Comp. & Maths. with Appls. Vol. 12B, No. 5/6, pp. 1217–1224, 1986
Printed in Great Britain.
0886-9561/86 $3.00 + 0.00

MOBILE JACOBI SCHEMES FOR PARALLEL COMPUTATION†

J. J. MODI
Engineering Department, Trumpington St, Cambridge, Cambs. CB2 1PZ, England

J. D. PRYCE
Computer Science Department, School of Mathematics, University of Bristol, Bristol, Avon BS8 1TW, England

(*Received* 17 *September* 1984)

Abstract—Parallel computers (such as the distributed array processor and systolic arrays) bring into consideration the Jacobi method where several non-interacting rotations can be performed simultaneously. However, the design of the algorithm is much more crucial in a parallel environment; inefficiencies can lead to considerable organizational costs. This paper provides a general framework for the description of mobile schemes together with two specific schemes, the better of which reduces the organizational overheads for the Jacobi method to zero.

1. INTRODUCTION

We first give a brief introduction to parallel computers. Parallelism has been introduced in novel architectural design firstly by pipelining, in which overlapping subtasks are executed simultaneously. Coupled with pipelining, in many machines such as the Cray-1, there are also independent hardware units for performing operations, such as addition and multiplication, which allow greater number of simultaneous operations and this in turn results in an improvement in speed. This class of machines is often referred to as vector processors. Another concept is that of array processors, such as the distributed array processor (DAP), which basically performs the same instruction on sets of data. This class of machine is often referred to as single instruction stream multiple data stream (SIMD). In this paper we are concerned with the DAP, which consists of 64×64 processors p_{ij} in a square grid with nearest-neighbour interconnection, and data storage may be thought of as a set of planes below each processor, each holding a 64×64 array. A processor has fast access to the data in its "column" and those of its four neighbours. There is a single instruction stream broadcast to all processors. However, masks may be set so that only a subset of the processors perform a given operation.

The design of algorithms for SIMD and vector processors tend to be similar. In this respect the algorithm design developed here will be applicable to other vector machines and indeed on systolic arrays, which have recently received a lot of attention. The processors, like the DAP, have the nearest-neighbour connections, so that the mobile scheme suggested in Section 3 will also be applicable to systolic architecture. A detailed description on parallel architectures and algorithm design is given by Parkinson[1].

The Jacobi method for the computation of the eigenvalues and eigenvectors of a symmetric matrix applies a sequence of similarity transformations $A := R^{\mathrm{T}}AR$, where R is a plane rotation:

$$R = \begin{array}{c} \begin{bmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & \ddots & & & & \\ & & & \mathrm{C} & -\mathrm{S} & & \\ & & & \mathrm{S} & \mathrm{C} & & \\ & & & & & \ddots & \\ & & & & & & 1 \\ & & & & & & & 1 \end{bmatrix} \end{array} \begin{array}{l} \\ \\ \\ \text{—Row } p \\ \text{—Row } q \\ \\ \\ \\ \end{array} \tag{1}$$

where $\mathrm{C} = \cos\theta$ and $\mathrm{S} = \sin\theta$.

†Parts of the introductory description in sections 1 and 2 are taken from the authors' earlier paper[3].

Here R is chosen so as to decrease the off-diagonal sum of squares by reducing the element a_{pq}, usually to zero[5, 7]. The method is attractive for a multiprocessor machine such as the DAP, and systolic array, because at each stage one can perform up to $[n/2]$ rotations simultaneously on an $n \times n$ matrix.

The literature contains a number of different methods for determining a sequence of $[n/2]$ pairs specifying the pivot elements a_{pq} that are to be simultaneously annihilated at successive stages. Indeed, the specification of such sequences may be said to date back to "Kirkman's Schoolgirls problem", first posed in *The Lady's and Gentleman's Diary* in 1850; for an interesting discussion see Rouse Ball and Coxeter[2].

In an earlier paper[3] the term "mobile schemes" was introduced and the method briefly discussed. Here the rows and columns of A are permuted systematically to reduce the data movement overhead which, when rows p and q in equation (1) are far apart, can be serious in a mesh-connected architecture like that of the DAP.

The aim of the present paper is to provide a general framework for the description of mobile schemes together with two specific schemes for arbitrary n, the better of which (Scheme 1) reduces the relocation overhead to zero. We in fact arrange that just two groups of pivot elements are used alternatively at successive stages, such as $\{a_{12}, a_{34}, \ldots\}$ and $\{a_{23}, a_{45}, \ldots\}$ in Scheme 1 below.

2. THE PARALLEL JACOBI METHOD

We give a brief outline of this, in order to fix some notation and to provide the setting for what follows.

The element a_{pq} $(p \neq q)$ of a real symmetric $n \times n$ matrix A can be reduced by applying a plane rotation through an appropriate angle θ, $|\theta| < \pi/4$; i.e. by forming $R^{\mathrm{T}}AR$, where R is a block-diagonal matrix, defined as follows:

$$R_{pq} = -R_{qp} = -\sin\theta, \quad R_{pp} = R_{qq} = \cos\theta; \quad R_{ij} = \delta_{ij} \text{ otherwise.}$$

Usually, θ is determined by the formula

$$\tan 2\theta = 2\, a_{pq}/(a_{pp} - a_{qq})\,(p < q), \tag{2}$$

and then a_{pq} is annihilated (reduced to zero); we shall often write as if this is always the case. (If a_{pq} is already zero then no rotation is applied: R is the identity matrix.)

In the Jacobi method, A is reduced to diagonal form by applying an infinite sequence of such rotations $R_k = R(p_k, q_k;\ \theta_k)$, where (p_k, q_k), $k = 1, 2, \ldots$ runs through all pairs (p, q) with $1 \leqslant p < q \leqslant n$ infinitely often; the classical order is

$$(1, 2),\ (1, 3), \ldots, (1, n),\ (2, 3), \ldots, (2, n), \ldots, (n-1, n)$$

repeated indefinitely. Let $A_0 = A$, and

$$A_k = R_k^T A_{k-1} R_k \quad (k = 1, 2, \ldots).$$

The θ_k are determined at each stage from the elements a_{pp}, a_{qq}, a_{pq} $(p = p_k,\ q = q_k)$ in A_{k-1}, typically by formula (2). The overriding aim is to decrease the sum of squares of the off-diagonal elements at each stage, so that as $k \to \infty$, A_k converges to a diagonal matrix containing the eigenvalues of A_0 and $R_1 R_1 \ldots R_k$ to an orthogonal matrix whose columns are the corresponding eigenvectors.

Each rotation $R\ (p, q; \theta)$ affects only the pth and qth rows and columns of A while annihilating a_{pq} $(= a_{qp}$. This feature of the method is exploited on a parallel machine by annihilating many elements simultaneously.

A set of rotations $R_i = R(p_i, q_i;\ \theta_i)\ (i = 1, 2, \ldots, m)$ in which $\{p_i, q_i\} \cap \{p_j, q_j\} = \phi\ (i \neq j)$ is called disjoint; the compound rotation matrix $J = \Pi_i R_i$ annihilates m elements of A simultaneously.

We write

$$\operatorname{piv}(J) = \{\{p_i, q_i\}: i = 1, 2, \ldots, m\},$$

and we write $\mathbb{J}$ for the set of all such matrices J. As each rotation affects two rows (and columns) of A, the maximum number of rotations that can be performed simultaneously is clearly $[n/2]$.

The Jacobi method where compound rotations are performed at each stage we term the *parallel Jacobi method.*

3. MOBILE SCHEMES

In this section we use the following notation. A permutation σ is a one–one map of $\{1, \ldots, n\}$ onto itself. We denote by e the identity permutation, and by $\tau\sigma$ the composition of permutations σ, τ:

$$(\tau\sigma)(i) = \tau[\sigma(i)]. \tag{3}$$

We denote by σ^* the corresponding permutation–transformation of n-vectors (or its $n \times n$ matrix)

$$\sigma^* x = y, \quad \text{where } y_i = x_{\sigma(i)}. \tag{4}$$

Note that $(\sigma^*)^{\mathrm{T}} = (\sigma^{-1})^*$ and that if A is an $n \times n$ matrix, then

$$B = (\sigma^*)^{\mathrm{T}} A \sigma^* \tag{5}$$

is A with its ith row (column) moved to lie in the $\sigma(i)$th row (column) for each i. The result of doing the sequence of transformations (5), applying $\sigma_1, \sigma_2, \ldots, \sigma_k$ in that order, is equivalent to forming $(\sigma^*)^{\mathrm{T}} A \sigma^*$, where $\sigma = \sigma_k \ldots \sigma_2 \sigma_1$ (since both * and $^{\mathrm{T}}$ reverse the order of products).

Lemma 3.1. If σ is a permutation and $J \in \mathbb{J}$ then $K = \sigma^{*\mathrm{T}} J \sigma^*$ is in $\mathbb{J}$ and has pivot points

$$\operatorname{piv}(K) = \{\{\sigma(p), \sigma(q)\} : \{p, q\} \in \operatorname{piv}(J)\}. \tag{6}$$

Proof. It suffices to verify that

$$\sigma^{*\mathrm{T}} R(p, q; \theta)\, \sigma^* = R[\sigma(p), \sigma(q); \theta]. \tag{7}$$

Definition 3.2. A mobile (parallel Jacobi) scheme of length m is a sequence of $J_k \in J$ and permutations σ_k, $(k = 1, \ldots, m)$ which, given an $n \times n$ matrix A, define a sequence of transformed matrices $A_0, \ldots, A_m$ by

$$A_0 = A; \quad A_k = \sigma_k^{*\mathrm{T}} J_k^{\mathrm{T}} A_{k-1} J_k \sigma_k, \quad (k = 1, \ldots, m). \tag{8}$$

We speak of "the scheme $J_1\sigma_1 \ldots J_m\sigma_m$". An ordinary Jacobi scheme, henceforth called a *stationary scheme*, is one where all the σ_k equal e, and we speak of "the scheme $J_1 \ldots J_m$" in this case.

As we implement mobile schemes the σ_k and the pivots of the J_k are chosen in advance, leaving only the rotation angles to be determined as computation proceeds depending on the matrix A. This simplifies programming and there seems no advantage in organizing the algorithm differently. Strictly speaking the "scheme" is determined by the σ_k and the $\operatorname{piv}(J_k)$, different J_k with the same pivots giving "instances" of the scheme, but we shall rarely make this distinction.

We now show that, as one might expect, the entries in the A_k produced by a mobile scheme are merely rearrangements of those in a stationary scheme. In practice a scheme corresponds to a complete sweep, the sequence $J_1\sigma_1 \ldots J_m\sigma_m$ being repeated (with different rotation angles) until convergence occurs.

Definition 3.3. Two mobile schemes $J_1\sigma_1 \ldots J_m\sigma_m$ and $\tilde{J}_1\tilde{\sigma}_1 \ldots \tilde{J}_m\tilde{\sigma}_m$ are equivalent if there are permutations $\Pi_0 \ldots \Pi_m$ such that, for every A, the transformed matrices $A_0 \dots A_m$ and $\tilde{A}_0 \dots \tilde{A}_m$ produced by the schemes satisfy

$$A_r = \Pi_r^{*\mathrm{T}} \tilde{A}_r \Pi_r^* \quad (r = 0, \dots, m). \tag{9}$$

Obviously Π_0 must equal e.

Theorem 3.4. Any mobile scheme is equivalent to a (unique) stationary scheme.

Proof. Given the scheme $J_1\sigma_1 \dots J_m\sigma_m$, defining $A_0 \dots A_m$ as in equations (8), define the Π_k inductively by

$$\Pi_0 = e; \quad \Pi_k = \sigma_k \Pi_{k-1} \quad (k = 1, \dots, m). \tag{10}$$

Consider the stationary scheme $\tilde{J}_1 \dots \tilde{J}_m$ where

$$\tilde{J}_k = \Pi_{k-1}^* J_k \Pi_{k-1}^{\mathrm{T}} \tag{11}$$

which defines the sequence

$$\tilde{A}_0 = A; \quad \tilde{A}_k = \tilde{J}_k^{\mathrm{T}} \tilde{A}_{k-1} \tilde{J}_k \quad (k = 1, \dots, m). \tag{12}$$

It is easily verified by induction that equation (9) then holds. The equivalence is unique in the sense that equations (10) and (11) define the only Π_k, $\tilde{J}_k$ such that equation (9) holds for every A. This follows by induction, using the fact that if a matrix J is both a member of $\mathbb{J}$ and a permutation matrix, it equals the identity.

Corollary 3.5. With the above notation, the pivots of the matrix $\tilde{J}_k$ of the stationary scheme are $\{\Pi_{k-1}^{-1}(p_i), \Pi_{k-1}^{-1}(q_i)\}$ where $\{p_i, q_i\}$ are the pivots of J_k.

Proof. By Lemma 1.

We can display the pivots and data movement in a mobile scheme by a *migration table*, shown here for $n = 4$, $m = 2$ where $\mathrm{piv}(J_1) = \mathrm{piv}(J_2) = \{\{1, 2\}, \{3, 4\}\}$ and where σ_1 maps i to $i + 1$ (mod 4) and σ_2 maps i to $5 - i$:

$i \setminus k$	0	1	2
1	1	2	1
2	2	3	4
3	3	4	3
4	4	1	2
	J_1, σ_1	J_2, σ_2	

The i, k entry shows which original row (or column) of A has migrated to row (or column) i after σ_k is applied (the elements of the row or column having been subjected to the rotations of $J_1, \dots, J_k$ en route but not yet by those of J_{k+1}). It equals $\Pi_k^{-1}(i)$. The kth column, regarded as a vector, is $\sigma_k^{*\mathrm{T}}$ times the $(k - 1)$th. Lines joining pairs of indices $p_r q_r$ in column k, lying in rows i_r, j_r, show that $\{i_r, j_r\}$ are the pivots of J_{k+1} while $\{p_r, q_r\}$ are the pivots of the corresponding $\tilde{J}_{k+1}$ of the equivalent stationary scheme.

Definition 3.6. We call a mobile scheme a sweep if the corresponding stationary scheme is a sweep, i.e. if

$$\bigcup_{k=1}^{m} \mathrm{piv}(J_k)$$

is the full set of $n(n - 1)/2$ pivot points.

In the above example, $\bigcup \mathrm{piv}(\tilde{J}_k) = \{\{1, 2\}, \{3,4\}, \{2,3\}, \{1,4\}\}$.

We now describe a scheme which is well-suited to the DAP architecture. For given n, let

$$n' = [(n - 1)/2], \; n'' = [n/2], \; ([\;] = \text{integer part}).$$

Define the permutation ω (for odd) on $\{1, \ldots, n\}$ to be the product of interchanges $(1, 2)$, $(3, 4), \ldots, (2n'' - 1, 2n'')$, and define η (for even) to be the product of interchanges $(2, 3)$, $(4, 5), \ldots, (2n', 2n' + 1)$.

Let $\mathbb{Q}$ denote the set of $J \in \mathbb{J}$ having the form

$$\begin{bmatrix} C_1 & -S_1 & & & & & \\ S_1 & C_1 & & & & & \\ & & C_2 & -S_2 & & & \\ & & S_2 & C_2 & & & \\ & & & & \ddots & & \\ & & & & & C_{n''} & -S_{n''} \\ & & & & & S_{n''} & C_{n''} \end{bmatrix}$$

with a unit block [1] at the bottom right if n is odd,

i.e. the set of J having pivots at $\{1, 2\}, \{3, 4\}, \ldots, \{2n'' - 1, 2n''\}$.

Let $\mathbb{E}$ denote the set of $J \in \mathbb{J}$ having the form

$$\begin{bmatrix} 1 & & & & & \\ & C_1 & -S_1 & & & \\ & S_1 & C_1 & & & \\ & & & \ddots & & \\ & & & & C_{n'} & -S_{n'} \\ & & & & S_{n'} & C_{n'} \end{bmatrix}$$

with a unit block [1] at the bottom right if n if even,

i.e. the set of J having pivots at $\{2, 3\}, \{4, 5\}, \ldots, \{2n', 2n' + 1\}$. Clearly, J belonging to $\mathbb{Q}$ or $\mathbb{E}$ can be applied on the DAP with negligible data movement overhead.

Definition 3.7. The mobile Scheme 1 (MS1) for $n \times n$ matrices is a scheme $J_1\sigma_1 \ldots J_n\sigma_n$, of length n, with

$\sigma_k = \omega$ and $J_k \in \mathbb{Q}$ for k odd,

$\sigma_k = \eta$ and $J_k \in \mathbb{E}$ for k even.

Theorem 3.8. Scheme MS1 is a sweep, for any n.

Proof. We do not give a formal proof, but illustrate with the migration table (Table 1) for the case $n = 9$.

Table 1. Migration table for MS1 with $n = 9$

i \ k	0	1	2	3	4	5	6	7	8	9
1	1	2	2	4	4	6	6	8	8	9
2	2	1	4	2	6	4	8	6	9	8
3	3	4	1	6	2	8	4	9	6	7
4	4	3	6	1	8	2	9	4	7	6
5	5	6	3	8	1	9	2	7	4	5
6	6	5	8	3	9	1	7	2	5	4
7	7	8	5	9	3	7	1	5	2	3
8	8	7	9	5	7	3	5	1	3	2
9	9	9	7	7	5	5	3	3	1	1
		J_1, ω	J_2, η	J_3, ω	J_4, η	J_5, ω	J_6, η	J_7, ω	J_8, η	J_9, ω

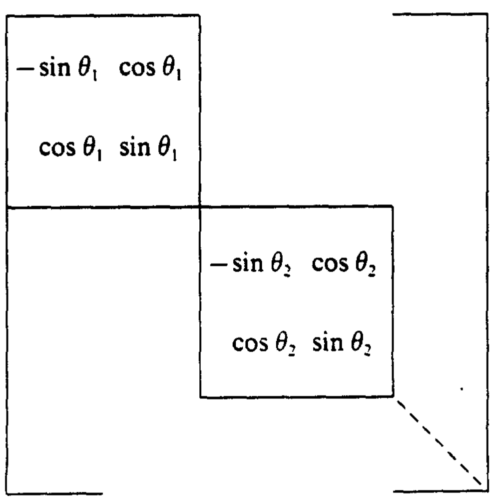
Each index migrates in a simple way, and it can easily be verified that for general n, the i, k entry μ_{ik} is given by

$$\mu_{ik} = \begin{cases} i + k \text{ if } i + k \leqslant n \\ 2n + 1 - (i + k) \text{ otherwise} \end{cases} \quad (i \text{ odd}) \qquad \begin{cases} i - k \text{ if } i - k \geqslant 1 \\ 1 - (i - k) \text{ otherwise} \end{cases} \quad (i \text{ even}).$$

Further, each pair of indices p, q, with $p \neq q$, are adjacent to each other in a pivot position exactly once. This makes the scheme a sweep. Finally, at the end of n stages the order of indices has been exactly reversed.

On the DAP, the data movement in MS1 can be completely overlapped with the numerical computation. This is because, at each odd-numbered stage, the transformation of A is $A := K^T AK$, where $K = J\omega^*$ with $J \in \mathbb{Q}$.

Such K are of the form



Similarly, even-numbered stages apply $A := K^{\mathrm{T}}AK$, where K has the form

$$\begin{bmatrix} 1 & & & & & \\ & -\sin\theta_1 & \cos\theta_1 & & & \\ & \cos\theta_1 & \sin\theta_1 & & & \\ & & & -\sin\theta_2 & \cos\theta_2 & \\ & & & \cos\theta_2 & \sin\theta_2 & \\ & & & & & \ddots \end{bmatrix}$$

It is as easy to apply such matrices K as to apply $J \in \mathbb{Q}$ or $J \in \mathbb{E}$, so MS1 on the DAP is an ideal scheme in that it incurs no data movement overhead at all.

The mobile Scheme 2 (MS2) is another scheme with simple, regular data movement, though slightly less efficient on the DAP than MS1. It has length n, and for odd-numbered stages k it has

$$\mathrm{piv}(J_k) = \{\{1, n\}\{2, n-1\}\cdots\{n'', n+1-n''\}\}$$

and σ_k is the identity. For even stages,

$$\mathrm{piv}(J_k) = \{\{2, n\}\{3, n-1\}\cdots\{n'+1, n+1-n'\}\}$$

and σ_k is the cyclic rotation $\sigma_k(i) = i + 1 \pmod{n}$.

The migration table for $n = 8$ is shown in Table 2. It is easy to see that MS2 is a sweep. On the DAP it can take advantage of the standard functions REVC, REVR which reverse the order of the columns or rows of a matrix.

4. CONCLUSION

Parallel computers bring into consideration the Jacobi method where several non-interacting rotations can be performed simultaneously[4]. However, in contrast to serial programming the design of parallel algorithm is much more crucial. In this paper a general framework of describing mobile schemes for the implementation of Jacobi's method on parallel computers has been presented.

Details of implementation on the DAP have been discussed previously[3], and implementation on systolic array architecture has been considered by Heller and Ipsen[6].

These techniques substantially reduce the organizational costs, and are applicable to other areas such as the generalized eigenvalue problem[7].

Table 2. Migration table for MS2 with $n = 8$

i \ k	1	2	3	4	5
1	1	1	2	2	
2	2	2	3	3	
3	3	3	4	4	
4	4	4	5	5	etc.
5	5	5	6	6	
6	6	6	7	7	
7	7	7	8	8	
8	8	8	1	1	

REFERENCES

1. D. Parkinson, Parallel processing architecture and parallel algorithms. In *New Advances in Distributed Computer Systems* (Edited by K. G. Beauchamp); *NATO Adv. Study Inst. Ser.* **80,** 367–377 (1981).
2. W. W. Rouse Ball and H. S. M. Coxeter, *Mathematical Recreations and Essays*. Macmillan, London (1939).
3. J. J. Modi and J. D. Pryce, Efficient implementation of Jacobi's diagonalization method on the DAP. *Num. Math.* **46,** 443–454 (1985).
4. A. H. Sameh, On Jacobi and Jacobi-like algorithms for parallel computers. *Maths Comput.* **25,** 579–590 (1971).
5. H. Rutishauser, The Jacobi method for real symmetric matrices. *Num. Math.* **9,** 1–10 (1966).
6. D. E. Heller and I. C. F. Ipsen, Systolic networks for orthogonal decompositions. *SIAM Jl scient. statist. Comput.* 261–269 (1983).
7. J. H. Wilkinson, *The Algebraic Eigenvalue Problem*. Clarendon Press, Oxford (1965).