SYSTOLIC IMPLEMENTATION FOR DECONVOLUTION ITERATIVE ALGORITM

J.J. NAVARRO GUERRERO
V. CASARES GINER

RR85/19

RESUMEN.

Para resolver algunos algoritmos de Procesado Digital de Señal donde se requiere gran velocidad de cálculo son necesarias arquitecturas de propósito especial con procesado paralelo. Los procesadores Systólicos son buenos candidatos para implementar estas aplicaciones. En este trabajo se presenta una implementación systólica de un algoritmo iterativo de deconvolución. Un array bidimensional implementa una iteración del algoritmo. Este módulo básico puede concatenarse repetidamente de forma que permite su uso en aplicaciones en tiempo real.

ABSTRACT.

Systolic architectures implement regular algorithms in hardware, in order to obtain high computational throughput. In this paper we provide a modular architecture for a deconvolution iterative algorithm. The basic module is a systolic array which implement one iteration of the algorithm recently proposed in 1 . The algorithm is a generalization of the method to invert non singular polynomial transfer function, previously published in 2 . The basic systolic module can be repeatedly concatenated in such a way that can be used in real time applications.

SYSTOLIC IMPLEMENTATION FOR DECONVOLUTION ITERATIVE ALGORITHM

Juan José Navarro-Guerrero *
Vicente Casares-Giner **

* Facultad de Informática (U.P.C.)
** E.T.S.I. Telecomunicación (U.P.C.)
C/Jordi Girona Salgado s/n. 08034 Barcelona, (Spain)

ABSTRACT

Systolic architectures implement regular algorithms in hardware, in order to obtain high computational throughput. In this paper we provide a modular architecture for a deconvolution iterative algorithm. The basic module is a systolic array which implement one iteration of the algorithm recently proposed in |1|. The algorithm is a generalization of the method to invert non singular polynomial transfer function, previously published in |2|. The basic systolic module can be repeatedly concatenated in such a way that can be used in real time applications.

INTRODUCTION

Special purpose system with parallel processing can be used for some Digital Signal Processing when high-speed processing is required. Systolic Array Processors allow to obtain high computational throughput for reasonable I/O requirements when it is possible to perform repeated calculations on each data item for one or more large input data streams. The simplicity and the regularity of Processing Elements (PEs), and their interconexions, as soon as the locality for communications, make this architectures very suitables for Very Larg Scale Integrations (VLSI). |7|,|3|,|8|.

Recursive and local data-dependent algorithms are good candidates to be implement with systolic architectures. This is the case of a deconvolution algorithm we present here, for which, we provide a systolic realization. In the next lines we introduce very briefly, the convolution and deconvolution problems.

There are some applications where the recorded information has been previously degraded. One of the most common model used, consist of noise added to the signal which has been distorted by an invariant linear system, that is

$$y = h*x+n \qquad (1)$$

In (1), x denotes the useful information, n the random noise, h the deterministic impulse response of the linear system and y the available information (* is the convolution operator). The deconvolution problem consist of recovering the information x, for a given degrading system h, a given random process n and a recorded information y.

This paper has two main parts. In the first one we present an iterative algorithm dealing with the inversion of (1) in a digital fashion. In addition, we assume that the transfer function of h is defined by a finite impulse response (FIR), free of zeros on the unit circumference Γ|4|. Also, in (1) we have considered n=0. Under such assumptions, for instance, we can always factorize the transfer function in a product of two factors of finite order, having their zeros inside and outside of Γ. This allow us to implement the inversion as a cascade of two systems of minimum and maximum phases |9|. However, if we are interested in a real time processing this method is not longer valid. The algorithm we present here, overcomes this deficiency.

The second part of the paper, deals with systolic architectures for the algorithm. In the literature, some systolic implementations has been proposed in the deconvolution area |5|. For instance, G.L. Li and B.W. Wah |6| provide a realization for the case that all zeros of the transfer function of h lie inside Γ. This and other cases, as we will see, can be obtained as particularizations of our systolic architecture.

THE GENERAL ALGORITHM

Let $H_M(z^{-1})$ be the transfer function of the degrading system, which consist of a real polynomial in z^{-1} of degree M, and it is free of zeros on Γ. Let R(0≤R≤M) be the number of roots outside Γ. It is well

This work was supported in part by CAICYT under grant nº 2906-83.

known that, in general, the ideal inverse is stable but not causal. So, if we want to implement the inverse of (h^{-1}) in a real time environment, we need to approximate the ideal impulse response by a causal one. For that purpose, in |1| it was presented a decomposition of $H_M(z^{-1})$ in order to carry out the refered real time inversion. Here, and for the sake of comprehensibility, we will repeat the basic arguments of |1|.

Given $H_M(z^{-1})$ we look for a 2D polynomial:

$$HP(z^{-1},W^{-1})=D_M(z^{-1})+\sum_{i=1}^{T} N_{Mi}(z^{-1})W^{-i} \qquad (2)$$

where $D_M(z^{-1})$ and $M_{Mi}(z^{-1})$ (1,2,...,T) are polynomials of degree M. Besides, the conditions we want to be satisfied are:

(2)-a: $HP(z^{-1},1)=H_M(z^{-1})=\sum_{m=0}^{M} h_m z^{-m}$

(2)-b: $F(z^{-1})=z^P D_M(z^{-1})$ be a minimum phase factor, i.e., all zeros and poles inside Γ. P is a positive constant to be determined.

(2)-c: $HP(\exp(j\omega),W^{-1})$ be a minimum phase factor with the variable W for any ω given.

From condition (2)-a, the sum of coefficients $\{d_m\}$ and $\{n_{im}\}$ (i=1,2,...,T) with the same power of z^{-1}, must be equal to the associated coefficient of $H_M(z^{-1})$, that is,

$$h_m=d_m+\sum_{i=1}^{T} n_{im};\ 0\leq m\leq M \qquad (3)$$

From the condition (2)-b is not difficult to see that

$$d_m=0,\ d_p\neq 0\ ;\ 0\leq m<P \qquad (4)$$

About the positive integer number P, we can very easy show from condition (2)-c that it must be equal to the number of roots of $H_M(z^{-1})$ outside Γ; i.e., P=R. In order to proof these, we write (2) as:

$$HP(z^{-1},W^{-1}) = D_M(z^{-1})(1-W_1(z^{-1})/W)\ldots(1-W_T(z^{-1})/W) \qquad (5)$$

,i.e., a factorial decomposition of $HP(z^{-1},W^{-1})$ in the variable W^{-1}, where the roots $W_i(z^{-1})$ are functions of z. Particulary, when $z=\exp(j\omega)$, and according to (2)-c, $|W_i(\exp(-j\omega))|<1$ for i=1,2,...,T. Then, when varies from 0 to 2π the argument of $HP(\exp(j\omega),1)$, i.e., of $H_M(\exp(-j\omega))$, varies from 0 to $-2R\pi$ which forces

$$D_M(z^{-1}) = \sum_{m=R}^{M} d_m z^{-m} \qquad (6)$$

, that is, P=R. Figure 1-a gives the array support of the proposed decomposition.

If the decomposition (2) exist, the solution to the convolution equation

$$y_m=\sum_{i=0}^{M} h_i\, x_{m-i} \quad (\text{or } Y(z^{-1})=H_M(z^{-1})X(z^{-1})) \qquad (7)$$

can be carried out by the following iterative algorithm in the frequency domain:

$$X^P(z^{-1}) = \left[Y(z^{-1})-\sum_{i=1}^{T} N_{Mi}(z^{-1}).X^{P-i}(z^{-1})\right]/D_M(z^{-1}) \qquad (8)$$

, where $X^P(z^{-1})$ denotes the p-th estimate of $X(z^{-1})$ obtained in the p-th iteration. The formulation of (8) in the time domain is given by:

$$x^P_{m-R}=\frac{1}{d_R}\left[y_m-\sum_{j=R+1}^{M} d_j x^P_{m-j}\ \sum_{i=1}^{T}\sum_{j=0}^{M} n_{ij} x^{P-i}_{m-j}\right] \qquad (9)$$

,where x^P_m denotes the p-th estimate of the sample x_m obtained in the p-th iteration.

A simple and general decomposition

Here, and in order to provide a systolic architecture, we show again a trivial decomposition according to (2), which was already presented in |1|. The mentioned decomposition can be obtained from the Algebra Fundamental Theorem. Let $H_+(z^{-1})$ and $H_-(z^{-1})$ be the two factor of $H_M(z^{-1})$ where the first one contain the M-R roots inside Γ and the second one the R roots outside Γ:

$$H_M(z^{-1})=H_+(z^{-1})\,H_-(z^{-1}) = \sum_{i=0}^{M-R} h_{+i}\, z^{-1} \sum_{i=0}^{R} h_{-i}\, z^{-i} \qquad (10)$$

If we identify

$$D_M(z^{-1}) = h_{-0} z^{-R} H_+(z^{-1}) \qquad (11\text{-a})$$

$$N_{Mi}(z^{-1}) = h_{-i}\, z^{i-R} H_+(z^{-1}) \qquad (11\text{-b})$$

$$i = 1,2,\ldots,T = R$$

it is clear that (1) is a valid solution satisfying (2). In figure 1-b we represent the array support for this simple and exact

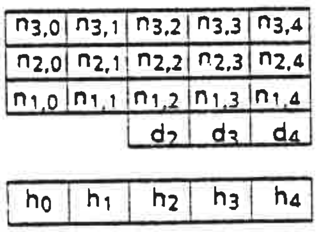

Fig.1.a: General decomposition. M = 4, R = 2, T = 3.

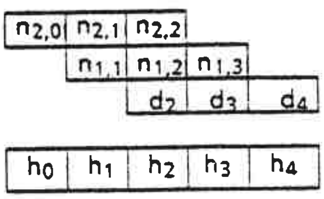

Fig.1.b: Simple decomposition. M = 4, R = T = 2.

decomposition. In this particular decomposition the time domain formulation of (9) becomes as:

$$x^p_{m-R} = \frac{1}{d_R}\left[y_m - \sum_{j=R+1}^{M} d_j x^p_{m-j} - \sum_{i=1}^{R} \sum_{j=R-i}^{M-i} n_{ij} x^{p-i}_{m-j}\right] \qquad (12)$$

SYSTOLIC ARCHITECTURES

In this section we propose a systolic architecture for (12). Figure 2-a shows the 2-Dimensional systolic array which compute one iteration $\{x^p\}$ we will name this systolic array, as a basic module; and its PEs are labeled as PE(i,j), $i=0,1,\ldots,R$; $j=0,1,\ldots M-R$. Three different kinds of PEs can be differentiated in the array; type A, B and C, (figure 2-b). Sub index sets $(i=0,1,\ldots,R-1; j=0,1,\ldots,M-R)$ $(i=R, j=1,2,\ldots,M-R)$ and $(i=R, j=0)$ define the PES of type A, B and C respectively.

In order to explain the computing spatial distribution in the array, we will use the expresion (12) transformated after some manipulations in:

$$x^p_m = \frac{1}{d_R}\{y_{m+R} - \sum_{j=1}^{M-R}\left[d_{R+j} x^p_{m-j} - \sum_{i=0}^{R-1} n_{R-i,i+j} x^{p-R+i}_{m+R-i-j}\right] - \sum_{i=0}^{R-1} n_{R-i,i} x^{p-R+i}_{m+R-i}\} \qquad (13)$$

Each PE(i,j) of type A compute the expression

$$\alpha_{ij} = \alpha_{i-1,j} + n_{R-i,i+j} x^{p-R+i}_{m+R-i-j} \qquad (14)$$

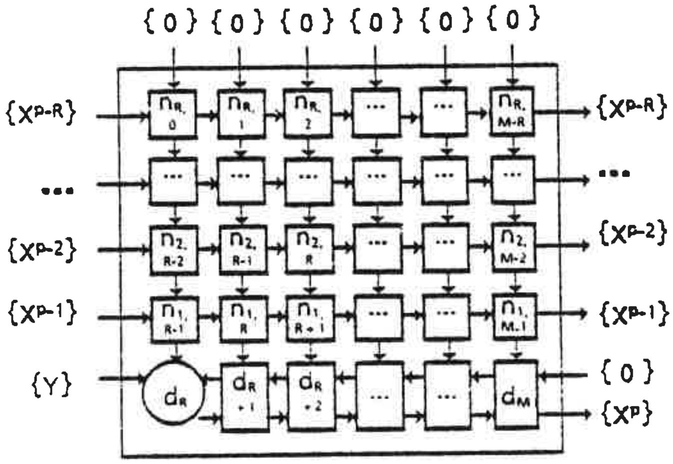

Fig. 2.a: Basic Module.

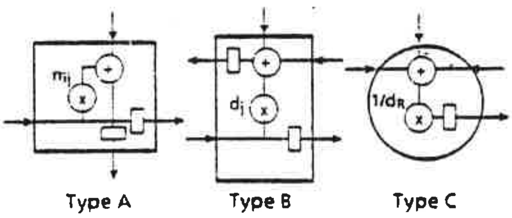

Fig. 2.b: Processing Elements.

in such a way that the partial result obtained for each PEs column of type A is

$$\alpha_j = \sum_{i=0}^{R-1} n_{R-i,i+j} x^{p-R+i}_{m+R-i-j} \qquad (15)$$

Each PE(i,j) of type B compute the expression

$$\beta_j = \beta_{j+1} + d_{R+j} x^p_{m-j} + \alpha_j \qquad (16)$$

in such a way that the PEs row of type B compute the expression:

$$\beta = \sum_{j=1}^{M-R}\left[d_{R+j} x^p_{m-j} + \sum_{i=0}^{R-1} n_{R-i,i+j} x^{p-R+i}_{m+R-i-j}\right] \qquad (17)$$

The PE(R,0) of type C compute the desired result

$$x^p_m = \frac{1}{d_R}\left[y_{m+R} - \beta - \alpha_0\right] \qquad (18)$$

In order to compute (14)-(18) correctly, the input data streams must be injected into the basic module with an empty clock cicle between consecutive data samples. Figure 3 shows the I/O data systems for M=7 and R=3.

If we want to obtain several sequences $\{x^p\}$ $(p=1,2,\ldots)$, we concatenate as many basic modules as number of iterations are desired. Figure 4 give us the interconection topology for 5 basic modules. Notice that this configuration performs a real time deconvolution; where the initial conditions are $\{x^0\}=\{x^{-1}\}=\{x^{-2}\}=\{y\}$.

When R=0 the deconvolution algorithm is only recursive, not iterative, since, and according to the general decomposition (2), $d_m=h_m$ $m=0,1,\ldots,M$. Consequently our algorithm becames the known method of successive substitutions. The resulting basic module has only PEs of type B and C, which is similar to the architecture proposed in |6|.

CONCLUSIONS

This paper concerns with a systolic architecture for a deconvolution iterative

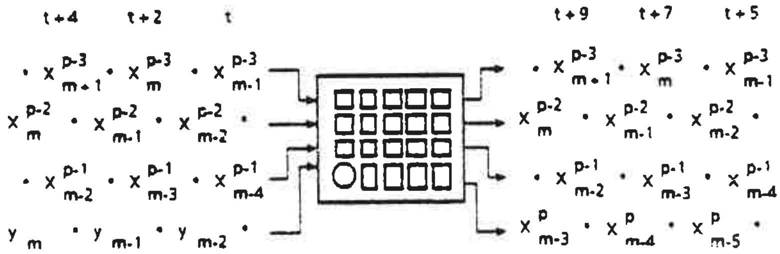

Fig. 3: Input / Output data flow.

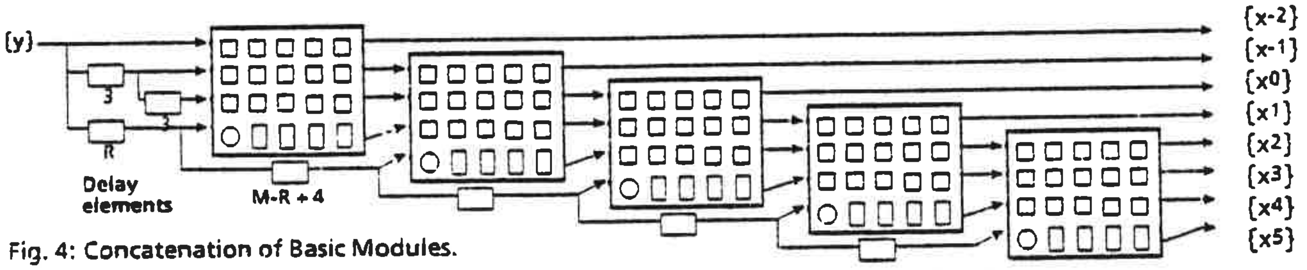

Fig. 4: Concatenation of Basic Modules.

algorithm. We have presented a modular configuration such that each basic module performs one iteration of the algorithm. With repeated concatenation of that basic modules we can obtain a deconvolution system for real time applications.

Further studies are needed in order to improve the efficiency of this and others systolic architectures. In that sense, the algorithms expresed by (12) and in |2| has been studied in |10|.

REFERENCES

|1| V. Casares-Giner, "Implementation of 1-D and 2-D Digital Filters by Iterative and Recursive Methods". Proceedings of the International Conference on Digital Signal Processing, pp. 18-21. Florence. September 1984.

|2| V. Casares-Giner et al., "Asymptotical inversion of non-singular digital symmetrical polynomial transfer functions". Signal Processing, vol. 4, pp. 5-16, January 1982.

|3| H.T. Kung, "Why Systolic architectures?". I.E.E.E. Trans. on Computers. Vol. C-15, pp. 37-46, January 1982.

|4| A.V. Oppenheim & R.W. Schaffer, "Digital Signal Processing". Prentice Hall 1978.

|5| R. Yoho & D. Preis, "Systolic Architectures for Deconvolution Algorithms". Proceedings of the ICASSP-1984, paper 25A.3.1-25A.3.4.

|6| G.J. Li and B.W. Wah, "The Design of Optimal Systolic Arrays". I.E.E.E. Trans. on Computers, vol. C-34, pp. 66-67, January 1985.

|7| H.T. Kung & C.E. Leiserson, "Systolic Arrays (for VLSI)". Sparse Matrix Proceedings 1978, Duff, I.S. and Stewart, G.W., eds. Society for Industrial and Applied Mathematics, pp. 256-282, 1979.

|8| C. Mead & L. Conway, "Introduction to VLSI Systems". Addison-Wesley 1980.

|9| T.H. Huang, "Stability of two-dimensional recursive filters". I.E.E.E. Trans. Audio Electroacoust. Vol. 20, pp. 158-163. June 1972.

|10| J.J. Navarro-Guerrero & V. Casares-Giner, "Improvement of the eficiency in deconvolution Systolic Architectures". Submited to the 2º Simposio de Electronica das Telecomunicaçoes. Forum Piscoas, Lisboa, 21,22 e, 23 de Maio de 1986.