Time objective functions for systolic arrays

Dragan M. Randjelović,Ratomirka M. Randjelović
Agricultural faculty
University of Priština
38219 Lešak, Kosovo and Metohia
SERBIA

Abstract: - Author discuss a problem of determining parameters suitable systolic arrays for implementation regular 3-nested loop algorithms. It is showed that on that occasion it has to use the characteristics of so called adaptable algorithms to the projection direction, if it has they. If this characteristics are not use, the difference in obtained results could be very significant. This characteristics can be space and time. The analyse a time objective functions for systolic arrays is very complex problem because the change of some space or other time objective function make influence on time objective functions. Therefore time objective functions are object for considering in this paper. Also obtained results in this paper are illustrated on the example of matrix multiplication.

Key Words: Systolic array, Adaptable algorithm, Time objective functions, Matrix multiplication

1 Introduction

For given regular p-nested loop algorithm exists many 2D systolic arrays (SA-s) suitable for its implementation. They could be different in significant measure in parameters, no matter if they have or not, some topological structure. To chose best suitable before his design and synthesis, it is good that we know in advance their characteristics.

In literature (see for example [4] and [17]), is defined big number of space-time characteristics, i.e. objective functions, and given procedure for theirs determining. The subject of interesting in this paper are some of this time characteristics and that: summary time necessary for realization of given algorithm-T, flow period time between two consecutively calculation some algorithm-t_p.

We must define and understend and some basis space's characteristics like :

number of processing elements PE-s in SA–Ω_p,area of SA-g_a, and also

space-time characteristics like AT measure.

Importance of this characteristics depends of boundary nested loop algorithm's and transformation matrix with whose help is SA synthesized. We can not to exert influence on boundary of nested loop algorithm but it is possible significant to exert influence on transformation matrix. This privilege gives the fact that one of projection directions corresponds to set of suitable transformation matrix's. Therefore, it is important to intensify criterion's for his determining so that the transformation matrix's which synthesize SA with bed characteristics to be automatic exclude.

2 Systolic arrays characteristics

Each regular 3-nested loop algorithm can be characterized by a pair (D,Pint), where

(1) $D = \lfloor \vec{e}_1 \quad \vec{e}_2 \quad \vec{e}_3 \rfloor$

is a dependency matrix, $P_{int} = \{(i,j,k) \mid 1 \le i \le N_1,\ 1 \le j \le N_2,\ 1 \le k \le N_3\}$ is index set where data are used or computed, and N_1,N_2 and N_3 are index boundaries.

The SA implementation can be obtained by a linear transformation

(2) $$T = \begin{bmatrix} \vec{\pi} \\ S \end{bmatrix} = \begin{bmatrix} \vec{\pi} \\ \vec{S}_1 \\ \vec{S}_2 \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix}$$

where first component of T determines time schedule and second - S is the space mapping function determining PE-s locations and the communication channels between them. Vector

(3) $$\pm \vec{\pi} = \left(\left(\vec{e}_2^{\,3}\right)^T - \left(\vec{e}_1^{\,3}\right)^T \right) \times \left(\left(\vec{e}_3^{\,3}\right)^T - \left(\vec{e}_1^{\,3}\right)^T \right)$$

is obtained from equation (3),for example see [3]. The sign + or - is determined from condition

$$\boldsymbol{\vec{\pi} \cdot \vec{e}_2^{\,3} \rangle\, 0\, (\langle\, 0)\ \forall\ i = 1,2,3.}$$

Matrix S which maps P_{int} into 2D SA is determined from the following conditions :

- Matrix T must be nonsingular, i.e.

(4) $\det T \neq 0$

This ensures a conflict free mapping.

- The corresponding projection direction is orthogonal to the projection plane, i.e.

(5) $\vec{S}_1 \cdot \vec{\mu} = 0$ and $\vec{S}_2 \cdot \vec{\mu} = 0$.

- The connection between the PE's in the 2D SA must be of near-neighbor type, and crossing is not allowed. This requirement means that elements of matrix $\Delta S = S \cdot D$ have to be from the set {-1,0,1}, i.e.

(6) $t_{ij} \in \{-1,0,1\},\ 2 \le i \le 3,\ 1 \le j \le 3$.

- Two arbitrary position vectors from P_{int} must not satisfy both

(7) $\vec{\pi} \cdot \vec{P}_1 = \vec{\pi} \cdot \vec{P}_2$ and $S \cdot \vec{P}_1 = S \cdot \vec{P}_2$,

simultaneously.

It is remarked that noticed condition for given projection direction

$$\vec{\mu} = \begin{bmatrix} \mu_1 & \mu_2 & \mu_3 \end{bmatrix}^T$$

not determine uniform matrix S as well as matrix T. We introduce for reduction the set of possible matrix S, and so abandon those for which would obtain worst space characteristics of synthesized SA, follow :

- We exchange condition (5) with stronger

(8) $\vec{\mu} = \vec{S}_1 \times \vec{S}_2$.

- In the case of planar 2D SA synthesis, i.e. for case

$\mu_i \in \{-1,0,1\},\ i = 1,2,3,$

we impose to elements matrix S one of follow alternate conditions.

If μ_1=1 elements matrix's S must satisfy equality

(9) $t_{22} \cdot t_{32} + t_{23} \cdot t_{33} = 0$,

and if μ_2=1,

(10) $t_{21} \cdot t_{31} + t_{23} \cdot t_{33} = 0$.

Conditions (9) and (10) have not to demand simultaneous.

So, for example

$$\vec{\mu} = \begin{bmatrix} 1 & \pm 1 & \mu_3 \end{bmatrix}^T$$

must to determine or (9) or (10).

Now, by standard projection procedure, see for example the papers [1]-[8] and [10], after selection the valid transformation T for given projection direction, synthesis of corresponding SA is on the base of mapping

(11) $T : (D, P_{int}) \rightarrow (\Delta, P_{int}^*)$.

We give now the procedures for determining basic characteristics, conform to mainly results from papers [2]-[4].In spite of all that it is considered 2D SA suitable for regular 3-nested loop algorithms implementation, with loop top borders N_1,N_2 and N_3.

2.1 Space characteristics

Number of PE-s in SA(notation Ω_p)is given with:

Theorem 1.

$$(12)\ \Omega_p = \begin{cases} N_1 N_2 N_3 & \text{if } a_i \rangle N_i \text{ for some } 1 \le i \le 3 \\ N_1 N_2 N_3 - (N_1 - a_1)(N_2 - a_2)(N_3 - a_3) \end{cases}$$

$$(13) \qquad a_i = \left| \frac{T_{1i}}{\gcd(T_{11} T_{12} T13)} \right|$$

T_{1i} is the (1,i) - cofactor of matrix T, $1 \le i \le 3$.

With gcd(T_{11},T_{12},T_{13}) is noticed the largest common divisor of numbers T_{11},T_{12} and T_{13}. For determining array of 2D SA, in notation g_a, is used following result :

Theorem 2.

(14)…g_a = $(N_1\text{-}1)(N_2\text{-}1)|T_{13}|+(N_1\text{-}1)(N_3\text{-}1)|T_{12}|$ + $+(N_2\text{-}1)(N_3\text{-}1)|T_{11}|$.

2.2 Time characteristics

From time's objective functions, see [4], we consider summary time is the time necessary for realization of given algorithm on synthesized SA and is calculated as sum of time for date input in SA- T_{in}, , time for algorithm executing - T_{exe}, and time necessary for dates leaving SA - T_{out}, i.e. $T = T_{in}+T_{exe}+T_{out}$,

Remark 1. T_{exe} is calculated from next equality:

$$\mathbf{T_{exe}} = \mathbf{1} + \max_{(t,x,y) \in P^*_{int}} t - \min_{(t,x,y) \in P^*_{int}} t$$

Theorem 3.

$$(15) \quad T_{exe} = 1 + \sum_{j=1}^{3} (N_j - 1) \cdot \left| t_{ij} \right|.$$

Also, t_p, called flow period of processor can be calculated on the basis of next equation:

Theorem 4.

$$(16) \qquad t_p = \left| \frac{\det(T)}{\gcd(T_{11}, T_{12}, T_{13})} \right|.$$

2.3 Time-space characteristics

AT-measure is defined with
$AT = \Omega_p T$ or $AT^2 = \Omega_p T^2$.
Efficiency E_p is defined with
$E_p = T_1/T\Omega_p$
where T_1 is time realization of given algorithm on one PE.
This characteristics is necessary like complement analysas while often the decreasing of space parameters (i.e. the number of PE-s) to improve SA sharacteristics causes the increasing of time parameters(i.e. the number of tacts for necessary calculations).

3 Main result

Let α be a regular 3-nested loop algorithm with index space
$P_{int} = \{(i,j,k) \mid 1 \le i \le N_1,\ 1 \le j \le N_2,\ 1 \le k \le N_3\}$. We introduce the
following subclasses of α..
Definition 1. If the ordering of computations in algorithm α, for some
fixed j, may be performed over arbitrary permutations of index
variables i and k, we say that α is α (i,k) adaptable.
Definition 2. If the ordering of computations in algorithm α, for some
fixed i, can be performed over arbitrary permutations of index
variables j and k, we say that α is α (j,k) adaptable.
Remark 2. If a given algorithm α satisfies both the Definition 1. and 2., we say that α is adaptable.
If the given algorithm is from some of defined classes, its adaptation
to the given projection direction working with linear mapping H=(F,G),
where F is 3x3 matrix whose elements are in function of elements of
vector μ and G is 3x1 vector with constant elements which provide
that after adaptation mapping :
(17)………$H : P_{int} \rightarrow P_{int}^{\wedge}$
space $P_{int}^{\wedge}$ again is in first octant coordinate system.
Now we are defining the mapping H.
Definition 3. Suppose that a given algorithm is of type α(j,k).
If

$$\vec{\mu} = \begin{bmatrix} 1 & \mu_2 & \mu_3 \end{bmatrix}^T$$

is allowable projection direction the mapping H=(F,G) is defined by

$$(18)\qquad F = \begin{bmatrix} 1 & 0 & 0 \\ \mu_2 & 1 & 0 \\ \mu_3 & 0 & 1 \end{bmatrix}, G = \begin{bmatrix} 0 \\ g_2 \\ g_3 \end{bmatrix}$$

where g_2 and g_3 are smallest integers determined such that for each $[i,j,k]^T$
from P_{int} the following equation's are valid
$v = \mu_2 i + g_2 + j > 0$ and $w = \mu_3 i + k + g_3 > 0$.
The elements v and w are defined according to

$$(19)\qquad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = F \cdot \begin{bmatrix} i \\ j \\ k \end{bmatrix} + G = \begin{bmatrix} i \\ \mu_2 i + j + g_2 \\ \mu_3 i + k + g_3 \end{bmatrix}.$$

Definition 4. Suppose that a given algorithm is of type α(i,k).
If allowable projection direction is

$$\vec{\mu} = \begin{bmatrix} \mu_1 & \pm 1 & \mu_3 \end{bmatrix}^T$$

than mapping H=(F,G) is defined by

$$(20)\qquad F = \begin{bmatrix} 1 & \pm\mu_1 & 0 \\ 0 & 1 & 0 \\ 0 & \pm\mu_3 & 1 \end{bmatrix}, G = \begin{bmatrix} g_1 \\ 0 \\ g_3 \end{bmatrix}$$

where g_1 and g_3 are smallest integers determined such that
for each [i,j,k]T from P_{int} the following equation's are valid :
$u = i \pm \mu_1 j + g_1 > 0$ and $w = k \pm \mu_3 j + g_3 > 0$.
The elements u and w are obtained according to

$$(21)\qquad \begin{bmatrix} u \\ v \\ w \end{bmatrix} = F \cdot \begin{bmatrix} i \\ j \\ k \end{bmatrix} + G = \begin{bmatrix} i \pm \mu_1 j + g_1 \\ j \\ k \pm \mu_3 j + g_3 \end{bmatrix}.$$

Now, instead mapping given with (18), we are making
suitable 2D SA with two mapping :
(22)H: $P_{int} \rightarrow P_{int}^{\wedge}$ and T: $(D, P_{int}^{\wedge}) \rightarrow (\Delta, P_{int}^{*})$
where elements of space $P_{int}^{\wedge}$ are defining with (19) or (21), in dependence of projection direction.
Let as now to determine
time characteristics these synthesized SA.
Theorem 5. Suppose that a given algorithm α is α(j,k) adaptable, the index space
$P_{int} = \{(i,j,k) \mid 1 \le i \le N_1,\ 1 \le j \le N_2,\ 1 \le k \le N_3$ and
the projection direction

$$\vec{\mu} = \begin{bmatrix} 1 & \mu_2 & \mu_3 \end{bmatrix}^T, \text{ then is:}$$

$$(23)\qquad t_p = \left| \vec{\pi} \cdot \vec{\mu} \right|.$$

Proof.
According (2) and the form of matrix H, (18), we can see that the valid transformation is

$$\textbf{(24)}\ \mathbf{M} = \mathbf{T}\cdot\mathbf{F} = \begin{bmatrix} \vec{\pi}\cdot\vec{\mu} & t_{12} & t_{13} \\ 0 & t_{22} & t_{23} \\ 0 & t_{32} & t_{33} \end{bmatrix}$$

and from other side from (2) and (18) we have
(25)$t=(t_{11}+\mu_2 t_{12}+\mu_3 t_{13}) + t_{12}j + t_{13}k + g_2 t_{12} + g_3 t_{13}$.
According remark 1. we have
$T_{exe}= 1 + | t_{11}+\mu_2 t_{12} +\mu_3 t_{13} |(N_1-1) + |t_{12}|(N_2-1) +$
$+ |t_{13}|(N_3-1)$ i.e.
$T_{exe}= 1 + | \vec{\pi}\cdot\vec{\mu} |(N_1-1) + |t_{12}|(N_2-1) + |t_{13}|(N_3-1)$.
Using (15) we have

$$\mathbf{t}_{\mathbf{p}}= |\vec{\pi}\cdot\vec{\mu}|.$$

By the some procedure we prove case

$$\vec{\mu} = [\mu_1 \quad \pm 1 \quad \mu_3]^T$$

Consequence 1.We have two different cases:

$$\mathbf{t}_{\mathbf{p}}= |\vec{\pi}\cdot\vec{\mu}| = 1,$$

when we have processing elements full engagement in algoritham realization of sinthetized SA and this time of realization T_{exe} is equal with the time in one space unoptomized SA (so we have already a big contribution in time-space minimization sinthetized SA, suitable for realization this considered algorithm) ;

$$\mathbf{t}_{\mathbf{p}}= |\vec{\pi}\cdot\vec{\mu}| > 1,$$

when we have t_p-1 empty steps in two successive calculations in one processing element of sinthetized SA and this is no the deficiency than advantage.
Let $1<t_p\le 5$ and let FP={0,1,2,…, t_p -1} is the set of moving factors of the index variable i. We mark

$$\textbf{(25)}\ \cancel{N}_1 = \begin{matrix} N_1, N_1 \neq t_p\cdot m \\ N_1+1, N_1 = t_p\cdot m \end{matrix}, \textbf{for } t_p \in \{2,3,5\}.$$

Each index variable i we can give uniform factor r_1, like the biggest element from set FP so that satisfy

$$\textbf{(26)}\ -t_p(i-1) + r_1 N_1 < 0, i = 1 \Rightarrow r_1 = 0.$$

Also, in this way each index variable j we can give uniform factor r_2, like the biggest element from set FP so that satisfy

$$\textbf{(27)}\ -t_p(i-1) - (j-1) + (r + r_2)N_1 \le 0.$$

On the basis (26) i (27) we give uniform each pair of index variables (i,j) the pair of moving factors (r_1,r_2). Now we move each input variable in the space of initial calculations or in the plane of SA for size $(r_1+r_2)\ \cancel{N}_1$ according to corresponding vector $\vec{e}_{\gamma}^{\,3}, \gamma \in \{1,2,3\}$.
So we havn't pauses between two successive calculations in each PE-s synthesized SA.
Advantage of results obtained according to Theorem 5. and suitable consequence, in relationship with knowen results from literature, we can see on two rectangular matrix multiplication, $A=(a_{ik})$ order $N_1 \times N_3$ and $B=(b_{kj})$ order $N_3 \times N_2$. We determine, free choice, for projection directions:

$$\vec{\mu} = [1 \quad 1 \quad 0]^T, \vec{\mu} = [1 \quad 1 \quad 1]^T, \vec{\mu} = [2 \quad 1 \quad 1]^T.$$

These SA are meeting very often in literature (see for example [1],[4],[17]) which is suitable for comparing with results from this paper. By comparing, we consider case N1≤N2 and the case $N_1=N_2=N_3=N$ will be given in parenthesis.
Synthesized array with standard procedure have characteristics noticed with WE and using procedure from this paper, given with (22), noticed with NR.
For direction

$$\vec{\mu} = [1 \quad 1 \quad 0]^T,$$

we obtain orthogonal 2D SA.

$$\boldsymbol{WE}:$$

$$\Omega_p = N_3\cdot(N_1+N_2-1), (\Omega_p = N\cdot(2N-1)),$$

$$g_a = (N_3-1)\cdot(N_1+N_2-1), (g_a = 2(N-1)^2),$$

$$t_p = 2,$$

$$T_{in} = N_2 - 1, (T_{in} = N-1),$$

$$T_{exe} = N_1 + N_2 + N_3 - 2, (T_{exe} = 3N-2),$$

$$T_{out} = N_2 - 1, (T_{out} = N-1).$$

$$\boldsymbol{NR}:$$

$$\Omega_p = N_3\cdot N_1, (\Omega_p = N^2),$$

$$g_a = (N_1-1)\cdot(N_3-1), (g_a = (N-1)^2),$$

$$t_p = 1,$$

$$T_{in} = N_1 - 1, (T_{in} = N-1),$$

$$T_{exe} = \cancel{N}_2 + N_3 - 1, (T_{exe} = 2N-1),$$

$$T_{out} = 0.$$

For direction

$$\vec{\mu} = [1 \quad 1 \quad 1]^T,$$

we obtain hexagonal 2D SA.

WE:

$$\Omega_p = N_1 \cdot (N_2 - 1) + N_2 \cdot (N_3 - 1) +$$
$$+ N_3 \cdot (N_2 - 1) + 1, (\Omega_p = 3N^2 - 3N + 1),$$
$$g_a = (N_1 - 1) \cdot (N_2 - 1) + (N_1 - 1)(N_3 - 1) +$$
$$+ (N_2 - 1)(N_3 - 1), (g_a = 3(N-1)^2),$$
$$t_p = 3,$$
$$T_{in} = N_2 - 1, (T_{in} = N - 1),$$
$$T_{exe} = N_1 + N_2 + N_3 - 2, (T_{exe} = 3N - 2),$$
$$T_{out} = N_2 - 1, (T_{out} = N - 1).$$

NR:

$$\Omega_{p=} N_3 \cdot N_2, (\Omega_p = N^2),$$
$$g_a = (N_2 - 1) \cdot (N_3 - 1), (g_a = (N-1)^2),$$
$$t_p = 1,$$

$$\boldsymbol{T_{in} = \max\{N_2, N_3\} - 1, T_{in} = (N-1),}$$
$$\boldsymbol{T_{exe} = N_2 + N_3 - 1, (T_{exe} = 2N - 1),}$$
$$\boldsymbol{T_{out} = 0.}$$

For projection direction
$$\boldsymbol{For\ \vec{\mu} = \begin{bmatrix}2 & 1 & 1\end{bmatrix}^T,}$$
we obtain 2D SA with crossing.

WE:

$$\Omega_p = N_1 \cdot (N_2 - 1) + 2N_2 \cdot (N_3 - 1) +$$
$$+ N_3 \cdot (N_1 - 2) + 2, (\Omega_p = 4N^2 - 5N + 2),$$
$$g_a = (N_1 - 1) \cdot (N_2 - 1) + (N_1 - 1) \cdot (N_3 - 1) +$$
$$+ 2(N_2 - 1) \cdot (N_3 - 1), (g_a = 4N^2),$$
$$t_p = 4,$$
$$T_{in} = \left\lceil \frac{3N}{2} \right\rceil - 1,$$
$$T_{exe} = N_1 + N_2 + N_3 - 2, (T_{exe} = 3N - 2),$$
$$T_{out} = N - 1.$$

NR:

$$\Omega_p = N_1 \cdot N_3, (\Omega_p = N^2),$$
$$g_a = (N_1 - 1) \cdot (N_3 - 1), (g_a = (N-1)^2),$$
$$t_p = 1,$$
$$T_{in} = \left\lceil \frac{3N}{2} \right\rceil - 1,$$
$$T_{exe} = N_1 + N_2 + N_3 - 2, (T_{exe} = 3N - 2),$$
$$T_{out} = 0.$$

On the base obtained values, we can see difference when we using or not the property of adaptability observed algorithm.
For other side obtained characteristics orthogonal and hexagonal 2D SA suitable for implementation in the matrix multiplication algorithm, are completely coincide with characteristics of static 2D SA. This gives now completely different picture in comparison with made in, papers [3]-[4] and [13]-[14].
For illustration, figures 1, 2, 3 show analyzed SA for projection directions

$$\boldsymbol{\vec{\mu} = \begin{bmatrix}1 & 1 & 0\end{bmatrix}^T, \vec{\mu} = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^T, \vec{\mu} = \begin{bmatrix}2 & 1 & 1\end{bmatrix}^T}$$

respectively, for case $N_1=N_2=N_3=3$ and when parameters are defining with Theorems 6.,. i.e. synthesized on the basis (22).

4 Conclusion

In the paper is showed that on the occasion of determining space and time parameters SA suitable for implementation regular 3-nested loop algorithms,also on occasion of synthesis these arrays, it have big sense to use adaptability of algorithm to the projection direction, when it allow that. For algorithms with this property is described one adaptability procedure. This one is apply before using of transformation when synthesize single SA. Obtained results are illustrated on example of algorithm for rectangular matrix

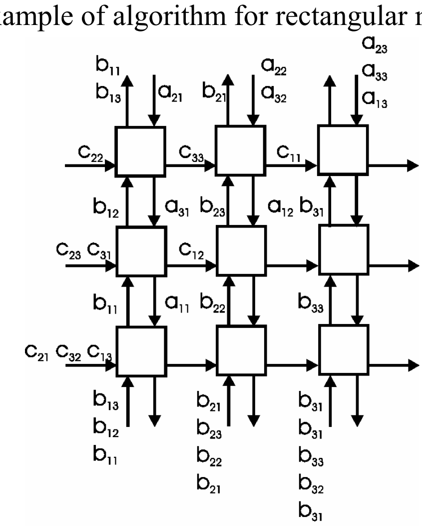

Figure 1

multiplication.

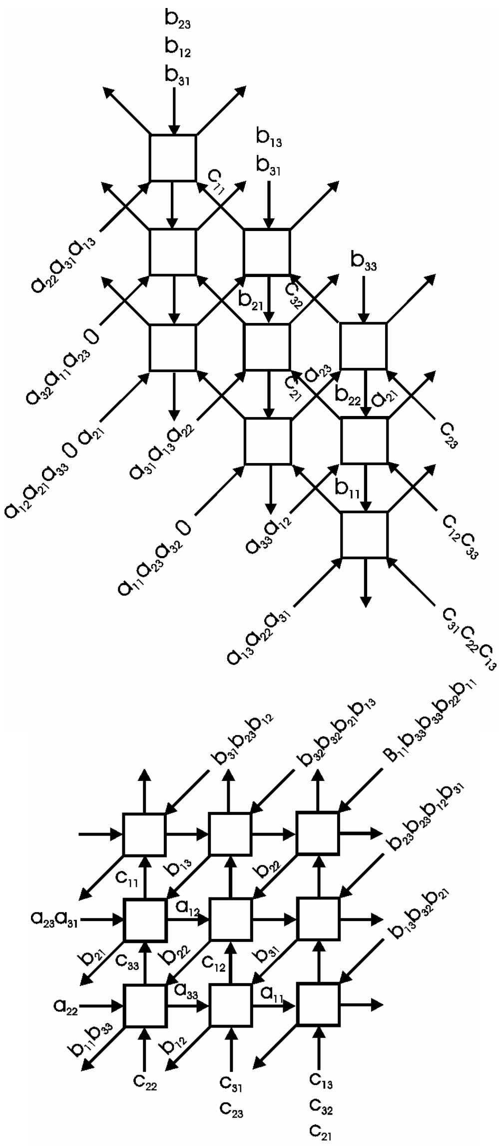

Figure 2

References

[1] M.O.Esonu, J.Al-Khalili, S.Hariri, D.Al-Khalili, Systolic arrays : How to chouse them, *IEE Proceedings*, Vol.139, No.3,1992, pp.179-188.

[2] C.N.Zhang, Systematic Design of Systolic array for computing multiple problem instences, *Microelectronics Journal*,No.23, 1992, pp.543-553.

[3] C.N.Zhang, T.M.Bachtiar, W.K.Chon, Optimal fault-tolerant design approach for VLSI array processors, *IEE Proc. Comput. Digit. Tech.*, Vol. 144, No.1, 1997, pp.15-21.

[4] C.N.Zhang, J.H.Weston, Y.F.Yan, Determining objective functions in systolic array designs, *IEEE Trans. VLSI Systems.*, Vol.2, No.31, 1994, pp.357-360.

[5] S.G.Sedukhin, The designing and analysis of systolic algorithms and structures,*Programing*, No.2, 1991, pp. 20-40.

[6] G.J.Li, B.W.Wah, The design of optimal systolic array, *IEEE Trans. Comput.*,Vol.C-34, No.1, 1989, pp. 66-77.

[7] P.Z.Lee, Z.M.Kedem, Mapping nested loop algorithms into multidimensional systolic arrays, *IEEE Trans. Parallel Distrib. Syst.*, Vol. 1, 1(1990), pp. 69-76.

[8] C.M.Liv, C.W.Jen, Design of algorithm-based fault-tolerant VLSI array processor, *IEE Proc.*, Vol. 136, 6(1989),pp. 539-547.

[9]I.Z.Milentijević,I.Ž.Milovanović,E.I.Milovanović,M.K.Stojčev, The design of optimal planar systolic arrays for matrix multiplication, *Comput. Math. Appl.*, 3(6)(1997), pp. 17-35.

[10] C.R.Wan, D.J.Evans, Ninetin ways of systolic matrix multiplication, *Inter. J. Computer Math.*, Vol. 68(1998), pp. 39-69.

[11] E.I.Milovanović, I.Z.Milentijević, I.Ž.Milovanović, Designing of processor-time optimal systolic array for matrix multiplication, *Comput. Artificial Intellegence*, Vol. 16, 1(1997), pp. 1-11.

[12] H.B.Lee, R.O.Groudin, A comparsion of systolic architectures for matrix multiplication, *IEEE j. Solid State Cicuits*, Vol. 23, 1(1988).

[13] S.Louhani,Y.I.Wang,A.Milenković,V.Milutinović,2D-matrix multiplication on a 3D SA, *Microelectronics Journal*, 27(1996), pp.11-22.

[14] D.I.Moldovan, *Parallel processing, from applications to systems*, Morgan Kaufman Publishers, San Mateo, 1993.

[15] T.I.Tokić, I.Ž.Milovanović, D.M.Randjelović, E.I.Milovanović, *Determining VLSI array size for one class of nested loop algorithms*, Advances in computer and information

[16] D.M.Randjelović,Objective functions for systolic arrays, *WSEAS Transactions of Computers*, Issue 4, Vol. 2,(2003), pp. 1015-1020.

[17] D.M.Randjelović,Space bjective functions for systolic arrays, *WSEAS Transactions of Mathematicss*, Issue 1, Vol. 3,(2004), pp. 7-13.

[18] D.M. Randjelović, Object function 1D systolic arrays, *Mathematica Balkanica*, Vol. 19, 1-2 (2005),

[19] D.M. Randjelović, Objective systolic arrays functions, *Mathematica Balkanica*, Vol. 20, (2006),

sciences `98,*IOS Press(1998),* , pp. 389-396, Istanbul Turkie.