Transaction Briefs

On the Synthesis of 2-D State-Space Digital Filters with Minimum Roundoff Noise

W.-S. LU AND A. ANTONIOU

Abstract—**Some results reported recently by Hinamoto, Hamanaka, and Maekawa concerning 2-D state-space digital filters with minimum roundoff noise are discussed.**

The synthesis of 2-D state-space digital filters has recently been considered in [1]–[3]. The techniques adopted in [2] to carry out the analysis and synthesis are very much the same as those reported earlier in [1] except that the Fornasini–Marchesini state-space model was used instead of the Roesser model.

Results reported in [2] for a specific example show that a filter structure based on the Fornasini–Marchesini model has lower roundoff noise than a corresponding structure based on the Roesser model. On the basis of these results, the authors concluded that their method enables one to attain more reduction in the roundoff noise.

In this brief, we show that the Roesser realization used in [2] is suboptimal and that another Roesser realization can be obtained that has approximately the same roundoff noise as the corresponding Fornasini–Marchesini realization used in [2]. In effect, the claim made about the possible reduction of roundoff noise by using the method in [2] remains unsubstantiated.

The 2-D transfer function used in [2], namely

$$H(z_1,z_2)=\frac{1+0.678\,76z_1^{-1}+0.703\,82z_2^{-1}+0.462\,09z_1^{-1}z_2^{-1}}{1+0.724\,34z_1^{-1}+0.681\,49z_2^{-1}+0.525\,71z_1^{-1}z_2^{-1}}$$

can be realized by the Roesser model

$$\begin{aligned}\begin{bmatrix}x^h(i+1,j)\\x^v(i,j+1)\end{bmatrix}&=\begin{bmatrix}-0.724\,34 & 1.054\,26\\-0.030\,43 & -0.681\,49\end{bmatrix}\\&\quad\times\begin{bmatrix}x^h(i,j)\\x^v(i,j)\end{bmatrix}+\begin{bmatrix}1\\1\end{bmatrix}u(i,j)\\&\equiv\begin{bmatrix}A_1 & A_2\\A_3 & A_4\end{bmatrix}x(i,j)+\begin{bmatrix}b_1\\b_2\end{bmatrix}u(i,j)\\y(i,j)&=[-0.045\,58\;\;310.022\,33]\begin{bmatrix}x^h(i,j)\\x^v(i,j)\end{bmatrix}+u(i,j)\\&\equiv[c_1\quad c_2]x(i,j)+du(i,j).\end{aligned}\tag{1}$$

If

$$K=\begin{bmatrix}K_{11} & K_{12}\\K_{21} & K_{22}\end{bmatrix},\quad W=\begin{bmatrix}W_{11} & W_{12}\\W_{21} & W_{22}\end{bmatrix}$$

are the positive definite matrices defined by (24) and (18) of [1], respectively, then K_{ii} and W_{ii} $(i=1,2)$ can be evaluated using a Lyapunov approach as proposed in [1]. This approach is especially efficient for lower order 2-D state-space filters. To find K_{11}, note that

$$K_{11}=\frac{1}{2\pi j}\oint_{|z_2|=1}\tilde{K}_1(z_2)\frac{dz_2}{z_2}\tag{2}$$

where $\tilde{K}_1(z_2)$ is the positive-definite Hermitian solution of the Lyapunov equation

$$\begin{aligned}\tilde{A}_1(z_2)\tilde{K}_1(z_2)\tilde{A}_1^*(z_2)-\tilde{K}_1(z_2)&=-\tilde{b}_1(z_2)\tilde{b}_1^*(z_2)\\\tilde{A}_1(z_2)&=A_1+A_2(z_2I-A_4)^{-1}A_3\\\tilde{b}_1(z_2)&=b_1+A_2(z_2I-A_4)^{-1}b_2.\end{aligned}\tag{3}$$

Solving (3) for $\tilde{K}_1(z_2)$, we obtain

$$\tilde{K}_1(z_2)=\frac{(z_2+1.735\,75)(1.735\,75z_2+1)}{z_2(z_2+1.358\,70)(z_2+0.7360)}.$$

K_{11} can now be calculated by applying the residue theorem to (2) as

$$K_{11}=6.761\,66.$$

Similarly, it is found that

$$K_{22}=1.911\,95$$

$$W_{11}=4.431\,15\times10^{-3}$$

and

$$W_{22}=3.147\,56\times10^{-3}.$$

Thus

$$G_0=\sum_{i=1}^{2}W_{ii}K_{ii}=0.032\,428.$$

The desired transformation matrix T is found by applying the algorithm presented in [1] as

$$T=\begin{bmatrix}\sqrt{K_{11}} & 0\\0 & \sqrt{K_{22}}\end{bmatrix}=\begin{bmatrix}2.600\,32 & 0\\0 & 1.382\,73\end{bmatrix}\tag{4}$$

and, consequently, the desired state-space representation of the filter is given by

$$\begin{aligned}\begin{bmatrix}x^h(i+1,j)\\x^v(i,j+1)\end{bmatrix}&=\tilde{A}x(i,j)+\tilde{b}u(i,j)\\y(i,j)&=\tilde{c}x(i,j)+du(i,j)\end{aligned}\tag{5}$$

where

$$\tilde{A}=T^{-1}AT=\begin{bmatrix}-0.727\,34 & 0.560\,61\\-0.057\,23 & -0.681\,49\end{bmatrix}$$

$$\tilde{b}=T^{-1}b=\begin{bmatrix}0.384\,57\\0.723\,21\end{bmatrix}$$

$$\tilde{c}=cT=[-0.118\,52\quad 0.030\,88].$$

Manuscript received July 26, 1989. This paper was recommended by Associate Editor T. R. Viswanathan.

The authors are with the Department of Electrical and Computer Engineering, University of Victoria, BC V8W 3P6, Canada.

IEEE Log Number 9038549.

0098-4094/90/1100-1424$01.00 ©1990 IEEE

Notice further that

$$\tilde{K} = T^{-1}KT^{-t} = \begin{bmatrix} 1 & * \\ * & 1 \end{bmatrix}$$

and

$$\tilde{W} = T^{t}WT = \begin{bmatrix} W_{11}K_{11} & * \\ * & W_{22}K_{22} \end{bmatrix} \equiv \begin{bmatrix} \tilde{W}_{11} & * \\ * & \tilde{W}_{22} \end{bmatrix}.$$

Hence

$$\tilde{G} = \sum_{i=1}^{2} \tilde{W}_{ii}\tilde{K}_{ii} = \sum_{i=1}^{2} W_{ii}K_{ii} = G_0 = 0.032\,428.$$

On comparing the above result with that in [2, sect. IV], it is observed that our $\tilde{G}$ is slightly smaller than the optimized unit noise in [2], which is 0.034 951. The small difference may be due to the truncation method used in [2] to obtain matrices $\boldsymbol{K}$ and $\boldsymbol{W}$, which may have resulted in an imperfect realization. The basic problem with the Roesser realization used in [2] is that it is of higher order than necessary, which has resulted in more multipliers and, therefore, more noise.

References

[1] W.-S. Lu and A. Antoniou, "Synthesis of 2-D state-space fixed-point digital-filter structures with minimum roundoff noise," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 965–973, Oct. 1986.

[2] T. Hinamoto, T. Hamanaka, and S. Maekawa, "A generalized study on the synthesis of 2-D state-space digital filters with minimum roundoff noise," *IEEE Trans. Circuits Syst.*, vol. CAS-35, pp. 1037–1042, Aug. 1988.

[3] T. Lin, M. Kawamata, and T. Higuchi, "A unified study on the roundoff noise in 2-D state-space digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 724–730, July 1986.

A Super-Parallel Sorting Algorithm Based on Neural Networks

YOSHIYASU TAKEFUJI AND KUO-CHUN LEE

***Abstract*—A new neural network parallel algorithm for sorting problems is presented in this paper. The proposed algorithm using $O(n^2)$ processors requires two and only two steps, not depending on the size of the problem, while the conventional parallel sorting algorithm using $O(n)$ processors proposed by Leighton needs the computation time $O(\log n)$. A set of simulation results substantiates the proposed algorithm. The hardware system based on the proposed parallel algorithm is also presented in this paper.**

I. Introduction

Sorting is one of the fundamental operations in computer science and engineering. The conventional sequential algorithm based on comparisons of pairs of elements must have complexity: $O(n \log n)$. In 1968, Batcher introduced a parallel sorting algorithm with time complexity $O(\log^2 n)$ using $2^{k-2}k(k+1)$ comparators where $n = 2^k$ are the unsorted elements [1]. Leighton shows the algorithm using n processors in time $O(\log n)$ [2]. Alon and Azar have claimed $\Theta(\log n / \log(1 + p/n))$ algorithms using p processors [3].

Since the advent of VLSI technology, the hardware cost has become negligible. In this paper a new parallel distributed algorithm to sort a list of n unsorted elements is presented. The algorithm using $O(n^2)$ processors requires two and only two iteration steps regardless of the size of the problem.

Processors used in the new algorithm are called neurons (where they perform the function of a simplified biological neuron), or binary neurons. Binary neurons have been successfully used for solving graph planarization problems [4] and tiling problems [5]. The output of the binary neurons is given by

$$V_i = f(U_i) = \begin{cases} 1, & \text{if } U_i > 0 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where V_i is the output of the ith neuron and U_i is the input to the ith neuron. Interconnection weights, called synaptic weights, between the neurons are determined by the predefined energy function $E(V)$, which describes the penalty quantity where V is an n-dimensional vector: $V = (V_1, V_2, \cdots, V_n)$.

Before discussing the details of our method, first we are going to show how the neural network can perform the parallel gradient descent method. As long as the motion equation of the binary neurons is given by $dU_i/dt = -dE/dV_i$, the predefined energy function E monotonically decreases. The following proofs claim that the state of our neural network is guaranteed to converge to the local minimum under the discrete numerical simulation.

Proofs:

$$\frac{dE}{dt} = \sum_i \frac{dV_i}{dt}\frac{dE}{dV_i} = \sum_i \frac{dV_i}{dt}\left(-\frac{dU_i}{dt}\right),$$

where dE/dV_i is replaced by $\left(-\frac{dU_i}{dt}\right)$

$$= -\sum_i \left(\frac{dU_i}{dt}\frac{dV_i}{dU_i}\right)\left(\frac{dU_i}{dt}\right) = -\sum_i \left(\frac{dV_i}{dU_i}\right)\left(\frac{dU_i}{dt}\right)^2.$$

Let $dV_i(t)/dU_i(t)$ be $(V_i(t+\Delta t) - V_i(t))/(U_i(t+\Delta t) - U_i(t))$. Let $dU_i(t)/dt$ be $(U_i(t+\Delta t) - U_i(t))$. It is necessary and sufficient to consider the following seven cases:

1) $U_i(t+\Delta t) > U_i(t)$, $U_i(t+\Delta t) < 0$, and $U_i(t) < 0$
2) $U_i(t+\Delta t) > U_i(t)$, $U_i(t+\Delta t) \geq 0$, and $U_i(t) < 0$
3) $U_i(t+\Delta t) > U_i(t)$, $U_i(t+\Delta t) > 0$, and $U_i(t) \geq 0$
4) $U_i(t+\Delta t) < U_i(t)$, $U_i(t+\Delta t) \geq 0$, and $U_i(t) > 0$
5) $U_i(t+\Delta t) < U_i(t)$, $U_i(t+\Delta t) < 0$, and $U_i(t) \geq 0$
6) $U_i(t+\Delta t) < U_i(t)$, $U_i(t+\Delta t) < 0$, and $U_i(t) < 0$
7) $U_i(t+\Delta t) = U_i(t)$.

If Condition 7) is satisfied, then $dU_i/dt = 0$ must be zero so that $dE/dt = 0$.

Manuscript received April 18, 1989. This work was supported by the National Science Foundation under Grant MIP-8902819. This paper was recommended by Associate Editor M. Ilic.

The authors are with the Department of Electrical Engineering, Center for Automation and Intelligent Systems Research, Case Western Reserve University, Cleveland, OH 44106.

IEEE Log Number 9038550.