

Design and Implementation of a Biologically Realistic Olfactory Cortex in Analog VLSI

JOSE C. PRINCIPE, FELLOW, IEEE, VITOR G. TAVARES, JOHN G. HARRIS, AND
WALTER J. FREEMAN, FELLOW, IEEE

Invited Paper

This paper reviews the problem of translating signals into symbols preserving maximally the information contained in the signal time structure. In this context, we motivate the use of nonconvergent dynamics for the signal to symbol translator. We then describe a biologically realistic model of the olfactory system proposed by Walter Freeman that has locally stable dynamics but is globally chaotic. We show how we can discretize Freeman's model using digital signal processing techniques, providing an alternative to the more conventional Runge-Kutta integration. This analysis leads to a direct mixed-signal (analog amplitude/discrete time) implementation of the dynamical building block that simplifies the implementation of the interconnect. We present results of simulations and measurements obtained from a fabricated analog very large scale integration (VLSI) chip.

Keywords—Analog VLSI implementation, digital simulation models, neural assemblies, nonlinear dynamics.

I. DESIGN OF SIGNAL TO SYMBOL TRANSLATORS

There are many important differences between biological and man-made information processing systems. Animals have goal-driven behavior and have explored inductive principles throughout the course of evolution to work reliably in a non-Gaussian, nonstationary, nonlinear world. Autonomous man-made systems with sensors and computational algorithms (animats) are still unable to match these capabilities. Engineered systems may be made faster, more accurate, but at the expense of specialization, which brings brittleness. Our present model of computation was inherited

from Turing and Von Neumann and is based on the theory of formal systems [26]. The formal model is invaluable and quite general for the purpose of symbolic processing [22]. However, we have to remember that symbols do not exist in the real world. The real world provides time varying signals, usually faint signals corrupted by noise. Hence, the critical issue for accurate and robust interpretation of real-world signals by animats is not only how to process symbols but also how to transform signals into symbols. The processing device that transforms signals into symbols is called here the signal-to-symbols translator (*S Σ T*). We can specify an optimal *S Σ T* as a device that is able to capture the goal-relevant information contained in the signal time structure and map it with as little excess irrelevant information as possible to a stable representation in the animat's computational framework.

There is no generally accepted theory to optimally design *S Σ T* for the unconstrained signals found in the real world. *S Σ T*s fall between two very important areas of research: symbolic and signal processing, which use very diverse tools and techniques. Symbolic manipulation is unable to deal with time signals, while signal processing techniques are ill-prepared to deal with symbols. Many hybrid approaches using the minimum description length principle [40] and optimal signal processing principles [42], pattern recognition [14], neural networks [41], or other machine learning approaches [28] have been proposed.

A framework where the *S Σ T*s is modeled as a dynamical system coupled to the external world seems a productive alternative. To bring a biological flavor, we will center the discussion in distributed, adaptive arrangements of nonlinear processing elements (PEs) called coupled lattices [28]. The appeal of these coupled lattices is that they have potentially very rich autonomous dynamics and the external world signals can directly influence the dynamics as forcing inputs. Von Neumann was the first to propose cellular automata, which are discrete-state and discrete-time coupled

Manuscript received January 1, 2000; revised February 1, 2001. This work was supported in part by the Office of Naval Research under Grant N00014-01-1-0405 and by the National Science Foundation under Grant ECS-9900394. The work of V. G. Tavares was supported by Fundacao para a Ciencia e Tecnologia and Fundacao Luso-Americana para o Desenvolvimento scholarships.

J. C. Principe, V. G. Tavares, and J. G. Harris are with the Computational NeuroEngineering Laboratory, University of Florida, Gainesville, FL 32611 USA.

W. J. Freeman is with the Department of Molecular and Cell Biology, University of California, Berkeley, CA 94720 USA.

Publisher Item Identifier S 0018-9219(01)05407-X.

lattices, as a paradigm for computation [52]. Grossberg [18] and later Hopfield [25] proposed dynamic content associative memories (DCAMs) as the simplest model for $S\Sigma T$ s by analogy with the Ising model of ferromagnetics. In order to contrast this method with statistical inference, we will briefly describe DCAM operation now. DCAMs are dynamical systems with point attractors in their dynamics that correspond to the memorized information. The input serves as an initial condition to the system state, and the stable dynamics relax to the closest fixed point, so the dynamical system's basins of attraction provide a natural similarity measure. Although mimicking statistical CAMs, the principle of operation of DCAMs are very different. DCAMs have limited capacity and display many spurious fixed points [24], so they are not directly usable as $S\Sigma T$ s. The reason DCAMs have a capacity limited to (a fraction of) the number of inputs is that their weights parameterize in a one-to-one manner the correspondence between the system attractors and the input patterns. This is not only a problem with the Hopfield model, but is equally shared by recurrent neural networks used for grammar inference (which have been proven Turing equivalent) [16], and other dynamic neural networks [8].

Although many other researchers have investigated dynamical principles to design and implement information processing systems (mainly in the biophysics [33], [37] and computational neuroscience communities [20]), this line of research is still a niche compared with the statistical approach. We are slowly realizing that the limited repertoire of dynamical behavior (fixed points) implemented by these DCAMs constrain their use as information processing devices for signals that carry information in their time structure. For instance, the point attractor has no dynamical memory (i.e., the system forgets all previous inputs when it reaches the fixed point) while the dynamic memory of the limit cycle is constrained to the period; only chaotic systems display long-term dynamic memory due to the sensitivity to initial conditions. This sensitivity carries the problem of susceptibility to noise, but a possible solution is to utilize a chaotic attractor created by a dynamical system with singularities of at least second order (third-order ODE). A chaotic attractor is still a stable representation, might exist in a high-dimensional space (much higher than the dimensionality of our three-dimensional world), and, more importantly, its dimensionality can be controlled by parameters of the $S\Sigma T$ s. Forseeably, such systems are capable of using the inner structure of trajectories within the attractor for very high information storage and rapid recall, but we still do not fully understand how to control the stability of recall, in particular, in the presence of noise.

Amazingly, signals with very complex time structure (consistent with chaotic generators) are measured from mammalian cortex during sensory processing [12]. Unlike existing computational systems, brains and the sensory cortices embedded within them are highly unstable. They jump from one quasi-stable state to the next at frame rates of 5–10/s under central control. The “code” of each frame is a 2-D spatial pattern of amplitude modulation (AM) of an aperiodic (chaotic) carrier wave. The AM contents of each frame are

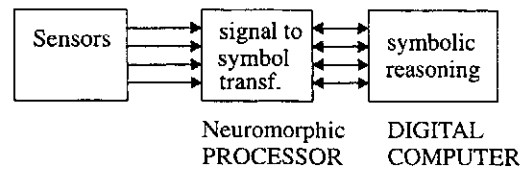


Fig. 1. Proposed computer architecture for real-world signal interpretation.

selected by input and shaped by synaptic connection weights embodying past learning. These frames are found in the visual, auditory, somatic, and olfactory cortices. Having the same “code,” they readily combine to form multisensory images (Gestalts) that provides a landscape of basins and attractors for learned classes of stimuli. Our thesis is that sensory cortices explore cooperatively the nonlinear dynamical behavior of neurons yielding global dynamics that are capable of codifying into spatial patterns minute changes perceived in the inputs. By creating “chaotic” dynamics, brains are offsetting their limitation of limited size with the richness of the cooperative behavior away from equilibrium conditions. This reasoning not only opens a new research direction to study computational models beyond symbolic processing, but also charts the construction of high-performance information processing systems for real-world signals.

Our aim is to construct a $S\Sigma T$ that operates in accordance with the neurodynamics of the cerebral cortex, and that has the sensitivity, selectivity, adaptiveness, speed, and tolerance of noise that characterizes human sensation. Fig. 1 shows our proposed methodology to attack the problem of interpreting real-world data. The processing chain is composed of a neuromorphic processor followed by a Turing machine (digital computer). The neuromorphic $S\Sigma T$ will accept signals and produce symbols as outputs, and will be the focus of this paper. The role of the neuromorphic $S\Sigma T$ is to serve as an interface between the infinite complexity of the real world and the countable infinite capacities of conventional symbolic information processors.

The problem is how to specify the properties and control the organization of high dimensional nonlinear (chaotic) dynamics to operate as an information processing system comparable to cortical systems.

II. FREEMAN’S COMPUTATIONAL MODEL

In a series of seminal papers spanning more than 25 years [11], [14], Freeman has laid down the basis for a biologically realistic computational model of the olfactory system. The brain is a vast pool of excitable cells with a dense interconnectivity that can be modeled as a spatio-temporal lattice of nonlinear processing elements (PEs) of the form

$$\frac{1}{a \cdot b} \cdot \left[\frac{d^2}{dt^2} x_i(t) + (a + b) \cdot \frac{d}{dt} x_i(t) + (a \cdot b) \cdot x_i(t) \right] = \sum_{j \neq i}^N [W_{ij} \cdot Q(x_j(t), q_j) + W'_{ij} \cdot f_j(Q(x_j(t), q_j), t)] + I_i(t) \quad i = 1, \dots, N \text{ (order)} \quad (1)$$

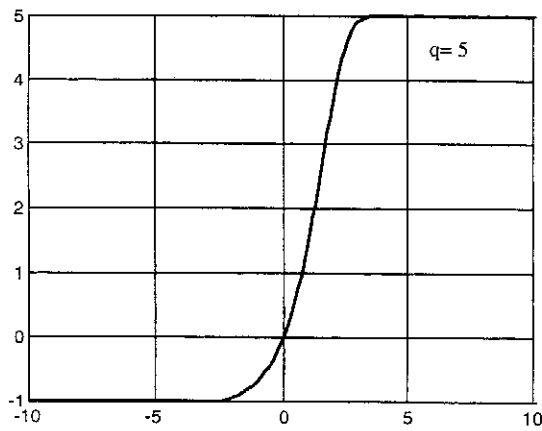


Fig. 2. Static nonlinear function sigmoidal characteristic.

where a and b are constants. Equation (1) presages Grossberg's additive model [19], and each PE is a second-order ordinary differential equation to take into consideration the independent dynamics of the wave density for the action dendrites and the pulse density for the parallel action of axons. No auto-feedback from the nonlinearity output $Q(x_i(t), q_j)$ is allowed. The operation at trigger zones is nonlinear (wave to pulse transformation) and is incorporated in the model through $Q(x_j(t), q_j)$, a static sigmoidal nonlinearity, which is nonsymmetric as opposed to the nonlinearities used in neurocomputing. This asymmetry is an important source of instability in the neural network because of asymmetric excitatory and inhibitory gain. $Q(x_j(t), q_j)$ is defined by (2) and plotted in Fig. 2 [12]. In the model, q_j is fixed at all levels to $q_j = 5$

$$Q(x(t), q) = \begin{cases} q(1 - e^{-(e^{x(t)} - 1)/q}), & \text{if } x(t) > x_0 \\ -1, & \text{if } x(t) < x_0 \end{cases} \Big|_{x_0 = \ln(1-q) \ln(1+1/q)} \quad (2)$$

A set of inputs may be prewired for temporal processing expressed by $f_j(\cdot, t)$ as in the gamma model [39]. Freeman's model is also an additive model because the topological parameters W_{ij} and W'_{ij} are independent of the input. The dynamical parameters a and b are real-time constants of two decaying eigenfunctions and are determined experimentally from brain tissue [11]. This is also true for all the other topological parameters. $I(t)$ is an external input.

If we follow the engineering literature, a neural network would be an arbitrary arrangement of these PEs. However, in the cortex there is a very tight relation between topological arrangement of neurons and function [44] that is totally missed in neurocomputing. Freeman's model incorporates neurophysiological knowledge by modeling the functional connection among groups of neurons, called neural assemblies. Although each neural assembly is built from thousands to millions of neurons, the functional connections have been catalogued in a hierarchy of topologies by Katchalsky [11]

among others. As a tribute to his contributions, the topologies are named K sets. The hierarchy levels are designated by K0, K1, K2, K3.

A. The K0 Set

The simplest structure in the hierarchy of sets is the K0 set. This set has the characteristic of having no functional interconnections among the elements of the neural assembly. The neural population in a K0 share common inputs and common sign for the outputs as shown in Fig. 3. It can accept several spatial inputs that are weighted and summed, and then convolved with a linear time invariant system defined by the left hand side of (1). The output state $x(t)$ resulting from the convolution is magnitude shaped by (2) to form the output. We will represent a K0 set by a square, as in Fig. 3. This is the basic building block necessary for the construction of all the other higher level structures. A K0 network is a parallel arrangement of K0 sets.

B. The K1 Network

The K1 set is formed from K0 units interconnected through lateral feedback of common sign; hence, we can have excitatory (denoted by + sign in Fig. 4) and inhibitory KI sets (denoted by a - sign). There is no auto-feedback. A number of KI sets connected only by forward lateral feedback connections also comprises a K1 network. Each connection represents a weight that can be constant or time varying and is obtained from neurophysiological measurements. These models are lumped circuit models of the dynamics.

C. The K2 Set and Network

The K2 set is built from at least two KI sets with dense functional interconnections [11]. The most interesting interconnectivity occurs when the K2 set is formed by 2KI (or 4 K0) sets with opposite polarity (i.e., two excitatory K0 and two inhibitory) (Fig. 5). K2 sets have fixed connecting weights, and they form oscillators with the parameter settings found in biology. The response measured at the output of the excitatory K0 to an impulse applied to its input is of a high-pass filter type (damped oscillation). However, with a sustained input it evolves into undamped oscillation, which vanishes after the input rests into a zero state. The K2 set is then an oscillator controlled by the input with an intrinsic frequency setup by the connection weights.

When the K2 sets are connected together they form a K2 network, which becomes a set of mutually coupled oscillators. Note that the excitatory K0 of each set is connected through weights to all the other excitatory K0 sets of the network. Likewise for the inhibitory K0 sets. The excitatory weights interconnecting the individual K2 sets in the network are adapted with Hebbian learning, while the inhibitory weights are fixed with a value obtained from physiological measurements [13].

The input to the K2 network is a vector of values, each applied to the excitatory K0 (named M1 in Fig. 5) in the K2 set. The output is spatially distributed over the network, as we discuss next. When an external excitation is applied to one

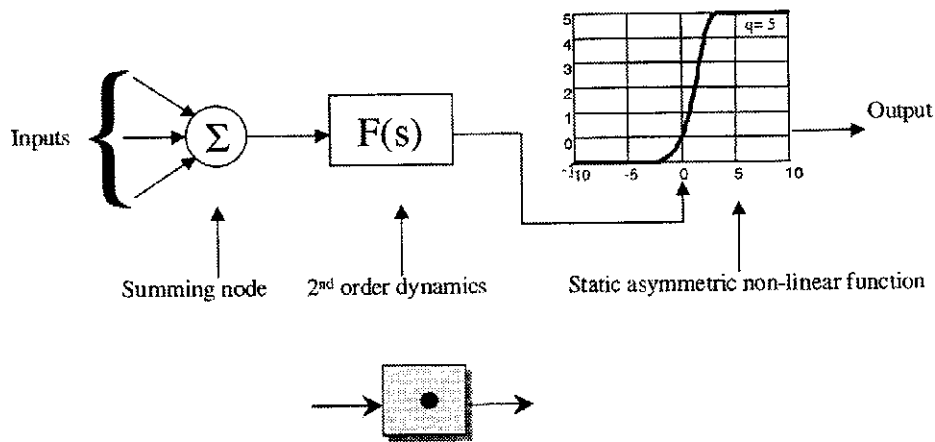


Fig. 3. The K0 set and its components.

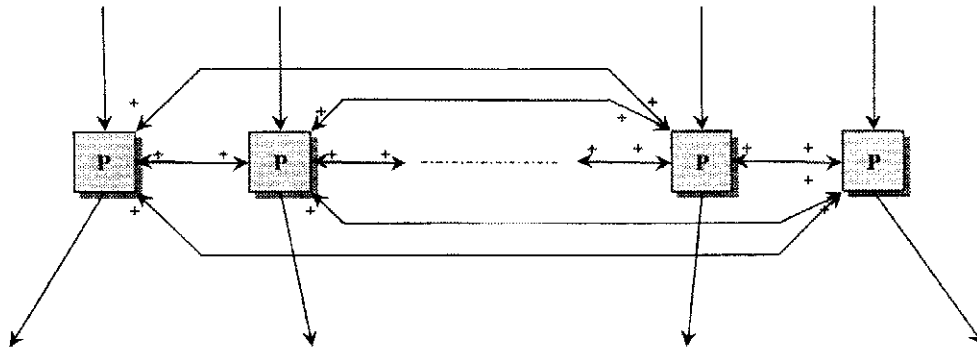


Fig. 4. The KI network made up of an arrangement of K0 sets (Periglomerular cells).

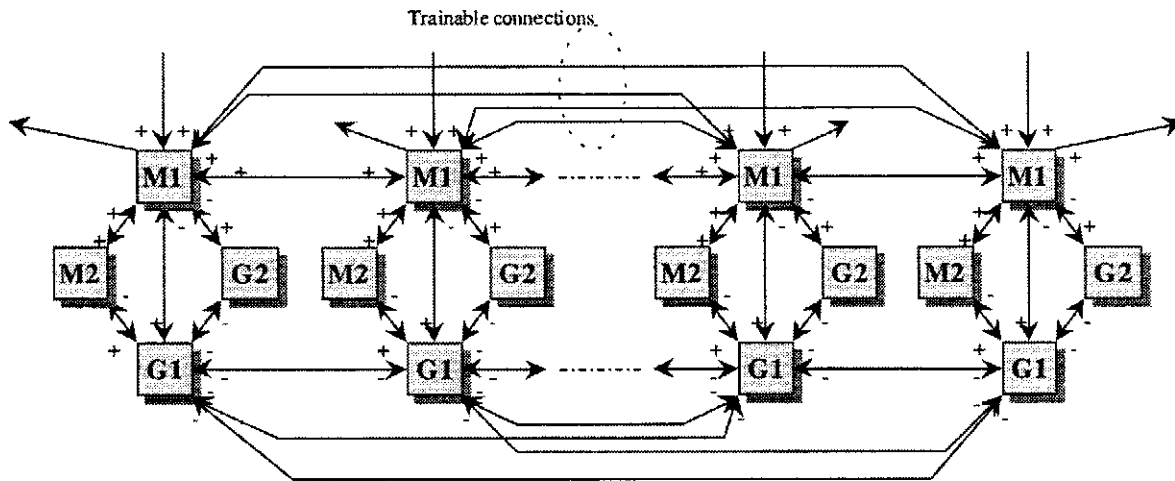


Fig. 5. KII network made up of tetrads of K0 sets. M: Mitral cells. G: Granular cells.

or more KII sets in the network, those sets tend to build up energy on the other sets through excitatory (positive) interconnections feedback. The oscillation magnitude of the sets that do not have any excitation will depend on the magnitude of the coupling gain defined by the interconnecting weights. If these interconnections are adapted to an input pattern, a distributed representation for that input is achieved. When a known pattern is applied to the system input after training, then a recognizable output oscillation pattern emerges across the KII sets. In this perspective, the KII network functions as an associative memory [30] that associates an input bit

pattern with the spatial distributed coupling that is stored in the interconnecting pathways, and that becomes "visible" from the outside as larger amplitude oscillations along the KII sets that have larger weights. The excitatory connections represent cooperative behavior while inhibitory connections introduce competitiveness into the system. They are crucial for contrast enhancement between the channels [13] (a channel is an input or output taken from a single set). For an "ON-OFF" (1/0) type of input pattern, the learning is accomplished with a modified Hebbian rule as follows [13]. If between any two input channels the product $I_j \cdot I_i$ is equal

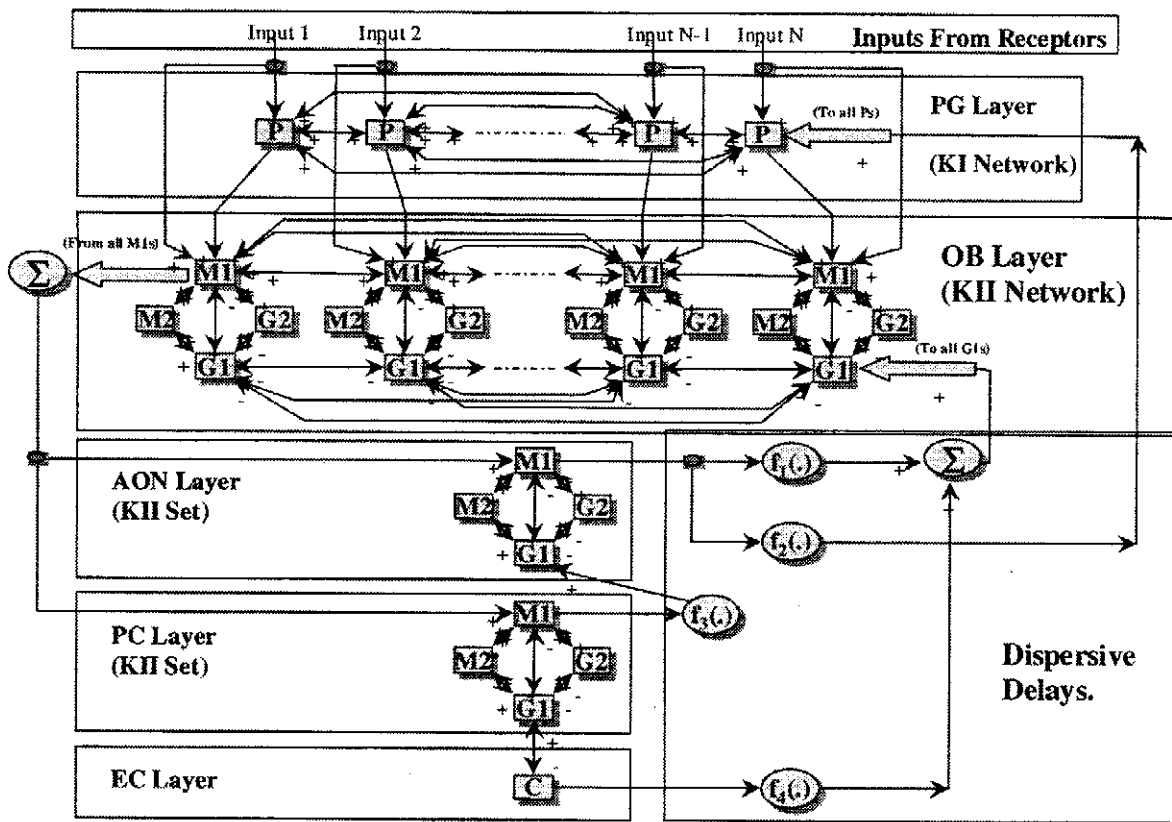


Fig. 6. KIII network as a model for the olfactory system.

to both “on” (1), then the weight between $M1_j$ and $M1_i$ is increased to a high value, otherwise its value is unchanged. At the beginning with no learned patterns, all these connection weights are at a low value. Observe that the KII network does not conform to traditional information processing systems that possess well-defined inputs and outputs. Here, the output is spatially distributed and exists briefly in time.

D. The KIII Network—A Model for the Olfactory System

The KIII network is topologically depicted in Fig. 6 and is a functional model of the olfactory system. The input to the model is a layer of excitatory neurons modeled as a KI network [periglomerular layer (PG)]. This layer projects to the olfactory bulb (OB), which is modeled by a KII network (with zero baseline) of a large size (N channels). The OB layer sends outputs to a KII set (positive baseline) that represents the anterior olfactory nucleus (AON) and to another KII set (negative baseline), the prepiriform cortex (PC), that in turn send its output to the entorhinal cortex (EC) and back to the OB and AON layers. Therefore, the KIII network is a variant of the KII network set with several layers of KII basic elements connected through dispersive delays $f_j(\cdot)$ in Fig. 6. The dispersive delays arise from the fact that the axons that interconnect the different layers in the brain have different thicknesses and lengths. As a consequence the action potentials received are dispersively delayed with respect to each other. The resulting operation is a low-pass filtering of the axon density pulses [49]. The way the overall network operates is very similar to the KII network set; however, the dy-

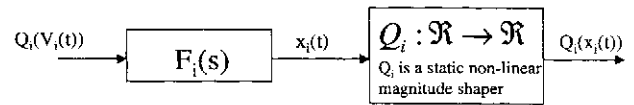


Fig. 7. Nonlinear cell embedded in a higher order system [$F_i(s)$ is the Laplace transform of the left hand side of (3)].

namics are qualitatively different. Because the intrinsic frequencies of oscillation of the KII sets in the different layers are incommensurable and the system is tightly coupled with dispersive delays, the different layers will never lock to the same frequency and chaotic behavior arises.

The best place to analyze the dynamic behavior of the overall system is the state space of the OB layer. The inputs will effectively “pull” the system to predetermined regions of the state space associated with learned input patterns. This chaotic motion can be “read-out” as spatial AM patterns of activity over the KII network. This output has been shown to provide a realistic account for the EEG obtained with an array of electrodes placed on the olfactory bulb [14], and also mimics the behavior of electrophysiological measurements made on the visual cortex of monkeys.

From the explanation of the KII network, we can see that it is unlike any studied in information processing. On one hand, it is a set of coupled-oscillators in a time-space lattice that is locally stable but globally chaotic. When an input is applied to the PEs, they oscillate with their characteristic frequency, and the oscillation is spatially propagated. On the other hand, the coupling in space is dependent upon the connection weights that are learned, so the information is coded

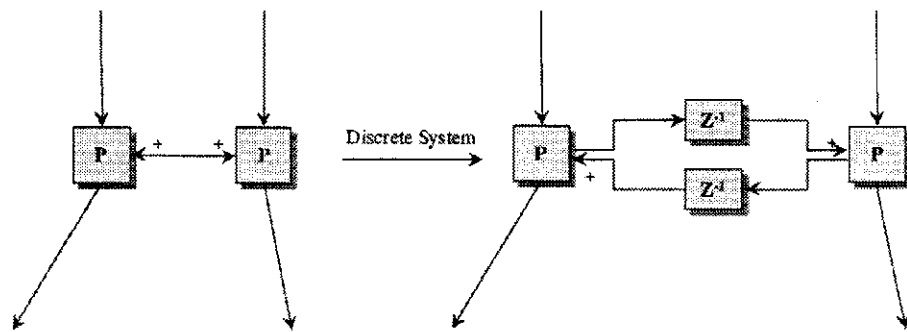


Fig. 8. Discrete representation of a K1 set feedback paths.

in spatial amplitude patterns of quasi-sinusoidal waves. Although we still cannot fully characterize the properties of the KII network for information processing, the dynamical characteristics of each piece have been specified by Freeman and co-workers [50], so we can build a system with similar dynamics in analog very large scale integration (VLSI). The next section will address the method of analysis and the implementation details.

III. DISCRETIZATION OF THE MODEL COMPONENTS

Freeman's model is normally simulated using Runge-Kutta integration [38] for each input, what is time consuming and it is not amenable to simple modifications in the topology. Here, we develop an equivalent digital model using digital signal processing (DSP) techniques at the system level. This digitalization approach has been successfully applied to physical phenomena that accept a linear model such as the vocal tract model in speech synthesis [40]. For nonlinear systems there is no transformation that can map phase space to a sampled time domain, like the conformal mapping from the S to the Z domains [46]. However, with the special structure of the K0 model (linear dynamics followed by a static nonlinearity) the following methodology to design equivalent discrete time linear systems was successful in preserving the general dynamical behavior of the continuous-time model. Similar methods have been applied to discretize the Hopfield model [3]. Equation (3) and Fig. 7 shows a general formal structure where a K0 set fits. The forcing input in the right-hand side of (3) represents the static nonlinear connections from other similar K0 sets, which themselves might also receive nonlinear excitation from this given i th K0 set. The differential equation in the left side of (3) represents a linear time invariant (LTI) system.

$$a_{in} \cdot \frac{\partial^n}{\partial t^n} x_i(t) + a_{i(n-1)} \cdot \frac{\partial^{(n-1)}}{\partial t^{n-1}} x_i(t) + \dots + a_{i1} \cdot \frac{\partial}{\partial t} x_i(t) + a_{i0} \cdot x_i(t) = Q_i(V_i(t)) \quad i = 1, 2, \dots, K. \quad (3)$$

The procedure that is proposed to discretize (3) decomposes this equation as topology and dynamics. The dynamics can be mapped with linear DSP techniques to the discrete time domain. The connecting structure between individual cells is preserved except that delays are incorporated in the

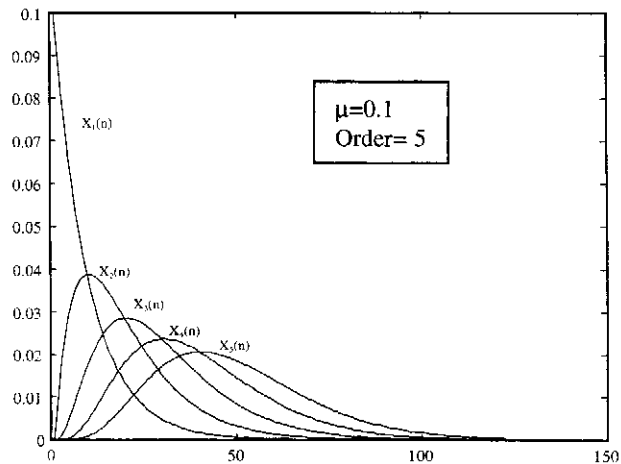


Fig. 9. Impulse response at different taps of a fifth-order gamma filter.

feedback pathways to avoid instantaneous time calculations. There are two basic problems with this methodology. First, it assumes that the continuous dynamical system is synchronized, which is unrealistic, but there are ways to mitigate this shortcoming [3]. Second, the dynamics between the analog and discretized models may differ. For the Hopfield network, the terminal dynamics of both (analog and discrete) systems have been shown equivalent [37]. However, the terminal dynamics for the KII set are a limit cycle, instead of a fixed point. The inclusion of a delay in the feedback path could change the frequency of oscillation between the analog and the digital counterpart, but the implementation discussed in Section IV avoids this problem.

One obvious advantage that arises from the proposed approach is that a *network of dynamical components* can implement the resulting discrete time model. This means that adding or deleting features or new components to the system is simple and does not imply rewriting the global state equations [as long as they follow the rule defined in (3)]. The digital system dynamics can also be probed at different components to understand the system better, and the resulting digital system computes the output solutions in a sample by sample basis for real time simulations. These characteristics are invaluable to perform testing of the analog simulator explained in the next section. An added advantage is that this methodology has been easily incorporated into standard commercial neural network packages.

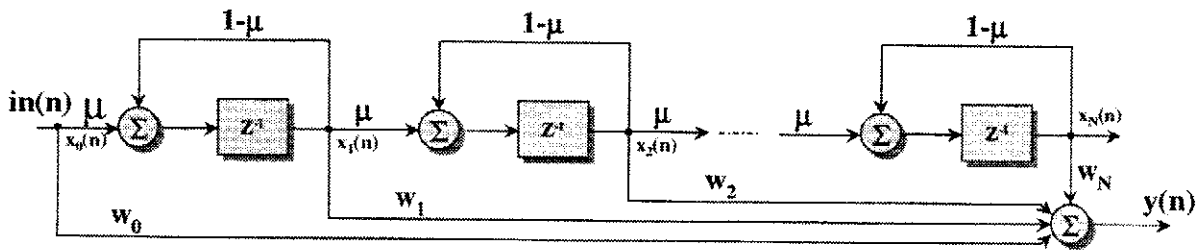


Fig. 10. Gamma filter structure.

A. Discrete Network Implementation of the Olfactory System Model

Each of the individual elements of the KII model follows the form of equation (3). The continuous-time instantaneous feedback is mapped into a delayed feedback operator. Fig. 8 illustrates the procedure for a KI set. Feedforward connections need not to be delayed. The exception is the time dispersion term $f_j(\cdot, t)$, but since this time operation is linear the procedure proposed in the previous section can still be applied to the overall system with delays.

IV. IMPULSE INVARIANCE TRANSFORMATION

The simplest method to discretize the dynamics of (1) is to take an impulse invariant [36] transformation that samples the impulse response of the dynamical system and preserves its shape. The discrete time system that results by sampling (4) with a suitable sampling frequency $f_s = 1/T$ is shown in (5)

$$\frac{1}{a \cdot b} \left(\frac{d^2}{dt^2} x(t) + (a + b) \frac{d}{dt} x(t) + a \cdot b x(t) \right) = \delta(t) \Leftrightarrow .$$

$$\Leftrightarrow h(t) = a \cdot b \left[\frac{1}{b - a} e^{-at} + \frac{1}{a - b} e^{-bt} \right] \quad (4)$$

$$x(nT) = T \cdot h(nT). \quad (5)$$

Since Freeman's model dynamics have a low-pass behavior, the potential aliasing error that arises from the impulse invariance transformation can be effectively controlled by decreasing T . In the results to be presented, the sampling frequency $1/T$ was chosen 20 times larger than the maximum frequency pole of the Laplace characteristic equation of (2). The resulting difference equation for Freeman's model is (6)

$$\begin{aligned} x(n) - (e^{-aT} + e^{-bT}) \cdot x(n-1) + (e^{-(a+b)T}) \cdot x(n-2) \\ = -T a \cdot b \left[\frac{1}{b-a} e^{-at} + \frac{1}{a-b} e^{-bt} \right] \cdot \theta(n-1). \end{aligned} \quad (6)$$

The function $\theta(n-1)$ is the delayed sampled version of the composite forcing input on the right-hand side of (1) or (3). Hence, the impulse invariant transformation naturally incorporates the delay required to avoid the instantaneous computations when the topological part of the dynamical model (3) is discretized.

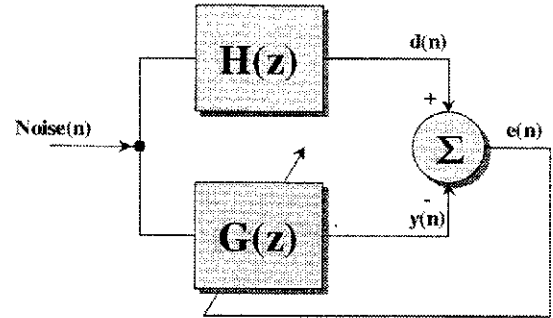


Fig. 11. System identification (ID) procedure.

A. Gamma Basis Decomposition

Global optimization methods or backpropagation through time (BPTT) [49] could be used to automatically set the parameters of the KII network provided input and output signals were available. However, this method would become difficult to implement due to the issue of stability. In fact, rewriting (6) as

$$x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) = a_1 \theta(n-1) \quad (7)$$

and taking the Z transform, we obtain an infinite impulse response system with a pair of poles

$$\frac{X(z)}{\Theta(z)} = \frac{a_1 \cdot z^{-1}}{1 + b_1 z^{-1} + b_2 z^{-2}}. \quad (8)$$

During adaptation of b_1 and b_2 , the system could become unstable. Alternatively, one can decompose the impulse response (7) over the set of real exponentials, which has been called the gamma bases (also known as the gamma kernels) given by (9) [39], [8]

$$g_k(n) = \binom{n-1}{k-1} \mu^k (1-\mu)^{n-k}. \quad (9)$$

The impulse responses of the gamma kernels (a cascade of low-pass filters of equal time constant) resembles many of the signals collected or modeled in neurophysiology (Fig. 9) [2]. The gamma kernels form a set of complete real basis functions, which means that any signal in L_2 (finite power) can be decomposed in gamma kernels with an arbitrarily small error [8]. The gamma kernels can be used as delay operators in substitution to the ideal delay operator of DSP. Weighting the values of each gamma delay operator creates a generalized feedforward filter called the gamma filter [39], which is written as (10).

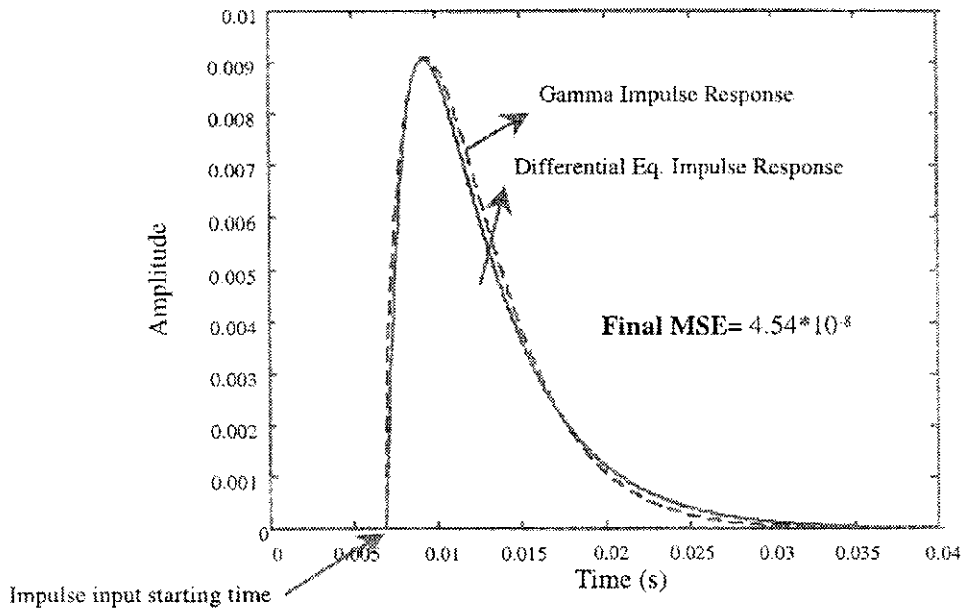


Fig. 12. Gamma basis approximation of the impulse response of (6).

The vector $W = [w_0 \ w_1 \ \dots \ w_N]^T$ in (10) is the projection vector. The gamma filter block diagram is shown in Fig. 10

$$x_k(n) = (1 - \mu) \cdot x_k(n-1) + \mu \cdot x_{k-1}(n-1)$$

$$y(n) = [w_0 \ w_1 \ \dots \ w_N] \begin{bmatrix} x_0(n) \\ x_1(n) \\ \dots \\ x_N(n) \end{bmatrix}. \quad (10)$$

Decomposing (5) over the gamma basis follows the well-established system identification framework, as represented in Fig. 11 [17]. A white noise source is simultaneously fed to the sampled differential equation (7) [denoted by $H(z)$] to be modeled and to the gamma filter $G(z)$. The outputs of these two systems are subtracted and the resulting value is used as an error for adaptation of the gamma parameters. Adaptation algorithms have been developed previously for the gamma filter parameters μ and the projection vector W [8].

The determination of the filter order for the case of (6) is relatively easy because we are decomposing the unknown system in the set of real exponentials. To gain a better understanding, we will address the characteristics of the gamma kernel next.

B. Gamma Filter Stability, Resolution, and Depth of Memory

If we consider the Z transform of (9) for N gamma kernels, we have a transfer function between the input and the last gamma tap as

$$G_T(z) = \left(\frac{\mu}{z - (1 - \mu)} \right)^N. \quad (11)$$

Therefore, the delay mechanism for the gamma filter is a cascade of identical first-order IIR sections (Fig. 10). For stability, the gamma filter feedback parameter has to be restricted to $0 < \mu < 2$. Yet, the condition is very easy to test since μ is common to all kernels in the filter.

Table 1
Gamma Parameters

w_0	-0.0250
w_1	1.0673
w_2	7.7234
μ	0.0237

The gamma filter has an infinite region of support, which means that a low-order filter can model long time dependencies. The region of support of the impulse response is very important for system identification, and for processing of time information it implements a short-term memory mechanism. Memory in the present context means how far in the past are the dependencies to compute the present output. The memory depth for the gamma filter with N taps is [8]

$$D = \frac{N}{\mu}. \quad (12)$$

The gamma filter is able to trade memory depth by resolution. In fact, we can write $N = D \cdot R$, where N is the order of the filter, D the memory depth given by (12), and R is the time resolution. In a FIR filter, the memory depth is the filter order N . From (12), we have the extra feedback parameter μ to change the memory depth of the gamma filter. However, there is a tradeoff between time resolution and memory depth, which is controlled by the order of the system. The resolution is problem dependent and most of the time the tradeoff is resolved heuristically. In our dynamical system modeling, we picked the lowest order due to implementation constraints.

C. Differential Equation Digitalization Results

From biological evidence, the parameters a and b assume the values 220/s and 720/s, respectively [11]. Since the poles are real in this case, we can obtain a good fit with a second-

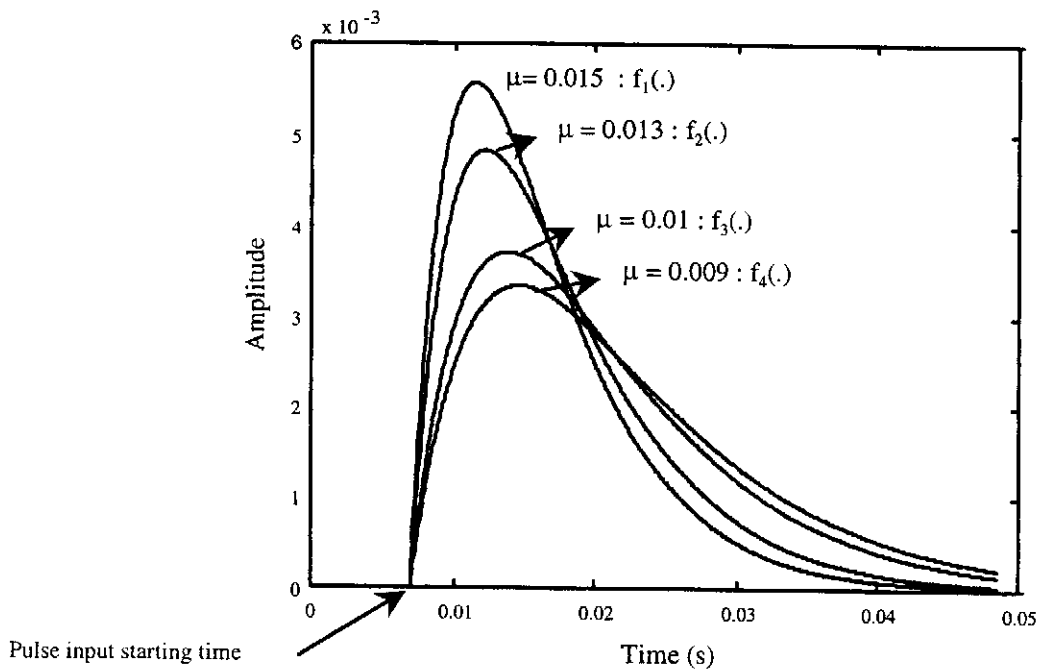


Fig. 13. Delay response for different dispersive delays of the KIII model.

order gamma filter. The sampling frequency $1/T$ was chosen as 20 times larger than the largest frequency pole. Fig. 12 represents the impulse invariant system response to an impulse. The gamma filter approximation of the sampled differential equation is also plotted. The resulting match is very good, although the approximation filter has only a double pole at $(1 - \mu)$. Table 1 summarizes the gamma filter parameters values found using the discussed system ID approach.

D. Dispersive Delays Digitization

As discussed earlier, the time dispersion was introduced in the KIII network to model the different axonal interconnection lengths and thicknesses. These functions can be modeled as in (13)

$$f(Q(x, q), nT) = \frac{1}{T_s - T_e} \sum_{nT - T_s}^{nT - T_e} Q(x, q) \quad (13)$$

which can be rewritten in a convolution form as

$$\begin{aligned} f(Q(x, q), nT) &= Q(x, q) \bullet \frac{1}{T_s - T_e} (u(nT - s)) - (u(nT - s)) \\ &= Q(x, q) \bullet D_d(nT) \end{aligned} \quad (14)$$

where \bullet denotes the convolution and $u(nT)$ is the step function. Equation (14) represents a FIR low-pass filter in discrete time and $D_d(nT)$ is recognized as a linear phase filter of Type I or II depending if the number of $D_d(nT)$ samples is an even or odd number, respectively. A linear phase system is nondispersive. To be more biologically realistic, the filter should have nonlinear phase. A simple low-pass IIR system like a first-order gamma kernel has nonlinear phase and so implements a more realistic dispersive delay.

The gamma kernel has the added feature of allowing for an analytical procedure to calculate the time dispersion. From (14), we may interpret $D_d(nT)$ as a memory structure with time depth of $T_s - T_e + 1$. A gamma kernel has a depth of memory given by (12), so we may take $D = (T_s - T_e + 1)/T$ and find a μ with a given order of the system. However, as mentioned before, time resolution is decreased with this design and may be a concern. In the present implementation, a cascade of two dispersive delays was utilized ($W = [0 \ 0 \ 1]^T$). Following this procedure, we get the dispersive delay response of Fig. 13.

E. Digital Simulation Results

After discretization, any software packages (DSP or neural networks) or even hardware simulators can be utilized in the simulation. The environment used for the simulation task is a commercial neural network package called "NeuroSolutions" [35]. The software has all the basic building blocks necessary to construct the K sets and the dispersive delays. The user interface enabled the creation of the topologies of each set using an icon-based interface. We performed many tests on each K set, but here we will report on the more interesting ones (forced responses). First, we show in Fig. 14 a comparison of the KII output spectrum to a "high" input obtained using Runge-Kutta (ODE45 in Matlab [32]) and our modeling method with a sampling period of $T = 1/(20 \cdot 720)$ s. The KII system response was simulated with an input square wave with an amplitude of six (arbitrary units) when excitation is present, and zero otherwise. The parameter set is summarized in Table 2 where K_{EI} are the weights between the mitral (E) and granular cell (I) denoted by M and G in Fig. 5.

We plot in Fig. 14 the magnitude spectrum of the oscillating time response using the fast Fourier transform (FFT)

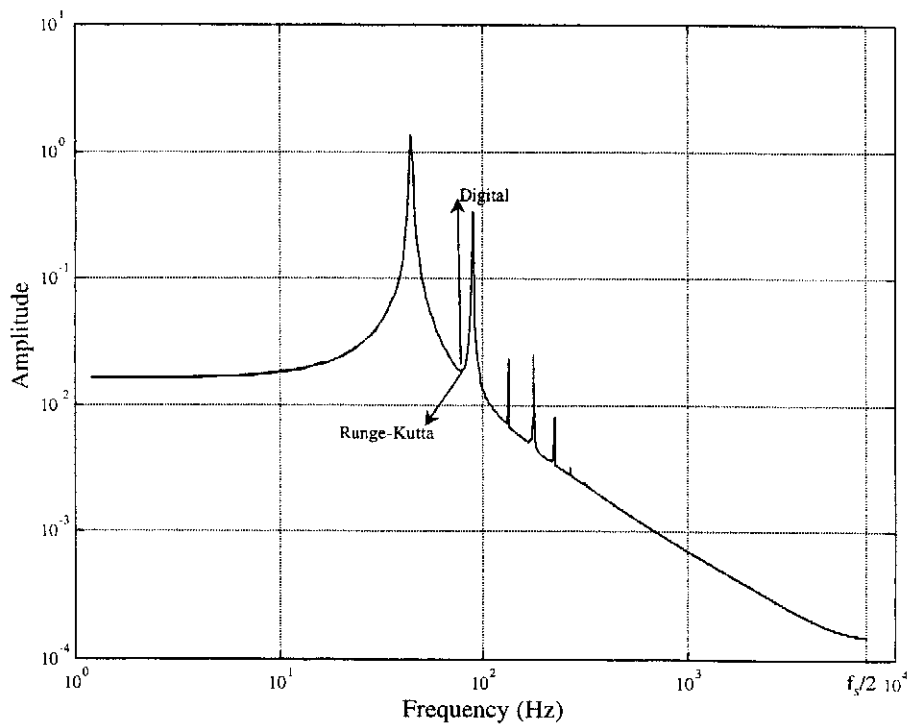


Fig. 14. Comparison of KII output spectrum using Runge-Kutta and our discretization.

Table 2
KII Parameter Set

K_{EE}	0.3
K_{EI}	5
K_{IE}	0.2
K_{II}	0.25

Table 3
Patterns Stored

Pattern 1	1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0
Pattern 2) 0 1 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 1 (
Pattern 3) 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 (
Pattern 4) 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 (
Pattern 5) 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 0 1 0 !

window method, with a 12000 sample window and 36 averages. As we can observe, there is a perfect match between the Runge-Kutta spectrum and our methodology.

We have further simulated a $N = 20$ channel KII network and stored five patterns in the network according to the modified Hebbian rule discussed earlier. The patterns are defined in Table 3. The results to be shown are for the recall of Pattern 5.

An excitation drives the system into a limit cycle that lasts while the input is applied. Fig. 15(a) shows the time response of channel 19 (driven by a "1") and Fig. 15(b) the limit cycle plotted between the excitatory versus the inhibitory states.

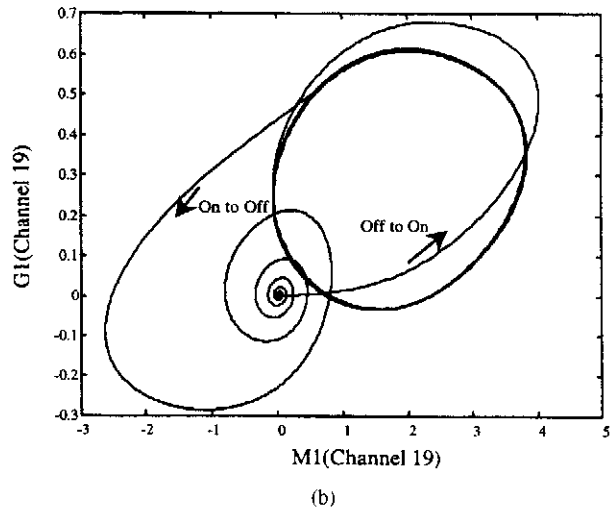
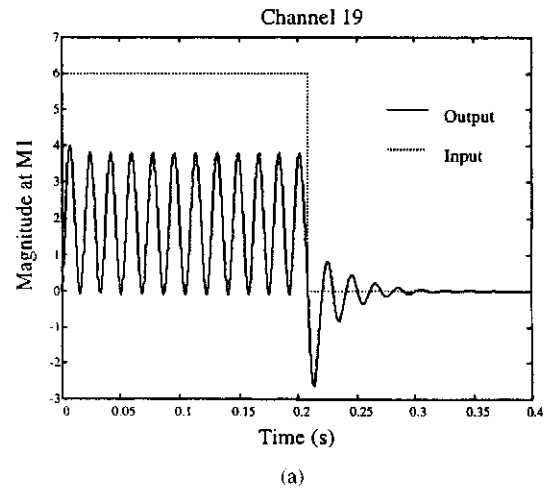


Fig. 15. Time series of KII with a square input and phase plot.

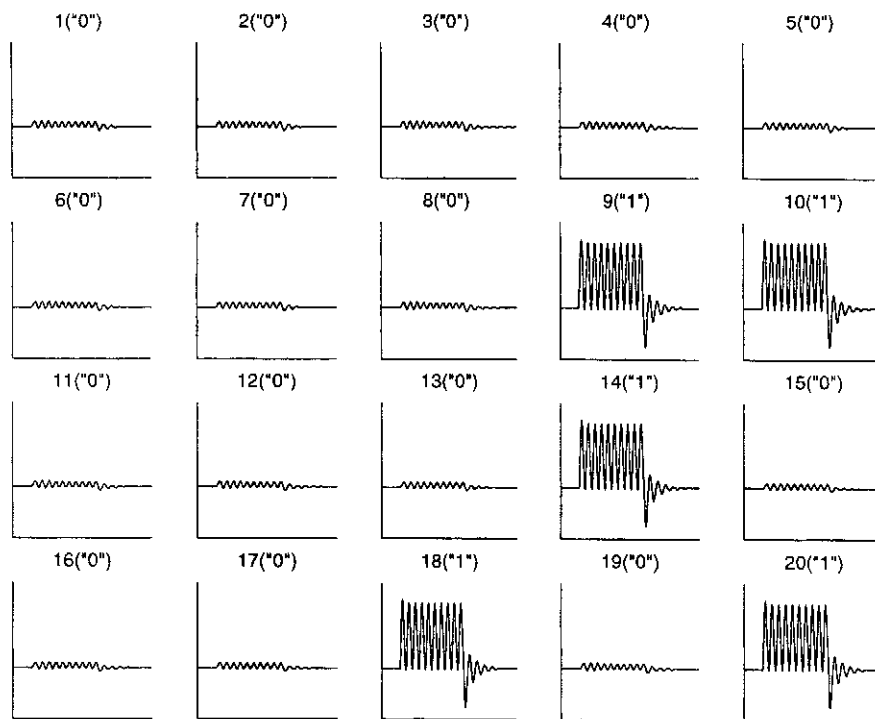


Fig. 16. KII 20 channel network response to input pattern 5 (Table 3) [channel response ("Input")].

to a large limit cycle centered away from the origin. It stays there for the duration of the input, and when the system input vanishes the output returns to zero.

The phase plot resembles a Shilnikov singularity, but we have not investigated further the local characteristics of the unstable point [45]. This result also indicates that the KII may be stable in the sense of Lyapunov, i.e., there is a potential function that models the system as energy conservative. Finally, Fig. 16 shows the outputs of all the KII sets in the KII network after the application of pattern 5 at the input (recall of pattern 5). The input pattern produces a large amplitude oscillation in the channels that have 1 in the input pattern, so we have the expected association between inputs and the AM oscillation that Freeman describes in the OB of the rabbit following a sniff, which was also simulated in [10].

One interesting property of this output is that it is transient in time, disappearing when the input vanishes. This behavior is very different from Hopfield networks, where the system dynamics stay in the fixed point after the input is removed. Extra energy would have to be supplied to the system to get away from the fixed point, while here the KII network is always ready for real-time processing of temporal patterns.

The KIII network was constructed in NeuroSolutions with only $N = 8$ channels in the OB layer because the simulation takes much longer. Table 4 summarizes the parameter set used for the simulation. All the results were generated for the recall of pattern 2. The input was a smooth rising pulse within the values of 2 and 8 as specified in [49].

In the full KIII system, we have many possible probe points. We will start by analyzing the resting state at the PG, OB, AON, and PC layers. If we observe the time signals at any of these locations in the absence of input, we will see random fluctuations of a quasi-periodic activity that resem-

Table 4
Parameters Set for the KIII Network Simulations (see Fig. 6)

External Interconnections	$[K_{MP}; K_{EM}; K_{AM}; K_{CB}; K_{BC}] = [0.5; 1.0; 1.0; 1.5; 1.0]$ $[K_{PP}; K_{MM-H}; K_{MM-L}; K_{GG}] = [0.2; 4; 0.5; 1.5]$
OB Layer KII	$[K_{MM}; K_{MG}; K_{GM}; K_{GG}] = [0.25; 1.5; 1.5; 1.8]$
AON Layer KII	$[K_{EE}; K_{EI}; K_{IE}; K_{II}] = [1.5; 1.5; 1.5; 1.8]$
PC Layer KII	$[K_{AA}; K_{AB}; K_{BA}; K_{BB}] = [0.25; 1.4; 1.4; 1.8]$
Gains Associated with dispersive delays (K_i Associated with $\tau_i(\cdot)$)	$[K_1; K_2; K_3; K_4] = [1.6; 0.5; 2; 1.5]$
T_e of equation 13	$[T_{e1}; T_{e2}; T_{e3}; T_{e4}] = [11; 15; 12; 24]$
T_s of equation 13	$[T_{s1}; T_{s2}; T_{s3}; T_{s4}] = [20; 26; 25; 39]$
Patterns Stored	$\begin{bmatrix} \text{Pattern1} \\ \text{Pattern2} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$

bles the electroencephalogram's background activity. When we plot the activity as phase plots between the inhibitory and excitatory states (Fig. 17), we see phase plots that resemble strange attractors of different shapes across the four layers. Therefore, the KIII has a high-order nonconvergent dynamical behavior compatible with a chaotic attractor. We have not made any tests for chaoticity yet, but a simple power spectrum estimation of the time series generated by the AON (E1 state) clearly shows a $1/f$ type of spectrum that is

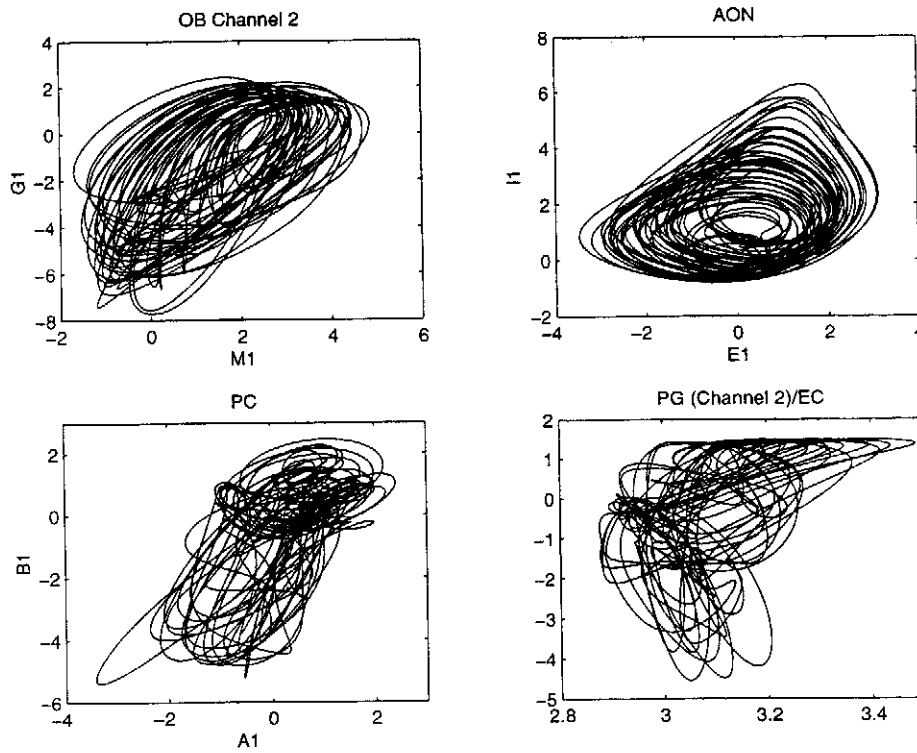


Fig. 17. Phase plots at different layers of the KIII network (no excitation).

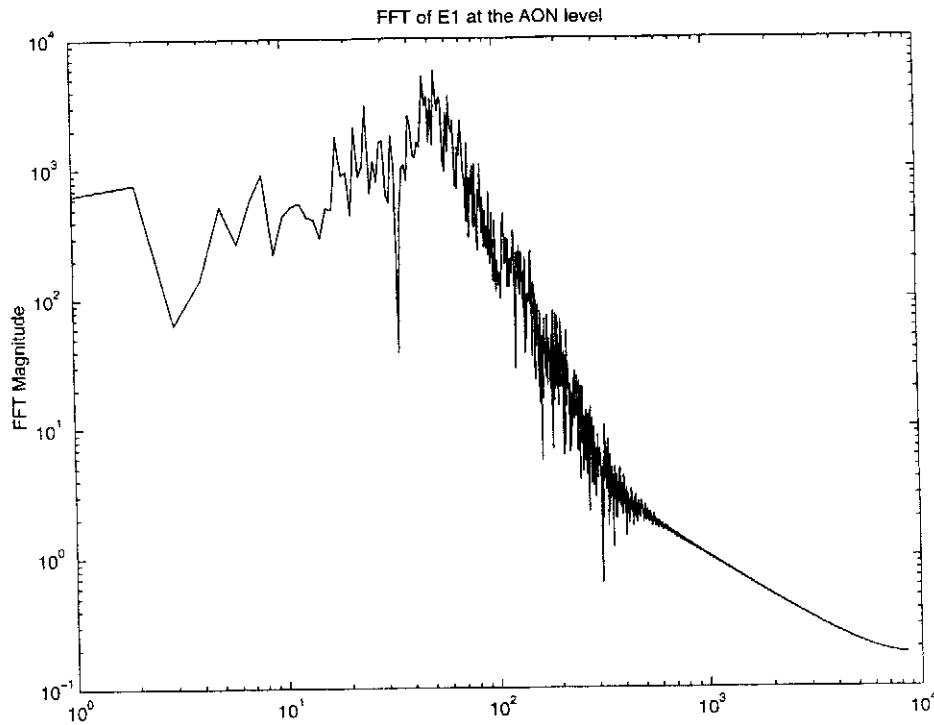


Fig. 18. FFT at the AON layer (no excitation).

once again compatible with a self similar attractor (Fig. 18). So we conclude that the expected behavior simulated using Runge–Kutta integration in [49] is again obtained in our discretized model.

Let us now turn to the OB layer and analyze the response to a pulse input. Fig. 19 shows the time series for the recall of pattern two. Notice that the response of the channels where

the “high” level of the input was stored do indeed show a dc offset during the time the pulse is applied. We were also expecting an increase in oscillatory amplitude during the pulse, but the small size of the OB layer (only eight channels) may explain the difference.

We investigated in more detail the structure of the oscillations among the different channels of the OB layer. When we

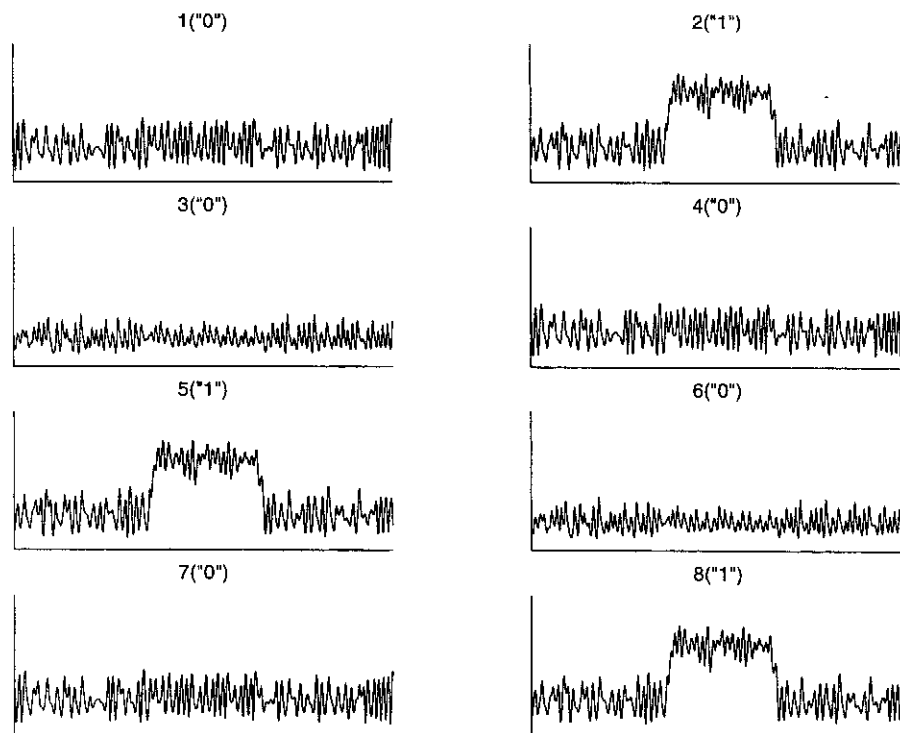


Fig. 19. Time Series for Pattern 2 recall [channel ("Input")].

create the phase plot between the excitatory and inhibitory state of channel 2, we clearly see [Fig. 20 (a)] that the basal attractor (for the "off" input) changes to a "wing" when the input switches to a high amplitude. Channels that have a stored pattern oscillate in perfect synchronization [Fig. 20 (b) and (c)] and share the same basal state evolution. This is also reported by Freeman [13]. However, channels belonging to different templates oscillate with different incommensurable phases [Fig. 20 (d)]. We remark that in the beginning of the simulation they start in phase, but they lose synchrony very quickly thereafter. Freeman states that this lack of coherence is not apparent in the signals collected from the rabbit nor in analog implementations of the model. He states that it corresponds to the breakdown of the chaotic attractor due possibly to quantization errors that create limit cycles in phase space [14].

Therefore, we conclude that when excited, the KIII system undergoes a transition from the basal to a different chaotic behavior characteristic of the pattern being recalled. When the excitation vanishes, the system always returns to the initial attractor (Fig. 20). The "on" channels are distinguishable by the higher offset and signal dynamics (Fig. 20). All these results are consistent with Freeman's results using Runge-Kutta numerical methods [49], while here we use our discretization of the linear part of the dynamics using the gamma approximation.

V. ANALOG CIRCUIT IMPLEMENTATIONS

The previous sections presented Freeman's model of the olfactory system and its DSP implementation. The flexibility, immunity to noise, and modularity of the DSP implementation is unsurpassed when compared to analog VLSI im-

plementations. However, in the general framework of $S\Sigma T$'s discussed in the introduction, an analog implementation also presents advantages of its own, such as:

- natural coupling to the analog real world;
- computation is intrinsically parallel;
- the computation time is practically instantaneous (only set by the delay imposed by each component);
- there are no roundoff problems, i.e., the amplitude resolution is effectively infinite;
- it generally renders smaller circuits;
- it is one order of magnitude more power efficient.

These benefits motivated the development of an analog VLSI implementation, but unlike the traditional design approach, we seek to preserve as much as possible the functional correspondence to the digital design to help us set in the digital simulator the parameters of the analog KIII model. Otherwise, finding appropriate parameters in the analog version of the KIII becomes a daunting task. Our analog design approach is centered on a *continuous amplitude and discrete time* implementation of the dynamic building blocks (mixed-signal implementation) for the following reasons.

- We wish to preserve a continuous amplitude representation due to the negative effect of the roundoff errors reported in implementing chaotic dynamical systems [50].
- The most serious design constraint in the VLSI implementation of the KIII is the size of the interconnect. If a full analog (i.e., analog amplitude and continuous time) implementation is selected, the area required for the interconnect would be prohibitive. Another notorious advantage of the mixed-signal solution is the control gained over the timing of events. The synchronous response update at discrete times provide free time slots

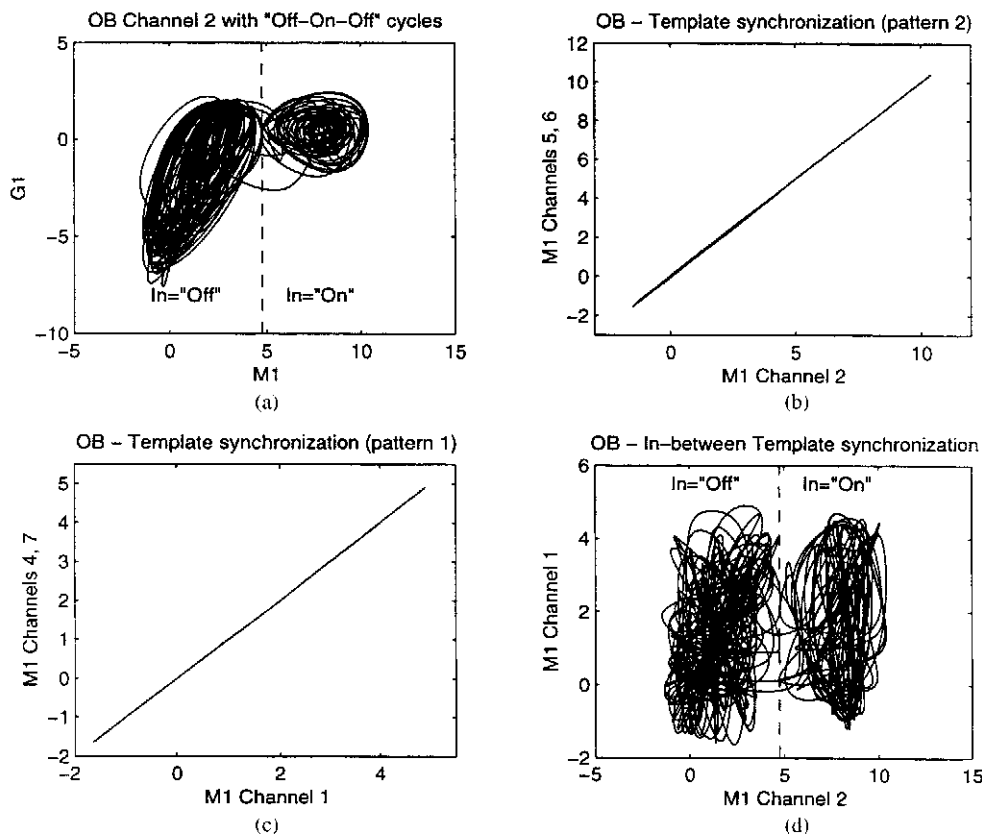


Fig. 20. Behavior with input "OFF-ON-OFF" cycles.

to implement other functions such as reading inputs, parameters, and transparent multiplexing with the consequent optimal resource management.

- Discrete time implementation also brings important advantages for testing since a one-to-one correspondence can be established between the DSP simulation and the chip components. This is evident from the fact that both have the same mathematical representations, difference equations, and Z transform. The difference resides on dimensionality that can easily be translated between both domains by substituting n by nT .
- Unexpectedly, time discretization also leads to a very efficient implementation of the dynamics in the form of a novel circuit called Filter and Hold (F&H, patent pending) that decreases the size of the capacitors and resistors for a given time constant. The discretization of the KIII system provides a clear roadmap for the analog implementation. As was discussed above, the only components required are gamma filters for the dynamics and dispersive delays, static nonlinearities, summing nodes, and the vast interconnect.

As a conclusion, the high-level formal simulation proposed in earlier sections becomes the centerpiece for the discrete-time analog implementation. We have access to a formal high-level representation that can, in a sample by sample basis, predict how the real system will behave. The VLSI system can be flexibly tested and the parameters refined with the digital simulator, which can be directly brought onto the designed chip by external active control.

Of course, this tight coupling between the mixed signal and the DSP implementations also creates extra difficulties because the exact mathematical functions are seldom implementable in VLSI. Hence, the design cycle should include a redefinition of the formal DSP system to include the small alterations imposed by implementation limitations. If both behave similarly, the analog implementation will be a good representation of the original continuous model, since the previous results showed that the digital model reproduces the Runge-Kutta simulation well. This is the procedure followed in the next sections, where we describe each of the building blocks and present measurements from circuits designed in MOSIS 1.2- μm AMI technology using the subthreshold CMOS regime for low power.

A. Nonlinear Asymmetrical Sigmoidal Function

The function responsible for the nonlinear characteristic of Freeman olfactory model is static and has the peculiarity of being asymmetric around the zero point (Fig. 2). A precise implementation of (2) is not an easy task, but we succeeded in approximating the sigmoid with a modified operational transconductor amplifier (OTA). The important aspects of the original function are preserved, such as exponential sigmoidal shape and asymmetry around the input axis. The exponential shape is set by placing the MOS transistor in the subthreshold operating region, where it displays an exponential relationship between the gate source voltage and the drain current. The asymmetry is accomplished by unmatching the current mirror to act as an active charge to a differential input

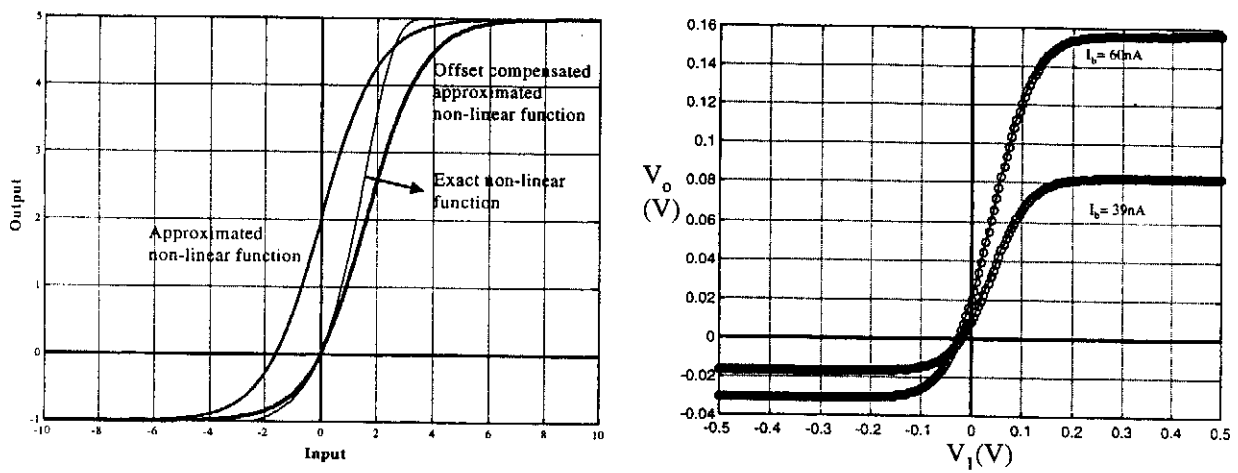


Fig. 21. Approximated nonlinear function (left with $N = 5$, $I_b = 1$, $VT = 1$) and measured responses in the chip for two different bias currents (the bias current of G_m is $I_{bG_m} = 600$ nA).

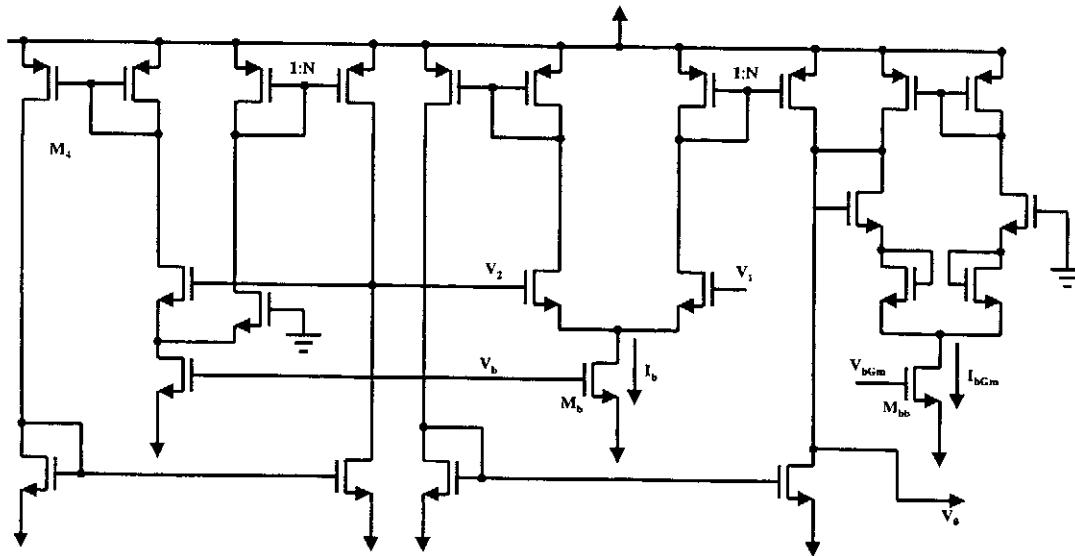


Fig. 22. Implemented asymmetric nonlinear function (voltage in/voltage out).

stage. The output current relation with the input differential voltage can be represented by (15), which has the required shape as Fig. 21 shows

$$Q(V_{in}) = I_b \times \frac{N \cdot e^{(V_{in}/VT)} - 1}{e^{(V_{in}/VT)} + 1}. \quad (15)$$

The offset that is evident from Fig. 21 (left panel) is produced by the current mirror unbalancing; however, it can be cancelled out by subtracting a voltage at the cell input equal to the offset value. This can be automatically done by placing feedback around a similar cell and applying the output to the inverting terminal [48]. The residual offset will be only due to the natural fabrication process. After the offset cancellation, the sigmoidal curve is shown in Fig. 21 (left panel). The final schematic is shown in Fig. 22. The current is converted to a voltage with a G_m amplifier and the OTA was implemented using a wide-range topology to prevent the inversion of current at the input differential pair when the output voltage drops below a certain value. Chip measurements with two different

bias currents are shown in Fig. 21 (right panel), and the overall shape matches the circuit simulations reasonably well.

Notice that the dynamic range of the DSP implementation and the chip are different. However, in the chip we preserve the 1/5 ratio between the dc unbalance and the saturation level that is implemented by Freeman's nonlinearity with $q_i = 5$. We choose an $I_B = 40$ nA to yield a ~ 60 -mV dynamic range, but different bias currents preserve the ratio while changing the gain and the saturating levels. We are only concerned with the slope of the nonlinearity that did not match (2). This will impact the parameter setting between the two implementations, but for comparison purposes (15) was also programmed into the digital simulator.

B. Dynamics—A Nanopower Analog Discrete Time Circuit

The olfactory system is complex, highly interconnected, with many similar blocks repeated many times. The power consumption increases proportionally to the system order, meaning that effectively power consumption is a strict con-

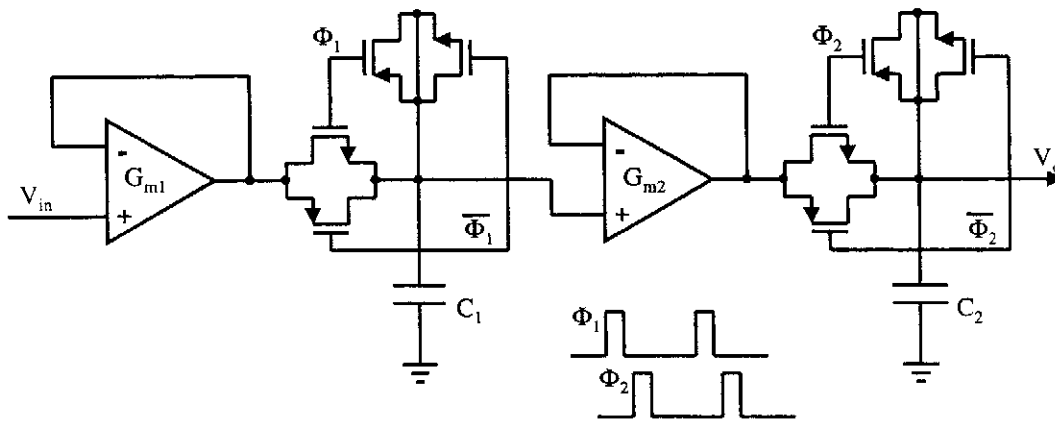


Fig. 23. F&H implementation for the KO dynamics.

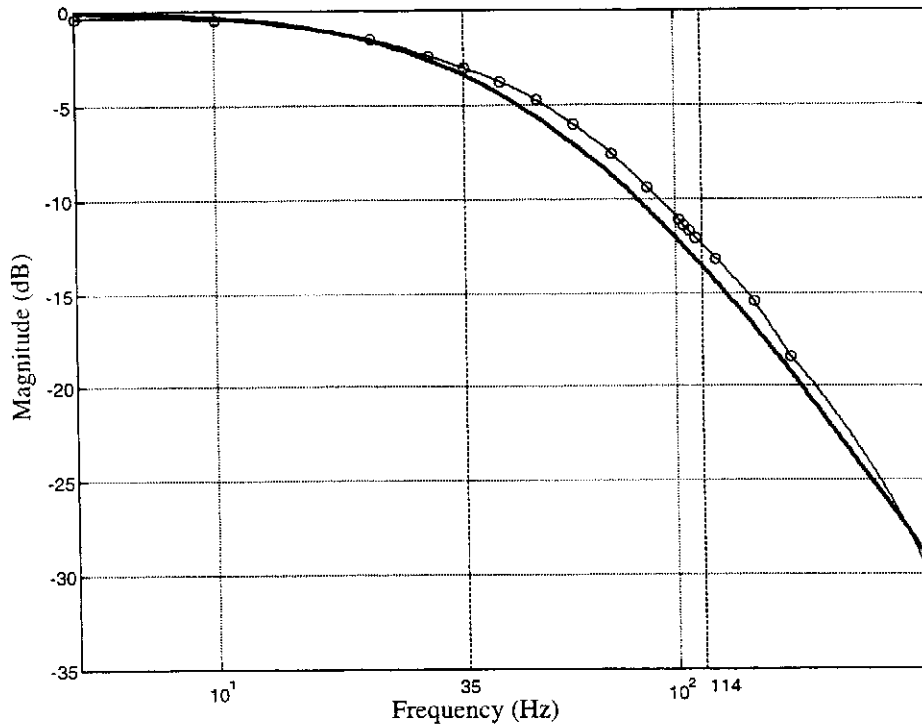


Fig. 24. F&H measured AC response together with predicted response by (16) (solid).

straint for the implementation. In discrete time, there are basically two filtering techniques available: the switched capacitor (SC) [5] and switched current (SI) [26], but they both utilize large areas and are not compatible with nanowatt consumption. A third alternative would use analog filtering with a sample and hold (S/H) at the input and output to implement continuous time processing of a discrete time signal [36]; however, it does not bring any advantage when compared with the above two approaches.

Small area and nanopower consumption usually go hand-in-hand. Simpler circuits take less elements and in general will require less power. We present an example of a novel discrete-time technique, that in essence is analog in magnitude, but uses time discretization in a very simple way to get very long time constants with little area and low power. The technique was named “filter and hold (F&H)” and it is a mixed

analog/discrete time design that combines analog continuous time conventional filtering with sampling [47], [48].

The F&H technique is based on the very intuitive idea of allowing a capacitor to integrate a current during τ seconds and halting it during a time of $T - \tau$ seconds. This process is repeated every T seconds. The resulting time constant value is then multiplied by the duty cycle, which is defined as the ratio of τ over the sampling period T . The multiplicative effect to the time constant is solely based on the duty-cycle of the clock and not on the sampling period. Therefore, large time constants can be achieved without a decrease in sampling frequency or with high capacitor (or transistors in SI systems) ratios. The switching scheme is simple and both the area and power requirements can be made small. Further details about F&H are out of the scope of this paper; however, we have shown that the procedure is general for any

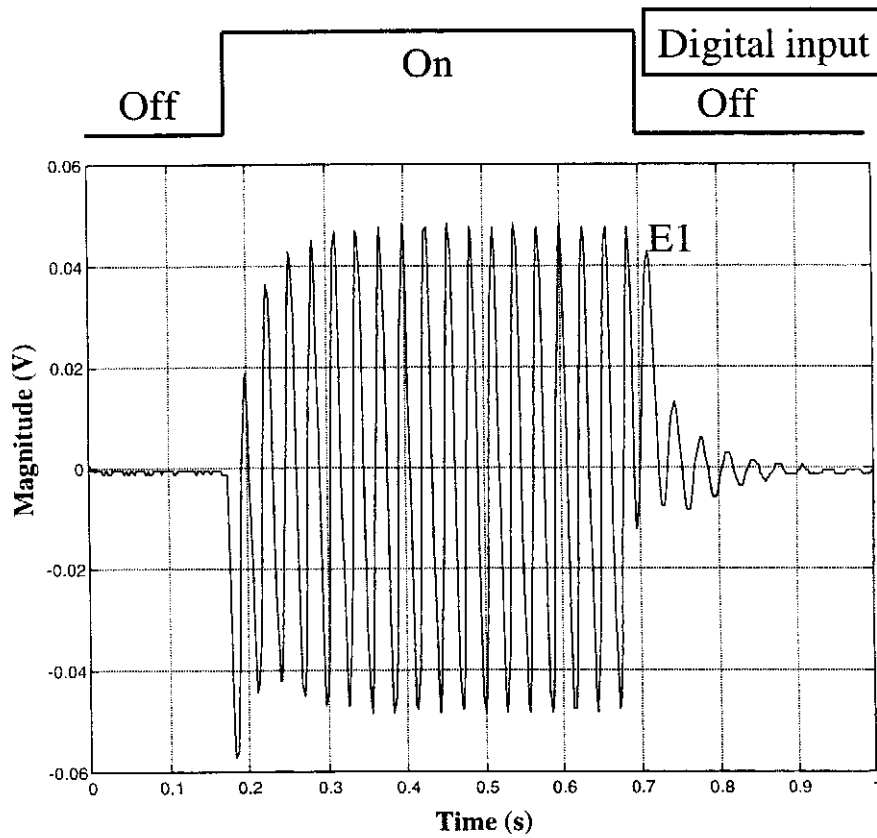


Fig. 27. Measured result of KII (E1) output for a pulse input.

D. A Coupled KII–K0 Measured Results

A KII set was implemented in analog VLSI with the components described in the previous sections. On the same chip, we also implemented several K0 sets and for the tests, we connected one of the K0 sets to the KII as shown in Fig. 26. All the resistors are external, and for this setup we have two more coupling parameters K_{GI} and K_{IG} linking the KII and the K0. Our first experiment was to find the values of the coupling coefficients that made sense from an analog circuit point of view (all the values in the digital simulations are dimensionless multipliers). For dynamic modeling, the ratio of resistors is what is important for the simulations, so it is the quantity presented in Table 5. The feedback resistor is $R_F = 100 \text{ K}\Omega$.

Due to the F&H implementation of the dynamics and differences in the nonlinearity, the coupling coefficients of Table 2 did not provide the sought after dynamics for the KII, so a tuning of parameters was necessary. We present in Table 5 a parameter set that produced the expected dynamical behavior. This reparametrization was anticipated, but according to our methodology, we now need to make sure that we redefine the DSP model to match the new dynamical behavior.

First, we measured the poles of the analog K0 to be $a = 110/\text{s}$ and $b = 360/\text{s}$. The bias current for the nonlinearity was $I_b = 40 \text{ nA}$, very close to the designed value. Next, we substituted in our digital implementation the gamma filter model by the F&H equation (16), and the original nonlin-

earity by (15). With these pole values and the same coupling parameters of Table 5, our digital simulation produced the expected dynamical behavior that we report next. Hence, we have reinforced our conviction that the digital simulation can be used as a flexible test bed to set analog system parameters; we just need to model all the components more accurately.

The waveform measured in the VLSI chip with a forcing input pulse is plotted in Fig. 27. Here, we plot the digital input and the measured response at the excitatory state (ac measurement). This figure should be compared with Fig. 15 obtained from the digital simulation.

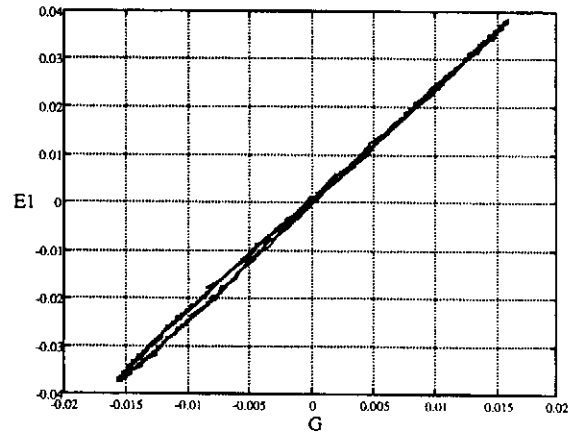
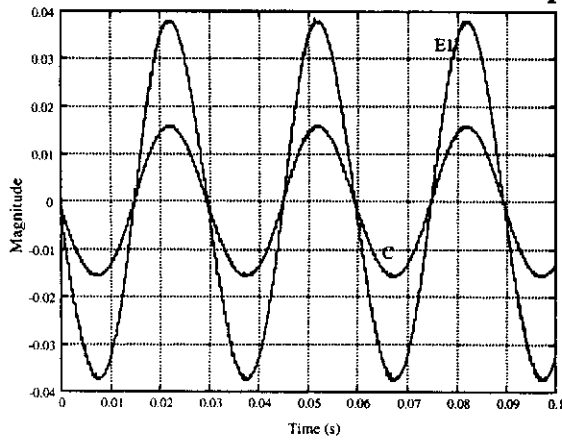
In Fig. 28, we show the response between the K0 and the excitatory state of the KII set, for both the digital simulation and the waveforms collected from the chip.

In Fig. 29, we select the KII inhibitory state to show the expected out-of-phase response. We conclude that the new digital simulation correctly follows the KII VLSI implementation time evolution with forced inputs both for the phase, frequency, and relative magnitudes.

We conclude that the major sources of error responsible for some observable differences are the F&H implementation and the shape of the nonlinearity. The G_m amplifiers were designed as a simple differential pair with emitter degenerating diode connected transistors to increase the linear input range and diminish the G_m value. It can be shown that for each stage in Fig. 23 when the switch is “on,” the differential equation is [33]

$$C \cdot \frac{dv_o}{dt} = I_b \tanh \left(\frac{v_{in} - v_o}{4V_T} \right). \quad (17)$$

Chip Measurements



Digital simulation result

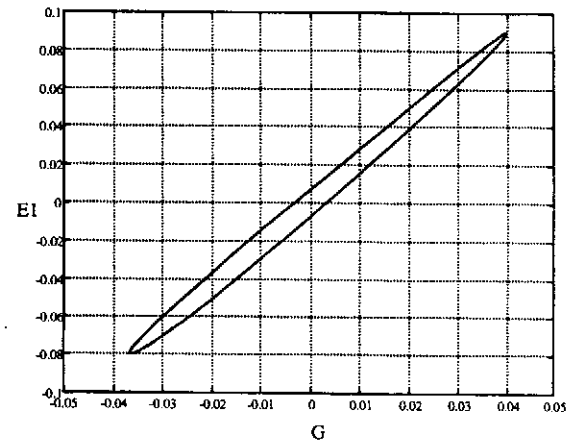
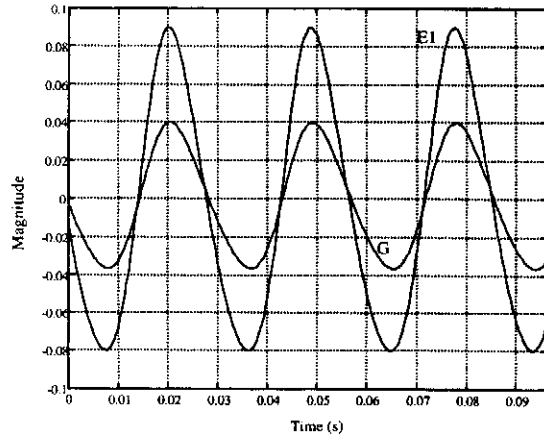


Fig. 28. Measured and simulated KII (EI and G) time sequence and phase plot. All K0 cells oscillate at the same frequency and in phase.

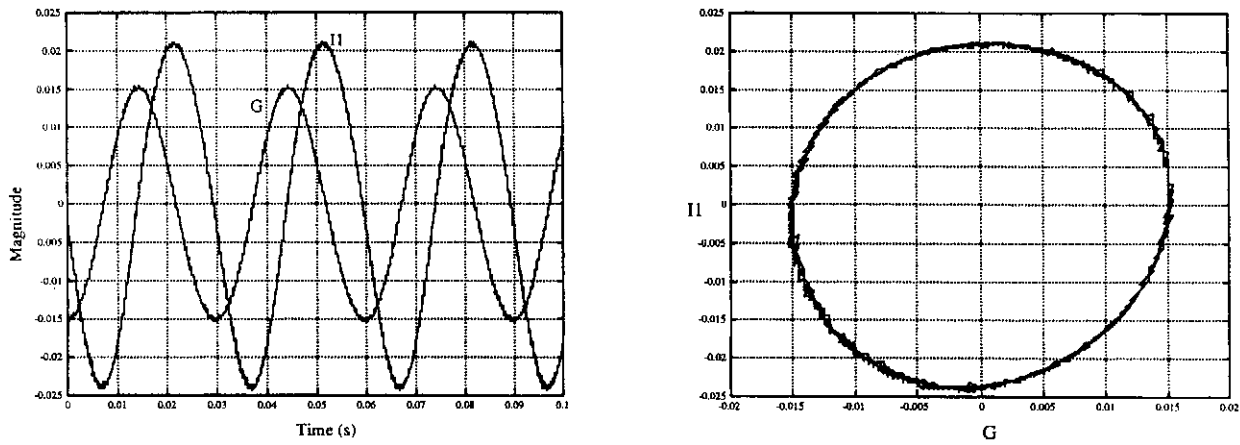
The nonlinear hyperbolic tangent function will have a re-shaping effect on the signals and will effectively limit the signals' magnitudes (the amplitude difference we see in the simulations of Figs. 28 and 29). For a better matching between the simple and flexible digital implementation and the real chip implementation, this effect needs to be included in the model or a more linear G_m or other more linear F&H topology should be designed.

We are beginning to understand the effect of the amplitude nonlinearity on the overall dynamics, which is the missing link to go back and forth between the parameterization of the digital simulator and the analog implementation. Observing Table 5, the ratio of K_{EI}/K_{IE} is smaller than the scale factor for the other gains of Table 2. A more careful analysis of Fig. 21 reveals that the exact nonlinear function proposed by Freeman has a higher slope and consequently requires a smaller input for the same saturating output levels. This is the gain factor responsible for such a parameter change. In fact, the normalization factor α needed to match the slopes of the two functions is about $\alpha = 1.5$. Furthermore, the implemented nonlinearity has $V_T = 25$ mV, and a positive saturating level of 80 mV. To scale up this function to the -1 and 5 saturating levels used in the digital simulations of Freeman's nonlinearity, we have to make in (15)

$I_b = 1$, and the resulting normalizing factor becomes $V_T' = 5/(80 \times 10^{-3}) \times 25 \times 10^{-3} = 1.56$, where the numerator comes from the fact that Freeman's nonlinearity has a ratio of five between the saturating levels. This scaled-up function has the same saturating levels as the one plotted in Fig. 21, but with different slope due to the different normalization of $V_T' = 1.56$. If now we correct Freeman's nonlinearity slope with a normalizing factor of $\alpha' = \alpha \times V_T' = 1.5 \times 1.56 = 2.34$, and perform a digital simulation with the same conditions of Figs. 28 and 29, the same dynamics (including the oscillating frequency) are obtained except for the obvious scaled-up amplitudes.

Other higher order effects, such as clock feedthrough [53] and extra delays due to finite bandwidth of all the needed components, have not been accounted for in the digital representation. We conclude that our discretization methodology is a good prediction of how the mixed-signal analog system behaves dynamically, and will be a tremendous asset to the parametrization of future, more complex analog implementations. But with the present silicon technology, these chips will still have relatively few components to attack the problems of representing real-world signals. One very promising area of research is to seek implementations with nanotechnology components. They hold the promise of shrinking the

Chip Measurements



Digital simulation result

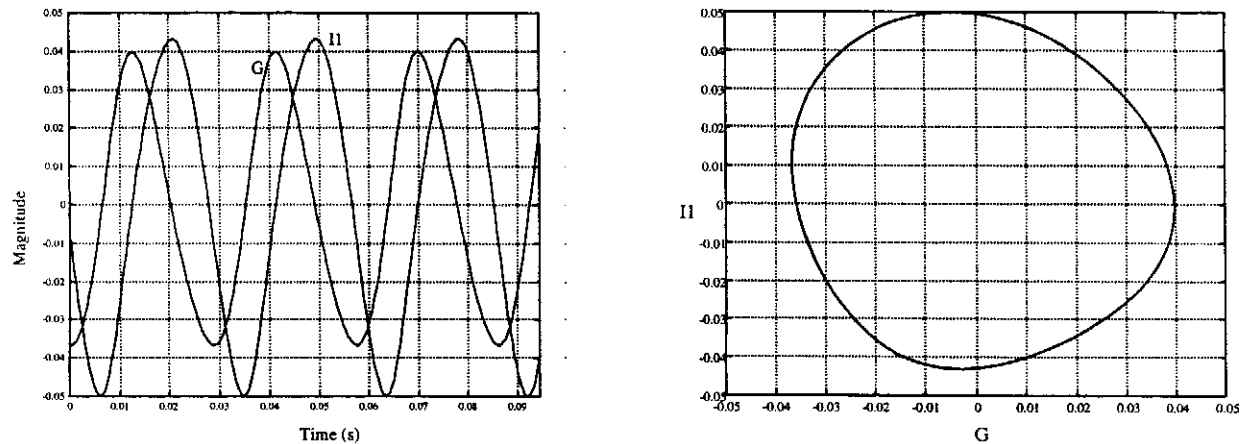


Fig. 29. Measured and simulated KII (II and G) time sequence and phase plot. All K0 cells oscillate at the same frequency but out of phase.

size and bring new properties such as short-term memory, which is very difficult to implement in analog VLSI. Biology can provide the insight for new computer architectures, and to take full advantages of the physics and chemistry of macromolecules, the language of dynamics should be utilized to describe the components of the next generation of nanoinformation processing devices.

VI. CONCLUSION

The work described in this paper encompasses several disciplines, and it is a prototype of the issues that will be faced in creating animats possessing the exquisite information processing capabilities found in animals. We think that effectively transferring information from real-world signals to stable representations is a key link in this process that requires new insights, mixed analog/digital solutions and a dynamic theory of information processing.

Translating signals into internal representations is not an easy task as the size of sensory cortices of mammals demonstrate. Understanding neurophysiology and computational neuroscience is the starting point, but the description language and the proper level of abstraction are very important to achieve a model that can be studied and implemented

using the tools of engineering. Freeman's model of the olfactory system shows how different the information processing solutions found in biology are from the conventional engineering approaches. Pragmatically, we should extract from this model what is essential, and discount the requirements of the "wetware." Unfortunately, we still are far from being able to understand the model to make this distinction. First and foremost, we lack a dynamic theory of information processing. Great strides and interest followed Hopfield's work on dynamic associative memories, but his model only utilizes the simplest of the singularities: the fixed point. Recently, analog processors have been built for image processing using dynamical processing elements [7], and chaos is being applied in communication systems [1] and signal processing [21]. Freeman [12], Haken [20], and many others are postulating nonconvergent dynamics for information processing, but we need to understand and quantify what is gained by increasing the order of the singularities, and the difficulties associated with chaotic attractors. The sensitive to initial conditions of chaotic attractors bring dynamic memory, but on the other hand require exactly the same inputs to reach the same attractor, which is obviously impossible with the variability of real-world signals. The role of network noise here seems crucial to smooth out the attractor

landscape. The intriguing aspect is that chaos appears as a control “knob” that can fine-tune the sensitivity-specificity curve of the system.

From an engineering point of view analyzing, simulating, and building coupled lattices of nonlinear oscillators is a challenge. Albeit an approximation, our method of digitizing the system dynamics is effective and yields digital models that describe to a first approximation the complex dynamic behavior. When the digital simulations are sufficiently accurate, we can do all this using digital computers. The issue is not the Runge–Kutta integration precision, but the discretization of phase space. Contrary to what is claimed in some papers [50], we do not see any constraint on using digital computers, provided integer arithmetic is utilized in the simulations. The bottleneck is the time to perform the simulations using integer arithmetic. Without a digital simulator, the parametrization of the analog model will be hopeless. Hence, further improvements to fully understand the nonlinear coupling are needed, as well as a mechanism to fine tune the parameters from data through adaptation.

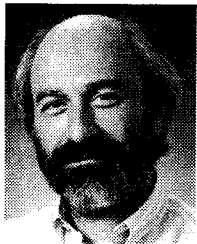
The analysis of the computational properties of the KIII model have barely started. We should quantify the capacity, the robustness to noise, and the speed of recall of the associative memory. For animats, we should also create a principled readout of information from the KIII model, something that biology did not bother to create since the information is used internally by the animal. We should also understand how to go from a simple associative memory paradigm to higher order cortical functions needed for roaming, goal driven behavior, and decision making. But the flexible architecture of the KIII model already incorporates coupling from central brain structures that will be implemented by the animats’ symbolic processing stage.

We showed that each piece of Freeman’s model is easy to implement, and the design has low power and small area. Our F&H circuit enables analog amplitude and discrete time implementations to facilitate interconnectivity. We are now testing a full KIII model using the components described in this paper that will yield a first-generation $\Sigma\Delta$.

REFERENCES

- [1] H. Abarbanel, *Analysis of Observed Chaotic Data*. New York: Springer, 1996.
- [2] M. Abeles, *Corticomics*. Cambridge, U.K.: Cambridge Univ. Press, 1991.
- [3] D. Amit, *Modeling Brain Function*. New York: Cambridge Univ. Press, 1989.
- [4] D. Ballard, *An Introduction to Natural Computation*. Cambridge, MA: MIT Press, 1997.
- [5] M. Beckerman, *Adaptive Cooperative Systems*. New York: Wiley, 1997.
- [6] R. Brodersen, P. Gray, and D. Hodges, “MOS switched-capacitor filters,” *Proc. IEEE*, vol. 67, pp. 61–75, Jan. 1979.
- [7] L. Chua, *CNN: A Paradigm for Complexity*. Singapore: World Scientific, 1998.
- [8] B. de Vries, “Temporal processing with neural networks: The development of the gamma model.” Ph.D. dissertation, Univ. Florida, 1991.
- [9] D. Donoho, “Sparse components of images and optimal atomic decomposition.” Stanford Univ., Tech. Rep., 1998.
- [10] J. Eisenberg, W. Freeman, and B. Burke, “Hardware architecture of a neural network model simulating pattern recognition by the olfactory bulb,” *Neural Networks*, vol. 2, pp. 315–325, 1989.
- [11] W. Freeman, *Mass Action in the Nervous System*. New York: Academic, 1975.
- [12] ———, “Tutorial on neurobiology from single neurons to brain chaos,” *Int. J. Bifurcation Chaos*, vol. 2, no. 3, pp. 451–482, 1992.
- [13] W. Freeman, Y. Yao, and B. Burke, “Central pattern generating and recognizing on olfactory bulb: A correlation learning rule,” *Neural Netw.*, vol. 1, pp. 277–288, 1988.
- [14] W. Freeman, “Mesoscopic neurodynamics: From neuron to brain,” *J. Physiol.*, 2000, to be published.
- [15] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. New York: Academic, 1990.
- [16] L. Giles, C. Miller, D. Chen, H. Chen, G. Sun, and Y. Lee, “Learning and extracting finite state automata with second order recurrent neural networks,” *Neural Comput.*, vol. 4, pp. 393–405, 1992.
- [17] G. Goodwin and K. Sin, *Adaptive Filtering Prediction and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
- [18] S. Grossberg, “Pattern learning by functional-differential neural networks with arbitrary path weights,” in *Delay and Functional Differential Equations and Their Applications*. Schmitt, Ed. New York: Academic, 1972, pp. 121–160.
- [19] S. Grossberg, *Studies of Mind and Brain*. Amsterdam, The Netherlands: Reidel, 1982.
- [20] H. Haken, *Principles of Brain Function*. New York, 1996.
- [21] S. Haykin and S. Puthusserypady, “Chaotic dynamics of sea clutter,” *Chaos*, vol. 7, no. 4, pp. 777–802, 1997.
- [22] J. Haugeland, *Artificial Intelligence, the Very Idea*. Cambridge, MA: MIT Press, 1975.
- [23] C. Helstrom, *Elements of Signal Detection and Estimation*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [24] J. Hertz, A. Krogh, and R. Palmer, *Introduction to the Theory of Neural Computation*. Reading, MA: Addison-Wesley, 1991.
- [25] J. Hopfield, “Neurons with graded response have collective computational properties like those of two-state neurons,” in *Proc. Nat. Acad. Sci.*, vol. 81, 1984, pp. 3088–3092.
- [26] J. Hopcroft and J. Ullman, *Introduction to Automata Theory, Languages and Computation*. Reading, MA: Addison-Wesley, 1979.
- [27] J. Hughes, N. Bird, and I. Macbeth, “Switched currents—A new technique for analog sampled data signal processing,” in *Proc. IEEE Int. Symp. Circuits and Systems*, 1989, pp. 1584–1587.
- [28] K. Kancko, *Theory and Applications of Coupled Map Lattices*. New York: Wiley, 1993.
- [29] M. Kearns and V. Vazirani, *An Introduction to Computational Learning Theory*. Cambridge, MA: MIT Press, 1994.
- [30] T. Kohonen, *Self-Organization and Associative Memory*. New York: Springer-Verlag, 1988.
- [31] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient based learning applied to document recognition,” *Proc. IEEE*, vol. 86, pp. 2278–2324, Nov. 1998.
- [32] *Matlab Users Manual*. Cambridge, MA: Mathworks, 1998.
- [33] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.
- [34] J. Mendel and K. Fu, *Adaptive Learning and Pattern Recognition Systems*. New York: Academic, 1970.
- [35] “Neurosolutions,” in *Neurosolutions Manual*. Gainesville, FL: NeuroDimensions Inc., 1995.
- [36] A. Oppenheim and R. Schaffer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [37] P. Peretto, *An Introduction to the Modeling of Neural Networks*. Cambridge, U.K.: Cambridge Univ. Press, 1994.
- [38] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical recipes: The art of scientific computing*. Cambridge, U.K.: Cambridge Univ. Press, 1986.
- [39] J. Principe, B. de Vries, and P. Oliveira, “The gamma filter a new class of adaptive IIR filters with restricted feedback,” *IEEE Trans. Signal Processing*, vol. 41, no. 2, pp. 649–656, 1993.
- [40] L. Rabiner and R. Schaffer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.
- [41] J. Rissanen, *Stochastic Complexity in Statistical Inquiry*. Singapore: World Scientific, 1989.
- [42] F. Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Washington, DC: Spartan, 1962.
- [43] L. Scharf, *Statistical Signal Processing*. MA: Addison-Wesley, 1991.
- [44] G. Shepard, *The Synaptic Organization of the Brain*. London, U.K.: Oxford Univ. Press, 1990.
- [45] L. Shilnikov, “Strange attractors and dynamical models,” *J. Circuits Syst. Comput.*, vol. 3, pp. 1–10, 1993.

- [46] K. Steiglitz, "The equivalence of digital and analog signal processing," *Inform. Contr.*, vol. 8, pp. 455–467, 1965.
- [47] V. Tavares, J. Principe, and J. Harris, "F&H filter: A novel ultra-low power discrete time filter," *Electron. Lett.*, vol. 35, pp. 1226–1227, July 1999.
- [48] V. Tavares, J. Principe, and J. Harris, "A silicon olfactory bulb oscillator," in *IEEE Int. Symp. Circuits and Systems*, vol. V, Switzerland, 2000, pp. 397–400.
- [49] P. J. Werbos, "Back propagation through time: What it does and how to do it?," *Proc. IEEE*, vol. 78, pp. 1550–1560, Oct. 1990.
- [50] Y. Yao and W. Freeman, "Model of biological pattern recognition with spatially chaotic dynamics," *Neural Netw.*, vol. 3, pp. 153–170, 1990.
- [51] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer, 1995.
- [52] J. Von Neumann, *Theory of Self-Reproducing Automata*, Burks, Ed. Urbana: Univ. Illinois Press, 1966.
- [53] W. B. Wilson, H. Z. Massoud, E. J. Swanson, R. George Jr., and R. B. Fair, "Measurement and modeling of charge feedthrough in n-channel MOS analog switches," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 1206–1213, Dec. 1985.



Jose C. Principe (Fellow, IEEE) is a Professor of Electrical and Computer Engineering and Biomedical Engineering at the University of Florida, Gainesville, FL, where he teaches advanced signal processing, machine learning and artificial neural networks (ANNs) modeling. He is a BellSouth Professor and the Founder and Director of the University of Florida Computational NeuroEngineering Laboratory (CNEL). His primary area of interest is processing of time varying signals with adaptive neural models.

The CNEL Lab has been studying pattern recognition principles based on information theoretic criteria (entropy and mutual information), the use of ANNs for dynamical modeling, and speech and object recognition applications. He is collaborating with neuroscientists in a computational model of the olfactory cortex toward its analog VLSI implementation. He has more than 70 publications in refereed journals, ten book chapters, and 160 conference papers. He directed 35 Ph.D. dissertations and 45 Master thesis. He recently wrote an interactive electronic book entitled *Neural and Adaptive Systems: Fundamentals Through Simulation* (New York: Wiley).

Dr. Principe is Secretary of the Technical Committee on Neural Networks of the IEEE Signal Processing Society, member of the Board of Governors of the International Neural Network Society, and Editor in Chief of the IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING. He is a member of the Advisory Board of the University of Florida Brain Institute.



Vitor G. Tavares received the Licenciatura and M.S. degrees in electrical engineering from the University of Aveiro, Portugal, in 1991 and 1994.

He is presently completing the Ph.D. degree in electrical engineering from the Computational NeuroEngineering Laboratory at the University of Florida, Gainesville. His interests are in the area of neuromorphic chip design and biomimetic computing.



John G. Harris received the B.S. and M.S. degrees in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge, in 1983 and 1986, respectively, and the Ph.D. degree in computation and neural systems from the California Institute of Technology in 1991. After a two-year post doc at the MIT Artificial Intelligence Lab, he joined the Electrical and Computer Engineering Department at the University of Florida, Gainesville, in 1993. He currently leads the UF Hybrid Signal Processing Group in researching analog circuits and DSP

for sensory processing.

Dr. Harris is the recipient of an NSF CAREER Award as well as the Teaching Improvement Program award at the University of Florida.



Walter J. Freeman (Fellow, IEEE) studied physics and mathematics at the Massachusetts Institute of Technology, English and philosophy at the University of Chicago, medicine at Yale University (M.D. 1954), internal medicine at Johns Hopkins University, and neurophysiology at the University of California, Los Angeles.

He has taught brain science at the University of California, Berkeley, CA, since 1959, where he is Professor of the Graduate School. He is the author of *Mass Action in the Nervous System* (1975), *Societies of Brains* (1995), *How Brains Make Up Their Minds* (1999), and *Neurodynamics: An Exploration of Mesoscopic Brain Dynamics* (2000).