

## 高次数規則トポロジの3D-NoCへのレイアウト

中原 浩† 安戸 僚汰† 松谷 宏紀† 鯉淵 道紘‡ 天野 英晴†

†慶應義塾大学大学院 ‡国立情報学研究所

## 1 はじめに

マルチコア・プロセッサの研究が盛んになっている昨今、Network-on-Chip(NoC)[1]は注目を集めている。NoCは、チップをタイルごとに分割して、タイル間の通信をルータを用いてパケット通信を行うことで、従来のバスと比較して高いバンド幅・並列性・スケラビリティを実現する。

NoCを実チップへ実装した多くのものは、2-D Meshのような低次数トポロジを用いて実装されている。低次数トポロジは各配線は隣接するタイルへの接続だけで済み実装がシンプルになるという利点があるが、ネットワークの直径・平均ホップ数が大きくなり、その結果として消費電力や性能の悪化を招く。[2]ではデータセンタ向けに提案されているFlattened Butterflyをチップ内に配置する方法を提案された。しかし、高次数トポロジをレイアウトする場合には、チップ内のコア数が増加するに従って、配線遅延の問題が存在する。配線長が長くなることによって、配線遅延がクリティカルパスになってしまい、結果的に性能の悪化を招く。

3D-NoCは三次元積層技術を用いてNoCを積層するものである。単一チップ上にすべてのコアを配置するのではなく、複数のチップに分けてコアを配置することで、配線遅延の長いチップ内通信を三次元方向通信によって隠蔽することで、結果的に配線遅延を小さく抑えることが可能となる。3D-NoCでは3D-Torus, 3D-Meshなどの低次数トポロジは研究されているが、高次数トポロジを配置する方法は提案されていない。

本論文では三次元積層されたチップにデータセンタ向けトポロジのFlattened Butterfly、Dragonflyを三次元チップ上にレイアウトする手法を提案する。これにより、最大配線タイル長を単一チップ上にレイアウトする場合と比べて小さくすることが可能になる。

## 2 NoCの配線遅延

NoCの配線遅延はルータ間をつなぐ配線の長さに大きく依存する。コア数が多くなるにつれて、高次数トポロジではルータ間リンクの物理的な距離が長くなり、それがクリティカルパスとなってしまう。例えば、図1は64コアの場合のMeshと2D-Flattened Butterflyのチップ内レイアウトであるが、Meshのレイアウトでは、各ルータ間の距離は高々タイル1つ分であるのに対し、

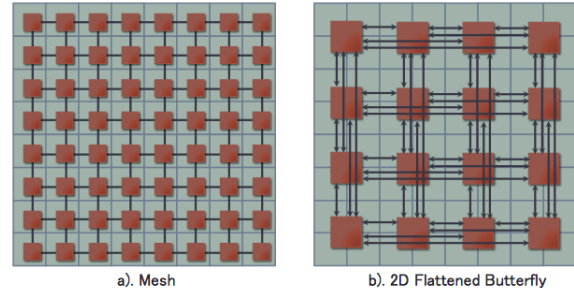


図1: Meshと2D-Flattened Butterflyのチップ内レイアウト

2D-Flattened Butterflyでは最大でタイル6つ分の距離がある。

タイル間をつなぐ最大の配線長が長くなるほど、そのタイル間の通信にかかる時間、配線遅延が長くなる。配線遅延がクリティカルパスとなるかどうかは、ルータのパイプライン段数や配線内に配置するリピータの数、1タイルの長さに依存する。高次数トポロジでは、コアの数が増えるほど最大配線タイル長は大きくなっていくことになる。

## 3 三次元レイアウト手法

この章で用いるパラメータを以下のように定義する。

- $N$ : コアの総数
- $M$ : チップ内のコア数
- $C$ : 積層するチップ枚数
- $L$ : 最大配線タイル長
- $D$ : ルータの次数
- $a$ : 一つのルータが接続しているコア数

## 3.1 Flattened Butterfly

2D-Flattened Butterflyのレイアウト方法は図1b)で示した通りである。3D-Flattened Butterflyを二次元チップ上にレイアウトする場合、最大配線長が長くなってしまふ。

次に3D-Flattened Butterflyの三次元レイアウト法を以下のように定義する。制約として、 $N = aC^3$ を満たすような $N$ 、 $C$ 、 $a$ を選択する。これは3D-Flattened Butterflyを満たすために必要な制約となっている。

定義1 3D-Flattened Butterflyの三次元レイアウト  
 $N = aC^3$ のとき、 $M = aC^2$ になるようにチップ内に

3D-NoC Topology Optimization for Average Energy Consumption and Average Shortest Path Length

†Hiroshi Nakahara †Ryota Yasuda †Hiroki Matsutani ‡Michihiro Koibuchi ††Koji Nakano †Hideharu Amano

†Keio University ‡National Institute of Informatics ††Hiroshima University

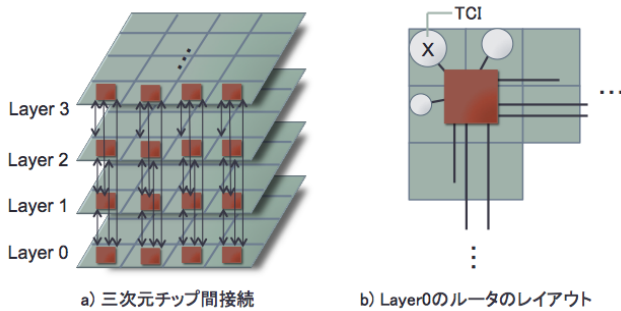


図2:  $N=256$  の 3D-Flattened Butterfly の三次元レイアウト

2D-Flattened Butterfly を配置する。チップ内の各ルータは他チップの垂直方向に隣接するルータと通信出来るように TCI を配置する。すなわち各ルータは  $C-1$  個の他チップの TCI と接続する。

例として、 $N=256$ 、 $a=4$  の 3D-Flattened Butterfly の三次元レイアウト法を図2に示す。 $N=256$ では、 $M=64$ のチップを4枚積層している。チップ内の各ルータは、その他のチップの水平方向の同じ位置に存在するルータと TCI を用いて通信する。図2a)のように、例えば Layer3 のチップは、TCI を用いて Layer2、Layer1、Layer0 のチップと通信している。図2b)はチップ内のルータのレイアウトを示しており、 $D=3(C-1)+a$  となる。

### 3.2 Dragonfly

Dragonfly [3] は、Off-Chip 用に提案された高次数トポロジである。各ノードをグループに分けて、グループ内ネットワークとグループ間ネットワークに分けることで直径が3となるように構成されている。Dragonfly は最大配線長ではなく配線長が長い配線数を少なくするために提案されていることから、On-Chip には適さないと考えられる。しかし、三次元レイアウトを行うことによって最大配線長を小さくすることが出来る。

Dragonfly の三次元レイアウト法のアイデアとしては、グループ間の配線を三次元方向の通信に用いることで物理的な距離を短くするようにしている。制約として  $C = \frac{M \cdot n}{a} + 1 (n \in \mathbb{N})$  を満たす  $M$ 、 $n$ 、 $a$  を選ぶ。この制約は Dragonfly でネットワークのバランスを考えると、各ルータから他グループに対して同じ本数のチャンネルが存在するほうがトラフィックが偏りが小さくなるためである。

#### 定義2 Dragonfly の三次元レイアウト

各チップをグループ、チップ間の TCI 通信をグループ間通信として考える。チップ内のネットワークには全結合ネットワークを用いる。チップ内の各ルータには、 $\frac{a \times C}{M}$  個の TCI と接続し、チップ内には合計で  $C-1$  個の TCI を配置する。チップ内の各 TCI はそれぞれ異なるチップとの通信を行う。

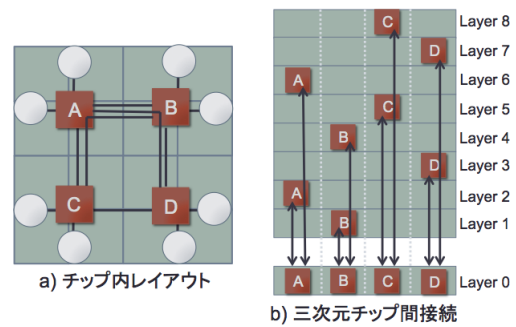


図3:  $N=144$  の Dragonfly の三次元レイアウト

例として、 $N=144$ 、 $a=4$  の Dragonfly の三次元レイアウトを図3に示す。チップ内には  $M=16$  となるように配置し、各ルータは全結合ネットワークで接続しており、 $D = (C-1)\frac{a}{M} + \frac{M}{a} + a$  と表せる。チップ間の通信は図3に示している。図3a)とb)のルータに書いてあるアルファベットはそれぞれ対応している。例えば、Layer 0 のルータ C から Layer 2 のルータ B にパケットを送信する場合、まずチップ内ルーティングでルータ A へ1ホップでパケットを送信し、次に TCI を通して Layer 2 へ送信する。最後に layer 2 内でチップ内ルーティングしてルータ B に送信する。

## 4 結論

本論文では 3D-NoC に対して、高次数トポロジである、Flattened Butterfly、Dragonfly をレイアウトする方法を提案した。この方法によって、二次元レイアウトの時と比べて最大配線タイル長が短く 3D-NoC にレイアウト出来るようになる。

### 謝辞

本研究の一部は JSPS 科研費 (S) ビルディングブロック型計算システムに関する研究の助成を受けたものである。

### 参考文献

- [1] William J. Dally and Brian Towles. Route Packets, Not Wires: On-Chip Interconnection Networks. In *Proceedings of the Design Automation Conference (DAC'01)*, pp. 684–689, June 2001.
- [2] J. Kim, J. Balfour, and W.J. Dally. Flattened butterfly topology for on-chip networks. In *40th Annual IEEE/ACM International Symposium on Microarchitecture, 2007. MICRO 2007.*, pp. 172–182, Dec 2007.
- [3] J. Kim, W.J. Dally, S. Scott, and D. Abts. Technology-driven, highly-scalable dragonfly topology. In *Computer Architecture, 2008. ISCA '08. 35th International Symposium on*, pp. 77–88, June 2008.