

マルチプロセッサシステムにおける並列データ 伝送問題の研究への一寄与†

横 山 正 明**

数十～数百のプロセッサから構成された並列計算機においては、プロセッサ相互間あるいはプロセッサ-メモリ間のデータ伝送の交通整理をする相互結合ネットワークの機能の優劣が計算機システム全体の性能に大きな影響を与える。このため、これまで数多くの相互結合ネットワークが提案されている。しかし、これらのネットワークにおいても、データ伝送による遅延を示すステップ数あるいはプロセッサ間距離に関しては上限あるいはオーダが与えられるのみで、確定したものはない。それゆえに、本研究は相互結合ネットワークの一つとして、端のない n 次元立方体状のネットワークを取りあげ、各軸方向のプロセッサの数とその結合方式を変化させて、ステップ数の確定を行っている。

1. 結 言

SOLOMON 計算機¹⁾ から現在研究開発中の工業技術院の科学技術用高速計算システムに至るまでに、数十～数百のプロセッサから構成された SIMD 型や MIMD 型の並列計算機が数多く計画され、あるものは試作された。このような高度に並列化された計算機ではプロセッサ相互間あるいはプロセッサ-メモリ間のデータ伝送の交通整理をする相互結合ネットワークの機能の優劣が計算システム全体の性能に大きな影響を与える。このため、これまでに数多くの相互結合ネットワークが提案されている²⁾⁻⁶⁾。しかし、提案された多くのネットワークにおいても、データ伝送による遅延を示す「ステップ数」あるいは「距離」に関しては上限あるいはオーダが与えられるのみで、全プロセッサを同時に動作させたときに干渉なくデータ伝送を行うためのステップ数あるいは距離について確定したものはないように見受けられる。

著者も技術計算用を目標に「連装計算機」と名づけた SIMD 型の並列計算機を計画したことがあるが⁷⁾、この計算機には、 n 次元立方体状にプロセッサを配列し、隣接プロセッサと結合した端のない相互結合ネットワークを採用した。図1に三次元立方体状のプロセッサの配置とプロセッサ間結合の一部を示すが、各軸方向の両端にある二つのプロセッサを、この図に二つの場合を示すように結合すれば著者の採用した端のな

い相互結合ネットワークになる。

この n 次元立方体状の相互結合ネットワークを用いれば、 $2n-1$ 回の手順の繰返しで全くランダムに配列されたデータの並列伝送が可能である。もちろん、 $2n-1$ ステップでデータ伝送が完了するのではなく、必要なステップ数は手順の繰返し数 $2n-1$ と各手順の実行に要するステップ数の積である。各手順の実行に要するステップ数は、各軸方向のプロセッサが完全グラフ状に結合されている場合は1であるが、そうでない場合は1より大きくなる。前の報告⁷⁾では4ⁿタイプのプロセッサ配列を考え、各軸方向の4ユニットのプロセッサは完全グラフ状に結合された場合を取り扱った。

本報告ではこれをより一般化するために、各軸方向のプロセッサ数を64ユニットまでふやし、また、完全グラフ状の結合以外の結合方式を取りあげ、ステップ数の確定ならびにデータ伝送路の決定法に関して行った研究の結果について報告する。

2. 並列データ伝送のステップ数確定のための基本定理

次章において各軸方向のプロセッサの総数、その結合方式と並列データ伝送のステップ数の関係をもとめるが、そのための準備として基本定理をのべておく。

本研究では各軸方向のプロセッサの結合は端のない構造をとるので、プロセッサ配列は有弦のリング状となる。すなわち、図2のようにプロセッサ PU_i ($i=1, 2, \dots, m$) を小円で表し、これを大円の円周上に等配する。PU間のデータ授受を行う伝送路は小円を結ぶ実線で表す。本研究では m は4の倍数とし、伝送路は双方向にデータを同時伝送できるものとする。ただ

† A Contribution to the Study of Parallel Transmission of Data in Multi-processors System by MASA AKI YOKOYAMA (Department of Precision Machinery and Systems Graduate School of Science and Engineering, Tokyo Institute of Technology).

** 東京工業大学総合理工学研究科精密機械システム専攻

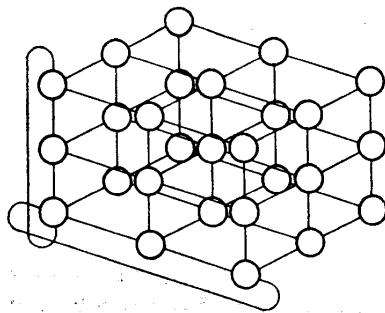


図1 三次元立方体状のプロセッサの配置
Fig. 1 Allocation of processors into a cube.

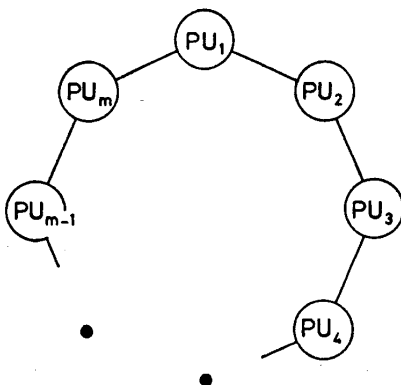


図2 弦結合のないリング状結合をもつ m ユニットのプロセッサ
Fig. 2 Interconnection of m units of processors with ring connection.

し、同時刻に同一の PU には2個以上は入り得ないものとする。

全データが目的 PU に到達するまでにいくつの PU を通過しなければならないかをステップ数で表すが、ある PU からみて1ステップでデータ授受できる PU の集合をその PU の近傍と呼ぶ。全 PU について近傍の数は同じものとする。PU の近傍を $\pm 1, \pm 2, \dots$ のように表す。たとえば近傍 ± 1 は図2のように両隣りの PU が近傍であることを示す。また、近傍 $\oplus 1$ のように複号を円で囲むと PU によって近傍のとり方が異なることを意味し、この場合は $+1$ の PU を近傍とする PU と -1 の PU を近傍とする PU に分かれる。たとえば、図3の各 PU の近傍は $\oplus 1, \oplus 2, \oplus 4$ で与えられる。

次に、基本定理をのべる。自明のものは証明を省略する。

[定理1] m ユニットの PU があり、どの PU も他の $m-1$ ユニットの PU と伝送路によって直接結合されている場合、並列データ伝送のステッ

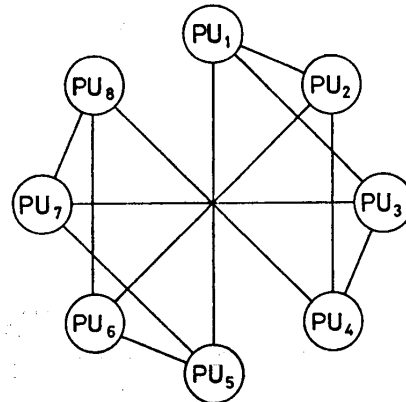


図3 近傍 $\oplus 1, \oplus 2, \oplus 4$ の結合状態
Fig. 3 Interconnection of eight processors with neighbours of $\oplus 1, \oplus 2, 4$.

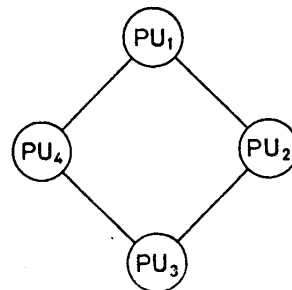


図4 近傍 ± 1 をもつ4ユニットのプロセッサ
Fig. 4 Interconnection of four processors with neighbours of ± 1 .

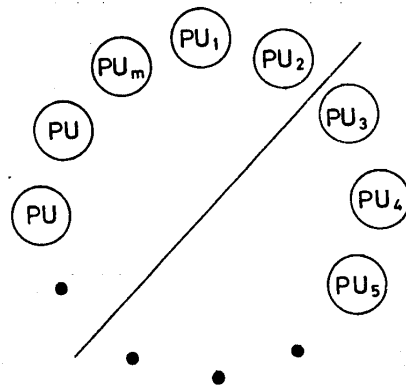


図5 中心を通る直線によるプロセッサの等数分割
Fig. 5 Division of processors into two groups by a straight line.

プ数は1である。

[定理2] 図4のような伝送路をもつ4ユニットの PU を、2ユニットずつの2グループに分けたとき、それぞれのデータを伝送目的 PU と同じグループに属する PU へ重複しないように1ステップで伝送することができる。

[定理3] 図5のように PU が m ユニットあると

き、中心を通る直線によって、伝送目的 PU の番号が奇数である PU と偶数である PU が $m/4$ ユニットずつ属する 2 グループに分けることができる。

[証明] m ユニットの PU を中心を通る任意の有向線分で 2 グループに分けたとき、有向線分の右側にある PU の中で、伝送目的 PU の番号が奇数である PU の数を a ($0 \leq a \leq m/2$) とすると、この有向線分を中心まわりに時計方向に 180° 回転するとこの PU の数は $m/2 - a$ になる。ところで、有向線分を中心まわりに時計方向に 1 PU 分だけ回転したとき、この PU の数の変化はたかだか 1 であるから、明らかに 180° 回転する途中に、この PU の数が $m/4$ になるところがある。また、そのとき 1 グループに属する PU の数は $m/2$ であるから、伝送目的 PU の番号が偶数である PU の数も $m/4$ となる。ゆえに、中心を通る直線によって、伝送目的 PU の番号が奇数である PU と偶数である PU が $m/4$ ユニットずつ属する 2 グループに分けることができる。

[定理 4] 図 6 のように、 m ユニットの PU が属する 2 グループが m 本の伝送路によって結合されている場合、グループ内での並列伝送が S ステップで完了すると仮定すると、 $2m$ ユニットの PU 間の並列伝送は

$S=1$ のとき 2 ステップ

$S \neq 1$ のとき $S+2$ ステップ

で完了する。

[証明] $S=1$ のとき：A グループに属する PU の中でその伝送目的 PU が B グループに属している PU の数を l ($l \leq m$) とすると、逆に B グループに属する PU の中でその伝送目的 PU が A グループに属している PU の数も l である。データの並列伝送は次の 2 ステップで完了する。

ステップ 1：それぞれのグループの中でその伝送目的 PU が A グループに属している PU の数も l である。データの並列伝送は次の 2 ステップで完了する。

ステップ 2：すべてのデータを伝送目的 PU へ伝送する。

$S \neq 1$ のとき： $2m$ ユニットが $2 \times m$ の二次元配列しているので、3 回の手順の繰返しで並列伝送が完了する。各手順は 1 ステップ、 S ステップ、1 ステップを必要とするから、合計 $S+2$ ステップで並列伝送が完了する。

3. 各軸方向のプロセッサの総数、その結合方式と並列データ伝送のステップ数の関係

3.1 各軸 4 PU のシステムの並列データ伝送

3.1.1 近傍 ± 1 の場合

この場合、近傍数=2、ステップ数=2 である。

[証明] (PU₁, PU₂) と (PU₃, PU₄) の 2 グループに分け、定理 4 を適用する。

3.2 各軸 8 PU のシステムの並列データ伝送

3.2.1 近傍 $\oplus 1, \pm 2, \oplus 4$ の場合

この場合、近傍数=4、ステップ数=2 である。

[証明] 8 ユニットの PU を奇数番と偶数番の 2 グループに分ける。各グループは完全グラフ状の伝送路をもつから、グループ内の並列伝送は 1 ステップで完了する。よって定理 4 より並列伝送は 2 ステップで完了する。

3.2.2 近傍 $\pm 1, \pm 2$ の場合

この場合、近傍数=4、ステップ数=3 である。

[証明] 定理 3 を適用し、中心を通る直線によって伝送目的 PU の番号が奇数である PU と偶数である PU が 2 ユニットずつ属する 2 グループに分ける。

ステップ 1：各グループの中で伝送目的 PU の番号が奇数である PU のデータを奇数番、偶数である PU のデータを偶数番の PU へ重複しないように伝送する。この伝送は定理 2 より 1 ステップで完了する。

ステップ 2, 3：奇数番の 4 ユニットが属するグループと偶数番の 4 ユニットが属するグループの中でそれぞれ 3.1.1 と同じ手順で伝送する。

3.2.3 近傍 $\pm 1, \pm 3$ の場合

この場合、近傍数=4、ステップ数=3 である。

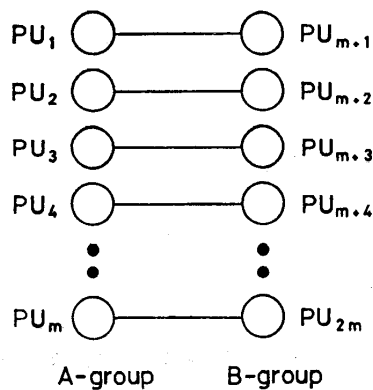


図 6 プロセッサの 2 グループ分割とグループ間結合
Fig. 6 Division of processors into two groups and interconnection between these two groups.

【証明】 PUの一部について次のように番号換えをする。 $PU_2 \rightarrow PU_3$, $PU_3 \rightarrow PU_2$, $PU_6 \rightarrow PU_7$, $PU_7 \rightarrow PU_6$ 。この新番号を使って3.2.2と同じ手順で伝送する。

3.2.4 近傍 $\oplus 1, \oplus 2, \oplus 4$ の場合

この場合、近傍数=3, ステップ数=4である。

【証明】 8ユニットのPUを奇数番と偶数番の2グループに分ける。各グループは3.1.1と同じ伝送路をもつから、グループ内での並列伝送は2ステップで完了する。よって、定理4より8ユニットのPU間の並列伝送は4ステップで完了する。

3.2.5 近傍 $\pm 1, \oplus 4$ の場合

この場合、近傍数=3, ステップ数=4である。

【証明】 8ユニットのPUを (PU_1, PU_2, PU_5, PU_6) と (PU_3, PU_4, PU_7, PU_8) の2グループに分ける。各グループは3.1.1と同じ伝送路をもつから、グループ内での並列伝送は2ステップで完了する。よって、定理4よりステップ数は4となる。

3.2.6 近傍 $\pm 1, \oplus 3$ の場合

この場合、近傍数=3, ステップ数=4である。

【証明】 3.2.5と同じ手順で伝送する。

3.3 各軸 16PU のシステムの並列データ伝送

3.3.1 近傍 $\oplus 1, \pm 2, \pm 4, \pm 6, \oplus 8$ の場合

この場合、近傍数=8, ステップ数=2である。

【証明】 16ユニットのPUを奇数番と偶数番の2グループに分ける。各グループは完全グラフ状の伝送路をもつから、グループ内での並列伝送は1ステップで完了する。よって、定理4よりステップ数は2となる。

3.3.2 近傍 $\oplus 1, \oplus 2, \pm 4, \oplus 8$ の場合

この場合、近傍数=5, ステップ数=4である。

【証明】 PUを奇数番と偶数番の2グループに分ける。各グループは3.2.1と同じ伝送路をもつから、グループ内の並列伝送は2ステップで完了する。よって、ステップ数は定理4より4となる。

3.3.3 近傍 $\oplus 1, \pm 2, \pm 4$ の場合

この場合、近傍数=5, ステップ数=5である。

【証明】 PUを奇数番と偶数番の2グループに分ける。各グループは3.2.2と同じ伝送路をもつから、グループ内の並列伝送は3ステップで完了する。よって、定理4よりステップ数は5となる。

3.3.4 近傍 $\oplus 1, \pm 2, \pm 6$ の場合

この場合、近傍数=5, ステップ数=5である。

【証明】 PUを奇数番と偶数番の2グループに分け

る。各グループは3.2.3と同じ伝送路をもつから、グループ内の並列伝送は3ステップで完了する。よって、定理4よりステップ数は5となる。

3.3.5 近傍 $\oplus 1, \oplus 2, \oplus 4, \oplus 8$ の場合

この場合、近傍数=4, ステップ数=6となる。

【証明】 PUを奇数番と偶数番の2グループに分ける。各グループは3.2.4と同じ伝送路をもつから、グループ内の並列伝送は4ステップで完了する。よって、定理4よりステップ数は6となる。

3.3.6 近傍 $\oplus 1, \pm 2, \oplus 8$ の場合

この場合、近傍数=4, ステップ数=6である。

【証明】 PUを奇数番と偶数番の2グループに分ける。各グループは3.2.5と同じ伝送路をもつから、グループ内の並列伝送は4ステップで完了する。よって、定理4よりステップ数は6となる。

3.4 各軸 32PU のシステムの並列データ伝送

3.4.1 近傍 $\oplus 1, \pm 2, \pm 4, \pm 6, \pm 8, \pm 10, \pm 12, \pm 14, \oplus 16$ の場合

この場合、近傍数=16, ステップ数=2である。

【証明】 PUを奇数番と偶数番の2グループに分ける。各グループは完全グラフ状の伝送路をもつから、各グループ内での並列伝送は1ステップで完了する。よって、定理4よりステップ数は2となる。

3.4.2 近傍 $\oplus 1, \oplus 2, \pm 4, \pm 8, \pm 12, \oplus 16$ の場合

この場合、近傍数=9, ステップ数=4である。

【証明】 PUを奇数番と偶数番の2グループに分ける。各グループは3.3.1と同じ伝送路をもつから、グループ内での並列伝送は2ステップで完了する。よって、定理4よりステップ数は4となる。

3.4.3 近傍 $\oplus 1, \oplus 2, \oplus 4, \pm 8, \oplus 16$ の場合

この場合、近傍数=6, ステップ数=6である。

【証明】 同様に、偶奇の2グループに分けると、各グループは3.3.2と同じ伝送路をもつので、定理4よりステップ数は6となる。

3.4.4 近傍 $\oplus 1, \oplus 2, \pm 4, \pm 8$ の場合

この場合、近傍数=6, ステップ数=7である。

【証明】 同様に偶奇の2グループに分けると、各グループは3.3.3と同じ伝送路をもつので、定理4よりステップ数は7となる。

3.4.5 近傍 $\oplus 1, \oplus 2, \pm 4, \pm 12$ の場合

この場合、近傍数=6, ステップ数=7である。

【証明】 同様に、偶奇の2グループに分けると、各グループは3.3.4と同じ伝送路をもつので、定理4よりステップ数は7となる。

3.4.6 近傍 ⊕1, ⊕2, ⊕4, ⊕8, ⊕16 の場合

この場合、近傍数=5、ステップ数=8 である。

[証明] 同様に、偶奇の2グループに分けると、各グループは3.3.5と同じ伝送路をもつので、定理4よりステップ数は8となる。

3.4.7 近傍 ⊕1, ⊕2, ±4, ⊕16 の場合

この場合、近傍数=5、ステップ数=8 である。

[証明] 同様に、偶奇の2グループに分けると、各グループは3.3.6と同じ伝送路をもつので、定理4よりステップ数は8となる。

3.5 各軸 64 PU のシステムの並列データ伝送

3.5.1 近傍 ⊕1, ±2, ±4, ..., ±28, ±30, ⊕32 の場合

この場合、近傍数=32、ステップ数=2 である。

[証明] 同様に、偶奇の2グループに分けると、各グループは完全グラフ状の伝送路をもつ。各グループ内の並列伝送は1ステップで完了するから、定理4よりステップ数は2となる。

3.5.2 近傍 ⊕1, ⊕2, ±4, ±8, ..., ±28, ⊕32 の場合

この場合、近傍数=17、ステップ数=4 である。

[証明] 同様に、偶奇の2グループに分けると、各グループは3.4.1と同じ伝送路をもつから、定理4よりステップ数は4となる。

3.5.3 近傍 ⊕1, ⊕2, ⊕4, ±8, ±16, ±24, ⊕32 の場合

この場合、近傍数=10、ステップ数=6 である。

[証明] 同様に、偶奇の2グループに分けると、各グループは3.4.2と同じ伝送路をもつから、定理4よりステップ数は6となる。

3.5.4 近傍 ⊕1, ⊕2, ⊕4, ⊕8, ±16, ⊕32 の場合

この場合、近傍数=7、ステップ数=8 である。

[証明] 同様に、偶奇の2グループに分けると、各グループは3.4.3と同じ伝送路をもつから、定理4よりステップ数は8となる。

3.5.5 近傍 ⊕1, ⊕2, ⊕4, ±8, ±16 の場合

この場合、近傍数=7、ステップ数=9 である。

[証明] 偶奇に分けた2グループが3.4.4と同じ伝送路をもつから、定理4よりステップ数は9となる。

3.5.6 近傍 ⊕1, ⊕2, ⊕4, ±8, ±24 の場合

この場合、近傍数=7、ステップ数=9 である。

[証明] 偶奇に分けた2グループが3.4.5と同じ伝送路をもつから、定理4よりステップ数は9となる。

3.5.7 近傍 ⊕1, ⊕2, ⊕4, ⊕8, ⊕16, ⊕32 の場合

この場合、近傍数=6、ステップ数=10 である。

[証明] 偶奇に分けた2グループが3.4.6と同じ伝送路をもつから、定理4よりステップ数は10となる。

3.5.8 近傍 ⊕1, ⊕2, ⊕4, ±8, ⊕32 の場合

この場合、近傍数=6、ステップ数=10 である。

[証明] 偶奇に分けた2グループが3.4.7と同じ伝

表1 並列データ伝送のステップ数

Table 1 Numbers of steps required for parallel data transmission.

8 PU			16 PU		
近傍	N_n	N_s	近傍	N_n	N_s
他の全 PU	7	1	他の全 PU	15	1
⊕1, ±2, ⊕4	4	2	⊕1, ±2, ±4, ±6, ⊕8	8	2
±1, ±2	4	3	⊕1, ⊕2, ±4, ⊕8	5	4
±1, ±3	4	3	⊕1, ±2, ±4	5	5
⊕1, ⊕2, ⊕4	3	4	⊕1, ±2, ±6	5	5
±1, ⊕4	3	4	⊕1, ⊕2, ⊕4, ⊕8	4	6
±1, ⊕3	3	4	⊕1, ±2, ⊕8	4	6

N_n : 近傍数, N_s : ステップ数

表2 並列データ伝送のステップ数

Table 2 Numbers of steps required for parallel data transmission.

32 PU			64 PU		
近傍	N_n	N_s	近傍	N_n	N_s
他の全 PU	31	1	他の全 PU	63	1
⊕1, ±2, ±4, ±6, ±8, ±10, ±12, ±14, ⊕16	16	2	⊕1, ±2, ±4, ±6, ..., ±28, ±30, ⊕32	32	2
⊕1, ⊕2, ±4, ±8, ±12, ⊕16	9	4	⊕1, ⊕2, ±4, ±8, ±12, ..., ±24, ±28, ⊕32	17	4
⊕1, ⊕2, ⊕4, ±8, ⊕16	6	6	⊕1, ⊕2, ⊕4, ±8, ±16, ±24, ⊕32	10	6
⊕1, ⊕2, ±4, ±8	6	7	⊕1, ⊕2, ⊕4, ⊕8, ±16, ⊕32	7	8
⊕1, ⊕2, ±4, ±12	6	7	⊕1, ⊕2, ⊕4, ±8, ±16	7	9
⊕1, ⊕2, ⊕4, ⊕8, ⊕16	5	8	⊕1, ⊕2, ⊕4, ±8, ±24	7	9
⊕1, ⊕2, ±4, ⊕16	5	8	⊕1, ⊕2, ⊕4, ⊕8, ±16, ⊕32	6	10

N_n : 近傍数, N_s : ステップ数

送路をもつから、定理4よりステップ数は10となる。

3.6 まとめ

前節までにおいて説明した各軸方向のプロセッサ数、近傍のとり方と並列データ伝送のステップ数の関係を表1、表2にまとめている。実際の n 次元立方体状の相互結合ネットワークにおけるステップ数は、これらの表中のステップ数に $2n-1$ を乗算したものになる。

4. 結 言

数十～数百のプロセッサから構成された並列計算機ではプロセッサ間あるいはプロセッサ-メモリ間のデータ伝送の交通整理をする相互結合ネットワークの機能の優劣が計算機システム全体の性能に大きな影響を与えるために、従来数多くの相互結合ネットワークが提案されている。しかし、これらのネットワークにおいて、データ伝送による遅延を示すステップ数あるいはプロセッサ間距離について確定したものはない。それゆえに、本研究においては n 次元立方体状にプロセッサを配列し、隣接プロセッサと結合した端のない相互結合ネットワークを取りあげ、各軸方向のプロセッサ数ならびにその結合方式を種々変化させて、並列データ伝送に要するステップ数の確定を行った。

謝辞 終りに、日頃有益な助言とご協力をいただいている本学大学院総合理工学研究科 林 国一教授に厚く謝意を表す。

参 考 文 献

1) Bouknight, W. J. et al.: The Illiac IV System, Proc. IEEE, Vol. 60, pp. 369-388 (1972).

- 2) Thompson, C. D.: Generalized Connection Networks for Parallel Processor Intercommunication, *IEEE Trans. Comput.*, Vol. C-27, pp. 1119-1125 (1978).
 - 3) Siegel, H. J.: Interconnection Networks for Parallel and Distributed Processing, *IEEE Trans. Comput.*, Vol. C-30, pp. 245-246 (1981)
 - 4) 田中善一郎: 並列処理システムの性能を左右する相互ネットワーク, 日経エレクトロニクス, 12月21日号, pp. 88-108 (1981).
 - 5) 高橋義造: 並列処理のためのプロセッサ結合方式, 情報処理, Vol. 23, No. 3, pp. 201-209 (1982).
 - 6) Hwang, K. and Briggs, F. A.: *Computer Architecture and Parallel Processing*, McGraw-Hill, New York (1984).
- 以上は主に解説記事であるが、これらに入っていない最近のわが国の研究者の論文の一部を列挙すると次のようになる:
- 成瀬 正, 雨宮真人: 科学技術計算向きデータフロー計算機に用いる相互結合ネットワークの性能評価, 電子通信学会技術研究報告, EC 82-36, pp. 31-42 (1982).
- 熊谷 毅, 池谷和夫: 並列オメガネットワークの性質について, 電子通信学会論文誌, Vol. J 68-D, No. 3, pp. 245-252 (1985).
- 小林 洋, 古賀義亮: 有弦環結合ネットワーク型コンピュータによるソーティング, 電子通信学会論文誌, Vol. J 68-D, No. 3, pp. 253-260 (1985).
- 7) 横山正明, ほか2名: 高速連装計算機による工学問題のダイレクトシミュレーション (第3報, 連装計算機の内部データ伝送方式), 日本機械学会論文集, Vol. 44, No. 382, pp. 2001-2008 (1978).

(昭和60年4月17日受付)

(昭和60年9月19日採録)