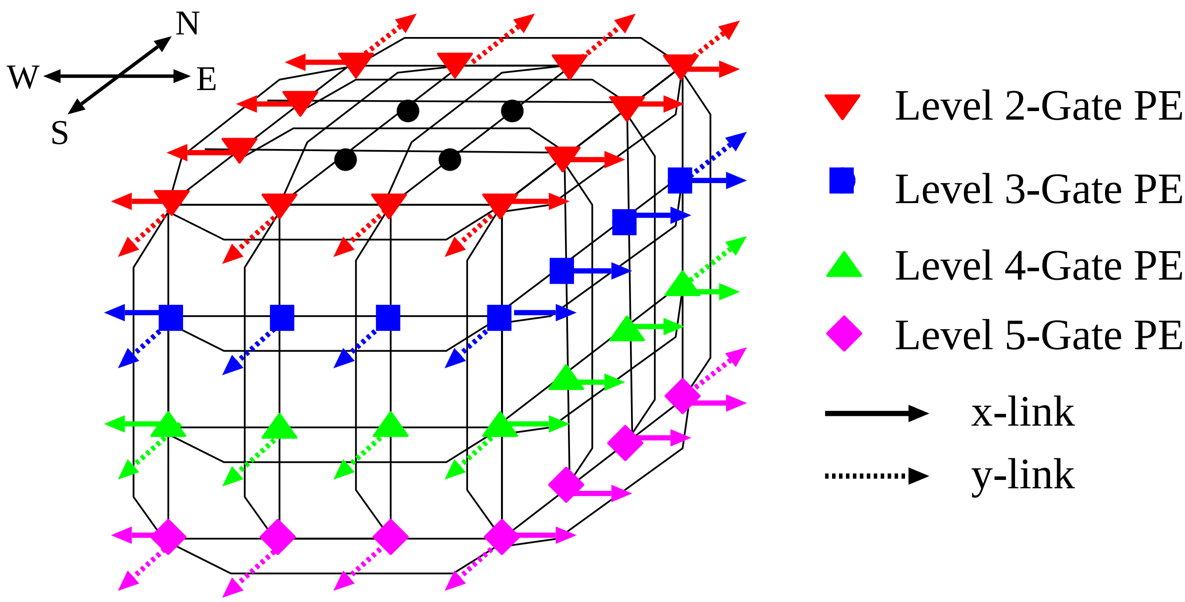


図 3.3: 3D 階層型メッシュの基本モジュール

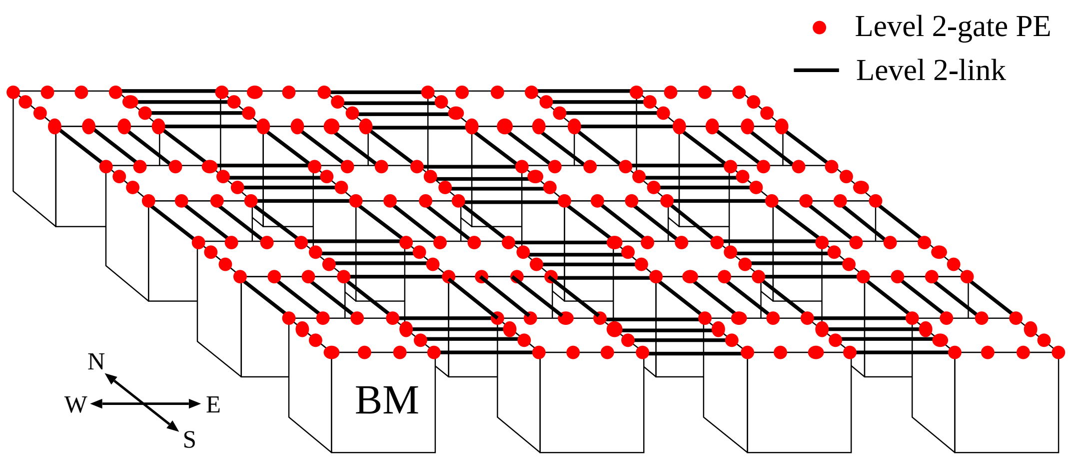


図 3.4: 3D 階層型メッシュのレベル 2 階層結合

3.3.2　ネットワーク特性

ここでは 3 次元階層型メッシュのルーティングを示し、直径を求める。3 次元階層型メッシュのルーティングは上位レベルから下位レベルの順序で行なう。レベル 2 以上のルーティングは y 軸、x 軸の順序で行ない、基本モジュールのルーティングは z 軸、y 軸、x 軸の順序で行なう。3 次元階層型メッシュは基本モジュール間を 4 本のリンクで結合する。軸移動を行なう際は、図 3.3.2に示すように $\{00_S, 30_W, 33_N, 03_E\}$,$\{01_S, 20_W, 32_N, 13_E\}$, $\{02_S, 10_W, 31_N, 23_E\}$,
$\{03_S, 00_W, 30_N, 33_E\}$ の組にした 4 つゲート同士で移動を行なう。階層間のパケット送信で移動方向を変える場合のゲート PE の選択を表 3.5に示す。これにより 4 本のリンクを同時に用いた場合に、パケットの衝突を避けることができる。また、3 次元階層型メッシュのルーティングはソースアドレスと目的アドレスのみで一意に決定できる。3 次元階層型トーラスのルーティングを以下に示す。ここでソースアドレスを $s_{n-1}s_{n-2}\ldots s_1s_0$、目的アドレスを $d_{n-1}d_{n-2}\ldots d_1d_0$とする。またルーティングに用いるタグを以下のよう設定する。

$$t_i = d_i - s_i \tag{3.15}$$

Routing Algorithm for Level-L 3D-hmesh Network：

Routing();
source node address：$s_{n-1}, s_{n-2}, \ldots, s_0$　destination node address：$d_{n-1}, d_{n-2}, \ldots, d_0$
tag：$t_{n-1}, t_{n-2}, \ldots, t_0$
　for i=n-1:2;
　　while($t_i \neq 0$) do
　　　if (i is even and $t_i > 0$),gate_node=North gate_node of Level-i/2;endif;
　　　if (i is even and $t_i < 0$),gate_node=South gate_node of Level-i/2;endif;
　　　if (i is odd and $t_i > 0$),gate_node=East gate_node of Level- $\lceil$i/2$\rceil$;endif;
　　　if (i is odd and $t_i < 0$),gate_node=West gate_node of Level- $\lceil$i/2$\rceil$;endif;
　　　send packet to next BM;$t_i = t_i - 1$;
　　endwhile;
　endfor;
　BM_tag$\{t_2, t_1, t_0\}$=receiveing node address -destinaton(d_2, d_1, d_0)
　for i=2:0;

```
    if (t_i > 0 and 2 ≥ t_i) or (t_i < 0 and t_i = −3),movedir=positive;endif;
    if (t_i > 0 and t_i = 3) or (t_i < 0 and t_i = −1),movedir=negative;endif;
    if (movedir=positive and t_i > 0),distance=t_i;endif;
    if (movedir=positive and t_i < 0),distance=4+t_i;endif;
    if movedir=negative,distance=1;endif;
  endfor;
  while(t_2 ≠ 0 or distance_2 ≠ 0) do
    if movedir=positive,move packet to +z node;distance_2 = distance_2 − 1;endif;
    if movedir=negative,move packet to −z node;distance_2 = distance_2 + 1;endif;
  endwhile;
  while(t_1 ≠ 0 or distance_1 ≠ 0) do
    if movedir=positive,move packet to +y node;distance_1 = distance_1 − 1;endif;
    if movedir=negative,move packet to −y node;distance_1 = distance_1 + 1;endif;
  endwhile;
  while(t_0 ≠ 0 or distance_0 ≠ 0) do
    if movedir=positive,move packet to +x node;distance_0 = distance_0 − 1;endif;
    if movedir=negative,move packet to −x node;distance_0 = distance_0 + 1;endif;
  endwhile;
end;
```

例えば $P_{(13)(211)}$ から $P_{(30)(111)}$ へのルーティングは以下のように行なわれる。まず、レベル 2 の N 方向のゲート PE である (13031) に移動する。次に y 軸方向のアドレスが一致するまで移動する。(33001) に達したら、x 軸方向のルーティングを行なうためレベル 2 の W 方向のゲート PE である (33020) に移動する。(30023) に達したら基本モジュール内のルーティングを行なうことでルーティングが終了する。

この 3 次元階層型メッシュのレベル L における直径は以下のようになる。

$$
\begin{aligned}
D \;=\;& D_{BM}^{to\ y\text{-}gate\text{-}L} + (L-1)D_{2D\text{-}mesh} \\
&+(2L-3)D_{BM}^{axis\text{-}move} + (L-2)D_{BM}^{level\text{-}move} \\
&+D_{BM}^{to\ x\text{-}gate\text{-}2} \qquad (3.16)
\end{aligned}
$$

表 3.3: 3 次元階層型メッシュの各階層での直径

$D_{BM}^{to\ z\text{-}gate\text{-}L}$	6	基本モジュール内でレベル L の y 軸方向ゲート PE までの直径
$D_{2D\text{-}mesh}$	6	4 × 4 の 2D メッシュの直径
$D_{BM}^{axis\text{-}move}$	3	$x \Leftrightarrow y$軸方向のゲート PE 間の直径
$D_{BM}^{level\text{-}move}$	5	レベル L_iの x 軸方向ゲート PE からレベル L_iの x 軸方向ゲート PE までの直径
$D_{BM}^{to\ x\text{-}gate\text{-}2}$	6	基本モジュール内でレベル 2 の x 軸方向ゲート PE までの直径

表 3.4: 3 次元階層型メッシュの直径

レベル L	2	3	4	5
総 PE 数 N	1024	16384	262144	4194304
直径 D	28	47	66	85

表 3.5: 3 次元階層型メッシュの移動方向変更の際のゲート選択

y 軸方向のゲート PE		x 軸方向のゲート PE	
$t_i > 0$	$t_i < 0$	$t_i > 0$	$t_i < 0$
33_N	00_S	30_W	03_E
32_N	01_S	20_W	13_E
31_N	02_S	10_W	23_E
30_N	03_S	00_W	33_E

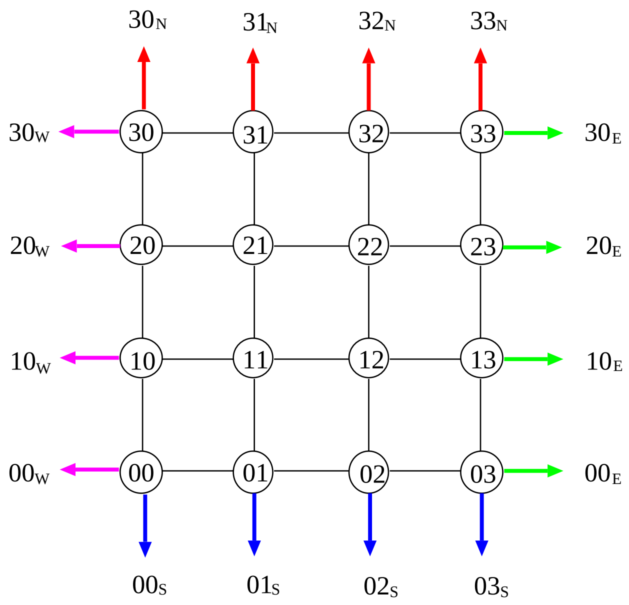


図 3.5: 3D 階層型メッシュの階層間リンク

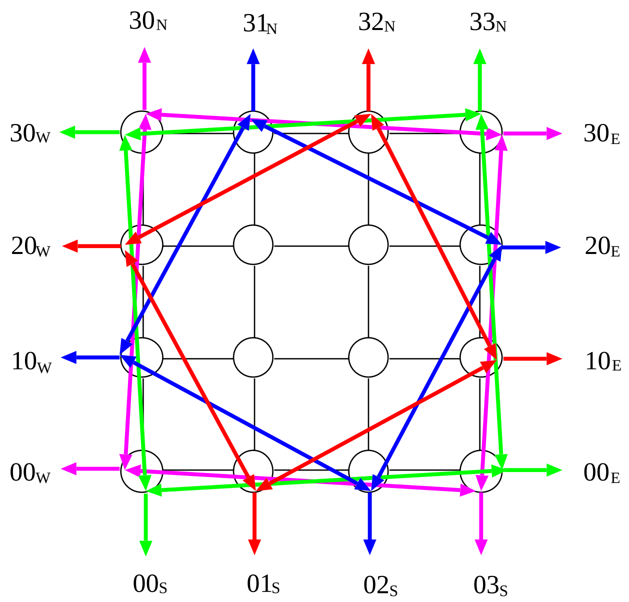


図 3.6: 3D 階層型メッシュの軸変更におけるゲート PE の選択

3.3.3 ウェーハ間最大結線数

3 次元階層型メッシュのウェーハスタック構造へのマッピングは以下のように行なう。この際、基本モジュールの PE 数は 64 と少ないため複数ウェーハへの分割は考慮しない。また表記に用いる記号は第 2.3節で定義したものを用いる。

1　レベル L_dをウェーハに s 個載せる。3 次元階層型トーラスは階層間を 2D メッシュで結合しているため、配置はアドレス順序で行なう。

2　レベル L の配置が終るまで 1 を繰り返す。

3 次元階層型メッシュも 3 次元階層型トーラスと同様に、サブネットにまたがるように上位レベルのリンクが行なわれるため、レベル L のウェーハ間最大結線数はレベル L 以下の各レベルのウェーハ間最大結線数の和となる。3 次元階層型メッシュはサブネット内の基本モジュール数に等しい 2D メッシュで階層間の結合が行なわれるため、レベル L_iの みを考慮したウェーハ間最大結線数は以下のようになる。

$$\begin{aligned} C_{max}^{L_i} &= C_{max}(2Dmesh, 16, s_{L_i})16^{L_i-2} \\ &= \begin{cases} 5(16^{L_i-2}) & s_{L_i} = 1, \\ 6(16^{L_i-2}) & s_{L_i} = 4 \end{cases} \end{aligned} \tag{3.17}$$

ただしレベル L_i階層のウェーハ 1 枚あたりのサブネット数を s_{L_i}で表す。したがって 3 次元階層型トーラスのウェーハ間最大結線数は以下のようになる。

$$\begin{aligned} C_{max} &= C_{max}(2Dmesh, 16, s)16^{L_d-2} + \sum_{L_i=L_d+1}^{L} C_{max}(2Dmesh, 16, 1)16^{L_i-2} \\ &= C_{max}(2Dmesh, 16, s)16^{L_d-2} + \sum_{L_i=L_d+1}^{L} 5(16^{L_i-2}) \end{aligned} \tag{3.18}$$

3.3.4 レイアウト面積

本研究では、レイアウト面積をウェーハ間最大結線数と行 (列) 当りのトーラスリンクの最大本数を用いて求める。ここでは、1 枚のウェーハに載るネットワークの最大トーラスリンク本数を求める。

3 次元階層型メッシュの最大トーラスリンク本数は、階層間の 2D メッシュ結合と 1 枚のウェーハに載る階層レベルによって決まる。1 つの基本モジュールがウェーハに載ってい

る場合は、$4 \times 4 \times 4$ の 3D トーラスを 1 枚のウェーハに配置した場合に等しく $t_x = t_y = 8$ である。レベル 2 階層を構成するリンクは $t_x = t_y$に寄与しないため $t_x = t_y = 8$ である。レベル 3 以降ではサブネット内の基本モジュール数に等しい 2 次元トーラスでサブネット間の結合が行なわれるため、トーラスリンク本数は 1 階層毎にサブネットの行 (列) あたりの基本モジュール数だけ増加する。したがって t_x, t_yは以下のようになる。

$$L_e = 1 + \lceil \log_{16} \frac{m}{64} \rceil \tag{3.19}$$

$$t_x = t_y = \begin{cases} 8 & L_e \leq 2, \\ 8 + 4 \sum_{i=L_e}^{L} 4^{i-2} & L_e > 2 \end{cases} \tag{3.20}$$

ここで L_eは 2 つ以上のサブネットが載っている最高階層レベルである。

3.4 ネットワーク特性の比較

ここでは階層型 3 次元ネットワークと第 2 章で検討したネットワークのネットワーク特性の比較を行なう。また比較結果からウェーハスタック構造に適したネットワークについて議論する。ただしハイパーキューブはウェーハ間結線数が非常に多く、ウェーハスタック構造への適用が困難であるため、ここでの検討には用いない。

本論文で提案した 3 次元階層型トーラスは直径の増加を抑えつつ、ウェーハ間結線数を低く保つことができた。これは、階層型ネットワークでは階層間の結線数を減らしつつ直径の増加を抑えることができるという特性を持つためである。2D トーラスのようにウェーハ間結線数が少ないネットワークに対しても数十万 PE まで、ウェーハ間結線を低く抑えることが可能であった。このため N=4096 の場合のレイアウト面積は 2D トーラスや TESH の 6 割、3 次元トーラスの半分以下に抑えることができた。3 次元階層型メッシュは基本モジュール間を 4 本の結線で結合したためウェーハ間結線が TESH の半分程度まで増加している。そのため、ウェーハ間結合線は、数万 PE で 2D トーラスと逆転しているが、3D トーラスに対しては数十万 PE まで低く保つことができた。N=4096 の場合のレイアウト面積は 2D トーラスや TESH の 8 割、3 次元トーラスの 6 割程度に抑えることができた。3 次元階層型ネットワークは直径に関しても 2D トーラスや 3D トーラスよりも増加が少なく、PE 数が非常に大きくなった場合でも、通信遅延を低く保つことができる。特に 3 次元階層型トーラスは、同じ階層型ネットワークの TESH よりも直径が

低い。これはレベル 4、5 において TESH は同方向の移動に基本モジュール内の移動が必要であるのに対し、3 次元階層型トーラスは同方向の移動では基本モジュール内での移動が必要ないことが影響している。

以上のように、ウェーハ間結線数が低いが直径が大きい 2 次元トーラスや直径がやや短い反面、ウェーハ間結線数が多い 3D トーラス、そして、階層型ネットワークの TESH と比較して、3 次元階層型ネットワークは直径を短く保ち、ウェーハ間結線数を少なく、そしてレイアウト面積を小さくできるという優れた性質を持つことが示された。

3.5 まとめ

本章ではウェーハスタック構造に適したネットワークとして、階層構造を用いた 3 次元階層型ネットワークを提案した。3 次元階層型ネットワークとして、ウェーハ間結線を重視した 3 次元階層型トーラスを提案し、基本モジュール間のデータ移動を重視した 3 次元階層型メッシュの提案を行なった。さらに提案した 3 次元階層型ネットワークのネットワーク特性を調べ、既存のネットワークとの比較を行なった。

提案した 3 次元階層型トーラスと 3 次元階層型メッシュは直径の増加を抑えつつ、ウェーハ間結線数低く保つことができた。特に 3 次元階層型トーラスはウェーハ間結線数が少ない 2D トーラスよりも、さらにウェーハ間結線数が少なく、レイアウト面積を小さく抑えることが可能である。また直径の増加も少なく、大規模超並列計算機システムへの適用が有効であることを確かめた。

次章では提案した 3 次元階層型ネットワークに基本的なアプリケーションのマッピングを行なう。

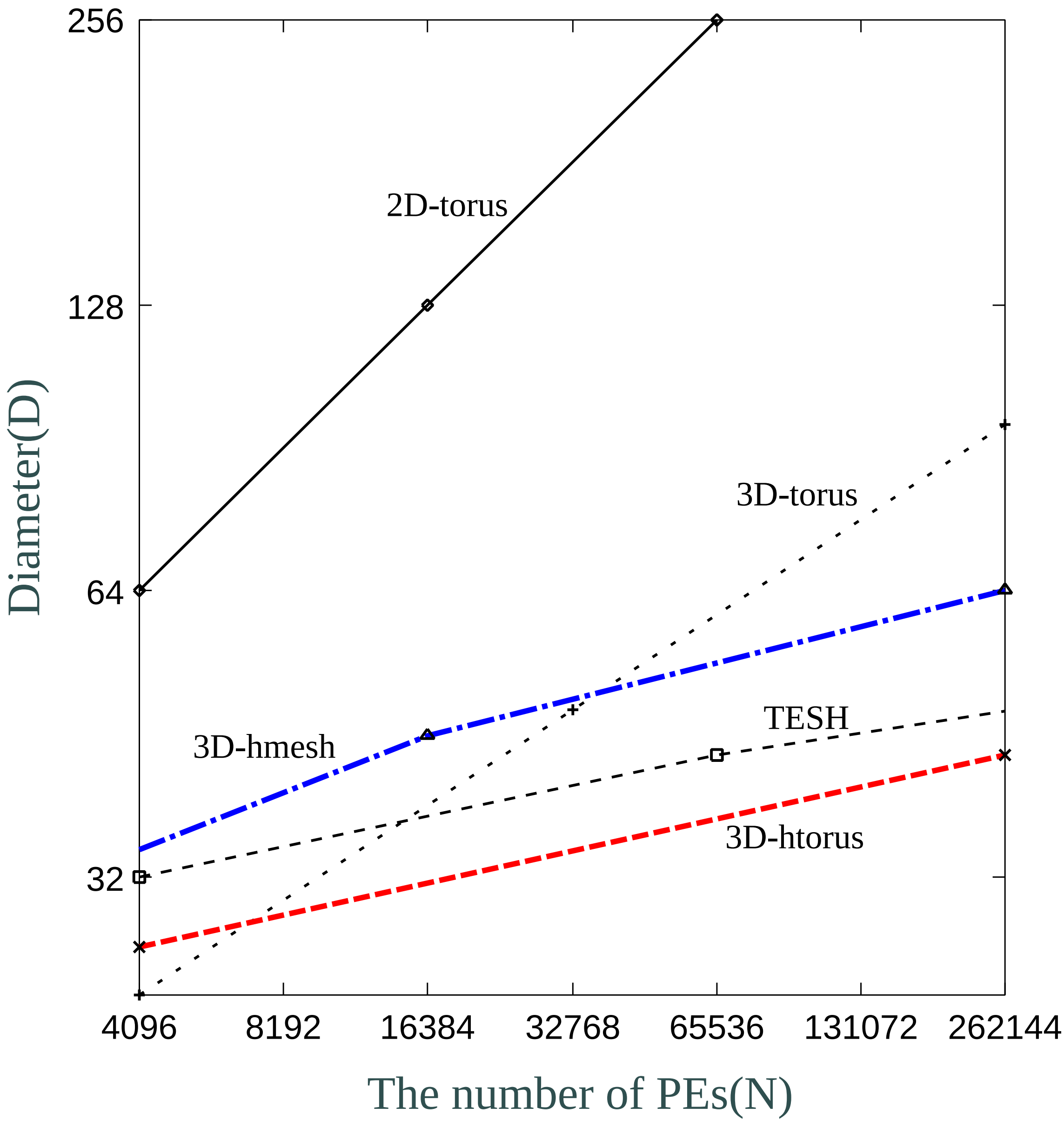

256
128
64
32
Diameter(D)
2D-torus
3D-torus
3D-hmesh
TESH
3D-htorus
4096
8192
16384
32768
65536
131072
262144
The number of PEs(N)

図 3.7: 直径 (D)

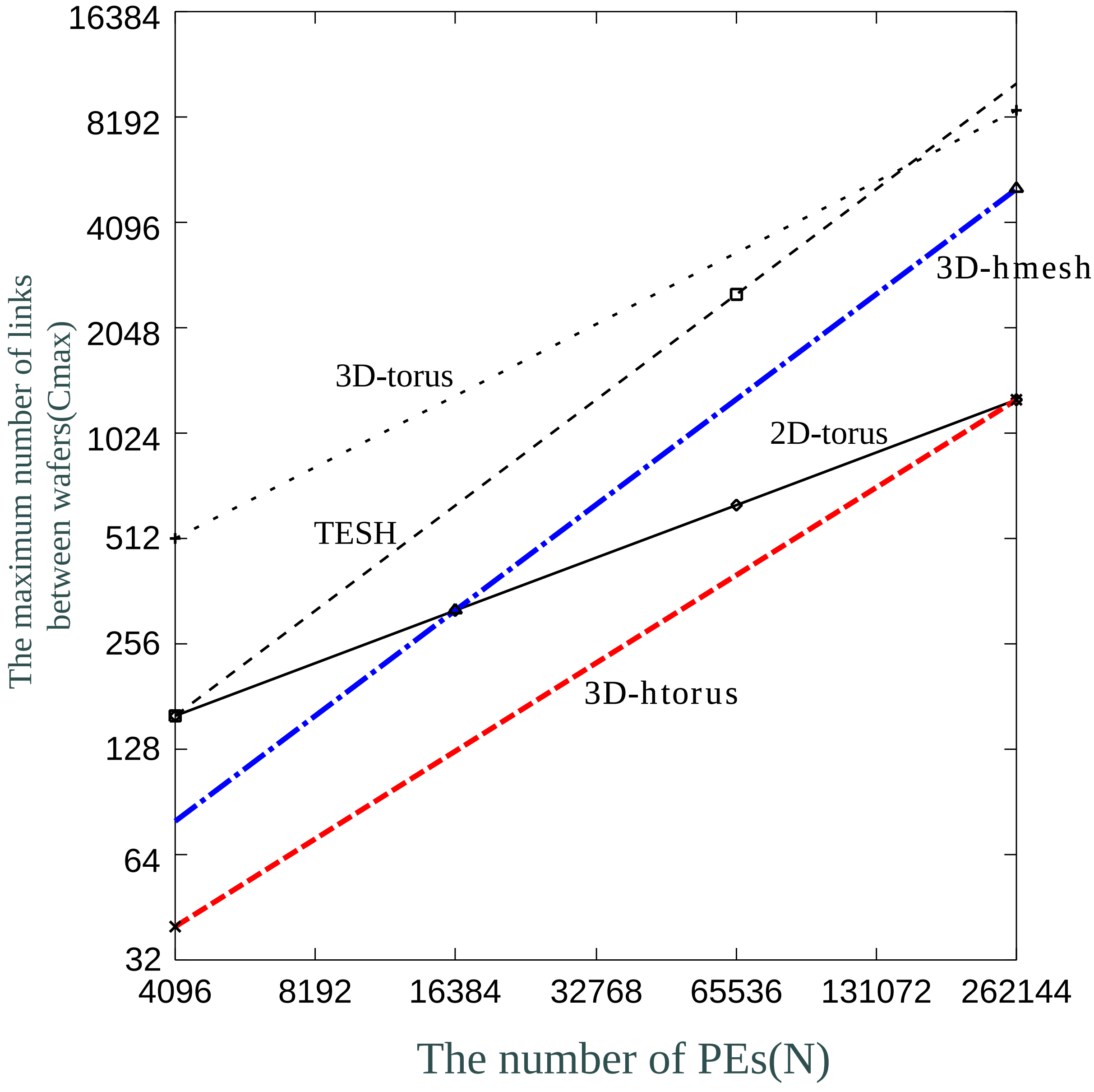


図 3.8: ウェーハ間最大結線数 (C_{max})

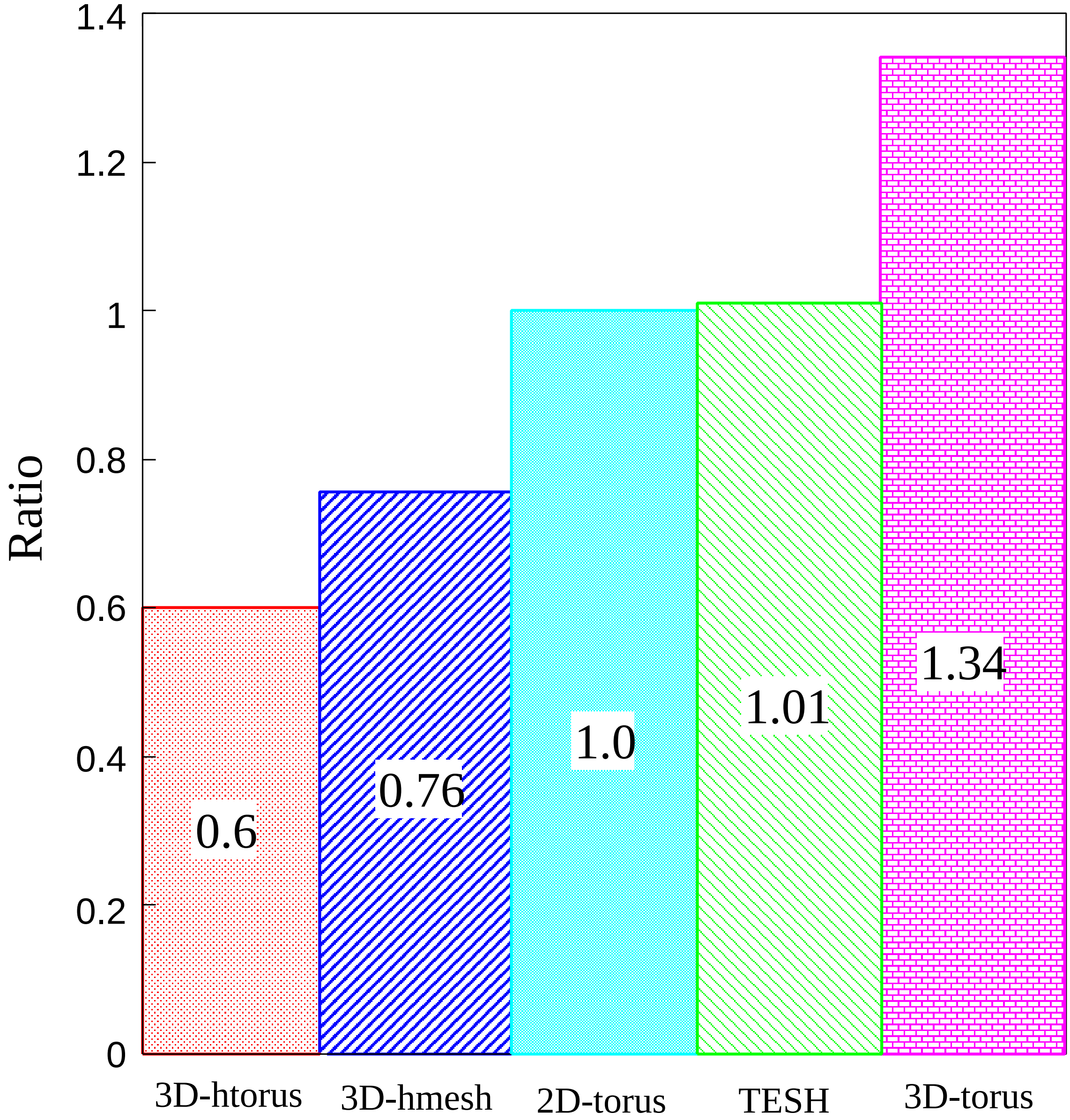


図 3.9: レイアウト面積 A(N=4096)

第 4 章

アプリケーションのマッピング

4.1　はじめに

本章では 3 次元階層型ネットワークに対して、基本的なアプリケーションのマッピングを行ない、実行にかかる総通信ステップ数を用いて評価を行なう。3 次元階層型ネットワークは格子状のネットワークで構成されるため、既存のメッシュ構造を持つネットワークに適用可能なアルゴリズムのマッピングが可能である。本論文では Divide&Conquer Scheme(分割倒置法) を用いて基本的なアプリケーションのマッピングを行なう [5][8][9]。この手法は細かく分割された問題に対して、複数のサブプログラムを並列に実行して、これを反復することで結果を得る手法である。この手法では、マージ、ソート、FFT などを簡単な命令の組合せで実行可能である。本論文では 3 次元階層型ネットワークに対して、以下のアプリケーションのマッピングを行なう。

- バイトニックソート
- FFT
- 最大値問題

4.2 Divide&Conqer Scheme

4.2.1 Divide&Conquer

Divide&Conqer Scheme は、細かく分割された問題に対して、複数のサブプログラムを並列に実行して、これを反復することで結果を得る手法である。ここでデータ数 N が 2 の累乗 ($N=2^k$) と仮定し、データの位置を m(m=0,1,...,N-1)、データの値を d[m] とする。DIVERGE 関数は、$2^0, 2^1, \ldots, 2^{k-2}, 2^{k-1}$離れたデータ間で演算を行なう。逆に CONVERGE 関数は、$2^{k-1}, 2^{k-2}, \ldots, 2^1, 2^0$離れたデータ間で演算を行なう。CONVERGE 関数と DIVERGE 関数を以下に示す。

```
  CONVERGE();
  for j=k-1:-1:0
    for 0 ≤ m ≤ N-1 do in parallel
      if a_j = 0,OPERATION(m,m+2^j);endif;
    endfor;
  endfor;
end;

  DIVERGE();
  for j=0:k-1
    for 0 ≤ m ≤ N-1 do in parallel
      if a_j = 0,OPERATION(m,m+2^j);endif;
    endfor;
  endfor;
end;
```

ここで、a_jはデータの位置をビット表現した場合の j ビット目を表し、OPERATION(m,m+2^j) は d[m] と d[m+2^j] に対する演算を表す。図 4.2.1に 4×4 の 2 次元メッシュの CONVERGE の実行課程を示す。

この CONVERGE を 3 次元階層型ネットワークの基本モジュールにマッピングすると

以下のようになる。

```
CONVERGE_BM();
for each element m in z-axis(a_z^1=0,1) do in parallel
    OPERATION(z[m,i],z[m,i+2]);
endfor;
for each element m in z-axis(a_z^1=0,2) do in parallel
    OPERATION(z[m,i],z[m,i+1]);
endfor;
for each element m in y-axis(a_y^1=0,1) do in parallel
    OPERATION(y[m,i],y[m,i+2]);
endfor;
for each element m in y-axis(a_y^1=0,2) do in parallel
    OPERATION(y[m,i],y[m,i+1]);
endfor;
for each element m in x-axis(a_x^1=0,1) do in parallel
    OPERATION(x[m,i],x[m,i+2]);
endfor;
for each element m in x-axis(a_x^1=0,2) do in parallel
    OPERATION(x[m,i],x[m,i+1]);
endfor;
```

DIVERGE の基本モジュールへのマッピングは演算の順序を反対にするだけで良い。したがって以降では DIVERGE のマッピングは省略する。次にレベル L_i階層に DIVERGE と CONVERGE のマッピングを行なう。3 次元階層型トーラスと 3 次元階層型メッシュでは階層間のネットワークが異なるため、別々にマッピングを行なう。

3 次元階層型トーラスは、階層間を 3D トーラスで結合する。CONVERGE を 3 次元階層型トーラスにマッピングすると以下のようになる。

```
CONVERGE_3Dhtorus-L_i();
for each element m in z-axis(a_z^{L_i}=0,1) do in parallel
```

```
    OPERATION(z[m,i],z[m,i+2]);
  endfor;
  for each element m in z-axis(a_z^{L_i}=0,2) do in parallel
    OPERATION(z[m,i],z[m,i+1]);
  endfor;
  for each element m in y-axis(a_y^{L_i}=0,1) do in parallel
    OPERATION(y[m,i],y[m,i+2]);
  endfor;
  for each element m in y-axis(a_y^{L_i}=0,2) do in parallel
    OPERATION(y[m,i],y[m,i+1]);
  endfor;
  for each element m in x-axis(a_x^{L_i}=0,1) do in parallel
    OPERATION(x[m,i],x[m,i+2]);
  endfor;
  for each element m in x-axis(a_x^{L_i}=0,2) do in parallel
    OPERATION(x[m,i],x[m,i+1]);
  endfor;
```

ただし、レベル 2 以降での z-axis,y-axis,x-aixs はサブネットワーク列を意味する。例えばレベル 2 階層では基本モジュール間で演算を行ない、レベル 3 階層ではレベル 2 階層間で演算を行なう。

3 次元階層型メッシュでは、階層間の結合が 2D メッシュで行なわれるため、3 次元階層型トーラスの z 軸に関する計算を省くだけででよい。したがって CONVERGE を 3 次元階層型メッシュにマッピングすると以下のようになる。

```
  CONVERGE_3Dhmesh-L_i();
  for each element m in y-axis(a_y^{L_i}=0,1) do in parallel
    OPERATION(y[m,i],y[m,i+2]);
  endfor;
  for each element m in y-axis(a_y^{L_i}=0,2) do in parallel
    OPERATION(y[m,i],y[m,i+1]);
```

endfor;

for each element m in x-axis($a_x^{L_i}$=0,1) do in parallel

  OPERATION(x[m,i],x[m,i+2]);

endfor;

for each element m in x-axis($a_x^{L_i}$=0,2) do in parallel

  OPERATION(x[m,i],x[m,i+1]);

endfor;

以降、この DIVERGE と CONVERGE を用いたアプリケーションのマッピングを行なう。

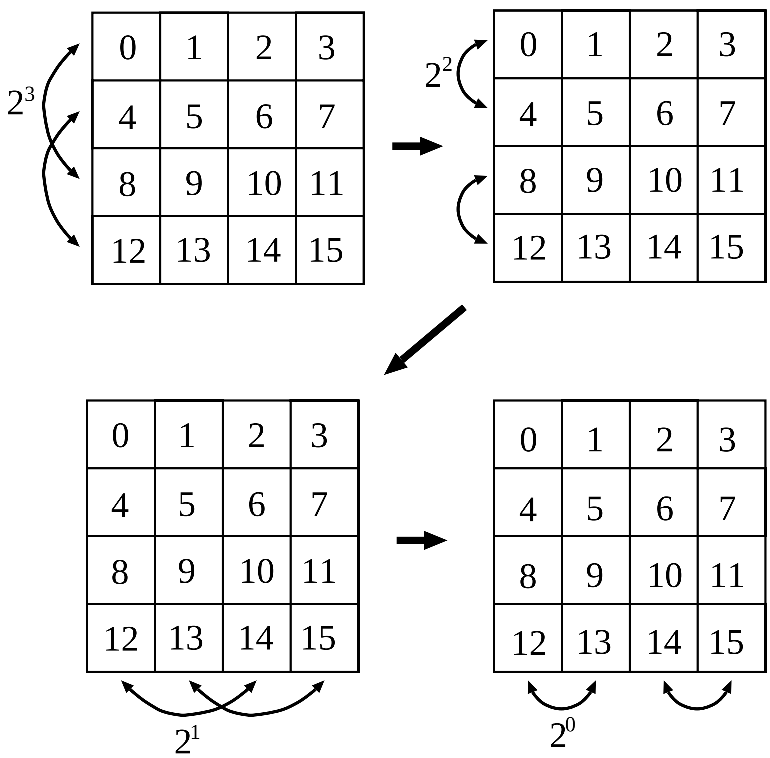


図 4.1: メッシュ上での CONVERGE の実行

4.2.2　バイトニック列

本節ではバイトニック列を計算する際の OPERATION を定義する。バイトニック列とは以下のような条件を満たす数列である。

定義

(1) $a_1 \le a_2 \le \ldots \le a_j \ge a_{j+1} \ge \ldots \ge a_{2n}$

(2) 数列が最初は条件 (1) を満たさなくても、巡回シフトを続けると条件 (1) が満たされる。

バイトニック列の計算は DIVERGE 関数を用いて行なう。小さい 2 つのバイトニック列を昇順と降順でソートして組み合わせることで 1 つのバイトニック列を得ることができる。これを繰り返すことで全要素を用いたバイトニック列を得ることができる。このバイトニック列を得る OPERATION は以下のようになる。

OPERATION(m,m+2^j){
　move $R_1(m+2^j)$ to $R_2(m)$;
　if a_{j+1}=0,
　　$[R_1(m), R_2(m)]$=[min$\{R_1(m), R_2(m)\}$,max$\{R_1(m), R_2(m)\}$];
　else
　　$[R_1(m), R_2(m)]$=[max$\{R_1(m), R_2(m)\}$,min$\{R_1(m), R_2(m)\}$] ;
　endif;
　move $R_2(m)$ to $R_1(m+2^j)$;
}

4.2.3　バイトニックマージ

本節ではバイトニック列をソートするバイトニックマージの OPERATION を定義する。バイトニック列は以下のような性質をもつ。

定理

$a_1, a_2, \ldots, a_{2n}$をバイトニック列とする。$1 \le i \le n$ に対して $d_i = min\{a_i, a_{i+n}\}, e_i = max\{a_i, a_{i+n}\}$ ならば

(1) $\{d_i\}, \{e_i\}$ はそれぞれバイトニック列である。

(2) $max\{d_i\} \leq min\{e_i\}$

(1) の性質により CONVERGE を用いて、バイトニック列を細かいバイトニック列に分解できる。(2) の性質より、反復演算によってバイトニック列がソートされる。以下にバイトニックマージの OPERATION を定義する。

OPERATION(m,m+2^j){
  move $R_1(m+2^j)$ to $R_2(m)$;
  $[R_1(m), R_2(m)]$=[min$\{R_1(m), R_2(m)\}$,max$\{R_1(m), R_2(m)\}$];
  move $R_2(m)$ to $R_1(m+2^j)$;
}

4.2.4 高速フーリエ変換 (FFT)

ベクトル a=$[a_0, a_1, \ldots, a_{N-1}]$ の離散フーリエ変換は、以下のようになる。

$$F(a) = [\sum_{k=0}^{N-1} a_k\omega^0 \quad \sum_{k=0}^{N-1} a_k\omega^k \quad \ldots \quad \sum_{k=0}^{N-1} a_k\omega^{ik} \quad \ldots \quad \sum_{k=0}^{N-1} a_k\omega^{(N-1)k}]$$

ここで$\omega = exp(\frac{-j2\pi}{n})$ である。FFT では以下のバタフライ演算を繰り返し行ない演算を行なう。

$$\begin{aligned} a_m[p] &= a_{m-1}[p] + a_{m-1}[p+2^{k-m}] \\ a_m[p+2^{k-m}] &= (a_{m-1}[p] - a_{m-1}[p+2^{k-m}])\omega^r \end{aligned}$$

k=log N、m:バタフライ算のステージ

したがって m 回目のステージでは 2^{k-m}離れたデータ間で演算を行なう。したがって CONVERGE 関数を用いることができる。この場合の OPERATION は以下のようになる。

OPERATION(m,m+2^j){
  move $R_1(m+2^j)$ to $R_2(m)$;
  $Temp(m) = R_1(m) + R_2(m)$;
  $R_2(m) = (R_1(m) - R_2(m)) * {W_N}^r$;

$R_1(m) = Temp(m)$;

move $R_2(m)$ to $R_1(m + 2^j)$;

}

4.2.5 最大値問題

最大値問題は CONVERGE を用いて、比較を繰り返し行なうことで解くことができる。比較した値が大きいデータのみを用いれば良いので、演算結果を返す必要はない。そのため、演算を行なった PE の半数は次の比較では必要ない。したがって最大値問題では余分な演算を省くため、CONVERGE を以下のように変更する。

```
CONVERGE();
  for j=k-1:-1:0
    for 0 ≤ m ≤ (N/2)^(k-j-1) - 1 do in parallel
      if a_j = 0,OPERATION(m,m+2^j);endif;
    endfor;
  endfor;
end;
```

また、最大値を求める OPERATION は以下のようになる。

OPERATION(m,m+2^j){

move $R_1(m + 2^j)$ to $R_2(m)$;

$R_1(m) = max\{R_1(m), R_2(m)\}$;

}

階層型ネットワークの TESH や 3 次元階層型トーラスのように PE 間のリンク数が少ないネットワークでは、ネットワーク全体に CONVERGE を用いると階層間のリンクに通信が集中し、性能が低下する。最大値のように必要なデータ数が少ない場合には各レベル毎に DIVERGE を適用して、得られた結果のみを階層間で演算することによって、通信の集中を避けることができる。実際に演算に用いる PE は各レベルでの最大値を保持してい

る PE のみでいい。そのため 3 次元階層型ネットワークの最大値は以下のようにして求める。

Maximum;

CONVERGE_BM();

for j=2:L

if($a_x^{[j-1:0]} = 0 \ \ and \ \ a_y^{[j-1:0]} = 0 \ \ and \ \ a_z^{[j-1:0]} = 0$),**OPERATION(m,m+$2^j$)**;endif;

CONVERGE_network-j();

endif;

endfor;

}

4.3 アプリケーションの性能評価

本節ではマッピングしたアプリケーションの実行ステップ数を求め、3 次元階層型ネットワークと 2D-トーラス、3D トーラスの比較を行なう。

DIVERGE と CONVERGE の計算ステップは、$\log N$である。したがってアプリケーションの実行時間を以下のように定義できる。

アプリケーションの実行時間：

$$T = \alpha T_{move} + \log N \ T_{OPER}$$

α：アプリケーション実行にかかる総通信ステップ数

T_{move}：隣接する PE 間のデータ移動時間

T_{OPER}：関数 OPERATION の実行時間

各ネットワークの計算ステップ数は等しい。したがって本研究では総通信ステップ数を用いてネットワークのアプリケーション実行性能を評価する。評価には PE 間で多数の通信が必要なバイトニックマージと一部の PE 間のみで通信を行なう最大値問題を用いる。各 PE にデータが分散配置されている場合の総通信ステップ数は以下のようになる。

2D トーラス：

$$S_{bitonic} = 4 \sum_{j=0}^{\sqrt{N}-1} \frac{\sqrt{N}}{2^j}$$

$$
\begin{aligned}
&= 4(\sqrt{N}-1) && (4.1)\\
S_{max} &= 2(\sqrt{N}-1) && (4.2)
\end{aligned}
$$

3D トーラス：

$$
\begin{aligned}
S_{bitonic} &= 6\sum_{j=0}^{N^{\frac{1}{3}}-1}\frac{N^{\frac{1}{3}}}{2^j} \\
&= 6(N^{\frac{1}{3}}-1) && (4.3)\\
S_{max} &= 3(N^{\frac{1}{3}}-1) && (4.4)
\end{aligned}
$$

最大値問題では値を返す必要がないため、通信ステップ数がバイトニックマージの半分になっているのが非階層型ネットワークの特徴である。これに対して、3 次元階層型ネットワークの総通信ステップ数は以下のようになる。

3 次元階層型トーラス：

$$
\begin{aligned}
S_{bitonic} &= (L-1)3(4+2)64+18 \\
&= 1152(L-1)+18 && (4.5)\\
S_{max} &= 18L && (4.6)
\end{aligned}
$$

3 次元階層型メッシュ：

$$
\begin{aligned}
S_{bitonic} &= (L-1)2(4+2)64/4+12 \\
&= 192(L-1)+18 && (4.7)\\
S_{max} &= 12(L-1)+18 && (4.8)\\
& && (4.9)
\end{aligned}
$$

階層型ネットワークでは、$O(\log N)$ の通信ステップ数でアプリケーションの実行が可能である。しかし、バイトニックマージのように全 PE 間の通信が必要なアプリケーションではオーダーにかかる係数が大きくなっている。これは基本モジュール間のリンクに通信の集中が起こるためである。図 4.2と図 4.3はバイトニックソートと最大値問題の実行に必要な総通信ステップ数である。基本モジュールを用いた結合方式ではレベルが上がるにつれ階層間のネットワーク数が増加する。例えば 3 次元階層型トーラスのレベル 2 階層の結合は 3D トーラスで行なわれる。レベル 3 階層の結合はサブネット内の基本モジュー

表 4.1: 総通信ステップ数

	バイトニックマージ		最大値問題	
	総通信ステップ数	オーダー	総通信ステップ数	オーダー
2D トーラス	$4(\sqrt{N}-1)$	$O(\sqrt{N})$	$2(\sqrt{N}-1)$	$O(\sqrt{N})$
3D トーラス	$6(N^{\frac{1}{3}}-1)$	$O(\sqrt{N})$	$3(N^{\frac{1}{3}}-1)$	$O(\sqrt{N})$
3 次元階層型トーラス	$1152(L-1)+18$	O($\log N$)	$18L$	O($\log N$)
3 次元階層型メッシュ	12(L-1)+18	O($\log N$)	12(L-1)+18	O($\log N$)

ル数に等しい 64 の 3D トーラスで階層間の結合が行なわれる。つまりレベルが増すにつれ階層の並列化が進む。3 次元階層型トーラスは基本モジュールの PE 数 (64) に対して、基本モジュール間の結合は 1 本で行なわれているため、基本モジュール間のリンクに通信の集中が起こり、更に階層化の規模が大きく PE 数に対しレベルの上昇が遅いため、2D メッシュよりも高速な演算を行なうには数十万 PE が必要であるという結果が得られた。それに対して、3 次元階層型メッシュは基本モジュールの PE 数 (64) に対して、基本モジュール間の結合を 4 本で行ない、階層化の規模も小さいため 3 次元階層型トーラスと比較して数万 PE で十分な性能を得ることができている。数十万 PE では 3D トーラスよりも高速な演算が可能であるという結果が得られた。最大値のように必要とするデータが少ない演算では階層型ネットワークのリンクの少なさが影響しないため、非常に少ないステップ数で演算が可能であった。

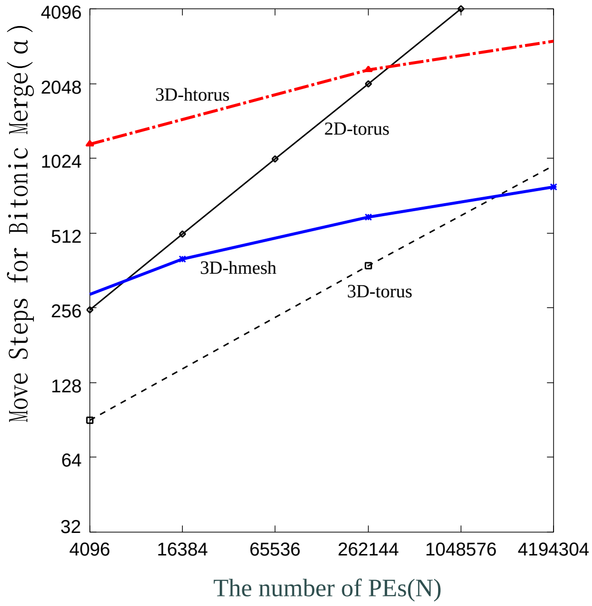


図 4.2: 総通信ステップ数 $S_{bitonic}$(バイトニックマージ)

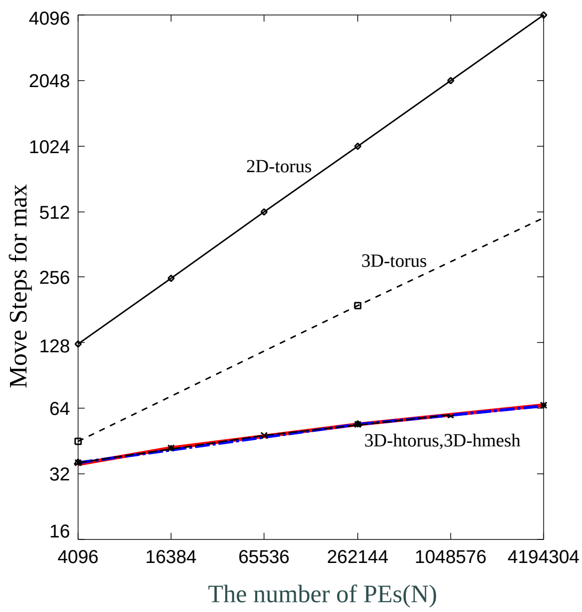


図 4.3: 総通信ステップ数 S_{max}(最大値問題)

4.4 まとめ

本章では 3 次元階層型ネットワークに対して基本的なアプリケーションのマッピングを行なった。Divide&Conquer を用いることで容易に既存のアプリケーションがマッピング可能であることを示した。3 次元階層型トーラスでは階層間のリンク数が少ないため、十分な性能を得ることができなかったが、階層間のリンク数を増加させた 3 次元階層型メッシュでは数千 PE で 2D トーラスよりも少ないステップ数での演算が可能であった。また最大値問題のように必要なデータ数が少ないアプリケーションでは非常に少ないステップ数で演算が可能であった。このような演算は階層型ネットワークの直径の小ささが有利に働いている。しかし、一般に Divede&Conquer を用いた場合、階層型ネットワークの能力を十分に活かすことは難しい。階層間のリンク数の少なさに影響を受けにくいアプリケーションのマッピングが今後の課題である。

第 5 章

まとめ

ウェーハスタック構造は、シミュレーションや動画像処理に必要とされる大規模超並列計算機システムを実現する WSI の手法である。しかしウェーハスタック構造ではウェーハ間の結合に大きな面積を必要とするため、ウェーハ間結線数を制限する必要がある。本研究では、ウェーハ間結線数を減らしつつ、直径やチップ面積の増加を抑えるネットワークとして、3 次元階層型トーラスと 3 次元階層型メッシュの提案を行なっい、その性能を評価した。第 2 章ではウェーハスタック構造におけるネットワーク特性を調べるため、既存のネットワークのウェーハ間最大結線数、ウェーハ間結線に必要な面積を考慮したレイアウト面積を求め、直径と合わせた検討を行なった。その結果、階層型ネットワークや低次数のネットワークがウェーハスタック構造に適していることが示した。第 3 章ではウェーハスタック構造に適したネットワークとして、基本モジュール間のリンクを減らした 3 次元階層型トーラスと基本モジュール間のリンクを増やし、通信の分散を行なった 3 次元階層型メッシュを提案した。また 3 次元階層型ネットワークのルーティングを示した。直径、ウェーハ間最大結線数、レイアウト面積を求め、第 2 章で用いた既存のネットワークとの比較を行ないその有効性を確かめた。第 4 章では 3 次元階層型ネットワークにアプリケーションのマッピングを行ない $O(\log N)$ で実行可能であることを示した。また総通信ステップ数を用いて実行性能を調べた。その結果 3 次元階層型トーラスでは階層間のリンク数が少ないため、通信の集中により性能が大きく低下すること分かった。しかし、3 次元階層型メッシュでは階層間のリンク数を増加させたことによって、数千 PE で 2D トーラスよりも少ないステップ数でアプリケーションの実行が可能であることも示し、階層型ネットワークの有効性を示した。

今後の課題としては、以下のものが挙げられる。

- 階層型冗長構成法の検討
- 階層型ネットワークに適したアプリケーションの選択
- 複雑なアプリケーションのマッピング

謝辞

本研究を進めるにあたり、終始熱心な御指導、御鞭撻をいただいた北陸先端科学技術大学院大学 堀口 進 教授、阿部 亨 助教授に深く感謝いたします。

サブテーマで熱心に御指導をいただいた 酒井 正彦 助教授に感謝いたします。

日頃より有意義な御教示、検討、議論をしてくださった沼田一成助手に感謝いたします。

また、日頃よりお世話になった堀口・阿部研究室の皆様に厚く御礼申し上げます。

参考文献

[1] 堀口 進：“ウェハ規模超密度集積回路について”,Hybrids,6,1,pp.16-21(1990).

[2] Little.M.J,J.Grinberg：“The 3-D Computer:An Integrated Satck of WSI Wafers,in Wafer Scale Integration”,Kluwer Academic Publishers,pp.253-317,1989.

[3] M.L.Campbell,S.T.Toborg：“3D wafer stack neurocomputing”,Proceedings of International Conference on Wafer Scale Integration,pp.67-74,1993.

[4] K.Hwang,J.Ghosh：
“Hypernet：A communication-efficient architecture for constructing massivlyparallel computers”,IEEE Trans. Comput.,36,12,pp1450-1466(Dec.1987).

[5] V.K.Jain,T.Ghirmai,S.Horiguchi：“Reconfiguration and field for TESH:A New Hierarchical Interconnection Network for 3-D Integration”,Proc. IEEE Int’l Conf. SISI(1996).

[6] 小柳　滋, 田辺　昇：“超並列マシンの実現技術”, 情報処理,Vol.32,No.4,pp565-376(Apl.1991).

[7] 福田 大, 堀口 進：“階層型ハイパーキューブ結合マルチプロセッサシステムの WSI 構成法”, 電子情報通信技報 WSIA94,no.12

[8] D.Nassimi,S.Sahni：“Bitonic sort on a mesh-connected parallel computer”,IEEE Transaction on Computers,Vol.27,pp.2–7,1979.

[9] S.Akl：“並列ソーティングアルゴリズム Parallel Sorting Algorithms”, 啓学出版, pp.89–121.

研究業績

[1] ウェーハスタック構造超並列ネットワークの特性
大木孝之，堀口進
電気関係学会北陸支部連合大会 pp279, Oct.1996.