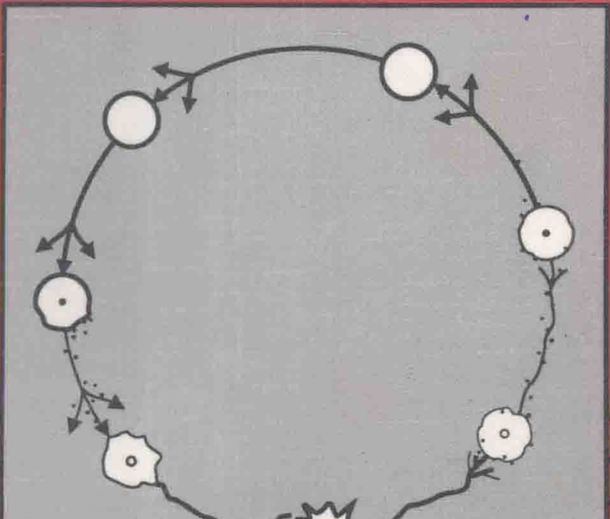
NEURAL NETWORKS: RESEARCH AND APPLICATIONS

Artificial Neural Systems

Foundations, Paradigms, Applications, and Implementations



Patrick K. Simpson

PERGAMON PRESS

ARTIFICIAL NEURAL SYSTEMS

Foundations, Paradigms, Applications, and Implementations

Patrick K. Simpson

General Dynamics Electronics Division,
San Diego

PERGAMON PRESS

Member of Maxwell Macmillan Pergamon Publishing Corporation

New York • Oxford • Beijing • Frankfurt • São Paulo • Sydney • Tokyo • Toronto

Pergamon Press Offices:

U.S.A.	Pergamon Press, Inc., Maxwell House, Fairview Park, Elmsford, New York 10523, U.S.A.
U.K.	Pergamon Press plc, Headington Hill Hall, Oxford OX3 0BW, England
PEOPLE'S REPUBLIC OF CHINA	Pergamon Press, Room 4037, Qianmen Hotel, Beijing, People's Republic of China
FEDERAL REPUBLIC OF GERMANY	Pergamon Press GmbH, Hammerweg 6, D-6242 Kronberg, Federal Republic of Germany
BRAZIL	Pergamon Editora Ltda, Rua Eça de Queiros, 346, CEP 04011, Paraiso, São Paulo, Brazil
AUSTRALIA	Pergamon Press Australia Pty Ltd., P.O. Box 544, Potts Point, NSW 2011, Australia
JAPAN	Pergamon Press, 8th Floor, Matsuoka Central Building, 1-7-1 Nishishinjuku, Shinjuku-ku, Tokyo 160, Japan
CANADA	Pergamon Press Canada Ltd., Suite 271, 253 College Street, Toronto, Ontario M5T 1R5, Canada

---

First edition 1990

Library of Congress Cataloging in Publication Data

Simpson, Patrick K.
Artificial neural systems : foundations, paradigms, applications, and implementations / Patrick K. Simpson.
p. cm. -- (Neural networks, research and applications)
Bibliography: p.
Includes index.
ISBN 0-08-037895-1 -- ISBN 0-08-037894-3 (pbk.)
1. Artificial intelligence. 2. Neural computers. I. Title.
II. Series.
Q335.S545 1989
006.3--dc20 89-33899
CIP

Printing: 2 3 4 5 6 7 8 9 Year: 1 2 3 4 5 6 7 8 9 0

Printed in the United States of America

The paper used in this publication meets the minimum requirements of American National Standard for Information Sciences -- Permanence of Paper for Printed Library Materials, ANSI Z39.48-1984

Preface

I was attending the University of California at San Diego (1983–1986) when I first became involved with artificial neural systems. My first exposure was a very frightening and frustrating experience. There was math on almost every page of every paper I found on the subject and every paper seemed to introduce a completely new set of terminology. What I really wanted, but could not find, was a book that described the fundamentals of artificial neural systems and outlined most of the primary paradigms. Quickly I discovered that no such text existed. In lieu of this deficit, I worked through several mathematical texts on topics that included differential equations, linear algebra, dynamical systems theory, probability and estimation theory, and calculus. Also, during this time, I read everything I could get my hands on that even remotely dealt with artificial neural systems.

Eventually these early efforts led to a technical document entitled, "A Survey of Artificial Neural Systems." I was working at Naval Ocean Systems Center (NOSC) at the time (1986–1987). NOSC was interested in learning more about artificial neural systems; so the group I was working for (Code 441) supported my efforts and allowed me to write a technical document on the subject.

The technical document was advertised on the ARPA-net's Neuron Digest and several hundred copies were distributed. One of these copies was read by Prof. Bourne at Vanderbilt University. Prof. Bourne liked the paper and invited me to expand it into a critical review for a new journal that he and Prof. Sztipanovits were editing for CRC Press entitled, *CRC Critical Reviews in Artificial Intelligence*. I eagerly accepted the professors' invitation and began to expand the NOSC technical document. Nine months and 270 pages later I had finished a two-part paper entitled, "A Review of Artificial Neural Systems." After the submission to CRC Press, I distributed copies of the paper at a conference and to other interested parties via the mail. One of these copies was sent to Prof. Gail Carpenter.

On the last day of August, 1988, I received a letter from CRC Press explaining that the journal was cancelled. Almost miraculously, during the same time period, Prof. Gail Carpenter contacted me and explained that she was one of the editors for Pergamon Press' *Neural Networks: Research and Applications* book series, and asked me if I would submit a prospectus. Of course I did, and the rest of the story is now self-evident.

The intention of this book is two-fold: (1) to provide the reader with a rapid introduction to the concepts and terminology of artificial neural systems, and (2) to provide a valuable reference guide to most of the primary artificial neural systems and the associated analyses, applications and implementations. In essence, I wrote the book I would have liked to have read when I started out in this field.

I have made a very diligent effort to give credit to the originator(s) of each paradigm. I have also tried to present the simplest form of each artificial neural system and provide references to the literature where the more complex versions can be found.

P.K.S.
San Diego, CA
April 30, 1989

Acknowledgments

There are many that have helped me to write this book in many different ways. I would like to thank General Dynamics Electronics Division for the use of their equipment and facilities to produce this book. I would like to thank J. Harold McBeth for his unwaivering support and his exceptional advice. I would like to thank Bruce Edson, Russell Deich, Richard McDuff, Karen Haines, and Stephen Luse for many hours of delightful and stimulating discussion. I would like to thank Steve Biafore for his able assistance in preparing the cover art. I would like to thank Prof. Bart Kosko for his invaluable advice, his unparalleled tutelage, and his supporting comments during all phases of this book's development. I would like to thank Prof. Gail Carpenter, my editor, for suggesting this book to Pergamon and working hard to see that it got equitable consideration. I would like to thank my son, Zachary, for helping me to unwind in those moments when the book was weighing heavily upon my mind. Most importantly, I would like to thank my wife, Christalyn, whose support, love, patience and advice truly made this book possible.

Contents

CHAPTER 1

Introduction

It has been a goal of science and engineering to develop intelligent machines for many decades. These machines were envisioned to perform all cumbersome and tedious tasks so that we might enjoy a more fruitful and enriched life. The technologies that have emerged to meet this challenge include cybernetics (Ashby, 1957; Weiner, 1948), machine learning (Nilsson, 1965), automata (Shannon & McCarthy, 1956; Tsetlin, 1973), bionics (WADD, 1960; Gawronski, 1971), mathematical biophysics (Rashevsky, 1948), general systems theory (Bertalanffy, 1968), self-organizing systems (Yovitz & Cameron, 1960; Yovitz, Jacobi & Goldstein, 1962), artificial intelligence (Minksy, 1961; Barr & Feigenbaum, 1981), cognitive science (Rumelhart & McClelland, 1986), and artificial neural systems.

Artificial neural systems (ANSs) are mathematical models of theorized mind and brain activity. ANSs are also referred to as neural networks, connectionism, adaptive systems, adaptive networks, artificial neural networks, neurocomputers, and parallel distribution processors. ANSs exploit the massively parallel local processing and distributed representation properties that are believed to exist in the brain. The primary intent of ANSs is to explore and reproduce human information processing tasks such as speech, vision, olfaction, touch, knowledge processing and motor-control. In addition, ANSs are used for data compression, near-optimal solutions to combinatorial optimization problems, pattern matching, system modeling, and function approximation.

The attempt to mechanize human information processing tasks highlights the classic comparison between the information processing capabilities of the human and the computer. The computer can multiply huge numbers at blinding speed, yet it cannot understand unconstrained speaker-independent speech. Human abilities complement those of the computer in that we can understand speech (even when heavily slurred in an extremely noisy environment), yet lack the ability to compute the square root of a prime number without the aide of pencil and paper—or a computer. The differences between the two can be traced to the processing methods each employs. Conventional computers rely on algorithm-based programs that operate serially, are controlled by a complex central processing unit, and store information at addressed locations in memory. The brain relies on highly distributed representations and transformations that operate in parallel, have distributed control through billions of highly interconnected neurons—or processing elements (PEs)—and appear to store their information in variable strength connections called synapses.

ANS theory is derived from many disciplines, including psychology, mathematics, neuroscience, physics, engineering, computer science, philosophy, biology, and linguistics. It is evident from this diverse listing that ANS technology represents a "universalization" among the sciences working toward a common goal—building intelligent systems. It is equally evident from the listing that an accurate and complete description of the work in all the listed disciplines is an impossible task. In light of this, we will focus upon ANS paradigms that have applications, application potential, or infrastructural significance. Hence, this book will not fully discuss the significant work in biology (Kandel, 1979; Kandel & Schwartz, 1985), philosophy (Churchland, 1986), linguistics (Arbib & Caplan, 1979), or psychology (Grossberg, 1982a; Rumelhart & McClelland, 1986).

This book contains an explanation of the relationship between real neural systems and artificial neural systems, an overview of the foundational concepts, constructs, and terminology used to describe ANS models, descriptions of 27 ANS paradigms organized into a coherent taxonomy, and a brief history of the field from the early 1900's to the present. Each ANS paradigm includes a mathematical, and sometimes an algorithmic, characterization; a discussion of its strengths and limitations; and a brief description of its current and potential applications and implementations.

The goal of this book is to provide an overview of ANS technology and several of its key paradigms. We have reserved judgment and opinion concerning each model's overall comparative worth, choosing instead to focus upon its mathematical and algorithmic principles, its applications, and its implementations.

We would be remiss not to mention previous surveys of this field. Each review varies in its emphasis and completeness. The list includes Arbib (1964, 1972), Barto (1984), Grossberg (1988a), Feldman & Ballard (1982), Levine (1983), Lippman (1987), Miller (1987), and Rumelhart & McClelland (1986).

CHAPTER 2

Comparing Real and Artificial Neural Systems

2.1. SIMPLIFIED NEURON

The human information processing system consists of the biological brain. The basic building block of the nervous system is the neuron, the cell that communicates information to and from the various parts of the body. Figure 2-1 shows a simplified representation of a neuron. The neuron consists of a cell body called a soma, several spine-like extensions of the cell body called dendrites, and a single nerve fiber called the axon that branches out from the soma and connects to many other neurons. Inside and around the soma are ions including sodium (Na^+), calcium (Ca^{++}), potassium (K^+), and chloride (Cl^-). The K^+ concentrates inside the neuron and the Na^+ concentrates outside. When the soma's membrane is electrically stimulated—usually by a voltage drop—its membrane allows the Na^+ and other ions such as Ca^{++} to pass across its membrane and change the soma's internal state. The connections between neurons occur either on the cell body or on the dendrites at junctions called synapses. A helpful analogy is to view the axons and dendrites as insulated conductors of various impedance that transmit electrical signals to the neuron (Arbib, 1964; Churchland, 1986; Kandel & Schwartz, 1985; Lindsay & Norman, 1977). The nervous system is constructed of billions of neurons with the axon from one neuron branching out and connecting to as many as 10,000 other neurons. All the neurons—interconnected by axons and dendrites that carry signals regulated by synapses—create a neural network.

The model neuron in its simplest form, shown in Figure 2-2, can be considered a threshold unit—a processing element that collects inputs and produces an output only if the sum of the input exceeds an internal threshold value. As a threshold unit, the neuron collects signals at its synapses and adds them together. If the collected signal strength is great enough to exceed the threshold, a signal is sent down the axon which abuts other neurons and dendrites. The soma adds all the signals, gated by the synapses from the impinging dendrites. The total signal is then compared to an internal threshold value of the neuron, and propagates a signal to the axon if the threshold is exceeded. Artificial neural systems are created by interconnecting many of these simple "neurons" into a network.

2.2. FORMAL ARTIFICIAL NEURAL SYSTEM DEFINITION

ANSs are neurally inspired models of brain and behavior (Arbib, 1964; Grossberg, 1982a & 1986a). Hecht-Nielsen (1988b) offers the following as a general, yet rigorous, definition of an artificial neural system:

> A neural network [ANS] is a parallel, distributed information processing structure consisting of processing elements (which can possess a local memory and carry out localized information processing operations) interconnected together with unidirectional signal channels called connections. Each processing element has a single output connection which branches ("fans out") into as many collateral connections as desired (each carrying the same signal—the processing element output signal). The processing element output signal can be of any mathematical type desired. All of the processing that goes on within each processing element must be completely local; i.e., it must depend only upon the current

values of the input signal arriving at the processing element via impinging connections and upon values stored in the processing element's local memory.

A simpler, but less rigorous, definition of an ANS is a nonlinear directed graph with weighted edges that is able to store patterns by changing the edge weights and is able to recall patterns from incomplete and unknown inputs (Simpson, 1987). The key elements of most ANS descriptions are the distributed representation, the local operations, and nonlinear processing. These attributes emphasize two of the primary applications of ANSs—*situations where only a few decisions are required from a massive amount of data* and *situations where a complex nonlinear mapping must be learned.*

It becomes obvious from these definitions that mathematics is the backbone of ANS theory, an idea perhaps best expressed by Arbib (1964, p. ii) in the following passage:

> We apply mathematics to derive far-reaching conclusions from clearly stated premises. We can test the adequacy of a model of the brain by expressing it in mathematical form and using our mathematical tools to prove general theorems. In the light of any discrepancies we find between these theorems and experiments, we may return to our premises and reformulate them, thus gaining a deeper understanding of the workings of the brain. Further, such theories can guide us in building more useful and sophisticated machines.

The mathematics most often used in ANS technology includes differential equations, dynamical systems, linear algebra, probability and statistics. Two excellent sources that cover the majority of these areas are texts by Hirsch & Smale (1974) and Papoulis (1965).

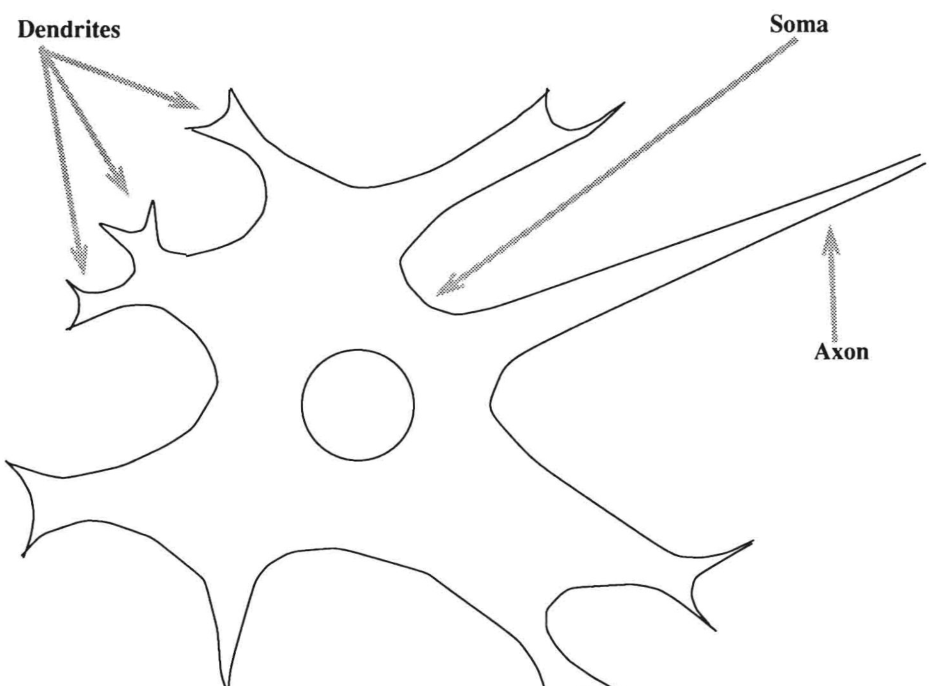


FIGURE 2-1. Simplified representation of a neuron. The neuron has a cell body with a nucleus, called the soma, an axon that carries the signal away from the neuron, and dendrites that receive the signal from other neurons.

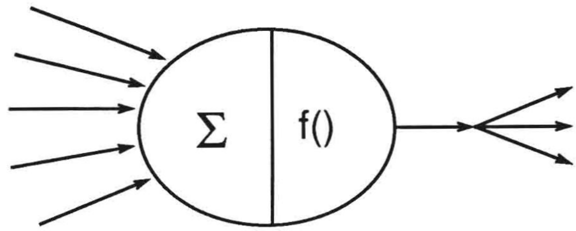


FIGURE 2-2. A neuron as a simple threshold unit. The incoming lines represent dendrites. Each line carries a signal that is added, Σ, together. After the addition, the signal is processed through a threshold function f(), which produces the output signal.

2.3. BRAIN VERSUS COMPUTER PROCESSING

Neural models are based on how the brain might process information. In reference to human information processing problems, the processes performed in the mind and in a conventional digital computer will now be compared to gain an understanding of why the mind is so adaptive and resilient and the computer is so rigid and precise. The reasons for using ANS models to solve human information processing and constraint optimization problems are highlighted by discussing these restrictions and assumptions of brain-style processing.

2.3.1. Processing Speed

Cycle time is the time taken to process a single piece of information from input to output. The cycle time of the most advanced computers, corresponding to processing one step of a program in the CPU during one clock cycle, occurs in the nanosecond range (~4.2 nanoseconds for a Cray 3). The cycle time for a neuron in the brain, corresponding to a neuronal event prompted by an external stimulus, occurs in the millisecond range (Cottrell & Small, 1984; Crick, 1979). The difference in speed is 10^6—the computer processes an element of information as much as a million times faster.

2.3.2. Processing Order

If the most advanced computers are able to process information one million times faster than the brain, why is the brain so superior for human information processing problems? The difference between the two can be traced to the processing order. The brain processes information in parallel and the computer processes information serially. Feldman (1985) has extended a constraint from this distinction called the 100-step program: "If the mind reacts in approximately half a second (500 milliseconds) to a given stimulus (i.e. answering a true-false question or naming a picture) and the cycle time of a neuron averages five milliseconds, then in 100 cycle times of a neuron a decision is reached." If we use the computer analogy that one step of a program is processed for each time cycle, then the brain runs parallel programs that are only 100 steps long. In contrast to large software programs operating in serial on conventional computers, the brain operates with massively parallel programs that have comparatively few steps, possibly explaining why the brain is superior at human information processing problems despite being as many as 6 orders of magnitude slower.

2.3.3. Abundance and Complexity

There are a massive number of neurons operating in parallel in the brain at any given moment. The number is estimated to be between 10^{11} and 10^{14}, each with between 10^3 and 10^4 abutting connections per neuron. If we consider the brain to be the ultimate ANS model, it appears that interesting applications will require a large number of processing elements (neurons). In addition to the number of neurons, studies have also found that the neuron is not a simple threshold unit, rather it is a complex computing device (Grossberg, 1982a & 1986a). Analysis has shown (Levy, 1982) that all the computing does not take place solely inside the soma; computations also occur outside the neuron body in the dendrites and at the synapses.

2.3.4. Knowledge Storage

Another distinction between the computer and the brain is knowledge storage. In the computer, a static copy of the knowledge being stored is placed in an addressed memory location. New information destroys old information. In contrast, because the number of connections between neurons in the brain is relatively fixed and very few new pathways are formed in the adult brain (Crick, 1979), information in the brain is thought to be stored in the interconnections between neurons (Grossberg, 1982a & 1986a). Moreover, it is felt that new information is added to the brain (i.e learned) by adjusting the interconnection strengths—the synaptic efficacy—between neurons. This adaptation premise provides a possible explanation for the brain's ability to generalize. In summary, knowledge in the brain is adaptable, while knowledge in the computer is strictly replaceable.

2.3.5. Fault Tolerance

The brain exhibits fault-tolerant characteristics. Damage (faults) to individual neurons can occur in the brain without a severe degradation of its overall performance (Hopfield, 1982; Hopfield, Feinstein, & Palmer, 1983; Hopfield, 1984; Lindsay & Norman, 1977). This graceful degradation is called fault-tolerance. Fault-tolerance supports the theory that the brain carries a distributed representation of the world and each concept or idea is not held in only one neuron, but rather spread across many neurons and their interconnections. If a portion of the brain is removed, the knowledge of the concept or idea is still retained through the redundant, distributed encoding of information. In contrast, most conventional computers are not fault-tolerant, instead they are fault-intolerant. Removing any processing component of a conventional computer leads to an ineffective machine, and the corruption of a conventional computer's memory is irretrievable and leads to failure as well.

2.3.6. Processing Control

The brain does not have any specific area with dictatorial control, rather the brain is an anarchic system (Crick, 1979; Feldman & Ballard, 1982; Feldman, 1985; Hecht-Nielsen, 1982 & 1983). There is no homunculus in the brain that monitors each neuron's activity. Each neuron's output is a sum of its synapse's activations thresholded by its signal functions; therefore, each neuron's output is a function of only its locally available information. Local information means each neuron only has access to the information contained in those neurons (or other neural processes) it is directly interconnected to and no others. In sharp contrast, the control in a conventional computer is completely autocratic. The computer's central processing unit monitors all activities and has access to global information, creating both a processing bottleneck and a critical point for failure.

CHAPTER 3

Foundations of Artificial Neural Systems

This chapter will introduce a fundamental nomenclature and the rudimentary mathematical concepts used to describe and analyze ANS processing. In a broad sense, ANSs consist of three elements: (1) an organized topology (geometry) of interconnected processing elements (i.e. a network or a neural system), (2) a method of encoding information, and (3) a method of recalling information. The following sections on processing elements and network topologies provide a description of neural system topologies. The sections on threshold elements and, again, processing elements provide an overview of ANS recall; and the sections on memory and learning will provide some insights concerning ANS encoding.

In addition to the three elemental ANS characteristics of topology, encoding, and recall, there are two other key concepts that play important roles and must be addressed—techniques for analyzing ANS dynamics and general taxonomy of all ANS paradigms that minimizes ambiguity. Each of these subjects will be addressed in separate sections.

3.1. PROCESSING ELEMENTS

Processing elements (PEs), also referred to as nodes, short-term memory (STM), neurons, populations, or threshold logic units, are the ANS components where most, if not all, of the computing is done. Figure 3-1 displays the anatomy of a generic PE. The input signals come from either the environment or outputs of other PEs and form an input vector $A = (a_1,\ldots,a_i,\ldots,a_n)$, where a_i is the activity level of the *ith* PE or input. Associated with each connected pair of PEs is an adjustable value (i.e., a system variable) called a weight (also referred to as a connection strength, interconnect, or long-term memory). The collection of weights that abuts the *jth* PE, b_j, forms a vector $W_j = (w_{1j},\ldots,w_{ij},\ldots,w_{nj})$, where the weight w_{ij} represents the connection strength from the PE a_i to the PE b_j. Sometimes there is an additional parameter Θ_j modulated by the weight w_{0j} that is associated with the inputs. This term is considered to be an internal threshold value that must be exceeded for there to be any PE activation. The weights W_j, their associated PE values A, and the possible extra parameter Θ_j, are used to compute the output value b_j. This computation is typically performed by taking the dot product of A and W_j, subtracting the threshold, and passing the result through a threshold function f(). (See next section for description of threshold functions.) Mathematically this operation (also shown in Figure 3-1) is defined as

$$b_j = f(A \cdot W_j - w_{0j}\Theta_j) \qquad \textbf{(3-1)}$$

or, in point-wise notation, as

$$b_j = f\left(\sum_{i=1}^{n} a_i w_{ij} - w_{0j}\Theta_j\right) \qquad \textbf{(3-2)}$$

3.2. THRESHOLD FUNCTIONS

Threshold functions, also referred to as activation functions, squashing functions, or signal functions, map a PE's (possibly) infinite domain—the input—to a prespecified range—the

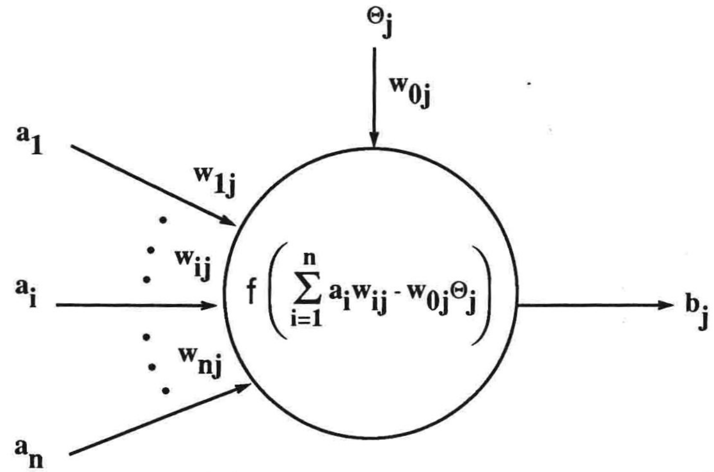


FIGURE 3-1. Topology of the generic PE b_j. Inputs form the vector $A = (a_1, \ldots, a_n)$. The threshold is Θ_j. Each input and threshold has a corresponding weight value w_{ij} which represents the connection strength from the *ith* input a_i to the *jth* output b_j with the threshold being assigned the weight w_{0j}. The collection of the weights forms the n + 1 dimensional vector $W_j = (w_{0j}, w_{1j}, \ldots, w_{nj})$. The output is shown as a function of its inputs, illustrating the local behavior of ANS PEs.

output. Four common threshold functions are the linear, ramp, step, and sigmoid functions. Figure 3-2 shows typical shapes of these functions.

The linear function, shown in Figure 3-2(a), has the equation

$$f(x) = \alpha x \tag{3-3}$$

where α is a real-valued constant that regulates the magnification of the PE activity x.

When the linear function is bounded to the range $[-\gamma, +\gamma]$, eq. 3-3 becomes the nonlinear ramp threshold function, shown in Figure 3-2(b) and described by the equation

$$f(x) = \begin{cases} +\gamma & \text{if } x \geq \gamma \\ x & \text{if } |x| < \gamma \\ -\gamma & \text{if } x \leq -\gamma \end{cases} \tag{3-4}$$

where γ $(-\gamma)$ is the PE's maximum (minimum) output value, a value commonly referred to as the saturation level. Note that although eq. 3-4 is a piece-wise linear function, it is often used to represent a simplified nonlinear operation.

If the threshold function responds only to the sign of the input, emitting $+\gamma$ if the input sum is positive and $-\delta$ if it is not, then the threshold function is called a step threshold function where δ and γ are positive scalars. The step threshold function shown in Figure 3-2(c) is mathematically characterized by the equation

$$f(x) = \begin{cases} +\gamma & \text{if } x > 0 \\ -\delta & \text{otherwise} \end{cases} \tag{3-5}$$

Often eq. 3-5 is binary in nature, emitting a 1 if $x > 0$, and 0 otherwise.

The final, and most pervasive, threshold function is the sigmoid threshold function. The sigmoid (S-shaped) function, shown in Figure 3-2(d), is a bounded, monotonic, non-decreasing function that provides a graded, nonlinear response. A common sigmoid function is the logistic function

$$S(x) = (1 + e^{-x})^{-1} \tag{3-6}$$

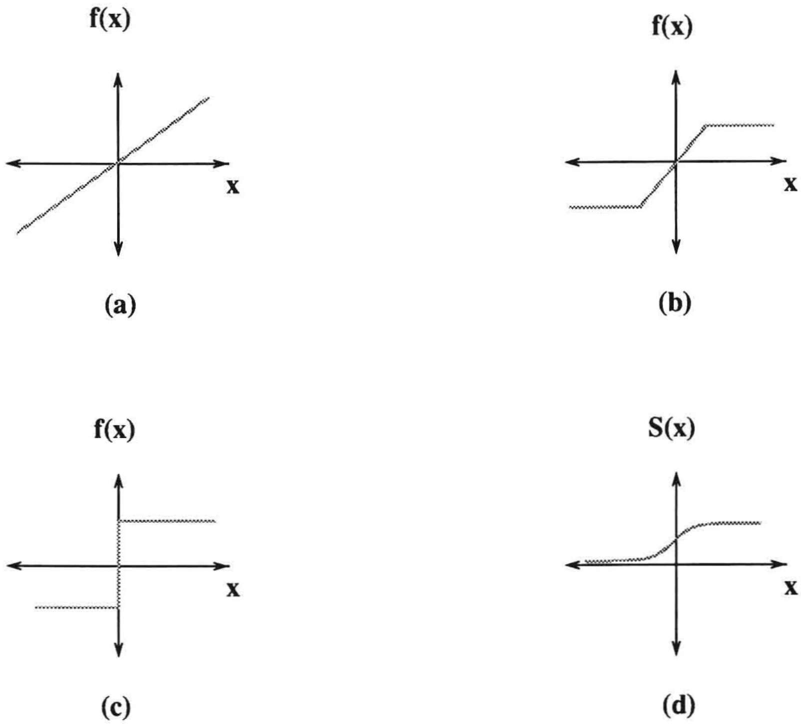


FIGURE 3-2. Four common threshold functions used in processing elements. These threshold functions are: (a) the linear threshold function, (b) the ramp threshold function, (c) the step threshold function, and (d) the sigmoid threshold function. Note that all except (a) are nonlinear functions. See text for a more detailed description.

This function is seen in statistics (as the Gaussian distribution function), chemistry (describing catalytic reactions), and sociology (describing human population growth). The saturation levels of eq. 3-6 are 0 and 1. Two other sigmoid functions are the hyperbolic tangent

$$S(x) = \tanh(x) \qquad \textbf{(3-7)}$$

which has saturation levels at -1 and 1; and the augmented ratio of squares

$$f(x) = \begin{cases} x^2/(1 + x^2) & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases} \qquad \textbf{(3-8)}$$

which has saturation levels at 0 and 1.

3.3. TOPOLOGY CHARACTERISTICS

ANS topologies, or architectures, are formed by organizing PEs into fields (also called slabs or layers) and linking them with weighted interconnections. Characteristics of these topologies include connection types, connection schemes, and field configurations.

3.3.1. Connection Types

There are two primary connection types, excitatory and inhibitory. *Excitatory* connections increase a PE's activation and are typically represented by positive signals. *Inhibitory* connections decrease a PE's activation and are typically represented by negative signals. These