Decoupling the Dimensions of a System of Affine Recurrence Equations

Yoav Yaacoby
Department of Electrical & Computer Engineering
University of California
Santa Barbara, CA 93106

Peter R. Cappello[*]
Department of Computer Science
University of California
Santa Barbara, CA 93106

November 18, 2008

Abstract

Most work on the problem of synthesizing a systolic array from a system of recurrence equations is restricted to systems of uniform recurrence equations. In this paper, this restriction is relaxed to include systems of affine recurrence equations. A system of uniform recurrence equations typically can be embedded in spacetime so that the distance between a variable and a dependent variable does not depend on the problem size. Systems of affine recurrence equations which are not uniform, do not enjoy this property. A method in another paper has been presented for converting a system of affine recurrence equations to an equivalent system of recurrence equations that is uniform, except for points near the boundaries of its index sets.

[*]This work supported by the National Science Foundation under grant MIP89-20598, and the Lawrence Livermore National Laboratories.

In this paper a procedure is presented for decoupling the dimensions of the system of affine recurrence equations, thereby simplifying the conversion to an equivalent system that is uniform (except for points near the boundaries).

Key Words: affine recurrence equation, concurrent computation, data dependence, decoupling, parallel computation, processor array, systolic array, uniform recurrence equation.

1 Introduction

An important property of any VLSI system is physical regularity. Systolic arrays [14, 13] and wavefront arrays [15] have an iterative form of physical regularity. Array architecture is suited to VLSI technology; replicating a processing element reduces design cost, and neighbor communication between processing elements reduces operation cost. Leiserson, Rose, and Saxe [17, 16] show how to convert a finite network without zero-delay cycles to an equivalent network that functions systolically (see also [28]). Melhem and Rheinboldt [19] give a mathematical model for the verification of systolic networks.

A system of uniform recurrence equations, as defined by Karp, Miller, and Winograd [12, 11], maps especially well onto a systolic/wavefront array. This is noted explicitly by Chen and Mead [4], and Quinton [25, 26], for example. Linearly mapping a system of recurrence equation's index sets into spacetime, has been pursued by Cappello and Steiglitz [1, 2, 3]. Fortes et al. [9] survey seventeen methodologies for the design of systolic arrays. Systematic translation from either a program fragment or a system of uniform recurrence equations to systolic/wavefront arrays has been studied by Moldovan [22] [21], Quinton [25, 26], Winkler and Miranker [20], Delosme and Ipsen [7], and Moldovan and Fortes [23]. Rao [27] introduces and analyzes a class of algorithms, called regular iterative algorithms, which contains systolic algorithms.

A system of uniform recurrence equations typically can be mapped linearly into spacetime so that interprocessor communication requires only a fixed amount memory, and fixed-length interconnections. There is no such linear mapping into spacetime for systems of *affine* recurrence equations which are not uniform. Delosme and Ipsen [8] present the first elements of a methodology for determining systolic array schedules for a 2-dimensional system of affine recurrence equations. Choffrut and Culik [5] treat a related problem. They apply a geometric transformation to a systolic array, such that an output can be fed back as an input via physically neighboring connections. They fold the array, eliminating long wires for connections between elements (in a 2-dimensional array) that are related by reflections and/or rotations. Yaacoby and Cappello [29] treat systems of affine recurrence equations of any finite dimension. They formulate a 'generalized fold', and provide a procedure for converting a system of affine recurrence equations to an equivalent system that is uniform except for points near its boundaries. These latter systems are called systems of *quasi-uniform* recurrence equations.

In this paper a method is presented which decouples the dimensions of the system before conversion whenever possible, thus simplifying the conversion. Such a decoupling entails finding a similarity transformation that can be applied to an integer matrix group which partially diagonalizes each matrix in the group. This partial diagonalization is reminiscent of the Jordan form. The Jordan form however cannot be used for three reasons:

- We have a group of matrices, not a single matrix;
- The similarity matrix must be integer;
- The transformed matrix group must be integer.

This decoupling also may be useful in other applications.

2 Definitions

The equations in Ex. 1 below are an example of a system of recurrence equations (SRE).

Example 1

$1 \le j \le n-1,$

$$1 \le i \le j-1, \qquad a_3(i,j) = a_3(i+1, j-1) \tag{1}$$

$$1 \le i \le j+1, \qquad a_2(i,j) = -a_3(i+1,j)a_2(j+2-i, j-1) + a_2(i, j-1) \tag{2}$$

These recurrence equations are used to illustrate some of the following definitions, which are related to an SRE.

Index set: The set of points where an array is computed or used.

Domain of computation: The set of points C_i where an array a_i is computed (e.g., $C_2 = \{(i,j) | 1 \le j \le n-1,\ 1 \le i \le j+1\}$ in Eq. (2)).

Dependence map: A function δ_{ij} from the domain of computation of array a_j to the index set of a_i, on which the computation of a_j depends (e.g., $\delta_{32}(p) = p + (\begin{array}{cc} 1 & 0 \end{array})^T$ in Eq. (2)).

Affine dependence: A dependence map of the form: $\delta_{ij}(p) = D_{ij}p + d_{ij}$ where $D_{ij} \in \mathbf{Z}^{n \times n}$, and $d_{ij} \in \mathbf{Z}^n$ (e.g., $\delta_{22}(p) = \begin{pmatrix} -1 & 1 \\ 0 & 1 \end{pmatrix} p + \begin{pmatrix} 2 \\ -1 \end{pmatrix}$ in Eq. (2)).

In the remainder of this paper, we assume that D_{ij} is nonsingular and integer, unless specified otherwise.

Uniform dependence: An affine dependence of the form: $\delta_{ij}(p) = p + d_{ij}$ (i.e., $D_{ij} = I$).

A system of affine [uniform] recurrence equations (SARE [SURE]): SRE where the dependence maps are affine [uniform], and every array is computed in one recurrence equation for its entire domain of computation (e.g., Eqs. (1,2) are an SARE).

3 Decoupling the Dimensions of an SARE

This section describes how the dimensions of an SARE can be decoupled. This procedure is applied before converting the SARE to a system of quasi-uniform recurrence equations (a system that is uniform except for points near it boundaries [30]). This reduces the dimension of the SARE that needs to be converted.

3.1 Decoupling a finite group of integral matrices

The following lemma was proved by C. Hermite in 1849 [18].

Lemma 3.1.1 *Let $a_1, a_2, \ldots, a_n$ be integer*[1] *numbers with greatest common divisor d_n (in case all numbers are zero, define $d_n = 0$). There exists an integral matrix of determinant d_n having $a_1, a_2, \ldots, a_n$ as its first row.*

The matrix constructed in the proof of the above lemma is almost lower triangular (except for the first row).

The following theorem is useful in decoupling the dimensions in SAREs that can be converted to a system of quasi-uniform recurrence equations.

Lemma 3.1.2 *Let G be a finite group of integral matrices $\{D_i\}$ (i.e., $G \subset GL(n, Z)$). If there exists a nonzero vector $\pi \in \mathbf{Z}^n$ which satisfies $\pi^T D_i = \pi^T \;\; \forall D_i \in G$, then there exists $M \in \mathbf{Z}^{n \times n}$ such that*

$$MD_iM^{-1} = \begin{bmatrix} 1 & 0 & . & . & . & 0 \\ 0 & & & & & \\ . & & & & & \\ . & & & B_i & & \\ . & & & & & \\ 0 & & & & & \end{bmatrix}, \quad \forall D_i \in G \tag{1}$$

where $B_i \in \mathbf{Z}^{(n-1)\times(n-1)}$, and the first row of M is π^T.

Proof. Suppose π is primitive[2]. According to Lemma 3.1.1 there exists a matrix A, having π^T as the first row, whose determinant is 1. A^{-1} thus is integral. The first row of

[1] The a_i, in general, can be elements of any principal ideal ring.

[2] That is, the greatest common divisor of its components is 1.

AD_iA^{-1} is $1, 0, 0, \ldots, 0$ for all $D_i \in G$. Also, since all matrices are integral, their product is integral. Denote the above product by $\begin{bmatrix} 1 & 0 \\ l_i & B_i \end{bmatrix}$, where l_i is a column vector, and B_i is an $(n-1) \times (n-1)$ sub-matrix. Since A is fixed for all $D_i \in G$, G is reducible [6]. From Maschke's theorem [24, Thm. 9 pp. 14], it follows that G is fully reducible. Thus, there exists a matrix M which satisfies Eq. 1. In particular, $M = T^{-1}A$, where A is the matrix mentioned above, and

$$T = \begin{bmatrix} 1 & 0 \\ \sum l_i/h & I \end{bmatrix}, \quad T^{-1} = \begin{bmatrix} 1 & 0 \\ -\sum l_i/h & I \end{bmatrix},$$

where h is the order of G. From this construction (given in [24]), the first row of M is π^T, as needed. Also, M is rational since A is integer, and T^{-1} is rational. By scaling all rows of M except the first, and corresponding columns of M^{-1} (by a positive integer), one gets an integral M, as needed. The first row of M is still π^T. Also, the new M and M^{-1} still satisfy Eq. 1.

Now, if π is not primitive, construct M' as above for $\pi/\gcd(\pi_i)$. Construct M as follows. Multiply the first row of M' by $\gcd(\pi_i)$, such that π^T will be the first row of M, and divide accordingly the first column of $(M')^{-1}$. The new M and M^{-1} satisfy Eq. 1. ■

Example 1A

Consider Ex. 1 given in § 2. In that example, the dependence maps are:

$$\delta_{22}^1(i,j) = \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix} \cdot \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} 2 \\ -1 \end{pmatrix}; \quad \delta_{22}^2(i,j) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} 0 \\ -1 \end{pmatrix};$$

$$\delta_{32}(i,j) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix}; \qquad \delta_{33}(i,j) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{pmatrix} i \\ j \end{pmatrix} + \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

The linear parts of the dependence maps generate the group G which includes the following two matrices: $D_0 = I$, $D_1 = \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix}$. The vector $\pi^T = \begin{pmatrix} 0 & 1 \end{pmatrix}$ is a common left eigenvector of eigenvalue 1 for D_0 and D_1. (This vector is constructed in Ex. 1B using Proc. 1. See § 3.2.) The matrix A, from Lemma 3.1.1, is $A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$. Its inverse is $A^{-1} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. We now have: $AD_0A^{-1} = I$, and $AD_1A^{-1} = \begin{bmatrix} 1 & 0 \\ -1 & -1 \end{bmatrix}$. Thus, $T = \begin{bmatrix} 1 & 0 \\ -1/2 & 1 \end{bmatrix}$, and $T^{-1} = \begin{bmatrix} 1 & 0 \\ 1/2 & 1 \end{bmatrix}$. We therefore get the following matrices: $M = T^{-1}A = \begin{bmatrix} 0 & 1 \\ -1 & 1/2 \end{bmatrix}$, and $M^{-1} = A^{-1}T = \begin{bmatrix} 1/2 & -1 \\ 1 & 0 \end{bmatrix}$. Since we want M to be integer, we multiply its second row by 2, and update M^{-1} accordingly, obtaining:

$$M = \begin{bmatrix} 0 & 1 \\ -2 & 1 \end{bmatrix}, \quad M^{-1} = \begin{bmatrix} 1/2 & -1/2 \\ 1 & 0 \end{bmatrix}.$$

M now is an integral matrix whose first row is π^T. $MD_1M^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, as needed.

Theorem 3.1 *Let G be a finite group of integral matrices $\{D_i\}$ (i.e., $G \subset GL(n, Z)$). If there exist k linearly independent vectors $\pi_1, \pi_2, \ldots, \pi_k \in \mathbf{Z}^n$ which satisfy $\pi_j^T D_i = \pi_j^T \;\; \forall j, \forall D_i \in G$, then there exists $M \in \mathbf{Z}^{n\times n}$ such that*

$$MD_iM^{-1} = \begin{bmatrix} I_{k\times k} & 0 \\ 0 & B_i \end{bmatrix}_{n\times n} \qquad \forall D_i \in G \tag{2}$$

where $B_i \in \mathbf{Z}^{(n-k)\times(n-k)}$, and the first k rows of M are $\pi_1^T, \pi_2^T, \ldots, \pi_k^T$.

Proof. We prove it by induction on k. For $k = 1$, the theorem follows from Lemma 3.1.2. Suppose it is true for $k = l$. We prove it for $k = l + 1$. By supposition, there exists M_l such that

$$M_lD_iM_l^{-1} = \begin{bmatrix} I_{l\times l} & 0 \\ 0 & B_i \end{bmatrix}_{n\times n} \qquad \forall D_i \in G \tag{3}$$

where $B_i \in \mathbf{Z}^{(n-l)\times(n-l)}$, and the first l rows of M_l are $\pi_1^T, \pi_2^T, \ldots, \pi_l^T$.

Let $p^T = \pi_{l+1}^T M_l^{-1}$, and denote the last $n - l$ entries of p by $\tilde{p}$. We claim that $\tilde{p} \neq 0$. Suppose otherwise. Then since $p^T M_l = \pi_{l+1}^T$, it follows that π_{l+1}^T is a linear combination of the first l rows of M_l. But these are $\pi_1^T, \pi_2^T, \ldots, \pi_l^T$, a contradiction; it is given that the $l+1$ vectors are linearly independent.

Let $\hat{p} = c \cdot \tilde{p} \in \mathbf{Z}^{n-l}$ (i.e., c scales $\tilde{p}$ to be a nonzero integer vector $\hat{p}$). Since $\{D_i\}$ form a finite group, the B_i do too. Using Lemma 3.1.2, construct a matrix $A \in \mathbf{Z}^{(n-l-1)\times(n-l)}$ such that

$$\begin{bmatrix} \hat{p}^T \\ A \end{bmatrix}_{(n-l)\times(n-l)} B_i \begin{bmatrix} \hat{p}^T \\ A \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & \hat{B}_i \end{bmatrix} \quad \forall B_i, \;\; \hat{B}_i \in \mathbf{Z}^{(n-l-1)\times(n-l-1)}. \tag{4}$$

Define M_{l+1} as follows:

$$M_{l+1} = \begin{bmatrix} \pi_1^T \\ \vdots \\ \pi_{l+1}^T \\ \begin{bmatrix} 0 & A \end{bmatrix} M_l \end{bmatrix}$$

where A is the matrix as defined above, and 0 is a zero submatrix of dimension $(n-l-1)\times l$. M_{l+1} is an $n \times n$ integer matrix as needed, and its first $l+1$ rows are the given $l+1$ vectors. Since the first $l+1$ rows of M_{l+1} are eigenvectors of eigenvalue 1 for all D_i in G, the first $l+1$ rows of $M_{l+1} D_i M_{l+1}^{-1}$ are $\begin{bmatrix} I & 0 \end{bmatrix}_{(l+1)\times n}$. The last $n-l-1$ rows are:

$$\begin{bmatrix} 0 & A \end{bmatrix} M_l D_i M_{l+1}^{-1} = \begin{bmatrix} 0 & A \end{bmatrix} M_l D_i M_l^{-1} M_l M_{l+1}^{-1} =$$

$$\begin{bmatrix} 0 & A \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & B_i \end{bmatrix} M_l M_{l+1}^{-1} = \begin{bmatrix} 0 & A \cdot B_i \end{bmatrix} M_l M_{l+1}^{-1}$$

From equation (4), $AB_i = \hat{B}_i A$. Thus,

$$\begin{bmatrix} 0 & A \cdot B_i \end{bmatrix} M_l M_{l+1}^{-1} = \begin{bmatrix} 0 & \hat{B}_i \cdot A \end{bmatrix} M_l M_{l+1}^{-1} = \hat{B}_i \begin{bmatrix} 0 & A \end{bmatrix} M_l M_{l+1}^{-1} = \begin{bmatrix} 0 & \hat{B}_i \end{bmatrix}$$

The last equality follows from the fact that $\begin{bmatrix} 0 & A \end{bmatrix} M_l$ are the last $n-l-1$ rows of M_{l+1}, and thus $\left[\begin{bmatrix} 0 & A \end{bmatrix} M_l\right] \cdot M_{l+1}^{-1} = \begin{bmatrix} 0 & I \end{bmatrix}$. So, we have $M_{l+1} D_i M_{l+1}^{-1} = \begin{bmatrix} I & 0 \\ 0 & \hat{B}_i \end{bmatrix}$. ■

3.2 Decoupling the dimensions of SAREs

The above theorem can be used to decouple the dimensions of an SARE whenever the SARE can be converted to a system of quasi-uniform recurrence equations, and the group of the linear parts of its cycle dependence maps[3] has common left eigenvectors of an eigenvalue 1. As proved in [31], there is one such vector whenever the SARE has an affine schedule.

The following procedure finds all left eigenvectors of eigenvalue 1, common to all matrices D_i in a finite group G. We are given that $D_1, D_2, \ldots, D_k$ are the generators of G.

Procedure 1: Find vectors v such that $\forall D_i \in G,\ D_i^T v = v$.

Defining the following matrix:

$$A = \begin{bmatrix} D_1^T - I \\ D_2^T - I \\ \vdots \\ D_k^T - I \end{bmatrix}_{kn \times n} .$$

one can see that the required vectors form a basis of the null space of A. Steps $1-3$ below determine such a basis:

1. Perform 1) Gauss-Jordan reduction, 2) a possible row and column permutation[4], and 3) deletion of zero rows, on matrix A. The resulting matrix is of the form:

$$\begin{bmatrix} I_{m\times m} & E \end{bmatrix}_{m\times n} .$$

2. If $m = n$ (i.e., E is empty), then return (no such vector exists).

3. There are $n-m$ independent vectors, for example, the columns of the following matrix:

$$\begin{bmatrix} -E \\ I \end{bmatrix} .$$

[3] A *cycle dependence map* is a composition of dependences such that a variable depends on [a different index value of] itself.

[4] For notational simplicity, we assume that no column permutation is needed.

The procedure above requires an exact solution. If all the matrices in the group are integer, then the submatrix E is rational, since Gauss-Jordan reduction does not introduce irrational numbers. In this case, these vectors can be scaled to integer vectors, and the computer's exact integer arithmetic can be used. However, as A gets larger, the word size for elements of A also must increase. In case A is small, this presents no problem, as can be seen in the next example.

Example 1B

Consider the SARE in Ex. 1. As mentioned in Ex. 1A, the linear parts of the dependence maps, which generate a finite group G, include the following two matrices: $D_0 = I \;\; D_1 = \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix}$. The corresponding matrix $A = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$.
After Gauss-Jordan reduction and deleting zero rows we get: $\begin{bmatrix} I_{1\times 1} & E \end{bmatrix}_{1\times 2} = \begin{pmatrix} 1 & 0 \end{pmatrix}$.
Taking the column of $\begin{bmatrix} -E \\ I \end{bmatrix}$, gives $v^T = \begin{pmatrix} 0 & 1 \end{pmatrix}$.

Proc. 2 uses the above theorem and procedure to decouple the dimensions of an SARE (that can be converted to a system of quasi-uniform recurrence equations [30]).

Procedure 2: Decouple the dimensions in a SARE.

1. Perform a tree conversion on the SARE[5].

2. Invoke Proc. 1 to find all common left eigenvectors for an eigenvalue 1 of the linear parts of the direct dependence maps (suppose there are k of them).

3. Generate the group using the linear parts of the direct dependence maps as generators.

[5] A *tree conversion* is a procedure that applies affine transformations to the index sets of the arrays [8] and [30].

4. Compute the matrix M as described in the proof of Thm. 3.1.

5. Linearly transform all index sets by M, and update the dependence maps accordingly.

6. Reduce the dimension by k (delete the first k components of every index point, the first k components of d_{ij} (the translation part of direct dependence δ_{ij}), and the first k rows and first k columns of every D_{ij} (the linear part of direct dependence δ_{ij})).

The new dependence maps have dimension which is reduced by k. The conversion of an SARE of reduced dimension [30] is simpler to construct and simpler to apply.

Example 1C

Consider the SARE in Ex. 1. Steps (1-4) in Proc. 2 have been done in Exs. 1B and 1A. Step 5 now can be applied to the SARE, resulting in the following SARE.

$$
\begin{aligned}
&1 \le i \le n-1, \\
&\quad 2-i \le j (j+i \text{ even}) \le i-2, \quad a_3(i,j) = a_3(i-1, j-3) && (1) \\
&2 \le i \le n, \\
&\quad -1-i \le j (j+i \text{ odd}) \le i-3, \; a_2(i,j) = -a_3(i-1, j-2) a_2(i-1, -j-5) + a_2(i-1, j-1) && (2)
\end{aligned}
$$

The dependences in the last dimension are now: $a_2(j)$ depends on $a_3(j-2)$, on $a_2(-j-5)$ and on $a_2(j-1)$. $a_3(j)$ depends on $a_3(j-3)$. It is now easier to convert the SARE to an equivalent system of quasi-uniform recurrence equations [30] because there is one less dimension.

3.3 Generalizations

The above results can be generalized in two ways:

1. Let $V = \{\pi_1, \ldots, \pi_k\}$ be a set of linearly independent vectors such that for odd k, $\forall D_i \in G,\; \pi_k^T D_i = \pm \pi_k^T$, and for $1 \le m \le \lfloor k/2 \rfloor$ and $\forall D_i \in G$ either

   (a) $\pi_{2m-1}^T D_i = \pm \pi_{2m-1}^T$, and $\pi_{2m}^T D_i = \pm \pi_{2m}^T$, or

   (b) $(\pi_{2m-1} + j \cdot \pi_{2m})^T D_i = \pm j \cdot (\pi_{2m-1} + j \cdot \pi_{2m})^T$

In this case, there exists an $M \in \mathbf{Z}^{n\times n}$ such that the first k rows of M are $\pi'_1 \dots \pi'_k$, where $\pi'_i = c\pi_i$ for some constant c, and

$$MD_iM^{-1} = \begin{bmatrix} \square & & & & \\ & \ddots & 0 & & \\ & & \square & & \\ & 0 & & \pm 1 & \\ & & & & B_i \end{bmatrix} \tag{5}$$

$$\text{where } \square = \pm \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \text{ or } \pm \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \text{ or } \pm I.$$

The row above matrix B_i with ± 1 on the diagonal is present only in case k is odd.

The proof of this follows the same arguments as those in the proof of Thm. 3.1, with induction on m (i.e., two vectors are added at a time), and by using Maschke's theorem as in lemma 3.1.2, but for a 2×2 block.

2. Let $V = \{\pi_1, \dots, \pi_k\}$ be a set of linearly independent vectors such that for odd k, $\forall D_i \in G,\ \ \pi_k^T D_i = \pm\pi_k^T$, and for $1 \le m \le \lfloor k/2 \rfloor$ and $\forall D_i \in G$ either

   (a) $\pi_{2m-1}^T D_i = \pm\pi_{2m-1}^T$, and $\pi_{2m}^T D_i = \pm\pi_{2m}^T$, or

   (b) $(\pi_{2m-1} + j \cdot \pi_{2m})^T D_i = \lambda_{i,m}(\pi_{2m-1} + j \cdot \pi_{2m})^T$, where $\lambda_{i,m} \in \mathbf{C}$.

   In this case, there exists an $M \in \mathbf{R}^{n\times n}$ such that the first k rows of M are $\pi'_1 \dots \pi'_k$, where $\pi'_i = c\pi_i$ for some constant c, and MD_iM^{-1} has the form in Eq. 5 with the exception that $\square$ can be any 'real Jordan block'. The proof of this is based on the real Jordan form [10], using the same arguments as those in the proof of Thm. 3.1.

In the first case, we preserve the integer entries in D and d (the parts of the affine dependence map), and the lattice remains in $\mathbf{Z}^n$. In the second case, these entries are not necessarily integer, and the lattice is not necessarily in $\mathbf{Z}^n$. In this latter case, we thus have to make a linear transformation M^{-1} at the end of the conversion. Also, in the second case we no longer can work with integer arithmetic.

The above decoupling further simplifies the conversion process, but it is more difficult to find the desired eigenvectors, since we no longer require that they be for the same eigenvalue (i.e., Proc. 1 cannot be used).

References

[1] Peter R. Cappello. *VLSI Architectures for Digital Signal Processing.* PhD thesis, Princeton University, Princeton, NJ, Oct 1982.

[2] Peter R. Cappello and Kenneth Steiglitz. Unifying VLSI array design with geometric transformations. In H. J. Siegel and Leah Siegel, editors, *Proc. Int. Conf. on Parallel Processing*, pages 448–457, Bellaire, MI, Aug. 1983.

[3] Peter R. Cappello and Kenneth Steiglitz. *Advances in Computing Research*, volume 2: VLSI theory, chapter Unifying VLSI Array Design with Linear Transformations of Space-Time, pages 23–65. JAI Press, Inc., Greenwich, CT, 1984.

[4] Marina C. Chen and Carver Mead. Concurrent algorithms as space-time recursion equations. In S-Y Kung, H. J. Whitehouse, and Thomas Kailath, editors, *VLSI & Modern Signal Processing.* Prentice-Hall, Englewood Cliffs, 1985.

[5] C. Choffrut and K. Culik II. Folding of the plane and the design of systolic arrays. *Information Processing Letters*, 17:149–153, 1983.

[6] C. W. Curtis and I. Reiner. *Representation Theory of Finite Groups and Associative Algebras.* Interscience Publishers (John Wiley & Sons), New York, 1962.

[7] Jean-Marc Delosme and Ilse C. F. Ipsen. An illustration of a methodology for the construction of efficient systolic architectures in VLSI. In *Proc. 2nd Int. Symp. on VLSI Technology, Systems and Applications*, pages 268–273, Taipei, 1985.

[8] Jean-Marc Delosme and Ilse C. F. Ipsen. Systolic array synthesis: Computability and time cones. Technical Report Yale/DCS/RR-474, Yale, May 1986.

[9] José A. B. Fortes, King-Sun Fu, and Benjamin W. Wah. Systematic approaches to the design of algorithmically specified systolic arrays. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 300–303, Tampa, 1985.

[10] R. A. Horn and C. A. Johnson. *Matrix Analysis*. Cambridge Univ. Press, 1985.

[11] Richard M. Karp, Richard E. Miller, and Shmuel Winograd. Properties of a model for parallel computations: Determinacy, termination, queueing. *SIAM J. Appl. Math*, 14:1390–1411, 1966.

[12] Richard M. Karp, Richard E. Miller, and Shmuel Winograd. The organization of computations for uniform recurrence equations. *J. ACM*, 14:563–590, 1967.

[13] H.-T. Kung. Why systolic architectures. *Computer*, 15(1):37–45, Jan 1982.

[14] H.-T. Kung and Charles E. Leiserson. Algorithms for VLSI processor arrays. In *Introduction to VLSI Systems*, pages 271–292. Addison-Wesley Publishing Co, Menlo Park, CA, 1980.

[15] S-Y Kung. On supercomputing with systolic/wavefront array processors. *Proceedings of the IEEE*, July 1984.

[16] Charles E. Leiserson, Flavio M. Rose, and James B. Saxe. Optimizing synchronous circuitry by retiming. In *Proc. Third Caltech Conf. on VLSI*, Rockville, MD, 1983. Computer Science Press.

[17] Charles E. Leiserson and James B. Saxe. Optimizing synchronous systems. In *Proc. IEEE 22nd Annual Symp. Foundations of Computer Science*, Oct 1981.

[18] C. C. MacDuffee. *The Theory of Matrices*. Chelsea Publishing Co., New York, 1946.

[19] R. G. Melhem and W. C. Rheinboldt. A mathematical model for the verification of systolic networks. *SIAM J. Computing*, 13(3):541–565, Aug. 1984.

[20] Willard L. Miranker and Andrew Winkler. Spacetime representations of computational structures. *Computing*, 32:93–114, 1984.

[21] Dan I. Moldovan. On the analysis and synthesis of VLSI algorithms. *IEEE Trans. Comput.*, C-31:1121–1126, Nov. 1982.

[22] Dan I. Moldovan. On the design of algorithms for VLSI systolic arrays. *Proc. IEEE*, 71(1):113–120, Jan. 1983.

[23] Dan I. Moldovan and José A. B. Fortes. Partitioning and mapping algorithms into fixed systolic arrays. *IEEE Trans. on Computers*, C-35(1):1–12, Jan. 1986.

[24] M. Newman. Matrix representation of groups. In *Applied Mathematics Series - 60*. Institute for Basic Standards, National Bureau of Standards, Washington D.C. 20234, July 1968.

[25] Patrice Quinton. Automatic synthesis of systolic arrays from uniform recurrent equations. In *Proc. 11th Ann. Symp. on Computer Architecture*, pages 208–214, 1984.

[26] Patrice Quinton. *The Systematic Design of Systolic Arrays*, pages 229–260. Princeton University Press, 1987.

[27] Sailash K. Rao. *Regular Iterative Algorithms and Their Implementation on Processor Arrays*. PhD thesis, Stanford University, October 1985.

[28] Jeffrey D. Ullman. *Computational Aspects of VLSI*. Computer Science Press, Inc, Rockville, MD 20850, 1984.

[29] Yoav Yaacoby and Peter R. Cappello. Converting affine recurrence equations to quasi-uniform recurrence equations. **Submitted to** *SIAM J. Computing*, February 1988.

[30] Yoav Yaacoby and Peter R. Cappello. Converting affine recurrence equations to quasi-uniform recurrence equations. In John H. Reif, editor, *VLSI Algorithms and Architectures*, pages 319–328. Springer-Verlag, June 1988.

[31] Yoav Yaacoby and Peter R. Cappello. Scheduling a system of nonsingular affine recurrence equations onto a processor array. *J. VLSI Signal Processing*, 1(2):115–125, 1990.