# Programmable Switched-Capacitor Neural Network for MVDR Beamforming

Wen-Hao Yang and Po-Rong Chang, *Member, IEEE*

*Abstract*— **In this paper, a real-time adaptive antenna array based on a neural network approach is presented. Since an array operating in a nonstationary environment requires a programmable synaptic weight matrix for the neural network, the switched-capacitor (SC) circuits with the capability of programmability and reconfigurability is conducted to implement the neural-based adaptive array. Moreover, the SC techniques can directly implement the neural network with less chip area and provide the ratio of SC-equivalent resistors with accuracy of 0.1 percent. Programming of the switched-capacitor values could be made by allocating each synaptic weight to a set of parallel capacitors with values in a digitally programmable capacitor array (PCA). A relatively wide range of values (5 to 10 binary bits resolution) can be realized for each synaptic weight. A simulation tool called SWITCAP is used to verify the validity and performance of the proposed implementation. Experimental results show that the computation time of solving a linear array of 5 elements is about 0.1 ns for 1 ns time constant and is independent of signal power levels.**

## I. Introduction

THE purpose of adaptive arrays is to suppress unwanted jamming interferences and to produce the optimal beamformer response which contains minimal contributions due to noise. The most commonly employed technique for deriving the adaptive weights uses a closed-loop gradient descent algorithm where the weight updates are derived from estimates of the correlation between the signal in each channel and the summed output of the array [1], [2]. The fundamental limitation for this technique is one of poor convergence for a broad dynamic range signal environment. Several different approaches for choosing optimum beamformer weights are summarized in [3]. In many applications, none of those approaches is satisfactory. The desired signal may be of unknown strength and may always be present, resulting in signal cancellation with the multiple sidelobe canceller and preventing estimation of signal and noise covariance matrices in the maximum SNR processor. These limitations can be overcome through the application of linear constraints to the weights. The basic concept of linearly constrained minimum variance (LCMV) beamforming is to constrain the response of the beamformer such that the desired signals are passed with specified gain and phase. The weights are chosen to minimize output power subject to the response constraint. When the beamformer has unity response in the look direction, the LCMV problem would become the minimum variance distortionless response (MVDR) beamformer problem which is a very general approach employed to control beamformer output.

The complex weights of an MVDR-based beamformer should be updated in real time in order to respond to the rapid time-varying environment. Meanwhile, the calculation of weights is computationally intensive and can hardly meet the real-time requirement. Systolic implementations of optimum beamformers have been studied to improve the computational speed by a number of investigators [4], [5]. As an alternative to the digital approach, an analog approach based on Hopfield-type neural networks could operate at much higher speed and requires less hardware than digital implementation.

Recently, Tank and Hopfield [6] have shown how a set of neural networks with symmetric connections between neurons presents a dynamics that leads to the optimization of a quadratic functional. Most recently, Chua and Lin [7] and Kennedy and Chua [8], [9] extended the design of Hopfield network and introduced a canonical nonlinear programming circuit which is able to handle more general optimization problems. They showed that a canonical neural network assigned to solve the optimization problem would reach a solution in a time determined by $RC$ time constant, not by algorithmic time complexity. Therefore, the convergence speed of reaching the optimal solution is dramatically improved. Chang, Yang, and Chan [10] showed that an MVDR-based neural analog circuit is able to quickly attain its optimal performance, and works satisfactorily under the stringent environment of strong jammers. The primary concern of the neural circuit using $RC$ design is that an array operating in a nonstationary environment requires a programmable synaptic weight matrix for the neural network. The $RC$-active design is not the best suited for monolithic implementation, especially taking into account that accurate and wide dynamic range resistors and large $RC$ values are required. In this paper, we try to overcome this drawback by focusing on the design of implementing the MVDR beamforming neural network using switched-capacitor techniques. The inherent programmability and reconfigurability of switched-capacitor circuits together with the maturity of this technique [12] in the field of analog VLSI would be conducted to improve the conditional design. We use the dynamically and digitally programmable capacitor array (PCA's) [11], [13] to realize the time-varying synaptic weight matrix on a real-time basis. Moreover, a relatively wide range of values (5 to 10 binary bits resolution) can be realized for each synaptic weight.

In this paper, we first introduce the real-valued quadratic optimization problem derived from MVDR beamforming problem. Moreover, a discrete-time neural circuit dynamic equation

Manuscript received March 1, 1995; revised July 30, 1995. This work was supported in part by the National Science Council, Republic of China.

The authors are with the Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan, R.O.C.

Publisher Item Identifier S 0364-9059(96)00795-9.

for the quadratic optimization is established. In Section IV, we propose a switched-capacitor realization for implementing the system dynamics. In addition, the programmable capacitor array (PCA) is introduced to match the time-varying covariance matrix of the adaptive antenna array. Experimental results are shown in Section V, and conclusions are given in the last section.

## II. Problem Formulation

For a linear array composed of $L$ isotropic antenna elements which receive narrow-band signals from sources of variation frequency $f_0$ located far from the array, $x_l(t)$ is defined as a complex output of the $l$th element at the sampling time $t$, and can be expressed as

$$x_l(t) = m(t)e^{j2\pi f_0(t+\tau_l(\theta,\phi))} + n_l(t) + x_{Il}(t) \tag{1}$$

where

$$\tau_l(\theta, \phi) = \frac{\boldsymbol{r}_l \cdot \hat{s}(\theta, \phi)}{c} \tag{2}$$

is the time delay of the $l$th element relative to a reference point chosen at origin. $\boldsymbol{r}_l$ is the position vector of the $l$th element. $\hat{\mathrm{s}}(\theta, \phi)$ is a unit vector in the direction $(\theta, \phi)$ of the source, and $c$ is the propagation speed of the plane wave in free space.

The source amplitude $m(t)$ is characterized statistically by

$$E[m(t)] = 0 \tag{3}$$

$$E[m(t)m^*(t)] = p_s \tag{4}$$

where $E[\cdot]$ is the expectation operator, $p_s$ is the power of the source, and * denotes the complex conjugate. $x_{Il}(t)$ is the component of the directional interferences received by the $l$th element and possesses the same statistics as the source. In addition, $n_l(t)$ is a white random noise with properties

$$E[n_l(t)] = 0, \qquad l = 1, 2, \cdots, L \tag{5}$$

$$E[n_l(t)n_k^*(t)] = \sigma_n^2\delta_{lk}, \qquad l, k = 1, 2, \cdots, L. \tag{6}$$

Let the signal waveforms derived from the $L$ elements of a beamformer be represented by an $L$-dimensional complex vector

$$x \overset{\text{def}}{=} [x_1, x_2, \cdots, x_L]^T \tag{7}$$

and the weights of element outputs be represented by $L$-dimensional complex vector $\boldsymbol{w}$,

$$\boldsymbol{w} \overset{\text{def}}{=} [w_1, w_2, \cdots, w_L]^T \tag{8}$$

where $T$ denotes the transpose of the vector. Then the output of the beamformer can be written as

$$y(t) = \sum_{l=1}^{L} w_l^* x_l(t) = \boldsymbol{w}^H \boldsymbol{x}(t) \tag{9}$$

where $H$ denotes the complex conjugate transpose of a vector.

Since each component of $\boldsymbol{x}(t)$ is modeled as a zero mean stationary process, the mean output power of the beamformer is given by

$$\begin{aligned} p(\boldsymbol{w}) &= E[y(t)y^*(t)] \\ &= \boldsymbol{w}^H \boldsymbol{R}\boldsymbol{w} \end{aligned} \tag{10}$$

where $\boldsymbol{R}$ is the array correlation matrix.

In order to achieve the optimal utilization of the mean output power of the beamformer, the weights are chosen based on the statistics of the data received at the array such that the output contains minimal influence due to noise as well as interference signals arriving from other directions. Different criteria exist for choosing optimum beamformer weights, which are summarized in [3]. A general approach called minimum variance distortionless response (MVDR) beamforming is to constrain the response of the beamformer so that the desired signals are passed with unit gain and the weights are chosen to minimize the output power subject to the required constraints. The MVDR beamforming problem is usually formulated as

$$\underset{\boldsymbol{w}}{\text{Min}} \quad \phi(\boldsymbol{w}) = \boldsymbol{w}^H \boldsymbol{R}\boldsymbol{w} \tag{11}$$

$$\text{Subject to} \quad \boldsymbol{w}^H \boldsymbol{s}_0 = 1 \tag{12}$$

where $\boldsymbol{s}_0$ is the steering vector associated with the look direction and is given by

$$\boldsymbol{s}_0 = \left[1, \exp\left(j\frac{2\pi d}{\lambda_0}\cos\theta_0\right), \cdots, \exp\left(j\frac{2\pi d}{\lambda_0}(L-1)\cos\theta_0\right)\right]^T \tag{13}$$

where $d$ is the element spacing, $\lambda_0$ the wavelength of the plane wave in free space, and $\theta_0$ the look direction angle (the angle between the axis of the linear array and the direction of the desired signal source).

The method of Lagrange multipliers can be used to solve (11) and (12) and resulting in

$$\hat{\boldsymbol{w}} = \frac{\boldsymbol{R}^{-1}\boldsymbol{s}_0}{\boldsymbol{s}_0^H \boldsymbol{R}^{-1}\boldsymbol{s}_0}. \tag{14}$$

Note that, in practice, the presence of uncorrelated noise ensures that $\boldsymbol{R}$ is invertible.

The MVDR beamforming problem defined in both (11) and (12) is indeed a complex-value constrained quadratic programming problem, which cannot be solved by neural network directly. In order to meet the requirement of neural-based optimizer, one should convert it into a real-value constrained quadratic programming formulation. Since $\boldsymbol{R}$ is a positive-definite Hermitian matrix, it has been shown [10] that the above complex-valued MVDR beamforming problem can be transformed to be a real canonical quadratic nonlinear programming problem with linear equality constraints as follows:

$$\underset{\boldsymbol{v}}{\text{Min}} \quad \phi(\boldsymbol{v}) = \frac{1}{2}\boldsymbol{v}^T \boldsymbol{G}\boldsymbol{v} \tag{15}$$

$$\text{Subject to} \quad \boldsymbol{f}(\boldsymbol{v}) = \boldsymbol{B}\boldsymbol{v} - \boldsymbol{e} = 0 \tag{16}$$

where $\boldsymbol{v}$ is a $2L$-dimensional real weight vector, $\boldsymbol{G}$ is a $(2L) \times (2L)$ symmetric, positive-definite matrix, and $\boldsymbol{f}(\boldsymbol{v})$ is the constraint column vector. $\boldsymbol{B}$ and $\boldsymbol{e}$ are $2 \times (2L)$ matrix and $(2 \times 1)$ column vector. They are defined as follows:

$$\boldsymbol{v} = \begin{bmatrix} \boldsymbol{w}_r \\ \boldsymbol{w}_i \end{bmatrix} \tag{17}$$

$$\boldsymbol{G} = \begin{bmatrix} 2\boldsymbol{R}_r & -2\boldsymbol{R}_i \\ 2\boldsymbol{R}_i & 2\boldsymbol{R}_r \end{bmatrix} \tag{18}$$

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{s}_{0_r}^T & \boldsymbol{s}_{0_i}^T \\ -\boldsymbol{s}_{0_i}^T & \boldsymbol{s}_{0_r}^T \end{bmatrix} \tag{19}$$

and

$$\boldsymbol{e} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{20}$$

where $\boldsymbol{w}_r$, $\boldsymbol{R}_r$, $\boldsymbol{s}_{0r}$ and $\boldsymbol{w}_i$, $\boldsymbol{R}_i$, $\boldsymbol{s}_{0i}$ are the real parts and imaginary parts of $\boldsymbol{w}$, $\boldsymbol{R}$, and $\boldsymbol{s}_0$, respectively.

## III. A Hopfield Neural Network Approach to the MVDR Beamforming Problem

To solve the constrained nonlinear programming problem defined in (15) and (16), we convert it to an equivalent unconstrained problem. The way to do this is to define a pseudo-cost function $E(\boldsymbol{v})$ as follows:

$$E(\boldsymbol{v}) = \alpha\phi(\boldsymbol{v}) + \mu P(\boldsymbol{v}) \tag{21}$$

where $\phi(\boldsymbol{v})$ is the original cost function, $P(\boldsymbol{v})$ is referred to as the penalty function, and $\alpha$ and $\mu$ are called the acceleration factor and the penalty multiplier, respectively.

A valid penalty function must monotonically increase as the constraints $f_j(\boldsymbol{v})$'s deviate from the feasible region, which is the subspace of the multidimensional space defined by constraints. In general, the absolute value or the square operator fulfills this requisite. In this paper, the square operator is employed and the penalty function is defined by

$$P(\boldsymbol{v}) = \frac{1}{2}\sum_{j=1}^{2} f_j^2(\boldsymbol{v}). \tag{22}$$

It is interesting to note that the pseudo-cost function $E(\boldsymbol{v})$ can be identified as energy function of the circuit with the system of equations [10]

$$C\frac{d\boldsymbol{v}}{dt} = -\nabla E(\boldsymbol{v}), \qquad C > 0 \tag{23}$$

or

$$C\frac{dv_k}{dt} = -\alpha\frac{\partial\phi}{\partial v_k} - \mu\sum_{j=1}^{2} i_j\frac{\partial f_j}{\partial v_k}, \qquad k = 1, 2, \cdots, 2L \tag{24}$$

where $i_j = f_j(\boldsymbol{v})$.

The equilibrium point of the gradient system coincides with either a local extreme (minimum or maximum) or inflection point of $E(\boldsymbol{v})$. However, since $C > 0$, the time evolution of $\boldsymbol{v}$ will result in $E(\boldsymbol{v})$ decreased monotonically. Therefore, the circuit would seek a local minimum of the pseudo-cost function $E(\boldsymbol{v})$. Since the Hessian matrix of $E(\boldsymbol{v})$ is positive definite throughout the feasible region, it is shown in [14] that any local minimum of $E(\boldsymbol{v})$ is a global minimum over this feasible region. As a result, the circuit solution tends to a global minimum of the original cost function $\phi(\boldsymbol{v})$ when $dE/dt = 0$.

The MVDR-based neural network would include two particular modules. The first module is called the variable amplifier, which can perform the integral of a sum of input currents $(-\alpha\partial\phi/\partial v_k)$ and $(-\mu i_j\partial f_j/\partial v_k)$, and then produces the desired output variable $v_k$. Fig. 1(a) shows the circuit implementation of the variable amplifier consisting of an integrator and a unity gain inverting amplifier. The inverting amplifier can provide an output of the variable $-v_k$ which is required for the circuit implementation of negative weights. The second module is called the constraint amplifier, which is used to perform the constraint satisfaction function $f_j(\cdot)$. The circuit realization is shown in Fig. 1(b). Without loss of generality, the penalty multiplier $\mu$ may be included in $g_j(\cdot)$. Thus, the circuit yields the input–output relation: $O = -\mu I$, where $\mu$ represents the magnitude of the resistance, and $O$ and $I$ are an output voltage and an input current, respectively. If the input current $I$ is equal to $-f_j(\boldsymbol{v})$, then $O = -\mu(-f_j(\boldsymbol{v})) = \mu f_j(\boldsymbol{v}) = g_j(f_j(\boldsymbol{v})) = i_j$.

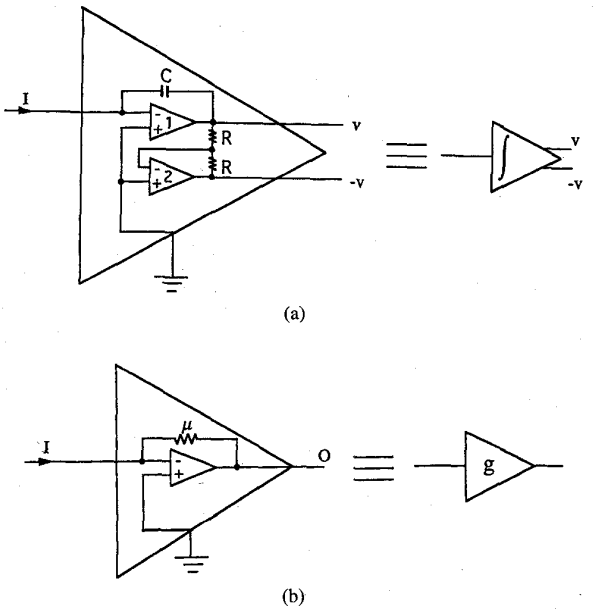



Fig. 1. Basic blocks of neural circuit using $RC$-design: (a) variable amplifier, (b) constraint amplifier.

Looking upon the circuit system dynamics of (24), the controlled current source $\partial\phi/\partial v_k$ and conductance $\partial f_j/\partial v_k$ are given by

$$\frac{\partial\phi(\boldsymbol{v})}{\partial v_k} = \sum_{i=1}^{2L} g_{ki}v_i \tag{25}$$

and

$$\frac{\partial f_j(\boldsymbol{v})}{\partial v_k} = b_{jk} \tag{26}$$

where $g_{ki}$ and $b_{jk}$ are the $(k, i)$ and $(j, k)$ entries of $\boldsymbol{G}$ and $\boldsymbol{B}$, respectively.

Equation (25) shows that the input current $\partial\phi/\partial v_k$ is a linear sum of the $v_i$'s weighted by conductances $g_{ki}$'s. In the case of linear constraints, the weights $\partial f_j(\boldsymbol{v})/\partial v_k$ are constants, and so may be implemented directly as conductances. Combining (24)–(26), one would obtain the state equations to the circuit implementation as

$$i_j = g_j(f_j(\boldsymbol{v})) = \mu\left(\sum_{i=1}^{2L} b_{ji}v_i - e_j\right) \tag{27}$$

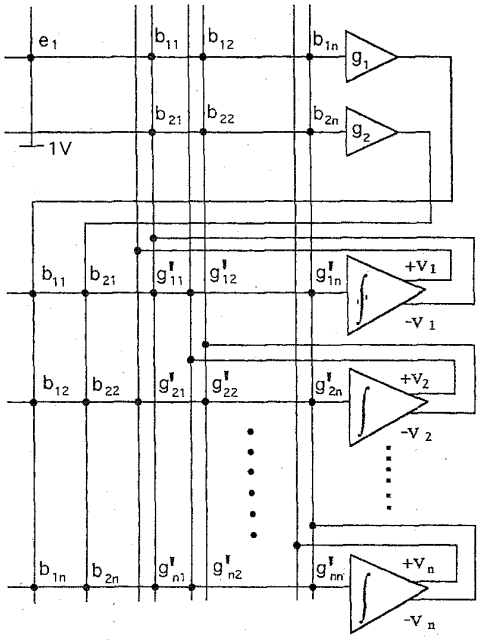



Fig. 2. Schematic diagram of neural circuit for MVDR-based beamforming problem using $RC$-design.

and

$$C\frac{dv_k}{dt} = -\sum_{i=1}^{2L}(\alpha g_{ki})v_i - \sum_{j=1}^{2} i_j b_{jk} = -\sum_{i=1}^{2L} g'_{ki}v_i - \sum_{j=1}^{2} i_j b_{jk}. \quad (28)$$

Note that the acceleration factor $\alpha$ is included in $g_{ki}$'s, and then $g'_{ki}$ is defined as $(\alpha g_{ki})$.

According to (27) and (28), a circuit realization is shown in Fig. 2. It should be noted that the elements of the $\boldsymbol{e}$, $\boldsymbol{G}'$, and $\boldsymbol{B}$ matrices are realized directly as resistive connections, and that their associated matrix entries correspond to conductance values.

## IV. Switched-Capacitor Realization for Neural-Based MVDR Beamforming

In the realization of the Tank and Hopfield neural network [6], they use the conventional $RC$-active design techniques. The main drawbacks arise for accurate resistors and large $RC$ values which are required for $RC$-active design. This implies that $RC$-active design is not the best suited for monolithic implementation. In this paper, we try to overcome this drawback by focusing on the design of implementing the MVDR-based neural network using switched-capacitor techniques. The inherent programmability and reconfigurability of switched-capacitor circuits together with the maturity of this technique [12], [13] in the field of analog VLSI would be conducted to improve the performance of conventional design. The switched-capacitor implementation (as shown in Fig. 3) is obtained by replacing the basic switched-capacitor modules into the $RC$-based neural circuit architecture in Fig. 2. The details about the schematic realization of each basic module will be discussed in the following.

### A. A Mapping Between Resistor and Switched-Capacitor Neural Network

In order to implement the MVDR-based neural network using switched-capacitor circuits, the circuit equations of (28) should be discretized by the forward-Euler formula, and are given by

$$v_i(n+1) = v_i(n) - \frac{T}{C}\left[\sum_{j=1}^{2L} g'_{ij}v_j(n) + \mu\sum_{j=1}^{2} b_{ji}f_j(n)\right], \quad i = 1, 2, \cdots, 2L \quad (29)$$

where $u_i(n) = u_i(t)|_{t=nT}$, and

$$f_j(n) = f_j(t)|_{t=nT} = \sum_{i=1}^{2L} b_{ji}v_i(n) - e_j \quad (30)$$

and $T$ denotes the period of the clocks which control the switches.

According to (29) and (30), Fig. 3 shows the schematic realization of switched-capacitor implementation for MVDR-based nonlinear programming problem. The SC circuit is on basis of the $RC$-active circuit architecture shown in Fig. 2 whose resistive elements are instead of capacitors and switches. Moreover, the details of the proposed SC circuit would be discussed in Subsection B.

Basically, the general SC circuit is composed of capacitors, op amps, and switches. The switches are controlled by two nonoverlapping clock phases S1 and S2. As shown in Fig. 3, there are three basic modules involved in the SC implementation. They are integrator, inverter, and summer. The schematic realization of each basic module would be discussed as follows.

*1) Switched-Capacitor Integrator:* The parasitic-insensitive SC integrator (variable amplifier) with multiple inputs is shown in Fig. 4(a). For the sake of describing the function of the circuit briefly, we consider a single-input SC integrator first. During the $n$th clock phase S1, the input capacitor $C_1$ is charged to the input voltage $v_1(n)$, while the integrator capacitor $C_f$ is held at $v_0(n-1)$. When $C_1$ is grounded during the clock phase S2, the charge $C_1v_1(n)$ from $C_1$ will be transferred to the capacitor $C_f$. The ideal output voltage $v_0(n)$ during clock phase S2 is

$$v_0(n) = v_0(n-1) + \frac{C_1}{C_f}v_1(n). \quad (31)$$

The voltage across the auxiliary hold capacitor $C_h$ compensates the offset voltage and dc gain error of the op amp. The recursive realization of (31) results in the first-order approximation to the integrator

$$v_0(n) = v_0(0) + \left[\sum_{i=1}^{n}\frac{C_1}{C_f}v_1(i)\right] \approx v_0(0) + \frac{1}{T}\int_0^{nT}\frac{C_1}{C_f}v_1(t)\,dt. \quad (32)$$

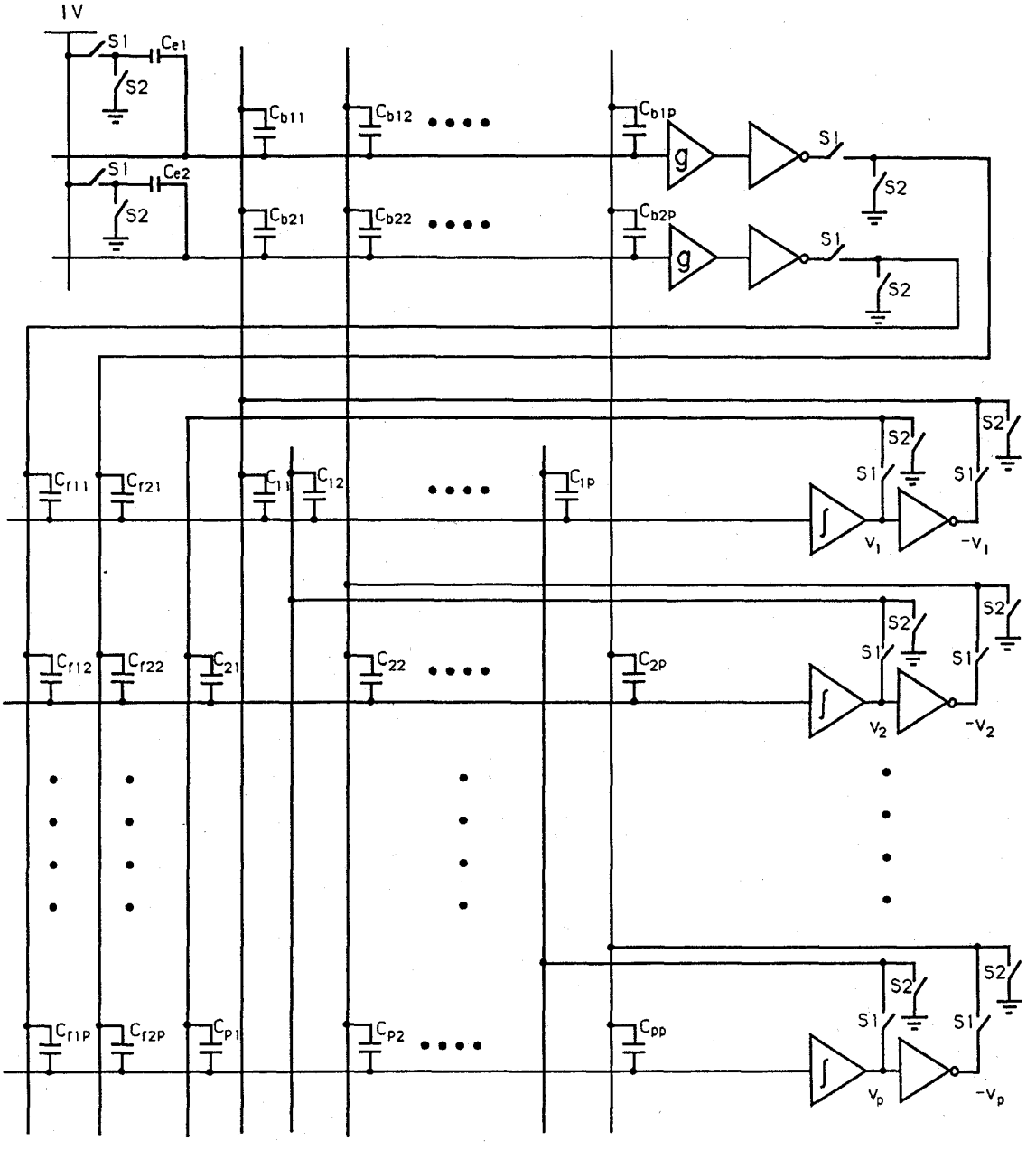



Fig. 3. A switched-capacitor neural circuit implementation for MVDR-based beamforming problem.

Similarly, the output voltage of multiple-input integrator shown in Fig. 4(a) during clock phase S2 is characterized by

$$v_0(n) = v_0(n-1) + \sum_{i=1}^{p} \frac{C_i}{C_f} v_i(n). \tag{33}$$

*2) Switched-Capacitor Inverter and Summer:* Since the negative weights are required in the beamforming problem, an SC inverter is used to achieve the negative terminal of the neuron's output voltage instead of the negative weights. From a precision SC inverter shown in Fig. 4(b), during the $n$th clock phase S1, the input capacitor $C_i$ is charged to $v_i(n)$, and then the output voltage becomes a scaled value of input voltage

$$v_0(n) = -\frac{C_i}{C_f} v_i(n). \tag{34}$$

At the mean time, the hold capacitor $C_h$ is charged to $v_0(n)$, while the output voltage of the inverter is kept at constant value during the full $n$th clock period. During the clock phase S2, the charges at both $C_i$ and $C_f$ are discharged to suppress the offset voltage of the op amp. This also reduces the possibility of the accumulated error due to the residued charges at $C_i$ and $C_f$.

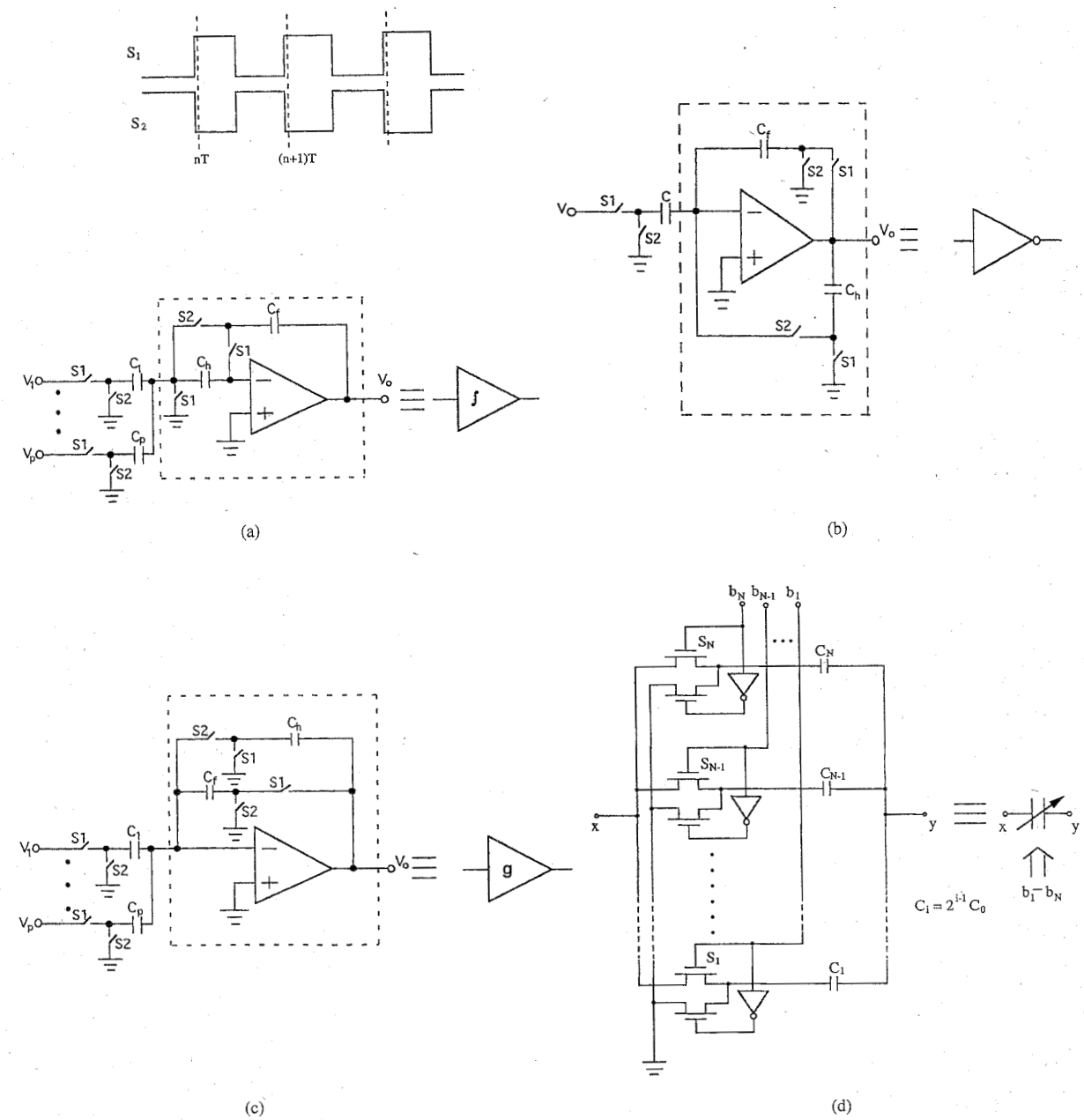



Fig. 4. The required clock signals and basic switched-capacitor blocks: (a) a parasitic-insensitive integrator (neuron); (b) a parasitic-insensitive inverter; (c) a parasitic-insensitive summer; (d) digitally programmable capacitor array (PCA) and simplified notation.

The schematics of a multiple-input SC inverting summer is shown in Fig. 4(c). The inverting summer produces an output voltage $v_0(n)$ at the $n$th clock period by

$$v_0(n) = -\sum_{j=1}^{p} \frac{C_j}{C_f} v_j(n). \tag{35}$$

*3) Programmable Capacitor Array:* Since the synaptic weight matrix for the MVDR-based neural network should be programmable when the array operates in a nonstationary environment, this subsection presents digitally programmable capacitor arrays (PCA's) which can be programmed on a real-time basis. The PCA's can be realized using a number of binary weighted capacitors. Fig. 4(d) shows several capacitors $(C_1, C_2, \cdots, C_N)$, each connected by a series switch $(S_1, S_2, \cdots, S_N)$ to a common node. The digital control word $(b_1, b_2, \cdots, b_N)$ determines whether an individual capacitor is connected with or disconnected from the circuit. If the bit $b_i$ has the logical value 1, then the switch $S_i$ on the capacitor $C_i$ is connected between nodes $x$ and $y$. Usually, the capacitances are weighted by powers of two (i.e., $C_1 = C_0$, $C_2 = 2C_0, \cdots, C_N = 2^{N-1}C_0$). It should be noted that the capacitor never floats, and the total capacitance loading node $y$ is constant. The value of the capacitance

between $x$ and $y$ in the $N$-bit PCA is given by

$$C_t = \sum_{i=1}^{N} b_i C_i = C_0 \sum_{i=1}^{N} 2^{i-1} b_i \tag{36}$$

where $C_0$ is the capacitance basis of PCA, and the practical value of $N$ can be up to ten.

### *B. Determination of the Capacitance Values in SC Neural Networks*

In this subsection, it is desired to design the MVDR-based neural circuit on the basis of the above-mentioned SC modules. The proposed SC neural network for MVDR beamforming is shown in Fig. 3. Next, we would like to validate our circuit which can exactly implement MVDR-based neural network. During the $n$th clock phase S1, the inverted outputs of the variable amplifiers (neurons), $v_i(n)$'s, are sent to the inputs of the constraint summers with associated weights. Thus, the $j$th summer's output is given by

$$g_j(n) = \left[ \sum_{i=1}^{2L} \frac{C_{bji}}{C_f} v_i(n) - \frac{C_{ej}}{C_f} \right] \tag{37}$$

where the capacitances $C_{ej}$ and $C_{bji}$ are connected to the constant voltage source 1 V and the output terminal of the $i$th variable amplifier, respectively. Note that $C_f$ is a constant capacitance of constraint amplifier.

Comparing (30) and (37), it can be shown that both equations become identical when $b_{ji} = (C_{bji}/C_f)$ and $e_j = (C_{ej}/C_f)$. During the clock phase S2, the charges at both capacitors, $C_{fji}$'s and $C_{ij}$'s, are transferred to the capacitor $C_f$ of the $i$th neuron or variable integrator amplifier. Since $C_{ij}$ and $C_{fji}$ are connected to the output terminal of the $i$th variable amplifier and the output of the $j$th constraint amplifier, respectively, the current flowing into the $i$th variable amplifier is obtained by

$$I_i(n) = \frac{2}{T} \left[ \sum_{j=1}^{2L} C_{ij} v_j(n) + \sum_{j=1}^{2} C_{fji} f_j(n) \right]. \tag{38}$$

The ideal output of the $i$th variable amplifier is the integral of the input current during clock phase S2. During the $n$th clock phase S1, the transferred charge and incremental output voltage can be obtained according to the law of the conservation of charges and are given by

$$Q_t(n) = I_i(n) \frac{T}{2} \tag{39}$$

$$\Delta v_i(n) = \frac{Q_t(n)}{C_f}. \tag{40}$$

Thus, the output voltage at $(n+1)$th clock phase becomes

$$\begin{aligned} v_i(n+1) &= v_i(n) + \frac{1}{C_f} \left[ I_i(n) \frac{T}{2} \right] \\ &= v_i(n) + \sum_{j=1}^{2L} \frac{C_{ij}}{C_f} v_j(n) + \sum_{j=1}^{2} \frac{C_{fji}}{C_f} f_j(n). \end{aligned} \tag{41}$$

Comparing (29) and (41), it is found that the proposed SC circuit can implement the MVDR-based neural network when the following conditions are satisfied, i.e.:

$$\frac{C_{ij}}{C_f} = -\frac{T}{C} \alpha g_{ij} \quad \text{and} \quad \frac{C_{fji}}{C_f} = -\frac{T}{C} \mu b_{ji}. \tag{42}$$

Notice that the negative values of both capacitance ratios represent the connection to the negative output terminal of the variable amplifier. From (25) and (26), it can be shown that $g_{ij}$ and $b_{ji}$ are dependent on the elements of the array correlation matrix and the steering vector, respectively. However, these parameters are not fixed when the array operates in a nonstationary environment. It requires the programming of their corresponding capacitance ratio values according to (42). Programming of the switched-capacitor ratio values could be made by allocating appropriate values to a set of parallel capacitors with values in PCA.

## V. Illustrated Examples

To verify the effectiveness of MVDR-based switched-capacitor neural circuit implementation, a linear array of five elements with half-wavelength spacing is considered in the following example. The variance of white noise present in each element is assumed to be equal to 0.1. In addition, there are two interference sources which fall in the mainlobe and the peak of the first sidelobe of the conventional uniform array pattern, respectively. The first interference makes an angle 55° ($\theta_1 = 55°$) with the line of the array and has the power which is taken to be 20 dB above the white noise power. The second interference makes an angle of 110° ($\theta_2 = 110°$) and has the power which is 10 dB more than the white noise power. The look direction of the signal is assumed to be orthogonal to the array. In our example, two signal power levels, 1 dB and 10 dB, are employed in the simulation.

The simulation of the MVDR-based switched-capacitor neural circuit is performed using the SWITCAP, which is a general simulation program for analyzing a switched-capacitor network [15]. The dynamics of the neural circuit is described by (29) and (30). The switching clock period is $10^{-13}$ s. The capacitance $C$ related to each variable amplifier of the $RC$ circuit and the capacitance $C_f$ of the switched-capacitor integrator are taken to be of the same value 1 pF. In addition, the penalty multiplier $\mu$ in each constraint amplifier and the acceleration factor $\alpha$ are taken to be 2 and 0.002, respectively. Fig. 5 shows the transient response of two output SNR's corresponding to their signal power levels 1 and 10 dB, respectively. It is noted that the converge time of each curve in Fig. 5 is almost independent of $p_s$ and equals 0.1 ns. Reference [10] showed that the converge time is characterized by the system time constant of the circuit. The comparisons of power patterns of a conventional uniform array and the resulting neural-based adaptive array are given in Fig. 6. It is observed that two sharp nulls presented in the pattern correspond to the directions of the interference sources, i.e., 55° and 110°. Furthermore, it should be mentioned that the values of capacitors $C_{ij}$'s and $C_{fji}$'s are dependent on the signal power level $p_s$ and its look direction, respectively. For example,

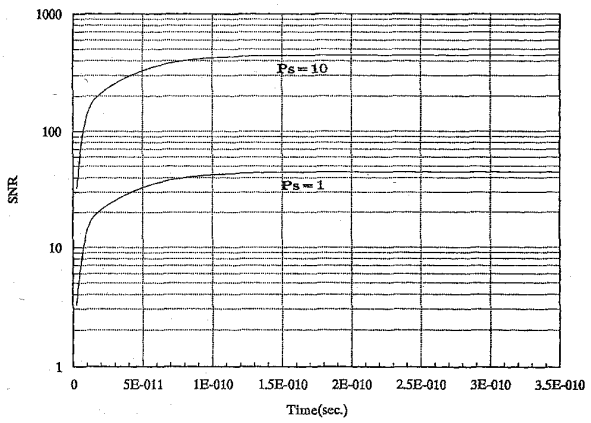



Fig. 5. The output SNR versus the response time for a five-element linear array.

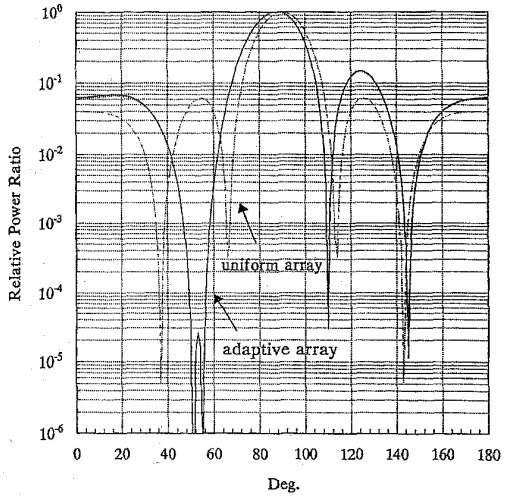



Fig. 6. Comparison of power patterns achieved by the conventional uniform array and neural-based array.

$C_{15}$ (or $C_{51}$) corresponding to their associated signal power levels 1 and 10 dB are 0.00265 and 0.00625, respectively. Fortunately, the adjustment of appropriate settings for the capacitors can be achieved by PCA's.

## VI. Conclusions

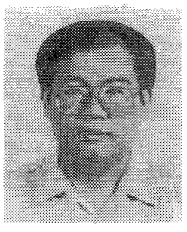
This paper presents the implementation of a digitally programmable analog neural adaptive array using SC integrated circuit techniques. The important advantage of SC circuits is that they can be digitally controlled on the synaptic weights to execute the programmable function by PCA's. The PCA's could obtain their capacitance values from binary numbers stored in the memory. A linear array of five elements with two interference sources is constructed accordingly to verify the performance of the proposed circuit. Simulation results using a tool called SWITCAP show that the MVDR-based neural circuit can solve this five-element array in 0.1 ns when the dominant time constant is 1 ns.

## References

[1] S. P. Applebaum, "Adaptive array," *IEEE Trans. Antennas Propagat.*, vol. AP-24, pp. 585–598, Sept. 1976.
[2] B. Widrow and S. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
[3] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Mag.*, pp. 4–24, Apr. 1988.
[4] J. G. McWhirter and T. J. Shepherd, "Systolic array processor for MVDR beamforming," *Proc. IEE*, vol. 136, Pt. F, no. 2, Apr. 1989.
[5] W. M. Gentleman and H. T. Kung, "Matrix triangularisation by systolic array," *Proc. SPIE, Real Time Signal Processing IV*, 298, 1981.
[6] D. W. Tank and J. J. Hopfield, "Simple 'neural' optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 533–541, May 1986.
[7] L. O. Chua and G. N. Lin, "Nonlinear programming without computation," *IEEE Trans. Circuits Syst.*, vol. CAS-31, pp. 182–188, Feb. 1984.
[8] M. P. Kennedy and L. O. Chua, "Unifying the Tank and Hopfield linear programming network and the canonical nonlinear programming circuit of Chua and Lin," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 210–214, Feb. 1987.
[9] ——, "Neural networks for nonlinear programming," *IEEE Trans. Circuits Syst.*, vol. CAS-35, pp. 554–562, May 1988.
[10] P. R. Chang, W. H. Yang, and K. K. Chan, "A neural network approach to MVDR beamforming problem," *IEEE Trans. Antennas Propagat.*, vol. 40, pp. 313–322, Mar. 1992.
[11] W. H. Yang, P. R. Chang, and K. K. Chan, "A programmable MVDR beamformer by switched-capacitor circuits," in *1993 Asia-Pacific Microwave Conf. Proc.*, vol. 1, Taiwan, R.O.C., Oct. 1993, pp. 1.68–1.73.
[12] A. Rodríguez-Vázquez *et al.*, "Nonlinear switched-capacitor 'neural' networks for optimization problems," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 384–398, Mar. 1990.
[13] R. Unbehauen and A. Cichocki, *MOS Switched-Capacitor and Continuous-Time Integrated Circuits and Systems*. New York: Springer-Verlag, 1989.
[14] D. G. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1984.
[15] *PC-SWITCAP Manual*, Columbia Univ., New York, 1987.

**Wen-Hao Yang** was born in Chyi, Taiwan, Republic of China, on June 12, 1964. He received the M.S. degree in communication engineering from the National Chiao-Tung University, Hsinchu, Taiwan in 1988, where he is currently pursuing the Ph.D. degree.

From 1984 to 1986 he fulfilled his military duty by serving in the Chinese Air Force. Since 1988 he has been an Assistant Research Engineer at the Telecommunication Laboratories, Ministry of Communications, Republic of China, where he works on satellite communication ground station establishment and microwave filter design. His research interests are in adaptive array, pattern synthesis, and microwave circuits.



**Po-Rong Chang** (M'87) received the B.S. degree in electrical engineering from National Tsing-Hua University, Taiwan, Republic of China, in 1980, the M.S. degree in telecommunication engineering from National Chiao-Tung University, Hsinchu, Taiwan, in 1982, and the Ph.D. degree from Purdue University, West Lafayette, IN, in 1988.

From 1982 to 1984, he was a lecturer in the Chinese Air Force Telecommunication and Electronics School for his two-year military service. From 1984 to 1985, he was an instructor of electrical engineering at National Taiwan Institute of Technology, Taipei, Taiwan. From 1989 to 1990, he was a project leader of SPARC chip design team at ERSO of Industrial Technology and Research Institute, Chu-Tung, Taiwan, Republic of China. Currently he is an Associate Professor of Communication Engineering at National Chiao-Tung University. His current interests include neural network, HDTV signal processing, and human visual and audio system.