"CHAOTIC RELAXATION" IN CONCURRENTLY ASYNCHRONOUS NEURODYNAMICS

Jacob Barhen[†] **Sandeep Gulati**[‡]

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 91109

ABSTRACT

A fundamental issue which directly impacts the scalability of current theoretical neural network models to applicative hardware embodiments is the inherent and unavoidable concurrent asynchronicity of massively parallel systems. We propose a mathematical framework for reconditioning additive-type models and derive a neuro-operator, based on the chaotic relaxation paradigm, whose resulting dynamics is neither "concurrently" synchronous nor "sequentially" asynchronous. Necessary and sufficient conditions guaranteeing concurrent asynchronous convergence are established in terms of contracting operators. Lyapunov exponents are also computed to charaterize the network dynamics and to ensure that throughput-limiting "chaotic" behavior in models reconditioned with concurrently asynchronous algorithms was eliminated.

1. Introduction

Advances in our understanding of physical, electrical and organizational processes occurring in biological systems, alongwith fundamental theoretical contributions by Grossberg [11], Kohonen [16], Amari [2], and Hopfield [13,14], have helped demonstrate the performance potential of artificial neural networks in solving traditionally hard problems in pattern analysis, associative recall, adaptive control and *in situ* learning. Hopfield's illuminating contributions have extended the applicability of neuromorphic techniques to the solution of combinatorially complex optimization problems [15]. In the areas of VLSI and opto-electronic implementations, major achievements have resulted from the efforts of Mead [19], Psaltis and Farhat [24] and others. This situation provides a strong incentive to explore the development of applicational artificial neural systems for problem domains where the traditional algorithmic paradigms [1], e.g PRAM, PPS, Universal TM etc. fail to provide effective solution strategies.

The bulk of existing neural network models are defined as an aggregation of adaptive dynamical systems, interacting through a densely interconnected synaptic network. Computation to be performed by the network is then encoded in terms of the connection strengths between pairs of neurons. Based on mathematical idealizations to biological behavior, the updating regimes of existing models, both discrete and continuous, may be classified into two basic algorithmic modes [9,13]. A ***concurrent synchronous*** mode, where all neurons are updated simultaneously and an ***asynchronous*** mode, wherein only one randomly selected neuron is allowed to update its state on the basis of its inputs. The firing decision for a particular neuron is allowed only after state information has been received from all other neurons to which it is connected. However, computational connotation of asynchronicity implies uncoordinated systemwide concurrent activity, while the biological manifestation of global asynchrony results from delays in nerve signal propagation, refractory periods and adaptive thresholding [10]. So, under their current framework, neural network models *per se* do not compute using a systemwide on-off uncoordinated switching (with random delays) of individual neurons. In fact, asynchrony as discussed by [9,13], in the context of existing artificial neural networks is essentially a "sequential randomness", lacking implicit concurrency. Thus, the paradigmatic advantages of neuronal processing, that essentially stem from an ability to perform massively parallel, asynchronous and distributed information processing, cannot be fully realized under the existing neurodynamical relaxation models.

We analyze the implications of this lack of ***true*** behavioral asynchrony at two levels, namely, problems encountered during digital VLSI, optical or opto-electronic computations and during discrete-time simulations on large-scale asynchronous computational ensembles. Concurrent synchronous activation requires complex global synchronization circuitry to neutralize the clock skew effects resulting from the variations in the physical [19] or optical path lengths [22,25] of the actual synaptic interconnections. This extra circuitry would limit the overall network performance to operate at the rate of slowest neuron. An alternate strategy involves clocking the system at a time constant slower

† Also with the Division of Engineering and Applied Sciences, California Institute of Technology.

‡ Visiting Scientist from the Robotics Research Lab., Dept. of Computer Science, Louisiana State University, Baton Rouge, LA 70803

than the slowest neuron, but again, throughput suffers. Not only does such circuitry lack a biological basis, it also enforces rigid firing sequences that are often difficult to sustain because of signal leakages and component instability. Macukow et al [18], showed that in large-scale networks such self-induced pathological activation could destabilize the entire neuromorphic system.

Even though the computational and paradigmatic gains expected from neuromorphic architectures will manifest from systems built around massively parallel and analog hardware, discrete-time simulations on asynchronous multiprocessors remain the primary benchmarking testbed for large-scale problems, and therefore, the algorithmic implications of the "sequential" nature of neuronal interrogation in simulations cannot be ignored. In general, during synchronous computation the processors must communicate their partial results to each other, at every instance of time specified by the precedence-constrained task graph obtained from the problem decomposition [4]. Hence, the distributed concurrent algorithms are mathematically equivalent to the sequential algorithms. These overheads, in the form of load imbalance due to processor inactivity, lower processor utilization and enhance resource contention due to communication and coordination requirements, and lead to a severe performance degradation in real-time neural network applications [5]. For instance, simulating the sequentially asynchronous nature of backpropagating networks on a concurrent computer introduces latencies, for which specific time bounds can be obtained in terms of the critical path of the corresponding task graph. Additionally, the existing approaches are lacking in fault tolerance, as updating decisions by a particular neuron require global interrogation, i.e., the status of each neuron to which it is connected. Failure to receive firing input from some inoperative neuron in the sequentially asynchronous setup could lead to blocking of the entire network. Thus, a model is necessitated wherein each neuron is associated with a decision algorithm that requires only local information to reach globally optimal decisions, as in the cellular automata approach. This also precludes the necessity for neural signals or activation potentials to remain stable for long intervals as in synchronous implementation.

In this paper we introduce a mathematical framework for reconditioning artificial neural network algorithms such that their embodiments are truly asynchronous. For illustrative purposes, the following discussion will focus on simple additive-type networks, but the ideas generalize in a straightforward fashion to more complex (e.g., shunting type) models. Hereafter, we do not distinguish between the simulation of a neural network on a concurrent computer or its subsequent hardware implementation, i.e., a neuron is considered as a "virtual" computing processor. Besides yielding a closer emulation of biological information processing, this approach is expected to providance guidance for large-scale fabrication of concurrent hardware. The organization of the remaining paper is as follows. In section 2 we present a characterization of asynchronous iterative computation and introduce the chaotic relaxation paradigm. Section 3 describes the reformulation of the Hopfield model in terms of contracting neuro-operators. We further define the necessary and sufficient conditions for convergence. To guide the implementation and validation of our asynchronous neuro-operator, we simulate an associative memory model in a concurrent environment in section 4. In section 5, we compute the Lyapunov exponents, for both the existing and modified model, to provide a fundamental isight to the network dynamcis and to dispel potential misgivings as to the main origin of oscillatory behavior observed hitherto.

2. Chaotic Relaxation Paradigm

In order to obviate the throughput-limiting "Feigenbaum bottleneck" arising from an extensive usage of the cooperative problem-solving approach, and the resulting "sequentiality" in the neurodynamical activation profile, we introduce a *chaotic relaxation* paradigm in the neural network dynamics. It is inspired from the seminal work of Chazan and Miranker [8], who showed that chaotic relaxation schemes could significantly reduce programming effort, communication overheads and turnaround time during concurrent computing on asynchronous multiprocessors. In our effort to design truly asynchronous neural networks we further draw motivation from *fixed point* techniques by Baudet [6], Miellou [20], Kung [17] and Bertsekas [4]. But before introducing the chaotic relaxation schema in conjunction to neurocomputation, we briefly summarize key attributes abstracted from concurrent asynchronous computational algorithms, to reinforce the paradigmatic divergence of neurodynamic relaxation in existing models from the biological phenomena and future hardware embodiments.

2.1 Concurrent Asynchronous Computation

In contrast to the synchronized iterative techniques (see Kung [17]), the execution profile of concurrent asynchronous algorithms is not constrained by the underlying task decomposition graph for the problem [4]. Concurrent tasks capable of uncoordinated execution are implemented as a collection of functionally, but not dynamically, cooperating processes, with no explicit dependencies to enforce waiting at synchronization points for the purpose of swapping partially computed results. Thus, instead of waiting for specific inputs from other tasks, they may continue, or terminate according to whatever information is available in the state variables. The computation *per se* is essentially iterative in nature, with the dynamics controlled by state variables and, possibly, previous history.

Thus, asynchronous computation provides an implicitly effective strategy for designing systems capable of delivering high throughput and real-time operational responses, since the synchronization and coordination restrictions are eliminated, and computations can be carried out without having to wait to receive all the messages implied by the precedence constraints. Also, the chaotic relaxations during asynchronous computation stagger data communication and memory accesses, alleviating the Von Neumann bottleneck [3]. In a neural network model, this implies that in contradistinction to the existing schema, computational functions could be implemented using neurons that are allowed to fire without having to wait to receive excitatory or inhibitory input signals from **all** other neurons to which they are connected, in order to evaluate if a firing threshold is exceeded.

In addition, asynchronous dynamics may lead to *true* fault tolerance, as it will enable neurons to remain idle for finite periods. This is analogous to the existence of "refractory" of recharging period in biological neurons [10]. But more important is the implication for hardware embodiments. Elimination of inter-neuron dependence, facilitates immediate replacement (or rerouting) of the failed segment, and resumption of processing without disturbing or reinitializing the entire configuration. Another advantage is the potential for implementation on large-scale heterogeneous computational ensembles, i.e., systems in which the different processing nodes may have different performance capabilities, to achieve hierarchical neuronal processing. The latter ability would lead to a reduction of complexity in interleaving operations, to provide for unpredictable activity fluctuations during neurocomputation. In summary, concurrently asynchronous dynamics defines an operational framework that implicitly confers to that essential for neurocomputation.

2.2 Concurrent Asynchronous Neurodynamics

In the subsequent development of our theory on concurrently asynchronous neural networks, we adopt a terminology in line with the generalized definition of chaotic iterations, originally introduced by Chazan and Miranker [8], and later generalized by Baudet [6]. Consider an additive-type neural network with N neurons, and let $\bar{u}$ denote the continuous-valued configuration vector of neuron activations in $\Re^n$. Let the components of $\bar{u}$ be given by u_i, for $i = 1, \cdots, N$. A temporal state sequence in terms of the neural coordinates u_i will be denoted by $u_i(t)$, for $t = 0, 1, \ldots$ Let $\bar{\varphi}$ be the nonlinear network operator from $\Re^n$ to $\Re^n$, whose components will be expressed as $\varphi_i(u_1, u_2, \cdots, u_N)$.

A concurrently asynchronous neural iteration, denoted by the tuple ($\bar{\varphi}$, $\bar{u}(0)$, ξ, ψ), corresponding to the neuro-operator $\bar{\varphi}$, and starting with a given vector $\bar{u}(0)$, is then a sequence of state iterates, $\bar{u}(t)$, of vectors on $\Re^n$, defined recursively by:

$$u_i(t) = \begin{cases} u_i(t-1) & \text{if } i \notin S_t \\ \varphi_i(u_1(x_1(t)), \ldots, u_n(x_n(t))) & \text{if } i \in S_t \end{cases} \tag{2.2.1}$$

where S_t: $1 \leq | S_t | \leq N$ denotes the set of neurons that update during the t–th iterate, and $x_i(t)$, indexes the availability of the i–th neuron's ***most recent*** updated state. Previous updating regimes implicitly assumed $x_i(t) = t - 1$. The set $\xi = \{S_t \mid t = 1, 2, \cdots \}$ is a sequence of ***nonempty*** subsets of neurons, that fired during each successive iterate. Also, $\psi = \{ (x_1(t), \cdots, x_N(t)) \mid t = 1, 2, \cdots \}$ denotes the latest update configuration for the network with respect to the t–th iterate. In addition, the following assumptions are made on sets ξ and ψ :

(a) $x_i(t) \leq t - 1$, $t = 1, 2, \cdots$, [6] i.e., each subsequent neuronal update uses only previously available state information;

(b) $x_i(t)$, considered as a function of t, tends to infinity as t tends to infinity [8], i.e., more and more recent state information must be used in evolving the set of neurons in the network;

(c) non-starvation condition [8], i.e., i occurs infinitely many often in the update-sets S_t, for $t = 1, 2, \ldots$, i.e., $\exists\, s < \infty$ such that each neuron is considered at least once in every s successive updates.

Manifestation of chaotic relaxation in networks with concurrent asynchronous updating can be intuited as follows: at some operating instant t, an idle neuron, i, initiates the update of its state to $u_i^t = \varphi_i(\bar{u}_{N-\{i\}}^{t-1})$. If the state $\bar{u}(t)$ differs from $\bar{u}(t-1)$ by a set of components $\{ u_i \mid i \in S_t \}$, then the i–th neuron may update itself using state information already available from the previous updates, and not wait to receive the results of ongoing activations. The precise strategy for selecting state information available depends on the degree of synchronization desired in the system. For example, in a fully asynchronous operation, the most recently available states could be selected. Alternately, in the vein of Chazan and Miranker [8], a more restricted selection could be considered, which limits the choice of available components to those that are produced no prior to some fixed number, k, of steps, such that for $t = 1, 2, \cdots$, the inequality $t - x_i(t) \leq k$ be satisfied. We now state a criterion for the asynchronous convergence of the neuro-operator, $\bar{\varphi}$.

2.3 Contraction Theorems

The concept of contraction plays a fundamental role in the iterative solution of nonlinear equations. It is most useful [23] to express contraction in terms of vector norms, which induces a partial ordering on $\Re^n$.

Definition [23] : An operator $\bar{\varphi} : D \subset \Re^n \rightarrow \Re^n$ is called a Φ-contraction on a set $D_o \subset D$, if there exists a linear operator $\Phi \in L(\Re^n)$ with the following properties :

[i] $| \bar{\varphi}(\bar{u}) - \bar{\varphi}(\bar{v}) | \leq \Phi | \bar{u} - \bar{v} | \quad \forall \bar{u}, \bar{v} \in D_o$

[ii] $\Phi \geq 0$

[iii] $\rho(\Phi) < 1$

The first property implies Lipschitz-continuity; indeed Φ is often referred to as the Lipschitz matrix of $\bar{\varphi}$. The latter requirements (non-negativity and spectral radius of Φ) generalize the typical specification of the contractive constant used in conjunction with the usual norm on $\Re^n$. The existance of a fixed point is then given by the following theorem.

Contraction-Mapping [23] : Suppose that $\bar{\varphi} : D \subset \Re^n \longrightarrow \Re^n$ is a Φ-contraction on the closed set $D_o \subset D$, such that $\bar{\varphi}(D_o) \subset D_o$. Then, for any $\bar{u}(0) \in D_o$, the sequence $\bar{u}(t+1) = \bar{\varphi}[\bar{u}(t)]$, $t = 0, 1, \cdots$ converges to the only fixed point of $\bar{\varphi}$ in D_o, and the error estimate

$$| \bar{u}(t) - \bar{u}(\infty) | \leq (\mathbf{I} - \Phi)^{-1} \Phi \, | \bar{u}(t) - \bar{u}(t-1) |$$

for $t = 0, 1, \cdots$, holds.

Chazan and Mirankar [8] first applied these concepts to establish the convergence of asynchronous iterations. Their results were later generalized by Baudet [3].

Baudet's Theorem : *If $\bar{\varphi} : \Re^n \longrightarrow \Re^n$ is a Φ-contraction mapping on a closed subset $D \subset \Re^n$ and if $\bar{\varphi}(D) \subset D$, then any asynchronous iteration corresponding to $\bar{\varphi}$ and starting with a vector $\bar{u}(0) \in D$, converges to a unique fixed point of $\bar{\varphi}$ in D.*

We now derive necessary and sufficient conditions for convergence of the concurrent asynchronous neurodynamics characterized by Eqs. (2.2.1).

3. Asynchronous Neuro-Operator

Amari [2] and others [12,13,15] have shown that, in general, the phenomenology of nonlinear neural networks, modeled as adaptive dynamical systems, is essentially a phase space flow towards static attractors. Associative recall, combinatorial optimization, learning, etc. on the other hand, are merely different functional manifestations of this phenomenlogy, wherein the nature of activation neurodynamics exercises an integral regulatory influence on functional efficacy, stability and scalability. We exploit this commonality in dynamical behavior to derive in the sequel a contracting neuro-operator, that significantly enhances the scalability of existing systems to massively parallel asynchronous embodiments.

Consider the temporal evolution of a fully connected, additive-type neurodynamical system, e.g., a Hopfield model defined by the following system of coupled differential equations:

$$\dot{u}_i + a_i u_i = \sum_j T_{ij} g_j(\gamma_j u_j) + I_i. \qquad (3.1)$$

Here u_i represents the internal state (e.g., mean soma potential) of the $i-$th neuron, T_{ij} denotes the synaptic coupling from the $j-$th to the $i-th$ neuron and I_i is the external input bias. The sigmoidal function g_j modulates the neural response, γ_j denotes the transfer function gain for the $j-$th neuron and a_i represents the inverse of a characterisitic time constant or the decay scaling term. Let $\varphi_i(\bar{u})$ denote the $i-$th component of the asynchronous operator introduced in (2.2). Using Euler's difference approximation to the above system of continuous-time differential equations, i.e., $\dot{u}_i = (u_i^{t+1} - u_i^t) / \Delta$, where Δ denotes the discretization stepsize, the $i-$th component of the above defined Hopfield operator is given by

$$\varphi_i(\bar{u}) = u_i + \Delta\left[- a_i u_i + \sum_j T_{ij} \, g_j(\gamma_j u_j) + I_i \right] \qquad (3.2)$$

Then for any two phase-space coordinates, $\bar{u}, \bar{v}$ in the domain of attraction

$$\varphi_i(\bar{u}) - \varphi_i(\bar{v}) = (u_i - v_i) \, (1 - \Delta a_i) + \Delta \sum_j T_{ij} \, [g_j(\gamma_j u_j) - g_j(\gamma_j v_j)] \qquad (3.3)$$

On taking the vector norm, the above system yields,

$$| \varphi_i(\bar{u}) - \varphi_i(\bar{v}) | \leq | u_i - v_i | \cdot | 1 - \Delta a_i | + \Delta \sum_j | T_{ij} | \cdot | g_j(\gamma_j u_j) - g_j(\gamma_j v_j) | \qquad (3.4)$$

We assume that for each neuron the response function, g_j: $\Re \rightarrow [-1,+1]$, is of class C^1, and that $| g_j' | \leq 1$. This is obviously the case for the usually considered neural response functions , i.e., $g(\gamma u) = \tanh(\gamma u)$ or $g(\gamma u) = [1 + e^{-\gamma u}]^{-1}$. Then the Mean Value Theorem implies that there exists a $z \in \Re$ such that,

$$g_j(\gamma_j u_j) - g_j(\gamma_j v_j) = g_j'(z) \, \gamma_j \, (u_j - v_j)$$

Thus

$$| g_j(\gamma_j u_j) - g_j(\gamma_j v_j) | < | \gamma_j | \, | u_j - v_j | \quad (3.5).$$

Regrouping all terms, we obtain

$$| \varphi_i(\bar{u}) - \varphi_i(\bar{v}) | \leq | 1 - \Delta a_i | \cdot | u_i - v_i | + \Delta \sum_j | T_{ij} | \cdot | \gamma_j | \cdot | \bar{u}_j - \bar{v}_j | \quad (3.6)$$

Let us now define a matrix Φ as follows:

$$\Phi_{ij} = | 1 - \Delta a_i | \, \delta_{ij} + \Delta \cdot | \gamma_j | \cdot | T_{ij} | \quad (3.7)$$

We see that Φ is nonnegative; furthermore, since

$$| \varphi_i(\bar{u}) - \varphi_i(\bar{v}) | \leq \sum_j \Phi_{ij} \, | u_j - v_j |$$

or equivalently

$$| \bar{\varphi}(\bar{u}) - \bar{\varphi}(\bar{v}) | \leq \Phi \cdot | \bar{u} - \bar{v} | \quad (3.8)$$

we deduce that the neuro-operator $\bar{\varphi}$ is Lipschitzian with Lipschitz matrix Φ. From Baudet's theorem, for $\bar{\varphi}$ to converge to a fixed point in an appropriate basin of attraction, the spectral radius of Φ must be less than one. Now, using Bechenbach and Bellman's theorem [23] we can write

$$\min_{1 \leq i \leq N} \left\{ \frac{\sum_{j=1}^{N} \Phi_{ij} y_j}{y_i} \right\} < \rho(\Phi) < \max_{1 \leq i \leq N} \left\{ \frac{\sum_{j=1}^{N} \Phi_{ij} y_j}{y_i} \right\} \quad (3.9)$$

where $\bar{y}$ denotes any positive vector. In particular, we can choose all vector components equal. The contraction then translates into

$$\rho(\Phi) < 1 \iff \begin{cases} \max_i \sum_j \Phi_{ij} < 1 \\ \Phi_{ij} > 0 \end{cases} . \quad (3.10)$$

for all i, j. This induces constrained interrelationships between the values of a_i, Δ, γ_j and T_{ij}, i.e.,

$$\max_i \left\{ | 1 - \Delta a_i | + \Delta \sum_j | \gamma_j | | T_{ij} | \right\} < 1 \quad (3.11)$$

and

$$| 1 - \Delta a_i | \, \delta_{ij} + \Delta | \gamma_j | | T_{ij} | > 0 \quad (3.12)$$

To fix the ideas, without loss of generality, consider simplest the situation where all gain parameters are equal to γ. Then convergence under a concurrently asynchronous regime will be guaranteed if one chooses, e.g.,

$$[i] \qquad a_i < \frac{1}{\Delta}$$

$$[ii] \qquad \gamma < \frac{a_i}{\sum_j | T_{ij} |} \quad (3.13)$$

Notice that the latter inequality invalidates the often made "high gain" approximation, at least for chaotic relaxation regimes.

4. Illustrative Example: Concurrently Asynchronous Associative Memory

The asynchronous methodology developed in the preceding section, was implemented on a hypercube multiprocessor [3] for Hopfield's content associative memory model [14]. A fully-connected network with 128 neurons and six orthogonal patterns were used. Despite its conceptual simplicity, this model encompasses the paradigmatic essence of additive neuronal interactions, and has been extensively benchmarked in terms of correctness, efficacy, capacity and scalability [9,18,21]. The study precluded a critique on the model's functionality and focussed instead on the activation mode implications. In particular, our experimentation was aimed at the following objectives: (a) verify algorithmic correctness and asynchronous convergence in a concurrent processing environment; (b) analyze the implications of ill-conditioned parameters, e.g., violation of conditions necessary for contraction, Eqs. (3.13), and; (c) benchmark computational efficacy with respect to the existing updating regimes.

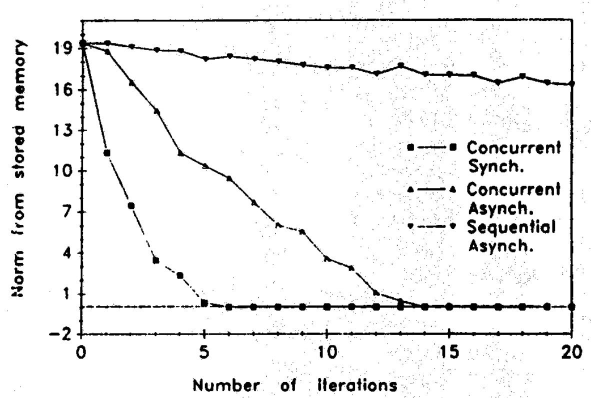

Figure 1: Concurent Simulation to benchmark relaxation rate with synchronous, sequentially asynchronous and concurrently asynchronous updating

Figure 1, juxtaposes the discretized temporal evolution of a noise-corrupted input probe to the nearest stored memory under synchronous, "concurrently" asynchronous (specified using Eqs. (2.2.1), (3.2) and (3.13)), and sequentially asynchronous updating. With proper conditioning, all three systems converged to the nearest stored memory. In this example, synchronous update yields the fastest rate of convergence. Recall, however, that it often suffers from, both ***horizontal*** as well as ***vertical*** oscillations [9]. In addition, it imposes severe clock synchronization constraints, the implications of which were outlined in section 1. Though the concurrently asynchronous update mode was simulated in a partially concurrent (e.g., 32 neurons / hypercube node) environment, convergence to the stored attractor indeed validates our methodology. Note that convergence in the latter case was achieved despite communication delays on the hypercube and globally inconsistent state information, i.e., neurons on different nodes operated assuming different states for the network. Also, as expected, the "sequentially" asynchronous mode led to the slowest convergence (over 1000 iterations).

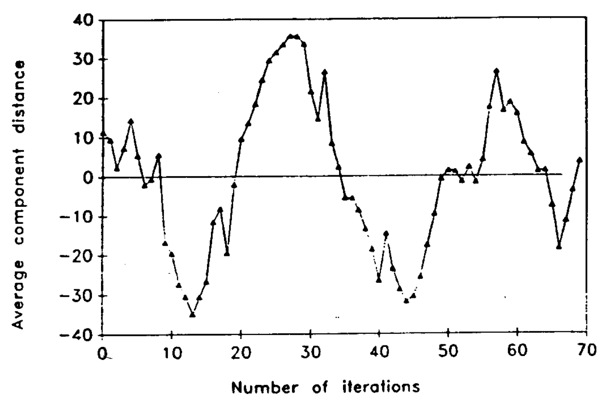

Figure 2 : Chaotic oscillations with improper conditioning

When the conditions, given by Eqs. (3.13) were violated, undesirable behavior abounded. The system failed to converge and oscillated instead, as depicted in figure 2. This behavior raises a hitherto unaddressed fundamental issue regarding network dynamics, i.e., the exact nature of noise observed in concurrent neural networks. Is it due to horizontal oscillations, vertical oscillations [9] or is it a manifestation of chaos, or merely numerical instability induced by discretizing continuous dynamical systems ? These issues are briefly taken up in the next section.

5. Elimination of "Emergent" Chaos

In contradistinction to prevalent notions on instability in neural networks [9], that attribute oscillatory behavior mainly to the tolpology of the interconnection matrix, we hypothesize that it is primarily a ***manifestation of "emergent chaos" induced by ill-conditioned parameters in the model.*** This hypothesis is strengthened by the following observations. Simulations have shown that the same type of model may exhibit radically different dynamical behavior with slightly different parameters. For example, small perturbations in time scales, delay distribution, transfer gain etc., may lead the system to oscillate back-and-forth from one basin of attraction to another. Also many authors commonly adopt a debugging strategy for neural network simulations which involve "anecdotal" manipulation of parameter space, essentially a combinatorial search for a system configuration that results in properly conditioned network dynamics.

To validate our hypothesis on the manifestation of "emergent chaos" we have computed the largest Lyapunov exponent of the time series obtained from a mapping which follows the evolution of the average component difference from a stored memory. A number of different approaches were additionally considered, and the detailed analysis is reported elsewhere [27]. Lyapunov exponents [26] essentially provide a dynamical diagnostic for measuring the exponential rates of convergence or divergence of phase trajectories. For a continuous dynamical system in n-dimensional phase space, the i-th dimensional Lyapunov exponent is defined as,

$$\lambda_i \;=\; \lim_{t\to\infty} \frac{1}{t} \log_2 \frac{p_i(t)}{p_i(0)} \qquad (5.1)$$

where $p_i(t)$ denotes the length of contracting / expanding principal ellipsoid axis, corresponding to the temporal deformation of the phase space. Any dynamical system characterized by a negative sum of Lyapunov exponents, but containing one or more positive terms is said to be chaotic, with the magnitude of such exponents reflecting the time scale on which the system dynamics becomes unpredictable [26]. Our simulations for an ill-conditioned neural model ($\Delta = 0.002$, $a = 1000$, $\gamma = 10000$ and $\sum \mid T_{ij} \mid = 28$) led to a value of $\simeq +91$ for the largest exponent. Since λ_1 is positive, we conclude the system to be if not chaotic, at least exponentially stochastic (since we did not compute the sum of all exponents). Also, when the largest Lyaponov exponent was determined for a contracting concurrently asynchronous network ($\Delta = 0.002$, $a = 100$, $\gamma = 1$), the value $\lambda_1 = -49$, was found, thereby proving that our conditioning methodology eliminates "emergent" chaos in concurrent neuromorphic models.

6. Conclusions

This paper presents a radically different insight into the neurodynamical implications of "sequentially" asynchronous and synchronous neuronal algorithms. Despite significant advances in concurrent hardware technology, full realization of the potential advantages of neural processing in solving real-life problems has been severely limited due to previous assumptions for asynchronicity. Electronic embodiments based on current mathematical frameworks lead to biological inconsistencies and require substantially complex circuitry. In a similar vein, such frameworks also limit the network scalability, stability and throughput in discrete-time simulations. It was hypothesized that, contrary to existing notions that attribute dynamical instability in the current models to the topology of the interconnection matrix, we ascribe it to "emergent chaos". Lyapunov exponents were computed to prove that improperly conditioned neurodynamical equations of motion do indeed exhibit chaotic relaxation behavior.

We exploited this insight to provide a strategy for systematically reconditioning the existing mathematical framework for additive networks, such that their VLSI, optical and opto-electronic embodiments are truly asynchronous, and thereby, eliminate the network instability ascribed to "emergent" chaos. We derived a neuro-operator that enables chaotic relaxations to achieve concurrently asynchronous updating. Necessary and sufficient conditions guaranteeing concurrently asynchronous convergence were defined in terms of Φ-contraction mappings. Lyapunov exponents were calculated for our proposed neuro-operator to ensure that the reconditioned system is devoid of chaotic behavior. Future directions include extension of our theory to shunting-type [8] neural networks. We also intend to theoretically analyze the implications of chaotic relaxation on network parameters, such as synaptic efficacies, transfer characteristics, network architecture and capacity.

Acknowledgements

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract NAS-918 with the National Aeronautics and Space Administration. Support for the work came from agencies of the U.S. Department of Defense, including the Innovative Science and Technology Office of the Strategic Defense Initiative Organization. We acknowledge the help of Phillip Alvelda in developing an early sequential simulation graphics package for the asynchronous memory model,and useful discussions with Mickail Zak from JPL and Jae Han from ORNL.

References

[1] A.V. Aho, J.E. Hopcroft and J.D. Ullman, "Design and Analysis of Computer Algorithms," Addison-Wesley, Readind, Mass., 1974.

[2] S. Amari, "Field Theory of Self-organizing Neural Networks," IEEE Trans. Syst. Man Cybern., Vol. SMC-13(5), pp. 741-748, 1983.

[3] J. Barhen and J. Palmer, "The Hypercube in Robotics and Machine Intelligence", Computers in Mech. Engg., Vol. 4(5), 39, 1986.

[4] J. Barhen and E.C. Halbert, " ROSES: An Efficient Scheduler for Precedence-Constrained Tasks on a Hypercube Multiprocessor," in Hypercube Multiprocessors '86, M.T. Heath, ed., SIAM Publishers, Philadelphia, Part III, Chap. 2, pp. 123 - 147, 1986.

[5] J. Barhen, N. Toomarian and V. Protopopescu, "Optimization of the Computational Load of a Hypercube Supercomputer Onboard a Mobile Robot," Applied Optics, Vol. 26(23), pp. 5007-5014, 1987.

[6] G.M. Baudet, "Asynchronous Iterative Methods for Multiprocessors," Journal of ACM, Vol. 25(2), pp. 226-244, 1978.

[7] D.P. Bertsekas, "Distributed Asynchronous Computation of Fixed Points", Mathematical Programming, Vol. 27, pp. 107-120, 1983.

[8] D. Chazan and W. Miranker, "Chaotic Relaxations," Linear Algebra and its Applications , Vol. 2, pp. 199-222, 1969.

[9] K.F. Cheung, L.E. Atlas and R.J. Marks II, "Synchronous vs Asynchronous Behavior of Hopfield's CAM," Applied Optics, Vol. 26(22), ,pp. 4808-4014, Nov. 87.

[10] M.Y. Choi, "Dynamic models of Neural Networks," Physical Review Letters, Vol. 61(24), pp. 2809-2812, Dec. 1988.

[11] S. Grossberg, "Some Nonlinear Networks Capable of Learning a Spatial Pattern of Arbitrary Complexity," Proc. of Nat'l Academy of Sciences , Vol. 29, pp. 368-826, 1983.

[12] M.W. Hirsch, "Convergence in Neural Networks," in Proc. of 1987 IEEE Int'l Conf. Neural Networks at San Diego, CA, June 19-24, 1987, Vol. II, pp. 115-125.

[13] J.J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Ability," Proc. of Nat'l Academy of Sciences, Vol. 79, pp. 2554-2558, 1982.

[14] J.J. Hopfield, "Neurons with Graded Response have Collective Computational Properties Like Those of Two-State Neurons," Proc. of Nat'l Academy of Sciences, Vol. 81, pp. 3058-3092, 1984.

[15] J.J. Hopfield and D.W. Tank, "Neural Computation of Decisions in Optimization Problems," Biological Cybernetics, Vol. 52, pp. 141-152, 1985.

[16] T. Kohonen, Self Organization and Associative Memory, Springer Verlag, New York: 1984.

[17] H.T. Kung, "Synchronized and Asynchronized Parallel Algorithms for Multiprocessors," Algorithms and Complexity: New Directions and Results, J.F. Traub, Ed. , Academic Press, New York, pp. 428-464, 1976.

[18] B. Macukow and H.S. Arsenault, "Modification of the threshold condition for a content-addressable memory based on the Hopfield Model," Applied Optics, Vol. 26(1), pp. 34-36, Jan. 1987.

[19] C. Mead, Analog VLSI and Neural Systems, Addison Wesley, Reading, MA, 1987.

[20] J.C. Miellou, "Asynchronous Iterations and Order Intervals," Parallel Algorithms & architectures, Elsevier Science Publishers B.V. (North Holland), ed. by M. Cosnard et al, 1986.

[21] C.M. Marcus and R.M. Westervelt," Stability of Analog Neural Networks with Delay," Physical Review A , (in press).

[22] S. Oh, L.E. Atlas, R.J. Marks II and D.C. Park, "Effects of Clock Skew in Iterative Neural Network and Optical Processors," in Proc. of 1988 IEEE Int'l Conf. on Neural Networks, San Diego, CA, July 24-27, 1988, Vol. II, pp. 429-435.

[23] J.M. Ortega and W.C. Rheinboldt, Iterative Solutions of Nonlinear Equations in Several Variables, New York: Academic Press, 1970.

[24] D. Psaltis and N.H. Farhat, "Optical Information Processing Based on an Associative-Memory Model of Neural Netws with Thresholding and Feedback," Optical Letters, Vol. 10, pp. 98-100, 1985.

[25] J. Shamir, "Fundamental Speed Limitations on Parallel Processing," Applied Optics, Vol. (26), pp. 1567-1568, 1987.

[26] I. Shimada and T. Nagashima, "A Numerical Approach To Ergodic Problem of Dissipative Dynamical Systems, " Progress of Theoretical Physics, Vol. 61(6), pp. 1605-1616, June 1979.

[27] J. Barhen, S. Gulati, and P. Alvelda, "Chaotic Neurodynamics in Massively Parallel Neural Networks," (in preparation).