

超並列向き de Bruijn 網の諸特性

奥川峻史

京都産業大学 工学部
情報通信工学科

2次元トーラスの次数(4)と2進ハイパキューブの直径($\log_2 N$: N =要素プロセッサ数)を合わせ持っている2進 de Bruijn 網は、直径 X 次数、即ち価格対性能比が、ハイパキューブの $O(\log_2^2 N)$ に対して、 $O(\log_2 N)$ であり、超並列向きである。本報告はこの網の、相互結合網に一般的に要求される諸特性、即ちフォールトトレランス、エンベダビリティ、スケーラビリティ、ルーティングアルゴリズム、レイアウト(以上、FESALと呼ぶ)を、階層構成、ダイグラフ、及びグラフの積による構成を含めて、実用的見地からまとめている。

Characteristics of the de Bruijn Network
for Massively Parallel Computers

Shunji OKUGAWA

Department of Information and Communication
Faculty of Engineering
Kyoto Sangyo University
Motoyama, Kita-ku, Kyoto 603, Japan

The binary de Bruijn network is massively-parallel-oriented, having node-degree of 2-D torus (4), and diameter of binary hypercube ($\log_2 N$: N = number of processor elements), and cost/performance of $O(\log_2 N)$. This paper reports various characteristics, such as fault-tolerance, embeddability, scalability, routing algorithm, and layout (so-called FESAL) of this network, including implementations by hierarchy, digraph, and Cartesian product of two graphs.

1. はじめに

多数の要素プロセッサ (PE) を結合する直接結合型の相互結合網としては、2進ハイバキューブ、2次元トラス (メッシュ) が一般的である。

PE数を $N = 2^n$ とすれば、2進ハイバキューブ HC (2, n) の直径と次数はともに $\log_2 N$ であり、2次元トラス HT (\sqrt{N} , 2) の直径は \sqrt{N} 、次数は4である。

ハイバキューブの長所は直径が短いことであり、短所は次数がNと共に大きくなる (拡張性を考えPEのI/Oポート数を大きくしておけば、小規模構成ではハードウェアに遊びが生じる) ことである。トラスの長所は次数が小さい定数であることであり、短所は大規模構成では直径が大変長くなる (通信が近傍のPE間に限られるアプリケーションなら良いが一般には低速) ことである。

2進 de Bruijn 網 UB (2, n) はこの両者の長所を兼ね備えており、直径はハイバキューブと同じ $\log_2 N$ 、次数はトラスと同じ4である。価格対性能比 (直径×次数) は $4 \log_2 N$ となる。⁽⁹⁾⁽¹⁰⁾⁽¹¹⁾

表1の3つの網の価格対性能比の比較より分かるように、de Bruijn 網はハイバキューブの $O(\log_2^2 N)$ に対して $O(\log_2 N)$ であり、N大 (超並列) 程ますます de Bruijn 網の優位が顕著になる (畳込みハイバキューブでは価格対性能比は約 $1/2$ になる⁽¹²⁾)。

de Bruijn 網は、NASAのGalileoプロジェクト (PE数8192, 1995年完成を目標)⁽⁹⁾、Jet Propulsion Laboratoryでビタビ符号デコーダ (PE数8192, 上と同一プロジェクト?)⁽⁹⁾、ヨーロッパで開発中のトランスペュータ (4つのI/Oポートを持ち2進 de Bruijn 網向き) をPEとするマルチコンピュータシステム⁽¹⁾、で採用されている。

価格対性能比の他に次に示す諸特性が重要である。

(1) フォールトトレランス (FT, 耐故障性)

故障PEのバイパスや予備PEによる再構成

(2) エンベダビリティ (EB, 埋込み可能性)

他のトポロジをエミュレートできる汎用性

(3) スケラビリティ (SB, 拡張性)

スケールアップの容易性

(4) ルーティングアルゴリズム (RA)

経路選択 (故障時を含む) の容易性

(5) レイアウト (LO)

チップ上への効率的配置の可能性

以上の5つの特性を FESAL と呼ぶ。⁽¹⁰⁾⁽¹¹⁾

本報告では簡単で実用性の高い2進網 (価格対性能比は2,3,4進が良い⁽¹⁰⁾⁽¹¹⁾) の FESAL 特性を、階層構成、ダイグラフ、グラフの積による構成も含めて、超並列向き結合網の見地から実際的にまとめている。

2. de Bruijn 網の基本構成

2進 de Bruijn 網の構成はシャフルエクステンジ網とよく似ている (2進 de Bruijn 網をシャフルエクステンジ網と呼んでいる文献すらある。⁽⁷⁾)。図1に de Bruijn 網とシャフルエクステンジ網、及びその相互関係を示す (N=8)。

有向 de Bruijn 網 B (2, n) は、シャフル (S) アークとシャフルしたものに対するエクステンジであるシャフルエクステンジ (SE) アークから成り、入力次数、出力次数ともに2である。

これに対して、シャフルエクステンジ網は、シャフル (S) アークと、シャフルする前のものに対するエクステンジであるエクステンジ (E) アークから成り、直径 $2 \log_2 N - 1$ 、次数3である。⁽¹²⁾ 無向 de Bruijn 網 UB (2, n) は、有向網のアーキをリンク (双方向) で置き換え、ノード 00...0 と 11...1 の自己ループを取り除き、

表1 ハイバキューブ、トラス、de Bruijn 網のPE数に対する価格対性能比の比較

n	$N=2^n$	HC(2, n)	HT(N, 2)	UB(2, N)
10	1024	100	128	40
12	4096	144	256	48
14	16384	196	512	56
16	65536	256	1024	64
18	262144	324	2048	72
20	1048576	400	4096	80

0101...と1010...間の多重アークを1本のリンクで置き換えたものである。00...0と11...1の2個のノードの次数は2, 0101...と1010...の2個のノードの次数は3となる。これら4個のノード以外のノードはS, SE, S⁻¹(逆シャフル), ES⁻¹(=(SE)⁻¹, 逆シャフルエクスチェンジ)で4つのノードとリンクで結合され, 次数は4である。

直径は, 有向の場合も無向の場合もlog₂Nであり, 平均距離は無向の方が当然小さくなる。例えばN=8の場合, 無向の平均距離は1.5, 有向のそれは2.04である。

なお, ノード00...0と11...1間, 11...1と1010...間, 0101...と00...0間に結合を付加し, すべてのノードの次数を4にした改良型⁶⁾も考えられる。

また, 2次元トラスの双方向リンクを単方向アークにして, 価格対性能比の向上を図った, マンハッターストリート網(MSN)²³⁾のように, 有向de Bruijn網の利用も検討する必要がある。

なお, 一般に次数4, 直径Dの有向グラフでは

$$N \leq 1 + 4 + 4^2 + \dots + 4^D = (4^{D+1} - 1) / 3$$

$$D \geq \log_4(3N + 1) - 1$$

の関係が成立するから, 上式の右辺, すなわち有向ム

ーア限界^{8) 10)}は, ほぼ(log₂N) / 2である。

3. フォールトレランス

多数のPEを大規模な相互結合網で結合したシステムでは, 当然のことながら故障の確率も大きくなる。PEチップはVLSI(WSI)化する必要があるが, 製造時の歩留りを向上させて製造コストを下げる為にも, 動作時の耐故障性の向上の為にも, 次のような機能を持つ構成が不可欠である。

- (1) 不良・故障PEやリンクをバイパスし, 故障が起こった場合でも, 性能は多少落ちるとしてもできるだけ本来の機能を維持できる (graceful degradation)。
- (2) 前もってスペアのPEやリンクを設けておき, 故障が起こったときスペアに切り替えて, 何等性能の低下を生じない。

当然のことながら, 後者の構成の方が望ましい。

ハイバキューブのフォールトレラント構成について, 多くの研究がある。^{17) 18) 24) - 27)}

de Bruijn網の場合, ネックレス(シャフルで結合されたノードのサイクル)に対して1つのスペアを設ける, レイアウトの観点からも効率的な方法が開発されている。^{4) - 7)}

ネックレスの数は, N / log₂Nで非常に良く近似でき, またネックレスは結合されてハミルトンサイクルを構成できることが, 明らかにされている。⁷⁾

図1のN=8の場合には, (0)(124)(356)(7)の4つのネックレスがあり, 近似式の値は3である。もっと大きなN, 例えば1024の場合108に対して

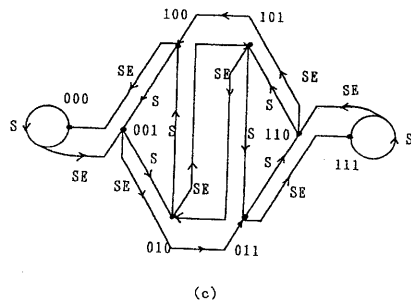
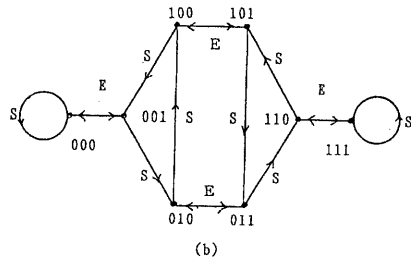
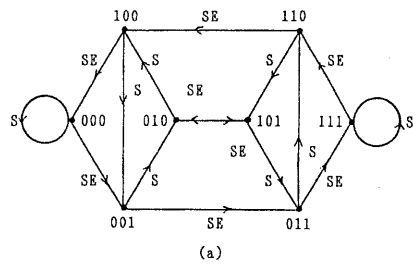


図1 de Bruijn網(a), シャフルエクスチェンジ網(b), 2つの相互関係(c) (N=8)

102, 8192の場合632に対して630, 65536の場合4116に対して4096となる。Nに対するスペアの個数の割合は、それぞれ、0.105, 0.077, 0.063である。図2に冗長なSリンクにより故障したPEのxがスペアPEのsにより置き換えられる様子を示す。⁷⁾

スペアの割合が大きいと考えられる場合には次のような構成も可能である。ネックレスを元のSEリンクと並列に設けられた、冗長なSEリンクによりチェイニングすることによって、大きなサイクルを構成し、このサイクルに1つのスペアを設けることもできる。この究極がハミルトンサイクル、即ち結合網全体に1つのスペアを設ける構成となる。

4. エンベダビリティ

基本的なエンベダビリティに関しては次のことが明らかにされている。^{2) 5)}

1つのハミルトンサイクルといくつかのハミルトンパスを持つので、

- (1) Nノードのリングをエミュレートできる。
- (2) Nノードの線形アレイをエミュレートできる。
- (3) 1つのノードが故障しても、N-1ノードの線形アレイをエミュレートできる。
- (4) 1つのリンクが故障しても、N-1ノードの線形アレイをエミュレートできる。
- (5) 1つのノードが故障しても、少なくともN-1 $\log_2 N-1$ ノードのリングをエミュレートできる。
- (6) 1つのリンクが故障しても、少なくともN-1ノードのリングをエミュレートできる。

シャフルエクステンジリンクは本質的に2進木であるから、

- (7) N-1ノード、高さ $\log_2 N-1$ の完全2進木を少なくとも4つ持つ。
- (8) $3N/4-2$ ノードの二重木を少なくとも4つ持つ。
- (9) 1つのリンクが故障しても、N-1ノード、高さ $\log_2 N-1$ の完全2進木を1つ持つ。
- (10) 1つのノードが故障したとき、 $N/2-1$ ノード、高さ $\log_2 N-2$ の完全2進木を構成できる。

なお、ハイパキューブは、dilation 1では完全2進木を構成できず、単なる中継だけのノードがどうしても生じる。¹⁵⁾

- (11) シャフルエクステンジ網をエミュレートできる。
- (12) メッシュのエミュレーションに関してはハイパキューブの方が優れている。^{15) 16)} de Bruijn網で効率良くメッシュをエミュレートするために、グラフの積による構成(第9節)が有力である。

5. スケーラビリティ

ハイパキューブでは、ノードの次数(PEのI/Oポート数)、即ちノード複雑度がNとともに大きくなるという大きな欠点がある。したがって、小規模構成から大規模構成まで、I/Oポートに無駄が生じないようにしようとするれば、スケーラビリティがなくなるし、逆に、スケーラビリティを持たせる為に、大きなI/Oポートを持つPEを統一的使用すれば、小規模構成ではI/Oポートに無駄が生じる。

de Bruijn網では、次数は4と定数であるから、"小規模構成から大規模構成まで、同じPEノードを使用できる"という意味ではスケーラビリティが良いことになる。

一般に $UB(r, D)$ は $UB(r+1, D)$ のサブグラフであり、 $UB(r, D)$ から $UB(r+1, D)$ を得るアルゴリズムが2つ確立されているが²⁾、これは実用的ではない。ハイパキューブのように、2つの $UB(2, n)$ から簡単に $UB(2, n+1)$ を構成することはできない。

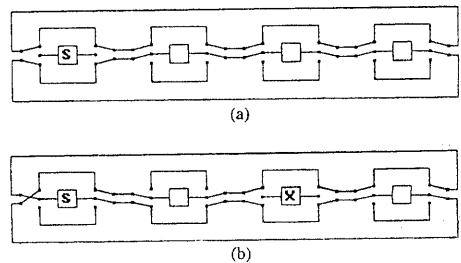


図2 スペアによる置換

” 現有の小規模構成を生かしつつ大規模構成に拡張できる” という意味ではスケーラビリティがないことになる。

6. ルーティングアルゴリズム

有向グラフにおいて、送信元ノード ($s_{n-1} \cdots s_1 s_0$) から受信先ノード ($d_{n-1} \cdots d_1 d_0$) への、原則として長さ n のルーティングは、 $(s_{n-1} \cdots s_1 s_0) \rightarrow (s_{n-2} \cdots s_0 d_{n-1}) \rightarrow \cdots \rightarrow (s_0 d_{n-1} \cdots d_1) \rightarrow (d_{n-1} \cdots d_1 d_0)$ と非常に簡単である。

各ノードで自分自身のアドレスと受信先アドレスを比較し、一致すればそのメッセージを受け入れる。異なっていれば、送信アドレスと受信アドレスのMSBから順に各ビットの排他的論理和をとったタグにより、1ならばSEアークのパスへ、0ならばSアークのパスへ転送すれば良い。

例えば図1(a)でノード(100)から(111)の転送を考えよう。上のアルゴリズムに従えば、排他的論理和は(011)であるからS, SE, SEアークを選択して、 $(100) \rightarrow (001) \rightarrow (011) \rightarrow (111)$ の長さ3のルートになる。

また、例えば(101)から(010)の転送では、タグは(111)であり、1回のSEで受信先ノードに到達する。

このように、次数が4でないノードも特別扱いは必要がなく、アルゴリズムは至極簡単である。

一般に最短パスは、 $(s_{n-1} \cdots s_1 s_0)$ の中で $(s_{n-1} \cdots s_0) = (d_{n-1} \cdots d_{i-1})$ となる最長のサブストリングを見つけることにより得られる。ある i についてそれが得られたとすれば、 $(s_{n-1} \cdots s_1 s_0) \rightarrow (s_{n-1} \cdots s_{n-i+1} d_{n-1} \cdots d_{i-1}) \rightarrow (s_{n-2} \cdots s_{n-i+1} d_{n-1} \cdots d_{i-2}) \rightarrow \cdots \rightarrow (d_{n-1} \cdots d_1 d_0)$ という長さ $i-1$ の最短パスが得られる。¹¹⁾

無向グラフのルーティングは、上で示した左シフトに対応するルーティング(S及びSEアーク)の他に、右シフトに対応するルート(S^{-1} 及びES $^{-1}$ アーク)も加わる。

右シフトによる原則として長さ n のルーティング ($s_{n-1} \cdots s_0) \rightarrow (d_0 s_{n-1} \cdots s_1) \rightarrow \cdots \rightarrow (d_{n-2} \cdots d_0 s_{n-1}) \rightarrow (d_{n-1} \cdots d_1 d_0)$ も存在する。これは上の例では $(100) \rightarrow (110) \rightarrow (111)$ の長さ2のルートとなる。この場合は排他的論理和のタグ(011)をLSBから順に見ていけば良い。

無向グラフでは有向グラフのアークの逆方向アークも利用することになるから、当然無向グラフの最短パスは有向グラフのそれと同じかそれより短い。しかし、無向グラフの最短パスを求める一般的な方法は知られていない。¹¹⁾

実際には、平均距離は直径と大して変わらないので、無理をして最短ルートを探査する必要も無く、非常に簡単な原則として長さ n (n 以下で到達する場合も多い)のルーティングアルゴリズムを使えば十分である。

左シフトに対応するルートと右シフトに対応するルートの短い方を選んでも良いし、更に簡単に無条件に前者(有向グラフの場合と同じ)を選んでも良い。

なお、放送時間は、ハイバキューブの $\log_2 N$ に対して、 $2 \log_2 N$ である。¹¹⁾

7. レイアウト

集積度の向上により1チップ(1ウエーハ)に数十個の簡単なPEを集積することも可能になりつつある。したがって、チップへの効率的なレイアウトの問題も重要である。

図1(c)より明らかのように、de Bruijn網はシャフルエクスチェンジ網のSエッジとEエッジを利用して構成できる。シャフルエクスチェンジ網の各ラインが2本になるので、シャフルエクスチェンジ網の高々2倍のチップ面積が有れば良い。¹²⁾

シャフルエクスチェンジ網を $O(N^2 / \log^2 N)$ の面積にレイアウトする大変効率的な方法が開発されている¹³⁾ので、これを利用すればよい。

なお、ハイバキューブのボード上への配置に関して最近の研究があるが、¹⁴⁾ 当然のことながら de Bruijn網より効率が悪い。

8. 階層構成

如何に集積度が向上しても、大規模並列システム全体を1チップ(1ウエーハ)に集積することはできないから、実際にはチップ内のPE間結合網とチップ間結合網の、少なくとも2層の階層構成になる(コネクションマシンでは、チップ内の4×4のメッシュと、チップ間の1次元ハイバキューブHC(2, 12)の2層)。

2層の階層構成を少し一般的に考察してみる。²²⁾

ノード総数Nを $N_1 = N / N_2$ 個のノードよりなる N_2 個クラスタに分割する。各クラスタの N_1 個のノードをレベル1の結合網で結合し、各クラスタ内の1つの代表ノード(インタフェースノード)をレベル2の結合網(ノード数 N_2)で結合する。

各ノードの $n (= \log_2 N)$ ビットアドレスは上位 $n_2 (= \log_2 N_2)$ ビットのクラスタアドレスと下位 $n_1 (= \log_2 N_1)$ ビットのクラスタ内アドレスから成る($n_1 + n_2 = n$)。

レベル1及びレベル2の結合網それぞれ自身の直径をそれぞれ d_1, d_2 、次数をそれぞれ k_1, k_2 とする。次数の総数の1/2であるリンク総数は、 $(Nk_1 + N_2k_2) / 2$ 、直径は $d_1 + d_2$ となる。

レベル1、レベル2ともハイバキューブの場合について、階層化による価格対性能比(平均距離×リンク総数)の改善率が報告されている。²²⁾ $N_1 = 16$ の場合が良く、約1/2に改善される。

両レベル共de Bruijnの場合には、それぞれの結合網はサブシャフル及びスーパーシャフルになる。de Bruijn網のレイアウトの効率が良いが、メッシュが一番良いのは当然であり、チップ内(レベル1)をメッシュ、チップ間(レベル2)をde Bruijn網にするのが実用的と思われる。

9. グラフの積による構成

一般に2つのグラフ $G_1 = (V_1, E_1)$ 、 $G_2 = (V_2, E_2)$ の積(Cartesian product)、 $G = G_1 \times G_2 = (V, E)$ は次のように定義されている。^{12) 17)}

点については、

$$V = V_1 \times V_2 = \{v_1, v_2\} \quad v_1 \in V_1, v_2 \in V_2$$

枝については(\langle 点1, 点2 \rangle は2つの点間の枝)。

$u_1 = v_1$ で、 $\langle u_2, v_2 \rangle \in E_2$ または、

$u_2 = v_2$ で、 $\langle u_1, v_1 \rangle \in E_1$ のとき、

$$\langle (u_1, u_2), (v_1, v_2) \rangle \in E$$

図3にグラフの積の一例として2つの1次元トーラスの積が2次元トーラスになることを示す。

グラフの積について次の性質がある。

(1) 2つの対称網の積も対称網である。

(2) $u \in G_1$ の次数が k_1 、 $v \in G_2$ の次数が k_2 とすれば、 $(u, v) \in G_1 \times G_2$ の次数は $k_1 + k_2$ である。

(3) G_1, G_2 の直径がそれぞれ d_1, d_2 ならば、 $G_1 \times G_2$ の直径は $d_1 + d_2$ である。

n 次元ハイバキューブHC(2, n)は、 $n-1$ 次元ハイバキューブHC(2, $n-1$)と2ノードの完全結合網 K_2 との積で定義でき、さらにこの操作を繰り返せば、 K_2^n と定義することもできる。

図3の例から明らかなように、2つのリング(1次元トーラス)HT($\sqrt{N}, 1$)の積は2次元トーラスHT($\sqrt{N}, 2$)になる。

de Bruijn網UB(2, n)と K_2 の積をとれば、UB(2, n)の各PEアドレスの頭に0及び1の1ビットを付加し、対応するPEノード同士をリンクで結合した2Nノード、次数5の結合網が得られるが、これは勿論UB(2, $n+1$)ではない。

2つのde Bruijn網の積に基づく結合網は、de Bruijn網が線形アレイ、リング(1次元トーラス)をエミュレートできることから明らかなように、メッシュ、2次元トーラス、木のメッシュもサブグラフとして含んでいる。¹¹⁾

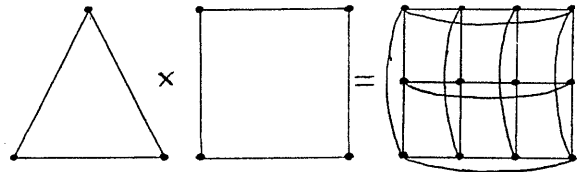


図3 グラフの積の一例

(リング×リング=2次元トーラス)

10. ダイグラフ (単向リンク) による構成

2次元トラスのリンクを単方向にしたマンハッタンストリート網では、直径が \sqrt{N} から $\sqrt{N}+1$ に、平均距離が $N^{3/2}/(2N-2)$ から $(N^{3/2}+2N-8)/(2N-2)$ になるだけで、性能低下は小さく、²²⁾ 価格対性能比を約半分にできる。

de Bruijn 網でもダイグラフ (単向リンク) で構成すれば、ノードの構成は図4に示すように簡単になり (クロスバ網のクロスポイント数は25から9に)、価格対性能比 (クロスポイント数 / 直径) は半分以下になる (クロスバ網以外のノード構成もある^{19) 20)})。

勿論、FT特性とEB特性の点で、双方向リンクの場合に比して劣るのは止むを得ず、要はトレードオフの問題である。

11. 間接網への展開

ノード数 N の有向 de Bruijn 網 $U(2, n)$ ($n = \log_2 N$) を間接網に展開すれば、 2×2 クロスバスイッチよりなる n 段オメガ網 (間接キューブ網と等価) になる。一般に N ノードの $U(2^k, n/k)$ ($k=1, 2, 3, \dots$) は、 $2^k \times 2^k$ クロスバスイッチ n/k 段よりなるオメガ網になる¹⁹⁾ (クレイ社の超並列コンピュータは 4×4 クロスバスイッチよりなるオメガ網を採用予定)。間接網の段数は常に直接網の直径に等しい。

$2^k \times 2^k$ クロスバスイッチ構成のオメガ網の必要とするライン数とクロスポイント数はそれぞれ $N(n/k+1)$ 、 $(2^k/k)Nn$ となる。クロスポイント数と段数 n/k の積、即ち価格対性能比は $k=3$ 、即ち 8×8 クロスバスイッチのとき最良となる。

これに対して、直接網 $U(2^k, n/k)$ の場合はそれぞれ、 $N(2^k+2)$ 、 $N(2^k+1)^2$ となり、同様にして $k=1$ 、即ち2進の場合が最良となる。

参考までにハイバキューブ $HC(2^k, n/k)$ の場合は、それぞれ、 $N((2^k-1)n+2)$ 、 $N((2^k-1)n+1)^2$ 、となり、やはり2進が最良であることが分かる。

さらに無向 de Bruijn 網 $UB(2^k, n/k)$ 、2次元トラス、及びマンハッタンストリート網の場

合はそれぞれ、ライン数は $2^{k+1}+2$ 、 $6N$ 、 $4N$ ；クロスポイント数は $N(2^{k+1}+1)^2$ 、 $25N$ 、 $9N$ となる。

12. おわりに

本報告では2進 de Bruijn 網の静的特性を考察した。静的な価格対性能比 (直径 \times 次数) の超並列における優位性は勿論のこと、FASAL 等の諸特性もハイバキューブに勝るとも劣らない。ソーティング網としても非常に優れていることが示されており²¹⁾、超並列向きの直接結合型の相互結合網として有力であると言って良いだろう。メッシュのエミュレーションの点でのみハイバキューブに劣るが、グラフの積による構成はこの点に関して大変有効である (次数は8になる)。さらにシミュレーションなどにより動的特性を総合的に評価することが今後の課題である。

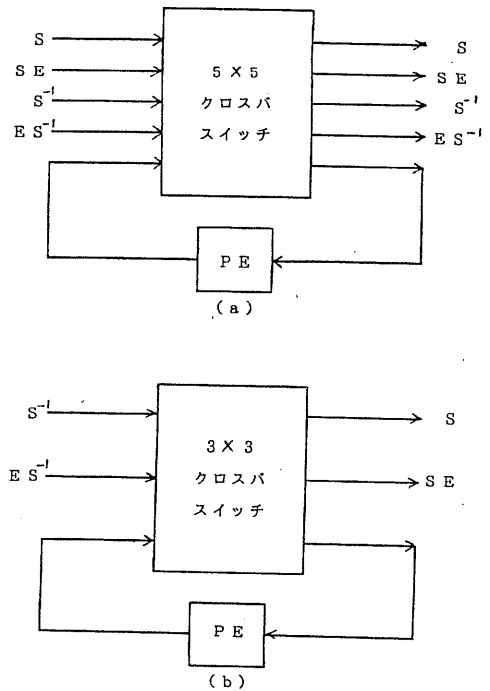


図4 双方向リンク (a) と単方向リンク (b) の場合のノード構成

謝辞

参考文献 5), 6), 及び参考文献 5) のフルペーパーを送って下さった, 慶応義塾大学 工学部 天野英晴先生の親切に感謝します。

参考文献

- (1) J-C Bermond, C. Peyrat: "De Bruijn and Kautz Networks: A Competitor for the Hypercube", pp.279-293, in F. Andre, J.P. Verjus (Eds.): "Hypercube and Distributed Computers", North-Holland (1989).
- (2) M.R. Samatham, D.K. Pladhan: "The de Bruijn Multiprocessor Networks: A Versatile Parallel Processing and Sorting Networks for VLSI", IEEE Trans. Comput., 38, 4, pp.567-581 (1989).
- (3) Correction to the above, IEEE Trans. Comput. 40, 1, pp.122-123 (1991).
- (4) R. Rowley, B. Bose: "On Necklaces in Shuffle-Exchange and Bruijn Networks", Proc. 1990 ICCP, pp. I-347-350.
- (5) -----: "Fault-Tolerant Ring Embedding in the de Bruijn Networks", Proc. 1991 ICCP, pp. I-710-711.
- (6) -----: "Edge-Disjoint Hamiltonian Cycles in de Bruijn Networks", Proc. 6th Distributed Memory Comput. Conf., pp.707-709 (1991).
- (7) W. Liu, et al.: "Hamiltonian Cycles in the Shuffle-Exchange Network", IEEE Trans. Comput., 38, 5, pp.745-750 (1989).
- (8) 今瀬 真, 他: "障害耐力のあるプロセッサ相互結合ネットワーク", 信学論 (D), J68-D, 8, pp.1449-1456 (1985).
- (9) 奥川峻史: "超並列コンピュータ向きアーキテクチャ", bit, 22, 7, pp.704-714 (1990).
- (10) -----: 並列計算機アーキテクチャ, 第3, 4章, コロナ社, 1991.
- (11) -----: "超並列コンピュータ向き結合網の検討", 信学技報, 91, 130, CPS91-13, pp.75-82 (1991).
- (12) V.P. Roychowdhury, et al.: "Decoding of Rate k/n Convolutional Codes in VLSI", Chap.33, in S.K. Tewksbury, et al. (Eds.): "Concurrent Computations", Plenum Press (1988).
- (13) D. Kleitman, et al.: "New Layouts for the Shuffle-Exchange Graph", Proc. 13th ACM Symp. Theory of Computation, pp.278-292 (1981).
- (14) T-H Lai, A.P. Sprague: "Placement of the Processors of a Hypercube", IEEE Trans. Comput., 40, 6, pp.714-722 (1991).
- (15) S. Ranka, S. Sahni: Hypercube Algorithms, Springer-Verlag, 1990.
- (16) M. A. A. Henriques, H. Ito: "Large Scale Rectangular Grids in Hypercubes: An Embedding Scheme and its Evaluation", IEICE Trans., E74, 6, pp.1705-1714 (1991).
- (17) 石川 勉: "切換え手段を不要としたフォールトトレラントハイパキューブの構成法" 信学論 (D-I), J72-D-I, 2, pp.117-125 (1989).
- (18) 伊藤秀男, 鈴木信行: "ウェーハスケールLSI上でのハイパキューブの構成法", 信学論 (D-I) J73-D-I, 3, pp.314-323 (1990).
- (19) S. Abraham, K. Padmanabhan: "Performance of the Direct Binary n -Cube Network for Multiprocessors", IEEE Trans. Comput., 38, 7, pp.1000-1011 (1989).
- (20) P.W. Dowd, et al.: "Static Interconnection Network Extensibility based on Marginal Performance/Cost Analysis", Proc. IEE, 136, Pt. B, 1, pp.9-15 (1989).
- (21) A. El-Amawy, S. Latif: "Properties and Performance of Folded Hypercubes", IEEE Trans. Parallel and Distributed Systems, 2, 1, pp.31-42 (1990).
- (22) S.P. Dandamuli, D.L. Eager: "Hierarchical Interconnection Networks for Multicomputer Systems", IEEE Trans. Comput., 39, 6, pp.786-797 (1990).
- (23) T.Y. Chung, D.P. Agrawal: "Cost-Performance Trade-offs in Manhattan Street Network versus 2-D Torus", Proc. 1990 ICCP, pp. I-169-172.
- (24) M. Chean, J. A. B. Fortes: "The Full-Use-of-Suitable-Spares (FUSS) Approach to Hardware Reconfiguration for Fault-Tolerant Processor Arrays", IEEE Trans. Comput., 39, 4, pp.564-571 (1990).
- (25) M. S. Alam, R.G. Melhem: "An Efficient Modular Spare Allocation Scheme and Its Application to Fault Tolerant Binary Hypercubes", IEEE Trans. Parallel and Distributed Systems, 2, 1, pp.117-126 (1991).
- (26) -----: "How to use an Incomplete Binary Hypercube for Fault Tolerance", pp.329-341, in F. Andre, J.P. Verjus (Eds.) "Hypercube and Distributed Computers", North Holland (1989).
- (27) H.P. Katseff: "Incomplete Hypercube", IEEE Trans. Comput., 37, 5, pp.604-608 (1988).