Vol. 27 No. 3 情報処理学会論文誌 Mar. 1986

格子状結合並列計算機において行優先に保持された行列の転置に要するプロセッサ間データ転送の最適化†

中野浩行††* 津田孝夫††

PE 数 P の二次元正方格子状結合計算機上に行優先に保持された $N\times N$ 行列を転置するアルゴリズムの，PE 間データ転送の最適化について論じる．$N=2^n$, $P=2^m$ (n: 整数，m: 偶数) とする．格子状結合の端のつながり方は Illiac Ⅳ の型と PAX-128 の型の2通りを考える．データ転送のための時間はデータが転送路を流れる時間（転送時間）と転送路設定のための時間（スイッチング時間）に分ける．まず転送時間の下界を示す．下界は $N^2P^{1/2}$ のオーダである．次に $P\leq N$ の場合について，転送時間に関してほぼ最適なアルゴリズムを提案する．

1. まえがき

並列計算機の processing element (PE) 間結合方式として近年様々のものが提案されているが，Illiac Ⅳ に代表される格子状結合方式は結合網のコストが低いことと，隣接 PE 間の転送遅れが全プロセッサの物理サイズの影響を受けないことから，非常に多数の PE を結合するための方式として現在でも有力な候補の一つである．

格子状結合方式においては，PE の間でデータの並べ換えを行うためのアルゴリズムが問題になる．Orcutt[3] や Nassimi and Sahni[2] は PE 一つ当り1個のデータを PE 間で置換 (permute) するクラスのデータ並べ換えアルゴリズムを論じている．特に Nassimi and Sahni は転送回数についての下界理論を導入してアルゴリズムの最適性を示している．

さて，Bhuyan and Agrawal[1] は格子状結合方式を含む三種の結合方式による並列計算機で2次元フーリエ変換を実行するときの実行時間を机上で評価している．そのアルゴリズムの途中に，行優先に保持された行列の転置が現れるが，彼らは格子状結合方式における効率のよい行列転置アルゴリズムを求めることを問題として提示している．

ここでの行列転置の問題を説明しておく．$N\times N$ 行列 ($N=2^n$, n は整数) が行優先に P 個に分割されて，順に PE 0～PE($P-1$) に保持されているとする．ここに P は PE の数で，$P=2^m$, m は偶数とする．文献1)で扱われたのは $P\leq N$ の場合で，この場合図1(a)のように行列が保持されている．$N<P<N^2$ の場合は図1(b)のようになる．この行列を転置することがここで考える問題である．

文献1)では暫定的に転置アルゴリズムを定め，それを用いて実行時間を評価している．その結果を見ると，2次元フーリエ変換の，行列転置を除いた計算時間は $O(N^2\log N\cdot P^{-1})$ で，一方行列転置に要する時間は $O(N^2)+O(P)$ である．PE 数 P がデータ量 N^2 に比例するとすると，前者は $O(\log N)$，後者は $O(N^2)$ となる．すなわち，問題が大きく，それに比例して計算機の PE 数が多いときには転置に要する時間が全計算時間を支配する可能性がある．したがって，効率よい転置アルゴリズムを求めることは重要である．

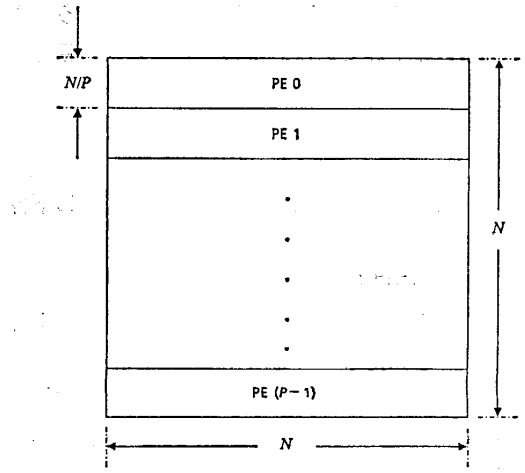
そこで本論文では，まず転置に要する時間のうち転送時間（次章で定義する）について，アルゴリズムによらない下界を示す．次に文献1)で提示された問題への解として，$P\leq N$ の場合について転送時間に関してほぼ最適なアルゴリズムを示す．

2. 格子状結合計算機

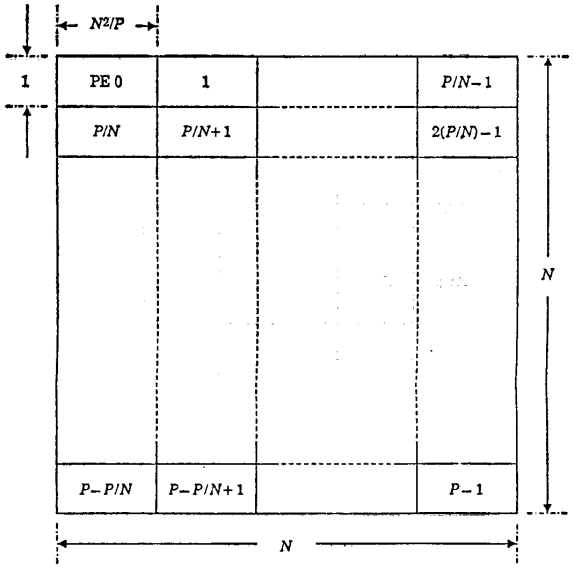
本論文で扱う格子状結合計算機は，端同士の結合と PE の番号づけを除いて文献1)の論じたものと同じである．PE の数は P で，$\sqrt{P}\times\sqrt{P}$ の二次元正方格子状とする ($P=2^m$, m は偶数)．各 PE はそれぞれプライベートメモリをもつとする．格子の端同士の結合のしかたはいくつか考えられるが，本論文では図2(a)のように全 PE が横の結合路で一列に並んでいる型 (propagating wraparound[2]) と，図2(b)のようなトーラス状トポロジーのもの (orthogonal wrap-

† Optimizing Inter-Processor Data Transfers in Transpositions of Matrices Stored Row-Wise on Mesh-Connected Parallel Computers by HIROYUKI NAKANO and TAKAO TSUDA (Department of Information Science, Faculty of Engineering, Kyoto University).

†† 京都大学工学部情報工学科
* 現在 ソニー(株)



(a) $P \leqq N$
(区切りは各 PE が保持する部分を示す)

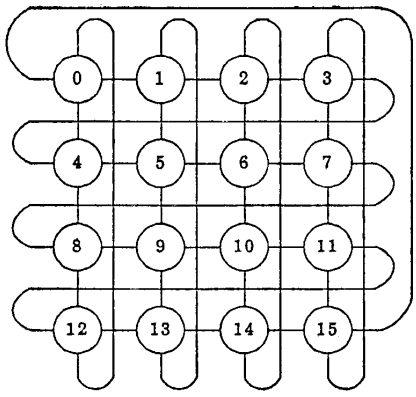


(b) $N<P<N^2$

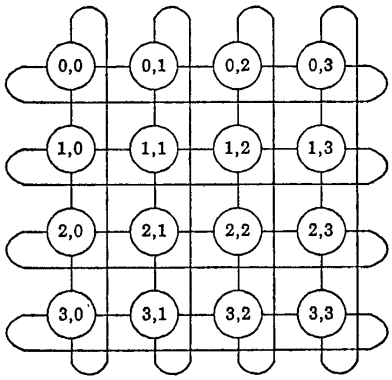
図 1 行優先に保持された行列
Fig. 1 Matrix stored row-wise.

around[2]) の二種類を扱う．前者は Illiac Ⅳ で採用されているので，以下 Illiac Ⅳ 型と呼ぶ．また後者は PAX-128[4] で採用されているので以下 PAX 型と呼ぶ．文献 1)で扱われたのは Illiac Ⅳ 型である．

隣接 PE 間のデータ転送は，隣接する二つの PE の間で転送路を設定したあと，ブロック転送によって行う（文献 2),3)の仮定とは異なる）．1 回の転送路設定に要する時間を α，1 個のデータが隣接する PE の間を転送される（これを以下単位転送と呼ぶ）時間を τ とする．ある PE がデータ転送の方向を変えるときには転送路設定時間 α が必要であるとする．また転送する方向が同じでも，他の PE から受け取ったデータ



(a) Illiac Ⅳ 型



(b) PAX 型

図 2 格子状結合計算機 ($P=16$)
Fig. 2 Mesh-connected computer ($P=16$).

をさらに隣の PE に転送する際には α の時間が必要であるとする．各 PE はデータの送信と受信を独立に並行して行えるとする．これによって計算機全体では最大 P 個の単位転送が同時にできる．

以下データそのものの転送に要する時間を**転送時間**と呼び，転送路設定に要する時間を**スイッチング時間**と呼ぶ．

PE には図 2 (a) のように行優先に 0～$P-1$ の番号をつける．PAX 型を考える際にはこのほかに図 2 (b) のように行番号・列番号の 2 次元の番号も用いる．PEi が PE(j, k) に当たるとすると，

$$i=\sqrt{P}\,j+k$$
$$(i=0,\cdots,P-1;\ j,k=0,\cdots,\sqrt{P}-1)$$

の対応がある．

次に PE 間の距離を定式化する．PEi から PEj へデータを転送するのに通るべき隣接 PE 間結合路の数の最小値を PEi と PEj の間の距離と呼び，$L(i, j)$ で表す．

Illiac Ⅳ 型の場合は $L(i, j)$ は $j-i$ の関数 $L_1(j-$

i) で表せる．$L_I(d)$ は

$$L_I(-d)=L_I(d) \tag{2.1}$$

$$L_I(P-d)=L_I(d) \tag{2.2}$$

の性質をもつ．すなわち $L_I(d)$ は周期 P の周期関数で，$d=0, P/2$ について対称である．$0\leq d\leq P/2$ についての $L_I(d)$ の値は

$$d=a\sqrt{P}+b$$

ただし $a=0,\cdots,\sqrt{P}/2$

$$b=-\sqrt{P}/2+1,\cdots,0,\cdots,\sqrt{P}/2$$

のとき，

$$L_I(d)=a+|b| \tag{2.3}$$

となる．$0\leq d\leq P/2$ 以外の範囲については式(2.1)，(2.2)によって $0\leq d\leq P/2$ の範囲に変換して求める．

PAX 型の場合は行と列が完全に独立なので，PE(i,j) と PE(k,l) の間の距離 $L((i,j),(k,l))$ は行方向の距離と列方向の距離の和 $L_P(k-i)+L_P(l-j)$ で表せる．$L_P(x)$ については

$$L_P(-x)=L_P(x) \tag{2.4}$$

$$L_P(\sqrt{P}-x)=L_P(x) \tag{2.5}$$

が成り立つ．すなわち $L_P(x)$ は周期 $\sqrt{P}$ の周期関数で，$x=0,\sqrt{P}/2$ について対称である．$0\leq x\leq\sqrt{P}/2$ のとき

$$L_P(x)=x \tag{2.6}$$

である．$0\leq x\leq\sqrt{P}/2$ 以外の範囲については式(2.4)と(2.5)によって $0\leq d\leq\sqrt{P}/2$ の範囲に変換して求める．

3. 転送時間の下界

データ並べ換えの問題に対して必要な転送時間等の下界を求めることは，並べ換えのアルゴリズムの最適性を示すために，あるいはアルゴリズムの最適化の目標として有用である．

あるデータ並べ換えを考え，これを実現するために転送されるデータの一つを x とする．このデータがもとあった PE を $s(x)$，行先の PE を $d(x)$ とすると，このデータは少なくとも $L(s(x),d(x))$ 回転送されなければならない．転送すべきデータすべてについてこの距離の総和をとると，これはデータ並べ換えに必要な単位転送の数の下界となる．これを u_{LB} とする．模式的に書くと，

$$u_{LB}=\sum_{x\in D}L(s(x),d(x))$$

(D は転送されるデータの集合)．

一方，同時に最大 P 個の単位転送ができると仮定したので，u_{LB} を P で割ったものは逐次に行うべき単位転送の数の下界である．この考え方による転送時間の下界を t_{LB} とする．

$$t_{LB}=\frac{u_{LB}}{P}\cdot\tau=\tau\cdot\sum_{x\in D}L(s(x),d(x))/P.$$

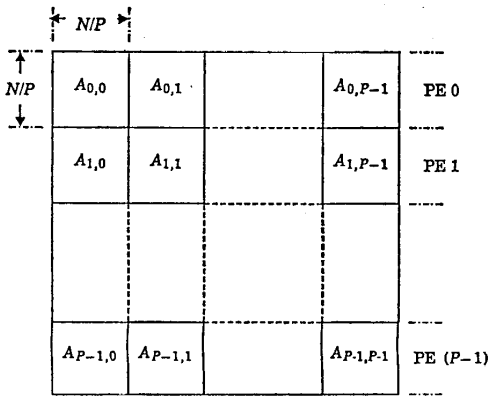
この方法により，行列転置に必要な転送時間の下界を計算する．

3.1 $P\leq N$ の場合

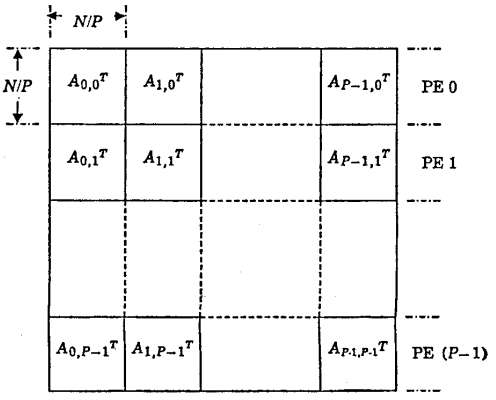
この場合図1(a)のように一つの PE が行列の1行ないし複数行を保持している．

初期状態の行列を**図3**(a)のように $(N/P)\times(N/P)$ の小行列 $P\times P$ 個に分けて考える．各小行列を $A_{i,j}$ $(i,j=0,\cdots,P-1)$ とする．

図3(a)の行列を転置すると図3(b)のようになり，初め PEi にある小行列 $A_{i,j}$ は転置によって PEj に移される $(i,j=0,\cdots,P-1)$．したがって初期状態で一つの PE に保持されていた N^2/P 個のデータは



(a) 初期状態
(a) Initial state.



(b) 最終状態
(b) Final state.

図 3 小行列に分割された行列とその転置 $(P\leq N)$
Fig. 3 Matrix partitioned into submatrices and its transposition $(P\leq N)$.

最終状態ではすべての PE に N^2/P^2 個ずつ分配されている．(PE 内での小行列の並べ換えや小行列内の転置は PE 内部の操作なので無視する．)

i) Illiac Ⅳ 型

任意の PE$i,j$ $(i,j=0,\cdots,P-1)$ について，i から j へ N^2/P^2 個のデータを転送しなければならないので，

$$u_{LB}=\sum_{i=0}^{P-1}\sum_{j=0}^{P-1}\frac{N^2}{P^2}L_I(j-i)$$
$$=\frac{N^2}{P}\sum_{d=0}^{P-1}L_I(d)$$
$$=\frac{N^2\sqrt{P}}{2}-\frac{N^2}{2\sqrt{P}}.$$
$$\therefore\quad t_{LB}=\frac{u_{LB}}{P}\cdot\tau=\left(\frac{N^2}{2\sqrt{P}}-\frac{N^2}{2P\sqrt{P}}\right)\tau. \qquad (3.1)$$

ii) PAX 型

i)と同様に考えると，

$$u_{LB}=\sum_{i=0}^{\sqrt{P}-1}\sum_{j=0}^{\sqrt{P}-1}\sum_{k=0}^{\sqrt{P}-1}\sum_{l=0}^{\sqrt{P}-1}\frac{N^2}{P^2}$$
$$\times\{L_P(k-i)+L_P(l-j)\}$$
$$=\frac{2N^2}{\sqrt{P}}\sum_{d=0}^{\sqrt{P}-1}L_P(d)$$
$$=\frac{N^2\sqrt{P}}{2}.$$
$$\therefore\quad t_{LB}=\frac{u_{LB}}{P}\cdot\tau=\frac{N^2}{2\sqrt{P}}\tau. \qquad (3.2)$$

Illiac Ⅳ 型の下界の方が PAX 型の下界より少し小さいことがわかる．

3.2 $N<P<N^2$ の場合

この場合は図1(b)のように行列の1行が複数 $(P/N$ 個) の PE に分割されて保持されている．

行列を**図4**のように，1PE にストアされているデータの数 (N^2/P) を一辺とする小行列 $(P/N)\times(P/N)$ 個に分割する．各小行列を $A_{i,j}(i,j=0,\cdots,P/N-1)$ とする．すると P 個の PE は，初期状態で保持している要素がどの小行列に属するかによって N^2/P 個ずつにグループ分けできる．ある PE の保持しているデータが $A_{i,j}$ に属するとき，この PE はグループ (i,j) に属するとする $(i,j=0,\cdots,P/N-1)$.

グループ (i,j) に属する PE の番号は，$iN+k(P/N)+j$ $(k=0,\cdots,N^2/P-1)$ と表せる．

転置を行うと，図4のaの部分のデータ (一つの PE に保持されている) はbの部分へ移動する．一般にはグループ (i,j) に属する PE の一つからグループ (j,i) に属する N^2/P 個の PE のすべてに1個ずつのデータが送られる．すなわち $N<P<N^2$ の場合の行列転置は $iN+k(P/N)+j$ $(i,j=0,\cdots,P/N-1;$ $k=0,\cdots,N^2/P-1)$ なる PE のそれぞれから，$jN+k'(P/N)+i$ $(k'=0,\cdots,N^2/P-1)$ なる PE のすべてに1個ずつのデータを転送するデータ並べ換えである．このことから u_{LB} は

$$u_{LB}=\sum_{i=0}^{P/N-1}\sum_{j=0}^{P/N-1}\sum_{k=0}^{N^2/P-1}\sum_{k'=0}^{N^2/P-1}$$
$$\times L\left(iN+k\frac{P}{N}+j,\ jN+k'\frac{P}{N}+i\right)$$

と表せる．

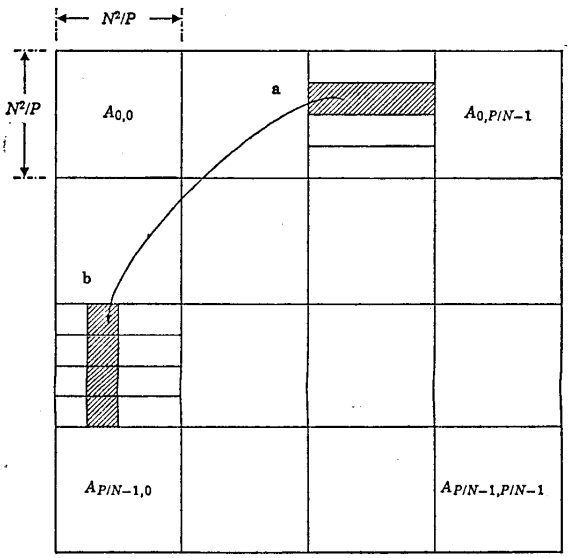


図4 小行列に分割された行列とその転置 $(N<P<N^2)$
Fig. 4 Matrix partitioned into submatrices and its transposition $(N<P<N^2)$.

i) Illiac Ⅳ 型

$$u_{LB}=\sum_{i,j,k,k'}L_I\left((j-i)N+(k'-k)\frac{P}{N}-(j-i)\right)$$
$$=\frac{N^2\sqrt{P}}{2}-\frac{N\sqrt{P}}{4}.$$

(途中の計算の概略は付録に示す．)

$$\therefore\quad t_{LB}=\frac{u_{LB}}{P}\cdot\tau=\left(\frac{N^2}{2\sqrt{P}}-\frac{N}{4\sqrt{P}}\right)\tau. \qquad (3.3)$$

ii) PAX 型の場合

まず PE 番号を行番号，列番号に分ける．$k=l_1(N/\sqrt{P})+l_2$ $(l_1,l_2=0,\cdots,N/\sqrt{P}-1)$ とおくと，

$$iN+k\frac{P}{N}+j=\left(i\cdot\frac{N}{\sqrt{P}}+l_1\right)\sqrt{P}+\left(l_2\frac{P}{N}+j\right)$$

より，$iN+k(P/N)+j$ は $(i\cdot(N/\sqrt{P})+l_1,\ l_2(P/N)+j)$ と表せる．同様に $jN+k'(P/N)+i$ は $k'=l_3(N/\sqrt{P})+l_4(l_3,l_4=0,\cdots,N/\sqrt{P}-1)$ とおいて，$(j\cdot(N/$

$\sqrt{P})+l_3, l_4(P/N)+i)$ と表せる．これを用いて，

$$u_{LB}=\sum_{i=0}^{P/N-1}\sum_{j=0}^{P/N-1}\sum_{l_1=0}^{N/\sqrt{P}-1}\sum_{l_2=0}^{N/\sqrt{P}-1}\sum_{l_3=0}^{N/\sqrt{P}-1}\times\sum_{l_4=0}^{N/\sqrt{P}-1}\left\{L_P\left(\left(j\frac{N}{\sqrt{P}}+l_3\right)-\left(i\frac{N}{\sqrt{P}}+l_1\right)\right)+L_P\left(\left(l_4\frac{P}{N}+i\right)-\left(l_2\frac{P}{N}+j\right)\right)\right\}=\frac{N^2\sqrt{P}}{2}$$

を得る（途中の計算の概略は付録に示す）．

$$\therefore\quad t_{LB}=\frac{u_{LB}}{P}\tau=\frac{N^2}{2\sqrt{P}}\tau. \tag{3.4}$$

$N<P<N^2$ の場合も Illiac Ⅳ 型の下界の方が PAX 型より少し小さい．また PAX 型では t_{LB} の P, N に関する表式は，$P\leq N$ のときと同じである．

4. 転置アルゴリズム

本章では $P\leq N$ の場合について転置アルゴリズムを提案し，その転送時間とスイッチング時間を評価する．

$P\leq N$ の場合は前章で触れたように任意の一つの PE から他のすべての PE へ N^2/P^2 個のデータが送られる．小行列 $A_{i,j}$ は初期状態で PEi にあって，PEj を行先とするデータの集まりであることに注意する．

PAX 型のアルゴリズムの方が考えやすいので，まずこちらを説明する．

i) PAX 型

小行列 $A_{\sqrt{P}i+j,\sqrt{P}k+l}$ をここでは $A_{(i,j),(k,l)}$ と書く $(i, j, k, l=0, \cdots, \sqrt{P}-1)$．$A_{(i,j),(k,l)}$ は初期状態で PE(i, j) にあって PE(k, l) を行先とするデータの集まりである．

アルゴリズム中で PE の行番号・列番号や小行列の添字は常に modulo$\sqrt{P}$ をとって $0\sim\sqrt{P}-1$ の範囲に直すものとする．

本アルゴリズムは格子状結合の列方向に転送を行うフェーズ1と行方向に転送を行うフェーズ2からなる．

フェーズ 1：本フェーズでは第 j 列の PE 群から第 k 行の PE 群へ行くべきデータを PE(k, j) に集める．すなわち，$A_{(i,j),(k,l)}$ を PE(i, j) から PE(k, j) に移動する $(i, j, k, l=0, \cdots, \sqrt{P}-1)$．この際，$0<(k-i)\bmod\sqrt{P}\leq\sqrt{P}/2$ なる (k, j) へは下向きに転送し，$\sqrt{P}/2<(k-i)\bmod\sqrt{P}<\sqrt{P}$ なる (k, j) へは上向きに転送する．

●下向き転送

ステップ 1　PE(i, j) は $A_{(i,j),(k,l)}$ $(k=i+1, \cdots, i+\sqrt{P}/2;\ l=0, \cdots, \sqrt{P}-1)$ を PE$(i+1, j)$ に転送する（転送するデータの数は $(N^2/P^2)\sqrt{P}\cdot(\sqrt{P}/2)$）．

ステップ h $(h=2, \cdots, \sqrt{P}/2)$　PE$(i+h-1, j)$ はステップ $(h-1)$ で送られてきたデータのうち第 $(i+h-1)$ 行の PE を行先とする $(N^2/P^2)\sqrt{P}$ 個のデータ $(A_{(i,j),(i+h-1,l)}, l=0, \cdots, \sqrt{P}-1)$ を自分のプライベートメモリに残し，それ以外の $(N^2/P^2)\sqrt{P}(\sqrt{P}/2-h+1)$ 個のデータ $(A_{(i,j),(k,l)}, k=i+h, \cdots, i+\sqrt{P}/2;\ l=0, \cdots, \sqrt{P}-1)$ を PE$(i+h, j)$ に送る．

（各ステップの操作はすべての $i, j=0, \cdots, \sqrt{P}-1$ について一斉に行う．）

●上向き転送

ステップ 1　PE(i, j) は $A_{(i,j),(k,l)}$ $(k=i-\sqrt{P}/2+1, \cdots, i-1;\ l=0, \cdots, \sqrt{P}-1)$ を PE$(i-1, j)$ に転送する．

ステップ h $(h=2, \cdots, \sqrt{P}/2-1)$　PE$(i-h+1, j)$ はステップ $(h-1)$ で送られてきたデータのうち第 $(i-h+1)$ 行の PE を行先とする $(N^2/P^2)\sqrt{P}$ 個のデータ $(A_{(i,j),(i-h+1,l)}, l=0, \cdots, \sqrt{P}-1)$ を自分のプライベートメモリに残し，それ以外の $(N^2/P^2)\sqrt{P}\times(\sqrt{P}/2-h)$ 個のデータ $(A_{(i,j),(k,l)}, k=i-\sqrt{P}/2+1, \cdots, i-h;\ l=0, \cdots, \sqrt{P}-1)$ を PE$(i-h, j)$ に送る．

（各ステップの操作はすべての $i, j=0, \cdots, \sqrt{P}-1$ について一斉に行う．）

フェーズ 2：フェーズ1の結果 PE(k, j) $(k, j=0, \cdots, \sqrt{P}-1)$ には $A_{(i,j),(k,l)}$ $(i, l=0, \cdots, \sqrt{P}-1)$ が保持されている．本フェーズではこれらを行方向に各 PE(k, l) に分配し，転置を完成する．

この際 $0<(l-j)\bmod\sqrt{P}\leq\sqrt{P}/2$ なる (k, l) へは右向きに転送し，$\sqrt{P}/2<(l-j)\bmod\sqrt{P}<\sqrt{P}$ なる (k, l) へは左向きに転送する．

●右向き転送

ステップ 1　PE(k, j) は $A_{(i,j),(k,l)}$ $(l=j+1, \cdots, j+\sqrt{P}/2;\ i=0, \cdots, \sqrt{P}-1)$ を PE$(k, j+1)$ に転送する．

ステップ h $(h=2, \cdots, \sqrt{P}/2)$　PE$(k, j+h-1)$ はステップ $(h-1)$ で送られてきたデータのうち $A_{(i,j),(k,j+h-1)}$ $(i=0, \cdots, \sqrt{P}-1)$ を自分のプライベートメ

モリに残し，それ以外の $A_{(i,j),(k,l)}$ $(l=j+h,\cdots,j+\sqrt{P}/2;\ i=0,\cdots,\sqrt{P}-1)$ を PE$(k,j+h)$ に送る．

（各ステップの操作はすべての $k,j=0,\cdots,\sqrt{P}-1$ について一斉に行う．）

●左向き転送

ステップ1 PE(k,j) は $A_{(i,j),(k,l)}$ $(l=j-\sqrt{P}/2+1,\cdots,j-1;\ i=0,\cdots,\sqrt{P}-1)$ を PE$(k,j-1)$ に転送する．

ステップ h $(h=2,\cdots,\sqrt{P}/2-1)$ PE$(k,j-h+1)$ はステップ $(h-1)$ で送られてきたデータのうち $A_{(i,j),(k,j-h+1)}$ $(i=0,\cdots,\sqrt{P}-1)$ を自分のプライベートメモリに残し，それ以外の $A_{(i,j),(k,l)}$ $(l=j-\sqrt{P}/2+1,\cdots,j-h;\ i=0,\cdots,\sqrt{P}-1)$ を PE$(k,j-h)$ に転送する．

（各ステップの操作はすべての $k,j=0,\cdots,\sqrt{P}-1$ について一斉に行う．） □

このアルゴリズムの転送時間・スイッチング時間を評価する．

転送時間

各ステップともすべての PE が同数ずつのデータを並列に送出するので，一つの PE が送出するデータののべ数に τ を乗じたものが転送時間になる．まず一つの PE の送出するデータののべ数を計算する．

1) フェーズ1

ステップ h で一つの PE が送出するデータ数は，下向き転送では $(N^2/P^2)\sqrt{P}(\sqrt{P}/2-h+1)$，上向き転送では $(N^2/P^2)\sqrt{P}(\sqrt{P}/2-h)$ である．

下向き転送：

$$\sum_{h=1}^{\sqrt{P}/2}\frac{N^2}{P^2}\sqrt{P}\left(\frac{\sqrt{P}}{2}-h+1\right)=\frac{1}{4}\frac{N^2}{P}\left(\frac{\sqrt{P}}{2}+1\right).$$

上向き転送：

$$\sum_{h=1}^{\sqrt{P}/2-1}\frac{N^2}{P^2}\sqrt{P}\left(\frac{\sqrt{P}}{2}-h\right)=\frac{1}{4}\frac{N^2}{P}\left(\frac{\sqrt{P}}{2}-1\right).$$

2) フェーズ2

左向き転送・右向き転送ではそれぞれ下向き転送・上向き転送と同数のデータを送出する．

以上から全転送時間は

$$2\left\{\frac{1}{4}\frac{N^2}{P}\left(\frac{\sqrt{P}}{2}+1\right)+\frac{1}{4}\frac{N^2}{P}\left(\frac{\sqrt{P}}{2}-1\right)\right\}\cdot\tau$$
$$=\frac{N^2}{2\sqrt{P}}\tau.$$

これは式(3.2)で示される転送時間の下界を実現している．

スイッチング時間

転送路設定はステップごとに1回必要である．全スイッチング時間は

$$2\left\{\frac{\sqrt{P}}{2}+\left(\frac{\sqrt{P}}{2}-1\right)\right\}\cdot\alpha=2(\sqrt{P}-1)\alpha.$$

ii) **Illiac Ⅳ 型**

この場合は1次元の PE 番号で考える．基本的には PAX 型のときと同様なアルゴリズムである．PE 番号，小行列の添字はすべて modulo P をとって $0\sim P-1$ の範囲に直すものとする．

フェーズ1：本フェーズでは $A_{i,i+\sqrt{P}k+l}$ を PEi から PE$(i+\sqrt{P}k)$ に移動する $(i=0,\cdots,P-1;\ k,l=-\sqrt{P}/2+1,\cdots,0,\cdots,\sqrt{P}/2)$．この際 $0<k\leq\sqrt{P}/2$ なる $(i+\sqrt{P}k)$ へは下向きに，また $-\sqrt{P}/2<k<0$ なる $(i+\sqrt{P}k)$ へは上向きに転送する．アルゴリズムの詳細は略すが，PAX 型の場合と同様に下向きに $\sqrt{P}/2$ ステップ，上向きに $(\sqrt{P}/2-1)$ ステップの一斉転送を行い，各ステップでそれぞれの PE は $(N^2/P^2)\sqrt{P}$ 個のデータをとり込む．

フェーズ2：フェーズ1の結果，PEi には $A_{i+\sqrt{P}k,i+l}$ $(k,l=-\sqrt{P}/2+1,\cdots,0,\cdots,\sqrt{P}/2)$ が保持されている．本フェーズではこれを行方向に各 PE$(i+l)$ に分配する．詳細は略すが，PAX 型の場合と同様に右向きに $\sqrt{P}/2$ ステップ，左向きに $(\sqrt{P}/2-1)$ ステップの一斉転送を行い，各ステップでそれぞれの PE は自分を行先とする $(N^2/P^2)\sqrt{P}$ 個のデータをとり込む． □

このアルゴリズムの各フェーズのステップ数および各ステップで一つの PE が送出するデータの数は PAX 型のアルゴリズムのそれと同じである．したがって転送時間・スイッチング時間も PAX 型のアルゴリズムのそれと同じである．

転送時間：$\dfrac{N^2}{2\sqrt{P}}\tau$.

スイッチング時間：$2(\sqrt{P}-1)\alpha$.

この転送時間は式(3.1)で示される下界よりも少し悪いだけなので，本アルゴリズムは転送時間に関してほぼ最適であるといえる．

下界に等しくない理由は最短経路で転送していないデータがあるためである．この例として $P=16$ の場合に初期状態で PE 0 にあるデータの転送経路を**図5**に示す．PE 10, 14 へは図の破線の経路が最短であるが，本アルゴリズムでは簡単のため実線の経路をとっており，この分だけ下界より悪い．

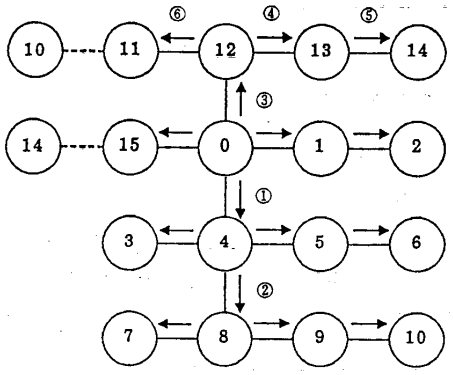


図 5 Illiac Ⅳ 型において初め PE 0 にあるデータの転送経路 ($P=16$, 破線は最短経路)
Fig. 5 Transfer path of data that is initially in PE 0 (Illiac IV-type, $P=16$, broken lines show shortest paths).

表 1 格子状結合方式における行列転置の下界と実現値
Table 1 The lower bounds and the realized values for matrix transposition on mesh-connected computers.

(a) $P \leqq N$, Illiac Ⅳ 型

	下界	Bhuyan らのアルゴリズム	本論文のアルゴリズム
転送時間	$\left(\frac{N^2}{2\sqrt{P}}-\frac{N^2}{2P\sqrt{P}}\right)\tau$	$\frac{N^2}{2P}(P-1)\tau$	$\frac{N^2}{2\sqrt{P}}\tau$
スイッチング時間	—	$(P-1)\alpha$	$2(\sqrt{P}-1)\alpha$

(b) $P \leqq N$, PAX 型

	下界	本論文のアルゴリズム
転送時間	$\frac{N^2}{2\sqrt{P}}\tau$	$\frac{N^2}{2\sqrt{P}}\tau$
スイッチング時間	—	$2(\sqrt{P}-1)\alpha$

(c) $N<P<N^2$

	Illiac Ⅳ 型 下界	PAX 型 下界
転送時間	$\left(\frac{N^2}{2\sqrt{P}}-\frac{N}{4\sqrt{P}}\right)\tau$	$\frac{N^2}{2\sqrt{P}}\tau$

まえがきで触れた Bhuyan and Agrawal[1] のアルゴリズムの転送時間・スイッチング時間はそれぞれ $(N^2/2P)\cdot(P-1)\tau, (P-1)\alpha$ である．これに比べると本章のアルゴリズムは大幅な改善となっている．

5. むすび

2次元格子状結合計算機上に行優先に分割されて保持されている行列の転置について，必要な転送時間の下界を得，さらに PE 数 P が行列の一辺 N 以下のときについてほぼ最適なアルゴリズムを示した．本アルゴリズムは Bhuyan らのものに比べ転送時間を約 $1/\sqrt{P}$ に，スイッチング時間を約 $2/\sqrt{P}$ に短縮している．結果を**表 1** にまとめておく．

まえがきにおける考察と同様に $P \propto N^2$ とおくと，本論文のアルゴリズムの転送時間，スイッチング時間はともに $O(N)$ となる．2次元フーリエ変換に適用した場合，これは依然として転置を除いた計算時間のオーダよりも高い．このことは，この種の問題ではやはりデータ転送に要する時間が性能のネックになる可能性があることを示している．

本論文で用いた転送時間の下界の求め方は一般のデータ並べ換えに適用できる．

$N<P<N^2$ のときの効率よい転置アルゴリズムは今後の課題である．またこの場合について本論文で得た下界は tight ではないと予想されるので，下界の改良も今後の課題である．

謝辞 日頃ご討論をいただいた津田研究室の諸兄に深く感謝する．

参考文献

1) Bhuyan, L. N. and Agrawal, D. P.: Performance Analysis of FFT Algorithms on Multiprocessor Systems, *IEEE Trans. Softw. Eng.*, Vol. SE-9, No. 4, pp. 512-521 (1983).
2) Nassimi, D. and Sahni, S.: An Optimal Routing Algorithm for Mesh-Connected Parallel Computers, *J. ACM*, Vol. 27, No. 1, pp. 6-29 (1980).
3) Orcutt, S. E.: Implementation of Permutation Functions in Illiac IV-Type Computers, *IEEE Trans. Comput.*, Vol. C-25, No. 9, pp. 929-936 (1976).
4) 影山隆久，阿部秀彦，白川友紀，星野 力：アレイ型並列計算機 PAX-128，情報処理学会第 27 回全国大会講演論文集 (I)，6 N-1, pp. 61-62 (1983).

付録 3.2 節における u_{LB} の計算

この計算は少し面倒であるが，概略のみ示しておく．

i) Illiac Ⅳ 型

$$S(x)=\sum_{k=0}^{N^2/P-1}\sum_{k'=0}^{N^2/P-1} L_I\left(xN+(k'-k)\frac{P}{N}-x\right)$$
$$=\sum_{t=-N^2/P+1}^{N^2/P-1}\left(\frac{N^2}{P}-|t|\right)L_I\left(xN+t\frac{P}{N}-x\right)$$

とおくと，

$$u_{LB}=\sum_{i=0}^{P/N-1}\sum_{j=0}^{P/N-1} S(j-i)$$

$$=\frac{P}{N}S(0)+2\sum_{x=1}^{P/N-1}\left(\frac{P}{N}-x\right)S(x) \qquad (A.1)$$

$(\because\ S(x)=S(-x))$

$S(x)$ $(0\le x<P/N)$ は a) $x=0$, b) $0<x<2P/N$, c) $x=2P/N$, d) $2P/N<x<P/N$ の四つの場合に分けて計算する.

a)

$$S(0)=2\sum_{t=1}^{N^2/P-1}\left(\frac{N^2}{P}-t\right)L_1\left(t\cdot\frac{P}{N}\right).$$

$t=u(N/\sqrt{P})+v$

$(u=0,\cdots,N/\sqrt{P}\,;$

$v=-N/2\sqrt{P}+1,\cdots,N/2\sqrt{P})$ (A.2)

とおくと, $t(P/N)=u\sqrt{P}+v(P/N)$, $-\sqrt{P}/2<v(P/N)\le\sqrt{P}/2$ より式 (2.3) が適用できる. 計算すると次式を得る.

$$S(0)=\frac{N^5}{3P^2\sqrt{P}}+\frac{N^4}{4P\sqrt{P}}-\frac{7N^3}{12P\sqrt{P}}.$$

b)

$$S(x)=\sum_{t=1}^{N^2/P-1}\left\{\frac{N^2}{P}L_1\left(xN+t\frac{P}{N}-x\right)-t\frac{N}{\sqrt{P}}\right\}.$$

a) のときと同様に式(A.2)のように分解すると, 式(2.3)が適用できる. 計算すると次式を得る.

$$S(x)=\frac{N^5}{P^2\sqrt{P}}x+\frac{N^4}{4P\sqrt{P}}-\frac{N^3}{2P\sqrt{P}}$$

$(0<x<P/2N).$

c)

$$S\left(\frac{P}{2N}\right)=\sum_{t=1}^{N^2/P-1}tL_1\left(\left(\frac{P}{2N}-1\right)N+t\frac{P}{N}+\frac{P}{2N}\right)+\sum_{t=1}^{N^2/P-1}tL_1\left(\left(\frac{P}{2N}-1\right)N+t\frac{P}{N}-\frac{P}{2N}\right)+\frac{N^2}{P}L_1\left(\frac{P}{2}-\frac{P}{2N}\right).$$

第1項は

$t=u(N/\sqrt{P})+v$

$(u=0,\cdots,N/\sqrt{P}\,;$

$v=-N/2\sqrt{P},\cdots,N/2\sqrt{P}-1)$ (A.3)

とおいて, また第2項は式(A.2)のようにおいて分解する. 計算すると次式を得る.

$$S\left(\frac{P}{2N}\right)=\frac{N^5}{3P^2\sqrt{P}}+\frac{3N^4}{4P\sqrt{P}}+\frac{N^3}{12P\sqrt{P}}.$$

d)

$$S(x)=\sum_{t=1}^{N^2/P-1}\left\{\frac{N^2}{P}L_1\left(\left(\frac{P}{N}-x\right)N+t\frac{P}{N}+x\right)-t\frac{N}{\sqrt{P}}\right\}.$$

式(A.3)のように t を分解して式(2.3)を適用する. 計算すると次式を得る.

$$S(x)=\frac{N^5}{P^2\sqrt{P}}\left(\frac{P}{N}-x\right)+\frac{N^4}{4P\sqrt{P}}+\frac{N^3}{2P\sqrt{P}}$$

$(P/2N<x<P/N).$

以上をもとに(A.1)から u_{LB} を計算すると本文中の結果を得る:

ii) PAX 型

$$u_{LB}=\frac{N^2}{P}\sum_{i,j,l_1,l_3}L_P\left((j-i)\frac{N}{\sqrt{P}}+(l_3-l_1)\right)+\frac{N^2}{P}\sum_{i,j,l_2,l_4}L_P\left((l_4-l_2)\frac{P}{N}+(i-j)\right)$$

$$=\frac{N^2}{P}\left\{\frac{P}{N}S_1(0)+2\sum_{x=1}^{P/N-1}\left(\frac{P}{N}-x\right)S_1(x)+\frac{N}{\sqrt{P}}S_2(0)+2\sum_{z=1}^{N/\sqrt{P}-1}\left(\frac{N}{\sqrt{P}}-z\right)S_2(z)\right\}.$$

ただし

$$S_1(x)=\sum_{y=-N/\sqrt{P}+1}^{N/\sqrt{P}-1}\left(\frac{N}{\sqrt{P}}-|y|\right)L_P\left(x\frac{N}{\sqrt{P}}+y\right),$$

$$S_2(z)=\sum_{x=-P/N+1}^{P/N-1}\left(\frac{P}{N}-|x|\right)L_P\left(z\frac{P}{N}+x\right).$$

$S_1(x)$ $(0\le x<P/N)$ は $x=0$, $0<x<P/2N$, $x=P/2N$, $P/2N<x<P/N$ の四つの場合に分けて, また $S_2(z)$ $(0\le z<N/\sqrt{P})$ は $z=0$, $0<z<N/2\sqrt{P}$, $z=N/2\sqrt{P}$, $N/2\sqrt{P}<z<N/\sqrt{P}$ の四つの場合にそれぞれ分けて計算すると, 式(2.6)が適用できて本文中の結果を得る.

(昭和60年 4 月 30 日受付)
(昭和60年11月21日採録)

中野 浩行

1960年生. 1983年京都大学工学部情報工学科卒業. 1985年同大学院修士課程修了. 現在ソニー(株)情報システム管理部に勤務.

津田 孝夫

1932年生. 1957年京都大学工学部電気工学科卒業. 現職は京都大学工学部情報工学科教授. 工学博士. 現在の主要研究テーマは, メモリ階層間データ転送量の下界とそれによるアルゴリズムの最適化, ベクトル計算機のための自動ベクトル化と自動並列化, 実時間オペレーティングシステムなど専用 OS の構成と実現法など.