International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:4, No:4, 2010

Some Characteristics of Systolic Arrays

Halil Snopce, Ilir Spahiu

***Abstract*—**In this paper is investigated a possible optimization of some linear algebra problems which can be solved by parallel processing using the special arrays called systolic arrays. In this paper are used some special types of transformations for the designing of these arrays. We show the characteristics of these arrays. The main focus is on discussing the advantages of these arrays in parallel computation of matrix product, with special approach to the designing of systolic array for matrix multiplication. Multiplication of large matrices requires a lot of computational time and its complexity is $O(n^3)$. There are developed many algorithms (both sequential and parallel) with the purpose of minimizing the time of calculations. Systolic arrays are good suited for this purpose. In this paper we show that using an appropriate transformation implicates in finding more optimal arrays for doing the calculations of this type.

***Keywords*—**Data dependences, matrix multiplication, systolic array, transformation matrix.

I. Introduction

Matrix multiplication plays a crucial role in many scientific disciplines. This multiplication can be thought of as the main tool for many other computations in different areas. Matrix multiplication in array of processors has been studied and a different arrays has been proposed [8, 10,12]. In this paper are used a special designs named systolic arrays which are suitable for matrix multiplication algorithm and offers both pipelineability and parallelism. Systolic approach and studying how to optimize these arrays is also studied extensively [1,3,4,5,9,10,11]. The main purpose of this paper is to discuss about characteristics of these arrays. It is done basically using different mathematical transformations for their construction. In addition there is done comparison of three different systolic arrays concluding about the optimality as well.

II. Designing Systolic Array for Matrix Multiplication Using Linear Transformation

Let A and B be two matrices of size NxN and we consider the problem of finding the resulting matrix C using the algorithm for matrix multiplication given below:

Algorithm 1:

for $i, j, k = 1$ *to* N

$a(i, j, k) = a(i, j-1, k)$

$b(i, j, k) = b(i-1, j, k)$

$c(i, j, k) = c(i, j, k-1) + a(i, j, k-1) \cdot b(i, j, k-1)$

end

where

$a(i,0,k) = a_{ik}$, $b(0, j, k) = b_{kj}$, $c(i, j,0) = 0$

Let $P_{ind} = \{(i, j, k)/1 \le i, j, k \le N\}$ be index space of used and computed data for matrix multiplication. Then we define the linear transformation matrix T given below:

$$T = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix} \tag{1}$$

Where $T_1 = [t_{11} \quad t_{12} \quad t_{13}]$ is the scheduling vector (in case of matrix multiplication is always $[1 \quad 1 \quad 1]$) and $S = \begin{bmatrix} T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix}$ is transformation which maps P_{ind} into 2-dimensional systolic array.

Data dependency matrix for algorithm 1 is given with:

$$D = \begin{bmatrix} \vec{e}^{\,3}_b & \vec{e}^{\,3}_a & \vec{e}^{\,3}_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The matrix T is associated with the so called projection direction $u = [u_1 \quad u_2 \quad u_3]^T$ (there are some possible allowable projection vectors, see [1]), so that the following conditions must satisfied:

$$1.\ \det T \ne 0 \tag{2}$$

$$2.\ T_2 u = 0 \quad \text{and} \quad T_3 u = 0 \tag{3}$$

$$3.\ \Delta_S = SD \in \{-1,0,1\} \tag{4}$$

The transformation matrix T maps the index point $(i, j, k) \in P_{ind}$ into the point $(t, x, y) \in T \cdot P_{ind}$ where:

$$t = T_1 [i \quad j \quad k]^T = i + j + k \tag{5}$$

$$[x \quad y]^T = S[i \quad j \quad k]^T \text{ For } (i,j,k) \in P_{ind} \qquad (6)$$

In this case t is time where calculations are performed, and (x, y) are the coordinates of processors elements on 2-dimensional systolic array.

Let us consider the case where $u = [1 \quad 1 \quad 1]^T$. From (3) there is:

$$T_2 u = 0 \Rightarrow t_{21} + t_{22} + t_{23} = 0 \qquad (7)$$

$$T_3 u = 0 \Rightarrow t_{31} + t_{32} + t_{33} = 0 \qquad (8)$$

Considering (1), (2), (4), (7) and (8), below are given all possible transformation matrices:

$$\begin{bmatrix} 1 & 1 & 1 \\ \pm1 & 0 & \mp1 \\ \mp1 & \pm1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 1 \\ \pm1 & \mp1 & 0 \\ \pm1 & 0 & \mp1 \end{bmatrix},$$

$$\begin{bmatrix} 1 & 1 & 1 \\ \mp1 & \pm1 & 0 \\ 0 & \mp1 & \pm1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 1 \\ 0 & \pm1 & \mp1 \\ \mp1 & \pm1 & 0 \end{bmatrix}$$

To implement the mapping $(i,j,k) \xrightarrow{T} (t,x,y)$, first there is defined a linear mapping $L = (L_1, L_2)$ such that $P_{ind} \xrightarrow{L} P^*_{ind} \xrightarrow{T} \bar{P}_{ind}$.

Let transformation matrix be:

$$T = \begin{bmatrix} T_1 \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \qquad (9)$$

If there is taken the matrix $L = (L_1, L_2)$ given with:

$$L_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \quad L_2 = \begin{bmatrix} 0 \\ -1 \\ -1 \end{bmatrix}$$

Then the elements $(u, v, w) \in P^*_{ind}$ are obtained from:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = L_1 \begin{bmatrix} i \\ j \\ k \end{bmatrix} + L_2 = \begin{bmatrix} i \\ i+j-1 \\ i+k-1 \end{bmatrix} \qquad (10)$$

From (6) for the new vector (u, v, w) the position of PEs can be found:

$$[x \quad y]^T = S \cdot \begin{bmatrix} i \\ i+j-1 \\ i+k-1 \end{bmatrix} = \begin{bmatrix} 1-j \\ 1-k \end{bmatrix} \qquad (11)$$

The new initial space is obtained:

$$\hat{P}_{in}(a) = \{(i, 0, i+k-1) / 1 \le i,k \le n\}$$
$$\hat{P}_{in}(b) = \{(0, i+j-1, i+k-1) / 1 \le i,j,k \le n\} \qquad (12)$$
$$\hat{P}_{in}(c) = \{(i, i+j-1, 0) / 1 \le i,j \le n\}$$

If for the new position of the vector γ, $\gamma \in \{a,b,c\}$ is taken $p^*_\gamma = p_\gamma - (i+j+k-2)e^3_\gamma$ then:

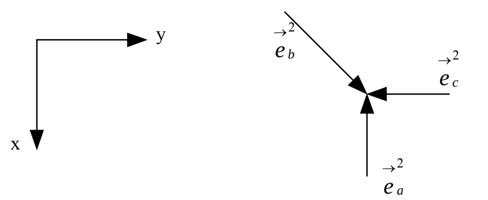
$$P^*_{in}(a) = [\, i,\ 3-2i-k,\ i+k-1]^T$$
$$P^*_{in}(b) = [4-2i-j-k,\ i+j-1,\ i+k-1]^T$$
$$P^*_{in}(c) = [\, i,\ i+j-1,\ 3-2i-j]^T$$

Finally the positions of input data and communication links in the array can be found:

$$\begin{bmatrix} x \\ y \end{bmatrix}_a = S \cdot P^*_{in}(a) = \begin{bmatrix} 3i+k-3 \\ 1-k \end{bmatrix},$$
$$\begin{bmatrix} x \\ y \end{bmatrix}_b = \begin{bmatrix} 5-3i-2j-k \\ 5-3i-j-2k \end{bmatrix}, \begin{bmatrix} x \\ y \end{bmatrix}_c = \begin{bmatrix} 1-j \\ 3i+j-3 \end{bmatrix} \qquad (13)$$

$$\Delta_S = S \cdot D = \begin{bmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} \vec{e}^{\,2}_b & \vec{e}^{\,2}_a & \vec{e}^{\,2}_c \end{bmatrix}$$

For the coordinate system and for the data flow given with:



There is obtained that b is flowing diagonally down, a is flowing up and c to the left. The corresponding hexagonal array for N=4 is given in the figure 1:

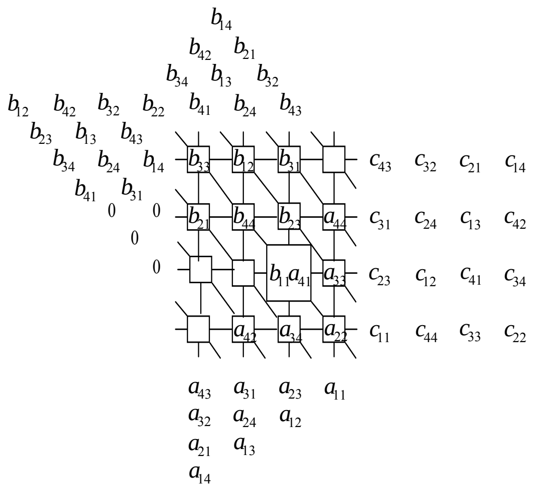


Fig. 1 Systolic array for N=4 using the mapping L

III. Standard Hexagonal Systolic Array

If there is taken the transformation matrix given with:

$$T_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix} \quad (14)$$

Then the obtained results are:

$$\begin{bmatrix} x & y \end{bmatrix}^T = S\begin{bmatrix} i & j & k \end{bmatrix}^T = \begin{bmatrix} i-k \\ j-k \end{bmatrix} \quad (15)$$

$$p_a^* = \begin{bmatrix} i \\ 0 \\ k \end{bmatrix} - (i+0+k-2)\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} i \\ 2-i-k \\ k \end{bmatrix}$$

$$p_b^* = \begin{bmatrix} 0 \\ j \\ k \end{bmatrix} - (0+j+k-2)\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2-j-k \\ j \\ k \end{bmatrix}$$

$$p_c^* = \begin{bmatrix} i \\ j \\ 0 \end{bmatrix} - (i+j+0-2)\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} i \\ j \\ 2-i-j \end{bmatrix}$$

The positions of input data in the array are given with:

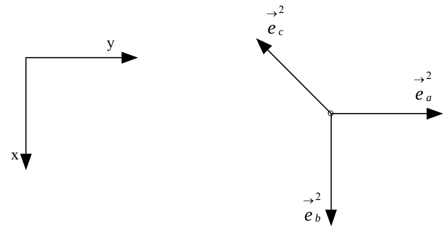
$$\begin{bmatrix} x \\ y \end{bmatrix}_a = S \cdot p_a^* = \begin{bmatrix} i-k \\ 2-i-2k \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix}_b = \begin{bmatrix} 2-j-2k \\ j-k \end{bmatrix}, \quad \begin{bmatrix} x \\ y \end{bmatrix}_c = \begin{bmatrix} 2i+j-2 \\ 2j+i-2 \end{bmatrix} \quad (16)$$

Communication links are given with:

$$\Delta_S = S \cdot D = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix} = \begin{bmatrix} \vec{e}_b^{\,2} & \vec{e}_a^{\,2} & \vec{e}_c^{\,2} \end{bmatrix} \quad (17)$$

For the coordinate system and for the corresponding data flow which is like below:



The conclusion is that b is flowing down, a to the right and c diagonally up. In the figure 2 this array is given for N=4. (This array is called standard hexagonal systolic array-SHSA, and first was proposed by Kung and Leiserson [10]).

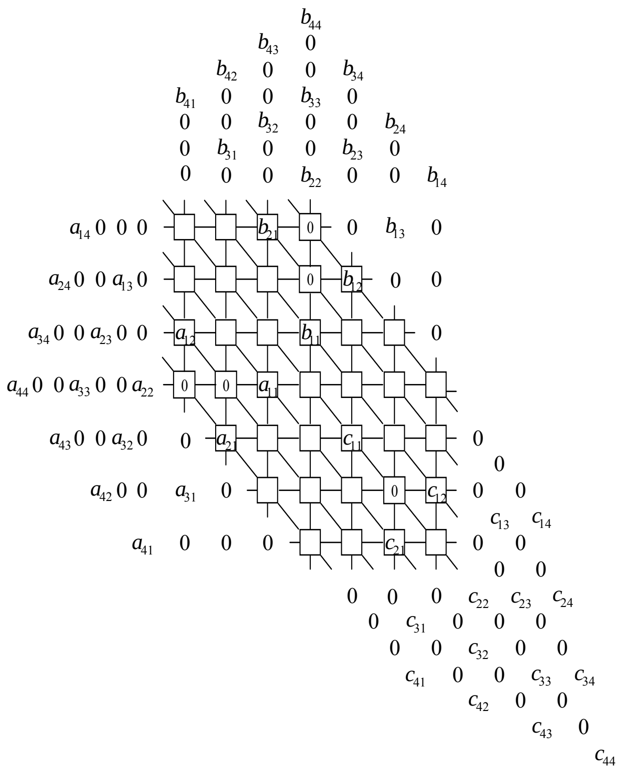


Fig. 2 the SHSA array for N=4

IV. Systolic Array for Matrix Multiplication Using Nonlinear Transformation

The transformation matrix which is used in this case is:

$$T_\lambda = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

To solve the problem of unwanted delays in processing and in the output of the results, we define the time scheduling compression function $t(p) = u + v + w + \lambda$ where λ is determined by the condition $t(p_{\min}) = 0$. The reposition of the elements is given with the following equation:

$$p^*(u,v,w) = p(u,v,w) - (t(p)+1)re^3 \tag{18}$$

In this case is added 1 in order to take non-negative values for the new produced time steps. e^3 ensures the appropriate direction of each element. $r \in \{1,-1\}$ and it takes the value -1 in the region of applying the nonlinear transformation when, with linear transformation the elements are placed in negative positions. The concept of two processing streams defined in [2] is used. Top stream (ts) and bottom stream (bt). For each of these streams there is used different transformation matrix. So, the new scheduling compression function is:

$$\begin{cases} t^{ts}(p) = u + v + w + \lambda^{ts}; & for\ ts \\ t^{bs}(p) = -(u + v + w) + \lambda^{bs}; & for\ bs \end{cases} \tag{19}$$

Where λ is determined by the condition:

$$\begin{cases} t^{ts}(p^{ts}_{\min}) = 0;\ for\ ts \\ t^{bs}(p^{bs}_{\min}) = j;\ for\ bs \end{cases} \text{where} \begin{cases} p^{ts}_{\min} = p(1,j,1);\ for\ ts \\ p^{bs}_{\min} = p(n,j,n);\ for\ bs \end{cases} \tag{20}$$

In equations given above is used the parameter j in place of 1, because transforming the equation of matrix multiplication $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$, such that the computations are represent into e three dimensional space, then j different computational planes will be created.

From (19): $t^{ts}(p) = t^{ts}(1,j,1) = 1 + j + 1 + \lambda^{ts}$ and from (20) $\lambda^{ts} = -2 - j;\ j = 1,2,...n$. Thus $\lambda^{ts} = -3$. In the same manner: $t^{bs}(p) = t^{bs}(n,j,n) = -(n + j + n) + \lambda^{bs}$, therefore $-2n - j + \lambda^{bs} = j \Rightarrow \lambda^{bs} = 2n + 2j$.

So, the final results for the constant λ are:

$$\begin{cases} \lambda^{ts} = -3; & for\ ts \\ \lambda^{bs} = 2n + 2j; & for\ bs \end{cases} \tag{21}$$

Finally applying the nonlinear transformation defined with:

$$T^{nl} = \begin{cases} T_\lambda xa;\ for\ ts \\ T_\alpha x T_\lambda xa;\ for\ bs \end{cases} \tag{22}$$

Where $T_\alpha = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ and it is symmetric matrix defined in [6], and $a = (i,j,k)^T$.

Now the position of input data can be determined:

$$p(i,j,k) = \xrightarrow{T^{nl}\ for\ ts} T_\lambda xa = \begin{bmatrix} -j \\ i-k \end{bmatrix} \text{ and}$$

$$p(i,j,k) = \xrightarrow{T^{nl}\ for\ bs} T_\alpha x T_\lambda xa = \begin{bmatrix} -j \\ -i+k \end{bmatrix}$$

The positions of systolic cells for top stream will be:

$$p^*_a(u,v,w) = p_a(u,v,w) - (t^{ts}(p)+1)\cdot(0,1,0)^T =$$
$$= (i,j,k)^T - (i + j + k - 2)\cdot(0,1,0)^T$$

$$p^*_a(u,v,w) = \begin{bmatrix} i \\ 2-i-k \\ k \end{bmatrix} \xrightarrow{T^{nl}\ for\ ts} T_\lambda xa^* =$$

$$= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} i \\ 2-i-k \\ k \end{bmatrix} = \begin{bmatrix} k+i-2 \\ i-k \end{bmatrix}$$

In similar manner:

$$p^*_b(u,v,w) = \begin{bmatrix} -j \\ 2-j-2k \end{bmatrix};\ p^*_c(u,v,w) = \begin{bmatrix} -j \\ 2i+j-2 \end{bmatrix}$$

The positions of systolic cells for bottom stream are:

$$p^*_a(u,v,w) = p_a(u,v,w) - (t^{bs}(p)+1)\cdot(0,1,0)^T =$$
$$= (i,j,k)^T - (2n + j - i - k + 1)\cdot(0,1,0)^T$$

$$p^*_a(u,v,w) = \begin{bmatrix} i \\ i+k-2n-1 \\ k \end{bmatrix} \xrightarrow{T^{nl}\ for\ bs} T_\alpha x T_\lambda xa^* =$$

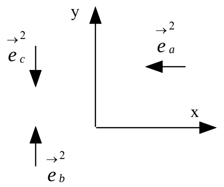
$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} i \\ i+k-2n-1 \\ k \end{bmatrix} = \begin{bmatrix} 2n+1-i-k \\ k-i \end{bmatrix}$$

And similarly:

$$p^*_b(u,v,w) = \begin{bmatrix} -j \\ 2k-2n-j-1 \end{bmatrix} \text{ and}$$

$$p^*_c(u,v,w) = \begin{bmatrix} -j \\ -2i+2n+j+1 \end{bmatrix}$$

For the coordinate system and for the corresponding data flow which is like below:



The corresponding systolic array is:

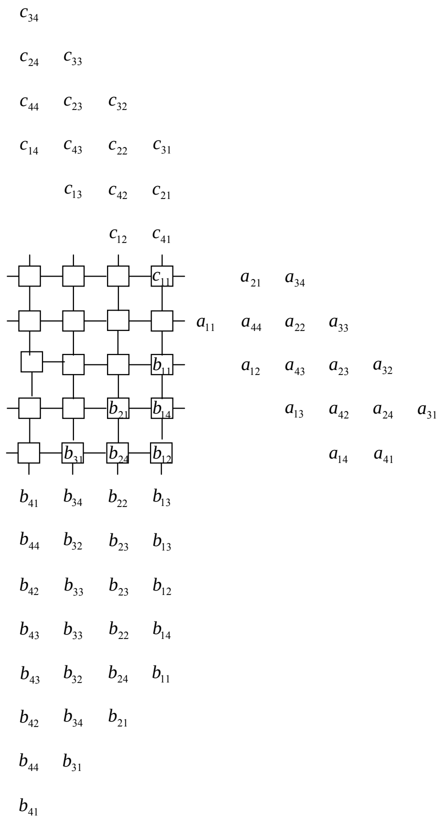


Fig. 3 the systolic for N=4 (with using the nonlinear mapping)

V. Comparison of Three Different Designing Method

Total running time is calculated using the formula $T_{tot} = T_{in} + T_{exe} + T_{out}$ (time of putting the input data, time of execution and time of obtaining the output results). In the case of SHSA array there can be calculated that $T_{tot} = 5n - 4$. The number of PEs for the same array is given by the formula $\Omega = 3n^2 - 3n + 1$. The speed up is calculated by the formula $S = \frac{T_S}{T_{tot}} = \frac{n^3}{5n-4}$, where $T_S = n^3$ is running time in sequential case. Efficiency is given by the formula $E = \frac{S}{\Omega} = \frac{n^3}{(5n-4)(3n^2-3n+1)}$.

These parameters in the case of array obtained by the using of linear transformation L are given below:

$$T_{tot} = 3n - 2;\ \Omega = n^2;\ \ S = \frac{n^3}{3n-2};\ E = \frac{n}{3n-2}$$

Obtained result after the calculation of these parameters for the systolic array using nonlinear transformation are:

$$T_{tot} = 4n - 2;\ \Omega = n \cdot \left\lfloor \frac{3n-1}{2} \right\rfloor;\ \ S = \frac{n^3}{4n-2}$$

And $E = \frac{n^2}{(4n-2)\cdot\left\lfloor \frac{3n-1}{2} \right\rfloor}$.

For the case of N=4 the discussion is already done. Analyzing two other cases for N=5 and N=10 (there can be taken each other integer value for n) the table of obtained results is given below:

TABLE I
Comparison of Three Different Parameters

	n=4			n=5			n=10		
	Ω	T	E	Ω	T	E	Ω	T	E
Using lin. Tr. L	16	10	40	25	13	38	100	28	35
Using nonlin. Tr	20	14	23	35	18	20	140	38	19
SHSA array	37	16	18	61	21	9.8	271	46	8

VI. Conclusion

In this paper are given models of three different systolic arrays. We emphasized the characteristics of each one giving detailed explanation of their construction. From the table 1 it is clear that in each case, the systolic array obtained using transformation matrix L is more optimal comparing with two others. An interesting conclusion is that even the array obtained by the use of nonlinear transformation offers better results comparing with the SHSA array where isn't used any type of transformation.

References

[1] M.P. Bekakos, Highly Parallel Computations-Algorithms and Applications, Democritus University of Thrace, Greece, 2001.
[2] Efremides, O.B., and Bekakos, M.P., A nonlinear Approach to Design Processor Time Optimal Systolic Arrays for matrix-vector multiplication, HERCMA '98, athenc, Greece, pp. 327-336, 1998
[3] Esonu, M.O., Al-Khalili, A.J., Hariri, S. and Al-Khalili, D., Systolic Arrays: How to choose them, pp. 179-188, 1992

[4] Milentijevic, I.Z., Milovanovic, I.Z., E.I. and Stojcev, M.K., The Design of Optimal Planar Systolic Arrays for Matrix Multiplication, Comput. Math. Appl., pp. 17-35, 1997
[5] Bekakos, M.P., Milovanovic, E.I., Milovanovic, I.Z. and Milentijevic, I.Z., An Efficient Systolic Array for Matrix Multiplication, Proc. of the Fourth Hellenic European Conference on Computer Mathematics and its Applications (HERCMA '98), Athens '98, pp. 298-317, 1999
[6] Gusev, M., and Evans, D.J., A new matrix vector Product Systolic Array, Parallel Algorithms and Aplications, 22, 346-349, 1994
[7] C.N. Zhang, J.H. Weston, Y. F. Yan: Determining object functions in systolic array designs. IEEE Trans. VLSI Systems 2, No. 3 (1994), 357-360
[8] Jagadish, H. V., and Kailath, T., A family of new efficient arrays for matrix multiplication. IEEE Trans. On Computers, 38(1), pp. 149-155, 1989
[9] Snopce, H., Elmazi, L., Reducing the number of processors elements in systolic arrays for matrix multiplication using linear transformation matrix, Int. J. of Computers, Communications and Control, Vol. III (2008), Suppl. issue: Proceedings of ICCCC 2008, pp. 486-490
[10] Kung, H.T. and Leiserson, C.E., Systolic arrays for (VLSI), Introduction to VLSI Systems, Addison-Wesley Ltd., Reading, MA, 1980.
[11] Yun YANG, Shinji KIMURA, The optimal architecture design of two-dimension matrix multiplication jumping systolic array, IEICE Transactions on Fundamentals of Electronics, Communications and computer sciences, Volume E91-A, pp. 1101-1111, 2008
[12] A.K., Oudjida, S. Titri, M. Hamarlain, Latency 2I/O-Bandwidth 2D-array matrix multiplication algorithm, The int. Journal for computation and mathematics in electrical engineering, volume 21, pp. 377-392, 2002.

Authors:_Halil Snopce[1], Ilir Spahiu[2]

[1,3] South East European University CST Faculty, Department of computer science, 1200 Ilindenska bb, Tetovo,Republic of Macedonia
[2] Faculty of Pedagogy "ss Clement of Ohrid", "Ss. Cyril and Methodius" University in Skopje, Republic of Macedonia

E-mail: [1]h.snopce@seeu.edu.mk, [2]i.spahiu.@seeu.edu.mk