# 多数個プロセッサ・アレイ－バス・アレイ

## A MANY-PROCESSOR ARRAY - BUS ARRAY -

相原玲二　岡田高幸　栄藤稔　阿江忠
Reiji Aibara　Takayuki Okada　Minoru Eto　Tadashi Ae
広島大学　工学部
Faculty of Engineering , Hiroshima University

## 1．まえがき

VLSI技術などの進歩のおかげで，高機能かつ高集積度のマイクロプロセッサやメモリが安く供給されるようになり，$10^3$～$10^4$個のマルチプロセッサの製作も可能となりつつある。このような現況に促して多数個マルチプロセッサの製作も行なわれている[1]。

我々も，まず，小規模のマルチプロセッサからつくり始め[2]，その実験結果からプロセッサ間通信のもつ重要さを定量的に指摘した[3]。多数個マルチプロセッサの有用性は，専用プロセッサにあるとしても，目的に合致し，かつ，もっとも有用な結合形状は必ずしも簡単には判明しない。

むろん，マルチプロセッサにおける結合形状については，多くの提案があり[4]，目的に応じてその選択を行なうことはできる。

本稿では，従来からよく製作される一つの形状であるアレイをとりあげる。ただし，従来のプロセッサ・アレイより，横のつながりに柔軟性をもたせた構造を新たに提案する。このようなプロセッサ・アレイ（バス・アレイと呼ぶ）は製造が簡単でかつ，応用範囲も広いという特徴をもつ。

## 2．結合ネットワークの形状

結合ネットワークは一般的には図1のような役割をもつ[5]。ここに

$P_i$ ： プロセッサ

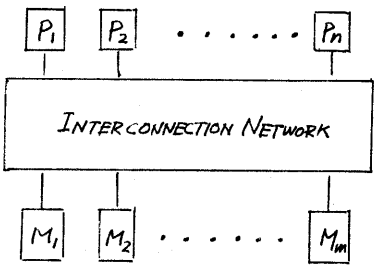



図1．結合ネットワーク

$M_j$ ： メモリ

である。

さらに細かい議論をする場合，メモリを

PM ： プロセッサに個有のメモリ

CM ： 共有メモリ

に分けた方がよい。むろん，この分け方は結合ネットワークを扱う上のもので，プロセッサ$P_i$に個有の$PM_i$を他のプロセッサ$P_j$がアクセスできるか否かという問題とは直接関係はない。

この表現を用いると我々が初期に製作し，実験してきたマルチプロセッサAMOVSTは図2のような結合ネットワークであらわされる[6]-[8]。図2からわかるように$P_i$と$PM_i$はまとめて1つのノードとしてあらわせる。問題は共有メモリCMを陽にあらわすか否かである。物理的には陽に存在するものであるが，図2のような小規模のマルチプロセッサは別として，ここで対象とする$10^3$～$10^4$をめざし

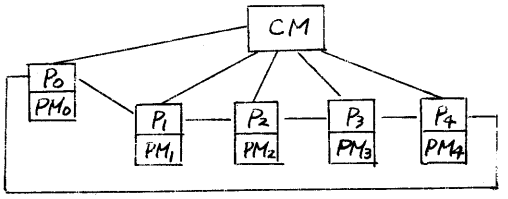



図2. AKOVSTの結合ネットワーク

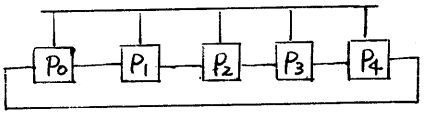



図3. 図2の別表現

また多数個マルチプロセッサではCMを陽に描くと本質を見通しにくくなるので，結合母線（バス）と同じようにあらわすことにする。例えば図2のAKOVSTの場合，図3のようになる。

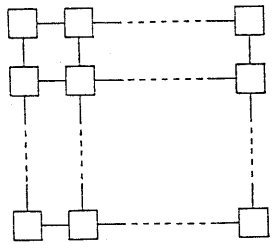
なお，さらにもう一種，切換えスイッチを陽にあらわす方法もある。が，我々はこの種のハードウェアもバス・スイッチと見なし，バス表現する。

このような前提で，代表的な結合ネットワークをあげると図4のような例がある。これらはいずれも実現性が高く多数個プロセッサの形状の有力な候補であるが，応用範囲の点でもう一つ満足できない。その点を改良する一つの方法として我々の提案するのはバスの導入である。むろん，ここでいうバスとはCMを含む広い意味でのネットワークの意味である。

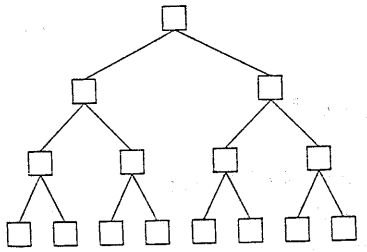
## 3. 処理を表現するグラフ S.C.f.g.

図4の代表的な結合ネットワークのうち我々は基本として(a)のアレイを望ぶ。(b)の木も良い所があるが，アレイは木の動作も行なうように修正しやすいが，逆は難しい。

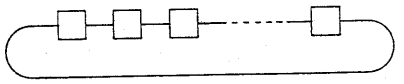
アレイを基本とする最大の理由は処理フローの埋め込みが容易なことである。多数個マルチプロセッサが専用プロセッサになるとしても，できるだけ多くの用途に適用できるアーキテク



(a) アレイ



(b) 木



(c) ループ

図4. 代表的な結合ネットワーク例

チャを備えていることが望ましい。そのためには広範な処理に対応できる形状がよい。

ここでは処理を記述する方式として，フローグラフを導入する。ただし，次のような条件を有している。

フローグラフを $G=(V,E)$ と書くとき，

(i) $V=\bigcup_{i=0}^{m} V_i$，ただし $V_i \cap V_j = \phi$ if $i \neq j$

各 $V_i$ をレベル $i$ と呼ぶ。

(ii) $\exists e=(u,v) \in E$

if and only if

$u \in V_i$ and $v \in V_{i+1}$ or

$v \in V_i$ and $u \in V_{i+1}$

(iii) $V_0 = \{n_0\}$ , $n_0$ は入枝をもたない。

$\forall v \in V_m$ は出枝をもたない。

つまり，フローグラフ $G$ は1つのノード（ルート）をもつグラフで†，ノードの集合はレベル化された部分集合に分割される。さらに，結合のための枝はすぐ隣のレベルのノードとの間にのみ存在する。

$G$ の各ノードには，処理ルーチン(procedure)が対応する。ここで，処理ルーチンごとの結合を考えると，そこには

(i) データの流れ

(ii) 制御の流れ

が存在する。ここまでのフローグラフ $G$ はデータの流れに着目しているので，これを $G_D$ と書く。図5(a)の例のように両方向にデータが流れてもよい。ただし，意味のある $G_D$ のみを対象とするため，$G_D = (V_D, E_D)$ において

$$\exists (u,v) \in E_D \text{ , where } u \in V_{i+1} \text{ , } v \in V_i$$
$$\text{only if } \exists (v,u) \in E_D$$

とする。

次に，制御の流れをあらわすグラフを $G_C = (V_C, E_C)$ とする。$G_C$ は

(i) $V_C = V_D$

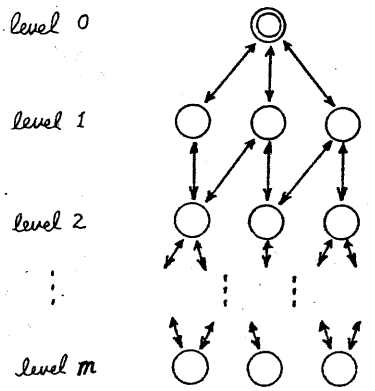
(ii) $E_C = E_D - E_R$ , where

$$E_R = \bigcup_{\text{all}} (u,v) \; (u \in V_{i+1}, v \in V_i)$$

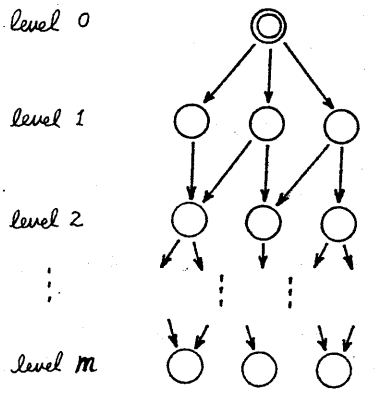
つまり，$G_D$ の枝のうちレベル $i+1$ からレベル $i$ へさかのぼる枝を除いたグラフが $G_C$ である。図5(b)のように，制御の流れはルートから下向きに一方向の流れとする。

なお，以下では $G_D$ と $G_C$ をともに描くことはやめ，$G_C$ のみでフローグラフをあらわす。このようなフローグラフ $G_C$ を **s.c.f.g.** (Sequential control-flow graph の略)と記す。

†) ルートつきグラフとしたのは付加プロセッサ(attached processor)としてホストコンピュータにつなぐという目的を明確にしたため。用途によってはこの条件をゆるめてもよい。




(a) データの流れ $G_D$




(b) 制御の流れ $G_C$

図5. フローグラフの細部

s.c.f.g. において，結合のセマンティクスをあらわすために，図6のような記法を導入する。

出枝に対する

OR-divide , AND-divide

入枝に対する

OR-merge , AND-merge

の各2種類づつである。

入出枝両方をもつノードでは入枝，出枝が複数ならばORまたはANDのどちらかを入，出のそれぞれについて定める。

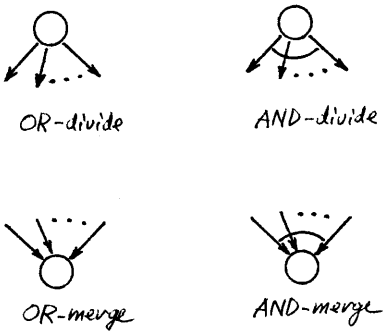



図6. S.C.f.g. におけるノードのタイプ

## 4. S.C.f.g. を実現するプロセッサ・アレイ — バス・アレイ

S.C.f.g. を実現するアーキテクチャを考えるにはノードへのプロセッサの割当てを考えればよい。

ここでは，バスを用いるプロセッサ・アレイでの実現のため，次のような割当て方法をとる†。

(1) 与えられた S.C.f.g. のレベル0を除く全レベルに各々1つづつプロセッサを割当てる（図7参照）。
(2) レベル $i$ ($i=1,2,\cdots,m$) に $l$ 個のプロセッサを割当てる（図8参照）††。
(3) (1)(2)の結果，図10のようなプロセッサ・アレイで S.C.f.g. が実現できる。行のプロセッサ数 $m$ が冗長ならば，この個数を減少させる（図11参照）。

図10のプロセッサ・アレイにおける結合ネットワーク I.N. の実現は，本稿ではバスを採

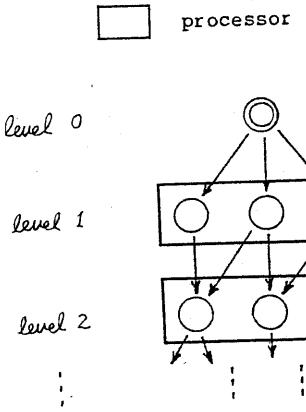



図7. ステップ(1)
— 各レベルにプロセッサを割当て —

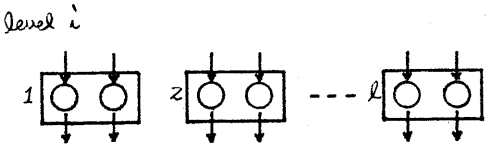



図8. ステップ(2)
— レベル $i$ に $l$ 個のプロセッサを割当て —

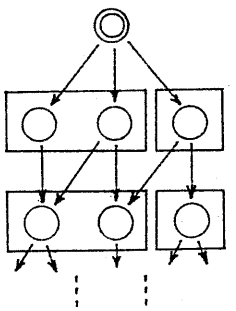
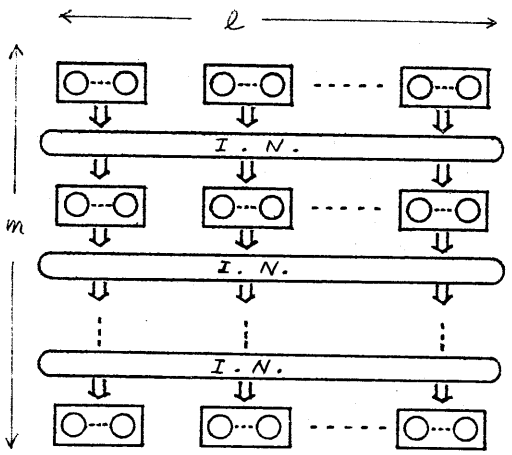


図9. ステップ(1)(2)の結果の例

†) 一般的にはかうつ S.C.f.g. 上のノード集合の分割問題になる。よく知られているスケジューリング問題との違いは，(i) プロセッサ数の制限なし，(ii) ノードに OR-divide/merge のタイプがある，(iii) 枝にもコストがつく場合がある（異なるプロセッサにまたがるとき），(iv) カットセットに上限あり。

††) ステップ(1)(2)の実行結果の例を図9に示す。




I.N. : INTERCONNECTION NETWORK

図10. $n=\ell \cdot m$ 個のプロセッサをもつプロセッサ・アレイ

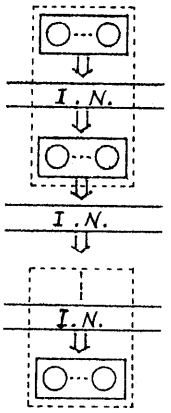



図11. 行のプロセッサ数の減少

用する。その結果，図12のように横方向にはバス，縦方向にはいわゆるポート結合を用いたプロセッサ・アレイが出来上がる。これをバス・アレイと呼ぶ。とくに，プロセッサ数を明示したいときは（$\ell$, $m$）バス・アレイという。実際には図13のようになる。

## 5. バス・アレイの最適化

結合ネットワークI.N.としてバスを用いるため横方向のデータ転送はすべて等距離に行なえるが，バスのためのボトルネックも生じる。ここではボトルネックにおける遅れも含んでs.c.f.g.上の枝にコストを付す。コストのつけ方は厳密には難しいが，与えられたs.c.f.g.の実行を最短時間で実行する（$\ell$, $m$）バス・アレイを求めることをバス・アレイの最適化という。この問題は最終的にはシミュレーションによらざるをえないが，横方向のプロセッサの数$\ell$と縦方向のプロセッサの数$m$がそれはどのような組合をもつかを次に述べる。

### 5-1 行のプロセッサ数$\ell$について

$\ell$の数について考察するため（$\ell$, 1）バス†）$\ell$と$m$の値を示すこと。

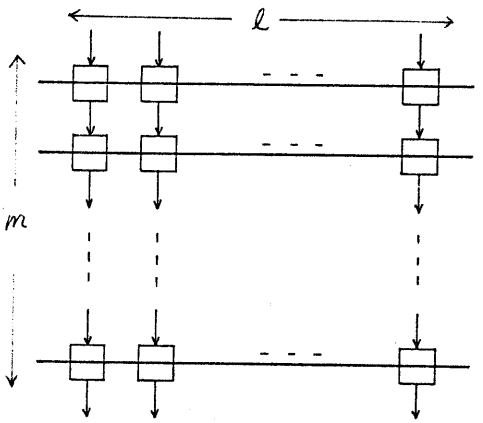



図12. 多数個プロセッサ・アレイの一形式 — バス・アレイ

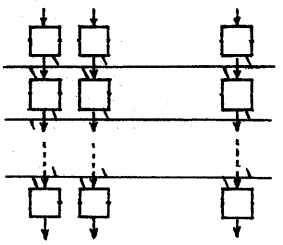


図13. バス・アレイの一例

アレイを対象とし，プロセッサ数 $l$ が速度向上に与える影響を調べる。ただし，次の条件をおく。

(1) どの処理も等しい処理時間 $t_p$ をもつ。

(2) 1つのプロセッサ内では処理と処理の間の転送コストは0とみるが，異なるプロセッサ間では $d$ とする。

(3) 通信オーバヘッド $r$

$$r = d / t_p$$

(4) 速度向上比 S.U.R.

$$S.U.R. = \frac{\text{1個のプロセッサでの処理時間}}{\text{n個のプロセッサでの処理時間}}$$

詳しい解析は文献(9)にゆずるが，$r$ をパラメータとしたとき，プロセッサ数 $n$ ($=l$) に対して S.U.R. がどう変化するかがわかる。S.U.R. $= 0.5n$ を飽和の限界と仮定すると

$$n \leq (1/r)$$

となり[9]，例えば，$r = 0.01$ ならば，プロセッサ数 $n$ は 100 に押さえてよいということになる。

このS.U.R.の飽和はバスの特性から生じているが，多数個マルチプロセッサ Cm* でも同じ傾向が見られる[10]。バスの階層化や複線化が一つの改善方法であるが[11]，定量的には未だ検討してはいない。むろん，2×2スイッチによる結合ネットワークが現実化すれば，その方がボトルネック解消の本筋ではある。

### 5-2 列のプロセッサ数 $m$ について

列方向のプロセッサの数をふやす効果はパイプライン処理において顕著にあらわれる。そこで，そのS.U.R.を用いて，2重木構造のふく.ま.の場合の例を図15に示す。パイプライン処理の特長はオーバヘッド $r$ の影響が小さいことで，$r = 100$ でも，プロセッサ数が16をこえると，ほとんどその影響は無視できるようになる。解析はやはり文献(9)にゆずる

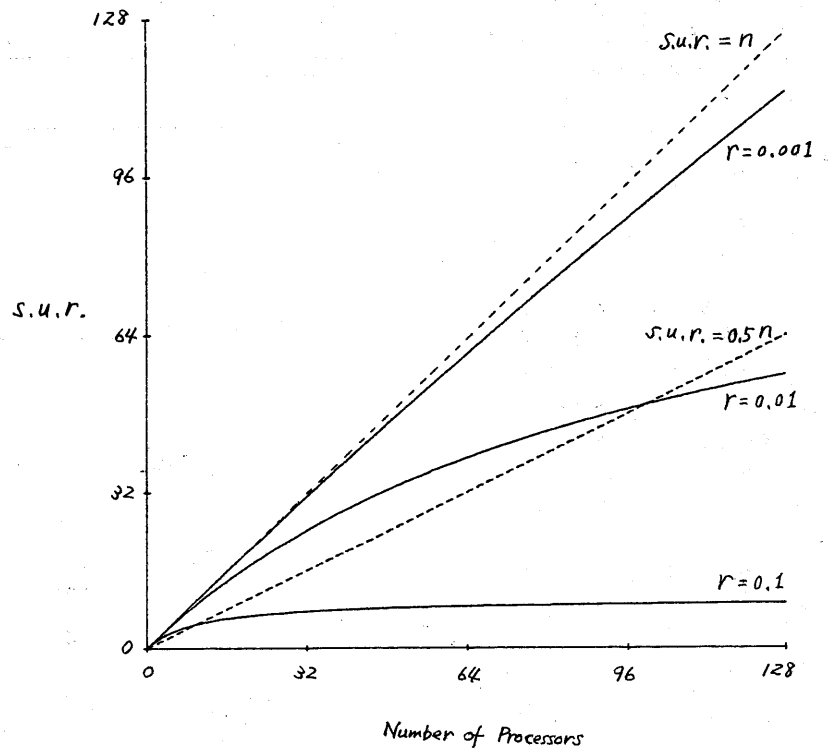



図14. $(l, 1)$ バス・アレイ の $l$ と S.U.R. の関係

が，パイプライン処理は図16のように多くのデータ列が順次入力する処理形態であり，データ数は2進木構造の各ノードに挿入されるのでk=2^m−1とした場合である。（プロセッサ数がふえるとデータ数が指数的に増加する。）

（1,m）バス・アレイにおいてプロセッサ数mの効果を論ずるには，データ数を考慮に入れる必要があり，ここでは，かなりパイプライン処理の効果の大きい一例を示した。この効果の大きいことは実験的にも確かめている。
（1,32）バス・アレイは先に報告したマルチマイクロプロセッサUNIP[11]を用いればよく，UNIPを用いたパイプラインソート実験の結果を付録に示す。

## 5-3 総プロセッサ数 ℓ・m について

以上はℓかmのどちらかを1とした特別な場合を論じたが，一般には，ℓ・m=n について考えなければならない。横方向にデータ転送のない場合については，総プロセッサ数一定のもとで，ℓとmの最適数を理論的に求めたことはある[12]。横方向にデータ転送のある一般的な場合は解析的には難しいため，現在，シ

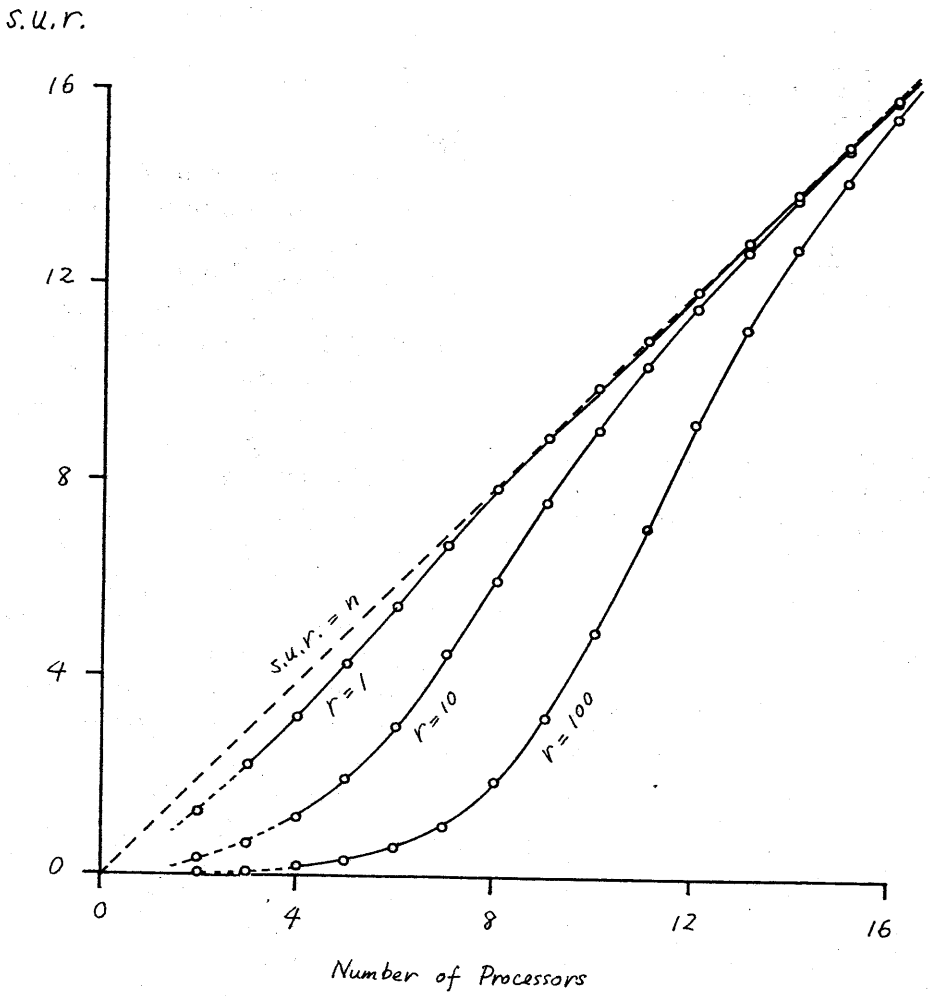



図15. (1,m) バス・アレイにおける S.U.r. の例

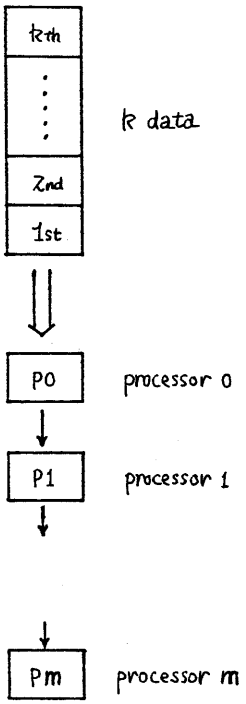



図16. パイプライン処理

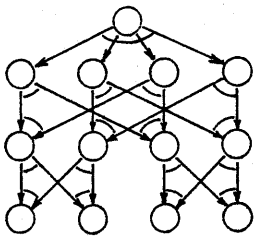


図17. FFTをあらわす S.C.f.g.

ミュレータを作成中である。(ℓ, m)バス・アレイの応用例として、FFTがあげられる。FFTの1つのS.C.f.g.表現は図17のようになる。データ数Kをパラメータとして、最適なℓとmを求めるのが目的である。

## 6. むすび

コスト・パフォーマンスの点でも実現性の高くなった多数個マルチプロセッサの一形式としてバス・アレイを提案し、概略と現状を述べた。とりあえず、$10^3$〜$10^4$個程度を仮定しているが、図17のFFTでは$O(n \log n)$の処理ノードが必要であり、将来、さらに個数を増加させるかも知れない。いずれにしても、その実現が有効となる応用範囲を拡大することが急務であろう。我々はリアルタイム処理の分野での専用プロセッサとして多数個マルチプロセッサの有用性を高く評価しており、今後、各種の応用例を追求したい。

## 文献

(1) T. Hoshino et al.: "Super freedom simulator PAX", Proc. 16th IBM Computer Science Sympo., pp. 43-55 (1982)

(2) 阿江, 大崎, Vuong: "小規模マルチプロセッサシステムの一方式", 信学技報, EC78-35 (1978-11)

(3) 阿江, 高橋, 松本: "共有メモリ結合によるマルチマイクロプロセッサの並列動作について", 信学論(D), J65-D, 3, pp. 322-329 (1982-03)

(4) 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, 23, 3, pp. 201-207 (1982-03)

(5) Tse-yun Feng: "A survey of interconnection networks", IEEE Computer, 14, 2, pp. 12-27 (Dec. 1981)

(6) T. Ae et al.: "Picture data processing on small parallel computer", Proc. 2nd IEEE Workshop on PDDM, pp. 266-271 (1980)

(7) T. Ae et al.: "Computer graphics on multiple microprocessor system", Eurographics 80, North-Holland Pub. Co., pp. 281-288 (1980)

(8) T. Ae et al.: "A Multiple Microprocessor System with Mutually Diagnosing Capability", Proc. ICS (Vol. 1), pp. 375-388 (1980)

(9) T. Ae and R. Aibara: "Experimentation and analysis of multiprocessor systems", IEEE Real-Time Systems Symposium (to appear on Dec. 1982)

(10) A. K. Jones and P. Schwarz: "Experience using multiprocessor systems — A status report",

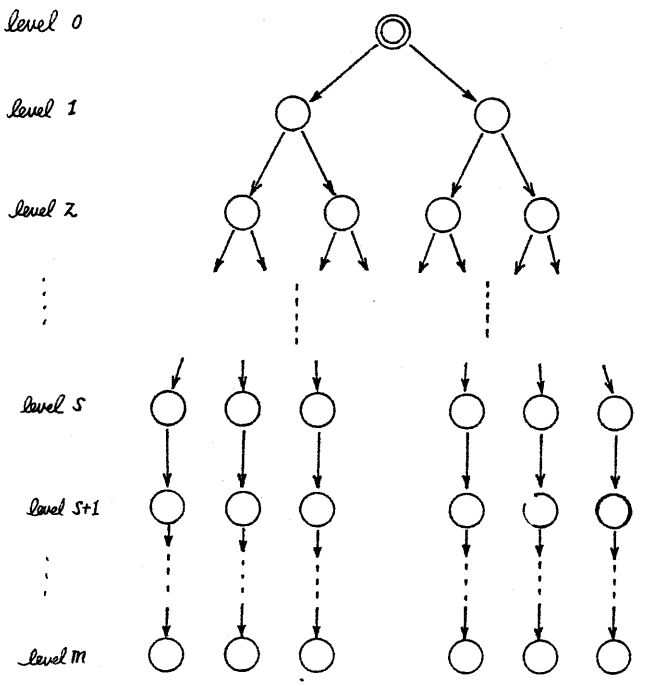
ACM Computing Surveys, 12, 2, pp.121-165 (1980)
(11) 相原, 阿江: "並列UNIXパイプラインプロセッサ UNIXの応用", 信学技報 EC82-31 (1982-06)
(12) 阿江ほか: "通信オーバヘッドを考慮した並列処理ネットワークにおけるソロー問題", 信学技報 AL81-31 (1981) および
松本健司: "分散システムにおける並列処理解析の一方法", 広島大学大学院工学研究科修士論文 (1982-02)

実際のソーティング の実行時間の測定例を図A-2に示す。(横軸はデータ数にとっている。データ数Nに対して, ごくデータ数の少ないところを除いて, ほぼ完全な O(N) の実行時間になっていることが確かめられる。)

## 付録

### (1,32) バス・アレイによるパイプラインソート実験

s.c.f.g. を図A-1に示す。




図A-1 パイプライン・ソートをあらわすs.c.f.g.

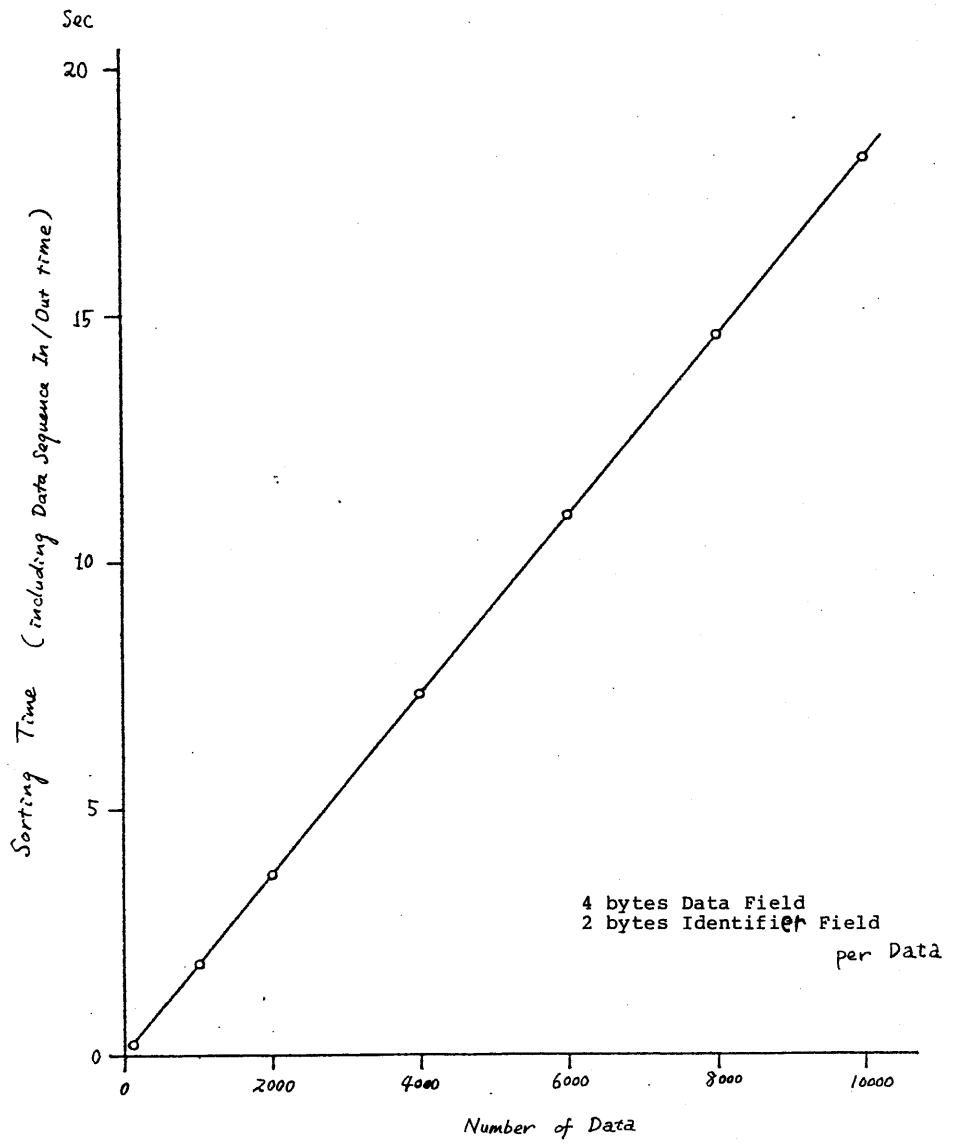



図 A-2 パイプラインソートの実行時間