A Problem of Selecting Systolic Algorithm for a Given Mathematical Method

E. I. Milovanović, I.Ž. Milovanović, B. M. Randjelović, N. M. Stojanović *

Faculty of Electronic Engineering, Beogradska 14, 18000 Niš, Serbia

Abstract. For a given mathematical method, which can be systolized, a number of systolic algorithms can be designed. If we want to design a systolic array with predefined topology and features, for some problem, we have to choose the most suitable systolic algorithm from a set of possible systolic algorithms. In general, this is not a simple problem. In this paper we define some criteria that can be applied to solve the above mentioned problem in some particular cases. These criteria are based on well understanding of systolic algorithms, systolic array design methodology and mathematical models for computing systolic array features. We will illustrate the methodology and the problems that arise during the decisions on the example of matrix multiplication algorithm.

1 Introduction

A number of systolic algorithms for a given mathematical model which can be systolized can be derived. Corresponding systolic arrays (SA) may differ in their interconnection pattern as well as spatial and time parameters. So, the question arise: Which one to choose? The answer is obtained by setting some constraints that the SA has to satisfy. To cope with this problem efficiently, one must have a good knowledge of systolic algorithm properties (see for example [1] - [2]), properties of the operations involved in systolic algorithm, as well as mathematical methods for calculating relevant parameters of the SA (see for example [3]-[6]). There is no general method for solving this problem. Instead, the problem is solved from case to case. We will illustrate this problem on the example of matrix multiplication. We will also point out to the situations when the problem can't be solved.

2 The Choice of Systolic Algorithm

Let $A = (a_{ik})_{N_1 \times N_3}$ and $B = (b_{ik})_{N_3 \times N_2}$ be two rectangular matrices. We want to design an orthogonal SA with optimal number of processing elements (PE) for a given problem size, i.e. $\Omega = N_3 \min\{N_1, N_2\}$, that implements $C = A \cdot B$.

Mathematical model for matrix multiplication that can be systolized is given by the following well-known recurrence relation

$$c_{ij}^{(k)} := c_{ij}^{(k-1)} + a_{ik} b_{kj}, \qquad k = 1, 2, \ldots, N_3 \tag{1}$$

* Research supported by Serbian Ministry of Science, Technologies and Development

for each $i = 1, 2, \dots, N_1$ and $j = 1, 2, \dots, N_2$, where $c_{ij}^{(0)} = 0$ and $c_{ij}^{(N_3)} = c_{ij}$.

As we have already mentioned, a number of systolic algorithms can be associated with this mathematical model. As an example we give the following three algorithms.

Algorithm_1
for $k := 1$ **to** N_3 **do**
for $j := 1$ **to** N_2 **do**
for $i := 1$ **to** N_1 **do**
$a(i, j, k) := a(i, j-1, k);$
$b(i, j, k) := b(i-1, j, k);$
$c(i, j, k) := c(i, j, k-1) + a(i, j, k) * b(i, j, k);$

where $a(i, 0, k) \equiv a_{ik}$, $b(0, j, k) \equiv b_{kj}$, $c(i, j, 0) \equiv 0$, $c_{ij} \equiv c(i, j, N_3)$.

Algorithm_2
for $k := 1$ **to** N_3 **do**
for $j := 1$ **to** N_2 **do**
for $i := 1$ **to** N_1 **do**
$a(i, i+j-1, k) := a(i, i+j-2, k);$
$b(i, i+j-1, k) := b(i-1, i+j-1, k);$
$c(i, i+j-1, k) := c(i, i+j-1, k-1) + a(i, i+j-1, k) * b(i, i+j-1, k);$

with the same initial values as in Algorithm_1. Here, we assume the following periodicity of data items $b(i, j+N_2, k) \equiv b(0, j, k)$ and $c(i, j+N_2, k) \equiv c(i, j, k)$, for each $i = 1, 2, \dots, N_1$, $j = 1, 2, \dots, N_2$ and $k = 1, 2, \dots, N_3$.

Algorithm_3
for $k := 1$ **to** N_3 **do**
for $j := 1$ **to** N_2 **do**
for $i := 1$ **to** N_1 **do**
$a(i+j-1, j, k) := a(i+j-1, j-1, k);$
$b(i+j-1, j, k) := b(i+j-2, j, k);$
$c(i+j-1, j, k) := c(i+j-1, j, k-1) + a(i+j-1, j, k) * b(i+j-1, j, k);$

with the same initial values as in Algorithm_1 and following periodicity of data items $a(i+N_1, j, k) \equiv a(i, 0, k)$ and $c(i+N_1, j, k) \equiv c(i, j, k)$ for each $i = 1, 2, \dots, N_1$, $j = 1, 2, \dots, N_2$ and $k = 1, 2, \dots, N_3$.

It is well-known that orthogonal SAs are obtained by the projection directions $\boldsymbol{\mu}[1\,1\,0]^T$, $\boldsymbol{\mu} = [1\,0\,1]^T$ and $\boldsymbol{\mu} = [0\,1\,1]^T$ (see for example [7]-[9]). In the analysis that follows we will mainly use the direction $\boldsymbol{\mu}[1\,1\,0]^T$ since it is best suited for the chosen algorithms, and give some notes concerning other two directions.

Suppose that we have chosen a valid transformation matrix (see for example [10]) which corresponds to the direction $\boldsymbol{\mu}[1\,1\,0]^T$. Denote by SA_1, SA_2 and SA_3 the orthogonal SAs synthesized according to Algorithms 1, 2 and 3, respectively. The number of processing elements in the synthesized SAs are determined according to

$$\begin{aligned} &\text{SA\_1: } \Omega_1 = N_3(N_1+N_2-1),\\ &\text{SA\_2: } \Omega_2 = N_2N_3,\\ &\text{SA\_3: } \Omega_3 = N_1N_3. \end{aligned} \tag{2}$$

It is obvious that $\Omega_1 > \Omega_2, \Omega_3$. If $N_1 > N_2$, then $\Omega_2 < \Omega_3$. It is also obvious that we cannot affect on the value N_3 since N_3 operations of the type $c + ab$ have to be performed to compute each element c_{ij} of the resulting matrix C. In other words, the value N_3 always contribute to the number of PEs in the SA. In the case $N_1 > N_2$, the array SA_2 has a minimal number of PEs for a given problem size and, consequently, the Algorithm_2 is the only right choice. Under the condition $N_1 > N_2$, Algorithm_2 is best suited to the direction $\boldsymbol{\mu}[1\,1\,0]^T$. To clarify this, observe the inner computation space of Algorithm_2,

$$P_{in} = \{[i\ i+j-1\ k]^T | 1 \le i \le N_1,\, 1 \le j \le N_2,\, 1 \le k \le N_3\}. \tag{3}$$

Denote with

$$\boldsymbol{p}_i = [i\ i+j-1\ k]^T \quad \text{and} \quad \boldsymbol{p}_{i+1} = [i+1\ i+j\ k]^T$$

the position vectors of two index points from the space (3), for some fixed j and k, and $i = 1, 2, \ldots, N_1$. Since $\boldsymbol{p}_{i+1} - \boldsymbol{p}_i = [1\,1\,0]^T = \boldsymbol{\mu}$, we conclude that all index points with the position vector $\boldsymbol{p}_i = [1\,i+j-1\,k]^T$, for $i = 1, 2, \ldots, N_1$ and some fixed j and k are placed on the line with the direction $\boldsymbol{\mu}[1\,1\,0]^T$. As a consequence, all these points are mapped into one point in the projection plane, i.e. in one PE of the synthesized SA. In other words, the value N_1 doesn't contribute to the size of the SA. Since $N_1 > N_2$, we conclude that Algorithm_2 is best suited for the direction $\boldsymbol{\mu}[1\,1\,0]^T$.

Similarly, when $N_1 < N_2$ we conclude that $\Omega_3 < \Omega_2$, i.e. that the array SA_3 is an optimal one for a given problem size. Since SA_3 is obtained from Algorithm_3, in the case $N_1 < N_2$ the Algorithm_3 is best suited for the direction $\boldsymbol{\mu}[1\,1\,0]^T$.

When $N_1 = N_2$ Algorithms 2 and 3 are both equally suited to the direction $\boldsymbol{\mu}[1\,1\,0]^T$, so both SA_2 and SA_3 are optimal with respect to a problem size.

Figures 1, 2 and 3 depict data schedule at the beginning of the computation in the arrays SA_1, SA_2 and SA_3, respectively, for the case $N_1 = 3$, $N_2 = 2$ and $N_3 = 4$.

It is nod difficult to see that Algorithms 1, 2 and 3 are not suited for other two directions, i.e. $\boldsymbol{\mu} = [1\,0\,1]^T$ and $\boldsymbol{\mu} = [0\,1\,1]^T$. The question is whether we can find the algorithms that are suited for these directions. The answer is: not for all relations between N_1 and N_2.

Suppose that $N_1 \ge N_2$. Then, the following algorithm is best suited to the direction $\boldsymbol{\mu} = [1\,0\,1]^T$:

Algorithm_4
  for $k := 1$ **to** N_3 **do**
    for $j := 1$ **to** N_2 **do**
      for $i := 1$ **to** N_1 **do**
      $a(i,j,i+k-1) := a(i,j-1,i+k-1);$
      $b(i,j,i+k-1) := b(i-1,j,i+k-1);$
      $c(i,j,i+k-1) := c(i,j,i+k-2) + a(i,j,i+k-1)b(i,j,i+k-1);$

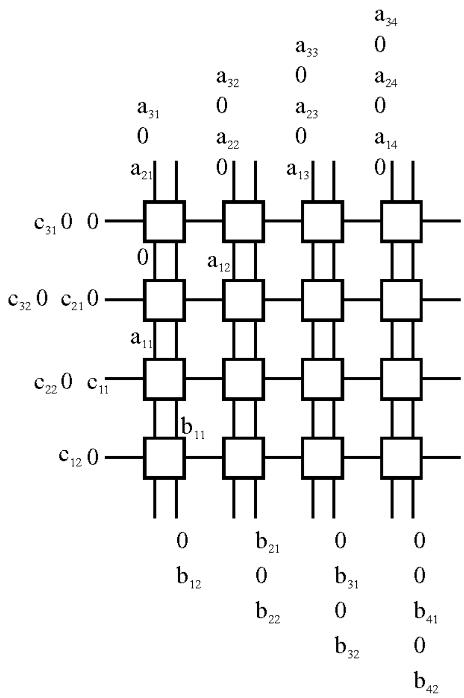

Fig. 1. Orthogonal array SA_1, for $N_1 = 3, N_2 = 2, N_3 = 4$

with the following periodicity of data items $a(i, j, k + N_3) \equiv a(i, 0, k)$ and $b(i, j, k + N_3) = b(0, j, k)$ for each $i = 1, 2, \ldots, N_1$, $j = 1, 2, \ldots, N_2$ and $k = 1, 2, \ldots, N_3$. The systolic array obtained according to Algorithm_4 and direction $\boldsymbol{\mu} = [1\,0\,1]^T$ will have $\Omega_4 = N_2 N_3$ processing elements, i.e. it will be optimal for a given problem size. When $N_1 < N_2$ it is not possible to construct an algorithm that can be used to obtain SA with optimal number of PEs.

Similarly, for the direction $\boldsymbol{\mu} = [0\,1\,1]^T$ it is possible to construct the best suited algorithm only in the case $N_1 \leq N_2$. The algorithm has the following form:

Algorithm_5
for $k := 1$ **to** N_3 **do**
for $j := 1$ **to** N_2 **do**
for $i := 1$ **to** N_1 **do**
$a(i, j, j + k - 1) := a(i, j - 1, j + k - 1);$
$b(i, j, j + k - 1) := b(i - 1, j, j + k - 1);$
$c(i, j, j + k - 1) := c(i, j, j + k - 2) + a(i, j, j + k - 1) b(i, j, j + k - 1);$

with the same periodicity of data items as in Algorithm_4. The array synthesized according to Algorithm_5 and direction $\boldsymbol{\mu} = [0\,1\,1]^T$ will have $\Omega_5 = N_1 N_3$ PEs. In the case $N_1 > N_2$ it is not possible to construct an algorithm which can be used to obtain SA with optimal number of PEs, that is $N_2 N_3$.

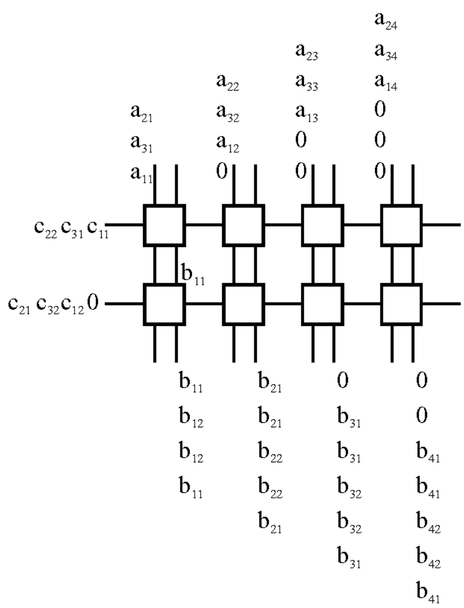

Fig. 2. Orthogonal array SA_2, for $N_1 = 3, N_2 = 2, N_3 = 4$

In general, to find out whether a given algorithm can be adjusted to some direction, we have to explore the properties of the operations involved in the algorithm. In the analysis we start from the basic algorithm. In the case of matrix multiplication it is Algorithm_1. The operations involved in matrix multiplication are addition and multiplication. Since the addition is both associative and commutative operation, the computations in the algorithm can be performed on arbitrary permutations of index variables $i \in \{1, 2, \ldots, N_1\}$, $j \in \{1, 2, \ldots, N_2\}$ and $k \in \{1, 2, \ldots, N_3\}$. Since index variable k is an iterative one, we keep it intact. Therefore we conclude that Algorithm_1 is adaptive on index variables i and j.

Algorithm_1 can be adapted to the projection direction $\boldsymbol{\mu} = \{\mu_1\, \mu_2\, \mu_3\}$ on index variable i, if $\mu_1 = 1$. We assume that directions $\boldsymbol{\mu}$ and $-\boldsymbol{\mu}$ are equal.

Similarly, Algorithm_1 can be adapted to the direction $\boldsymbol{\mu} = \{\mu_1\, \mu_2\, \mu_3\}$ on index variable j, if $\mu_2 = 1$. For more details concerning the adaptation one can refer to [5]-[6].

3 Conclusion

We have discussed a problem of selecting The most suitable systolic algorithm for designing orthogonal SA that implements matrix multiplication with optimal (i.e. minimal) number of PEs for a given problem size. We have shown that the solution depends on the relation between loop boundaries and selected projection direction vector.

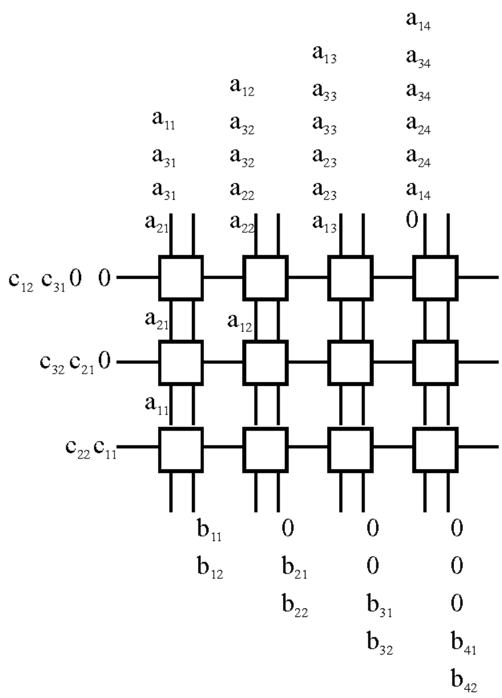

Fig. 3. Orthogonal array SA_3, for $N_1 = 3, N_2 = 2, N_3 = 4$

References

1. H.T. Kung, Why Systolic Architectures?, *IEEE Computer*, 15(1):37-46, (1982)
2. S.G. Sedukhin, The Designing and Analyzing Systolic Algorithms and Structures, *Programming* (in Russian), 2:20-40, (1991)
3. C.N. Zhang, J.H. Weston, Y.-F. Yan, Determining Objective Functions in Systolic Array Designs, *IEEE Transactions on VLSI Systems*, 2(3):357-360 (1994)
4. C.N. Zhang, Systematic Design of Systolic Arrays for Computing Multiple Problem Instances, *Microelectronics Journal*, 23:543-553 (1992)
5. T.I. Tokić, I.Z. Milovanović, D.M. Randjelović, E.I. Milovanović, Determining VLSI Array Size for one Class of Nested Loop Algorithms, In: *Advances in Computer and Information Sciences*, (U. Gudukbay, T. Dagar, A. Gursay, E. Gelembe, eds.), IDS Press, (1998), 389-396
6. I.Z. Milovanović, T.I. Tokić, E.I. Milovanović, M.K. Stojčev, Determining the Number of Processing Elements in Systolic Arrays, *Facta Univ., Ser. Math. Inform.*, 15:123-132, (2000)
7. I.Z. Milentijević, I.Ž. Milovanović, E.I. Milovanović, M.K. Stojčev, The Design of Optimal Planar Systolic Arrays for Matrix Multiplication, *Computer Mathematics Applications*, 33(6):17-35 (1997)
8. C.R. Wan, D.J. Evans, Nineteen Ways of Systolic Matrix Multiplication, *International Journal of Computer Mathematics*, 98:39-69 (1998)

9. S.G. Sedukhin, G.Z. Karapetian, Design of Optimal Systolic Systems for Matrix Multiplication of Different Structures (in Russian), Report 85, Comput. Center Siberian Division of USSR Academy of Science, Novosibirsk, (1990)
10. M. Bekakos, E. Milovanović, N. Stojanović, T. Tokić, I. Milovanović, I. Milentijević, Transformation Matrices for Systolic Array Synthesis, *Journal Electrotehnical Mathematics*, 7(1):9-15 (2002)