Journal of Pure and Applied Algebra 63 (1990) 29-47
North-Holland

A NEURAL NETWORK WHICH COMPUTES THE SQUARE ROOT

Gerald S. EISMAN

Department of Computer Science, San Francisco State University, 1600 Holloway Avenue, San Francisco, CA 941132, U.S.A.

Communicated by J. Rhodes
Received July 1987

A model for neural networks is presented in which the steady state firing frequency of a neuron is related to the amount of feedback in the network. One such network is shown to compute the square root function of the input to the network.

Introduction

Current research into the computational capability of neural networks makes great use of feedback to achieve such varied effects as learning (e.g. Hinton et al. [4] or Minsky and Papert [6]) or the onset of chaos (Ermentrout [3]).

In this paper a model of neural networks will be presented in which the degree of feedback bears a strong relationship to the computational complexity of the network. Instead of considering the firing frequency of a neuron, we consider the firing ratio, i.e. the ratio of on pulses to off pulses, and then show how to construct networks with one level of feedback which can compute the sum or product of input ratios and a network with two levels of feedback which can compute the square root of input (in the sense that the limit in time of the firing ratio of a neuron converges to the desired value). The measure of degree of feedback will be a modification of the cycle rank (as in Eggan [1]) of the underlying digraph of the network.

In particular, it will be shown that if all input ratios are rational, then: for networks of feedback level 1 or less, the output ratios are rational; for networks of level 2 the output ratios belong to a field which can be obtained by a chain of quadratic extensions of the rationals; networks of level 3 can obtain ratios whose minimal polynomial over the rationals is unsolvable by radicals; and for every polynomial of degree n with rational coefficients which has at least one non-negative real root, there exists a net of level $2n$ which can obtain a ratio which is a root of the polynomial.

The structure and operation of the neural networks is essentially that developed in [2] where it was also shown that the degree of feedback corresponded to the capability of the network considered as a finite state machine.

0022-4049/90/$3.50

In Section 1 of this paper, preliminary notation, definitions, and theorems from algebra and graph theory are presented.

In Section 2, the neural network model is developed along with the major results on output ratios.

1. Preliminaries

The following definitions are modifications of what appeared in [2]:

Graph theory preliminaries

1.1. Definition. Let G be a digraph. A *strongly connected component of* G is a subgraph G' of G such that for every pair of distinct vertices v_1, v_2 of G' there is a path in G' from v_1 to v_2 and a path in G' from v_2 to v_1. A *section of* G is a maximal strongly connected component.

1.2. Definition. A strongly connected component S of a digraph G has *circular feedback of level* 0 if it consists of a single vertex with no edges to itself, and *circular feedback of level* $m>0$ if

(i) S does not have circular feedback of a level $<m$,

(ii) S consists entirely of disjoint strongly connected components $S_1, \ldots, S_k$ and edges $e_{12}, e_{23}, \ldots, e_{(k-1)k}, e_{k1}$ such that for each i and j, e_{ij} connects a vertex in S_i to a vertex in S_j, and

(iii) for $i=1 \ldots k$, S_i has circular feedback of level $\leq m-1$.

A digraph G has *circular feedback of level* i, denoted $\mathrm{CF}(G)=i$, if i is the maximum of the circular feedback of the sections of G.

Note that if a component S has circular feedback 1, then it consists of a simple closed path of individual vertices (components of level 0).

It is not immediately clear from the inductive definition of circular feedback that every digraph has circular feedback of level k for some k. The following proposition shows however that this is indeed the case:

1.3. Proposition. *For every digraph*, G, *there exists an integer* k *such that* G *has circular feedback of level* k.

Proof. Consider a section S of G. If S consists of a single node with no edges to itself, then S has circular feedback of level 0. If there is an edge in S connecting vertex v_1 say to vertex v_2, then there must be a path in S from v_2 back to v_1. Therefore, S contains a simple closed path. Let S_1 be the subgraph consisting of the vertices and edges of one such path. If S_1 is not all of S, then there must be another edge of S say from v_3 to v_4 where v_3 is in S_1. Then there must be a simple path from v_4 back to S_1 which does not go through a vertex of S_1 except at its termination point. (If v_4 is in S_1, then the path consists of v_4 with no edges.) Let S_2 be the subgraph of S consisting of S_1, the edge from v_3 to v_4, and the path from v_4 back to S_1. If S_2 is not all of S, then there must be an edge of S from v_5 to v_6 say where v_5 is in S_2. As before, there must be a simple path from v_6 back to S_2. Continuing in this manner, a sequence of subgraphs $S_1, S_2, \ldots$ is created which eventually exhausts all of S. Clearly, $\mathrm{CF}(S_1)=1$. Suppose it has been shown that $\mathrm{CF}(S_k)\leq k$. Then, since S_{k+1} consists of a component with circular feedback of level at most k in a circle with individual vertices (which are level 0), $\mathrm{CF}(S_{k+1})\leq k+1$. Therefore, each section of G has circular feedback of some level and hence so does G. □

1.4. Definition. Let G be a digraph. *secgraph* G is the digraph whose vertices are the set $\{S$: S is a section of $G\}$ and whose edges are the set $\{(S_1, S_2)$: there exists an edge in G from a vertex of S_1, to a vertex of $S_2\}$. *height* G is the maximum number of sections of maximum circular feedback level over all paths in secgraph G.

Algebra preliminaries

1.5. Definition. A chain of fields $F_m \supseteq F_{m-1} \supseteq \cdots \supseteq F_0 = Q$ (the rationals) is a *radical tower* if, for each $j=1,\ldots,m$, there exists a positive integer, n_j, an element $a_j \in F_{j-1}$, and a root α_j of $x^{n_j}-a_j$ such that $F_j = F_{-1}(\alpha_j)$. If, for $j=1,\ldots,m$, the integer $n_j \leq 2$, then F_m is called a *quadratic extension of* Q. F_m is a *real quadratic extension of* Q if it is a quadratic extension for which α_j is real for each j. The *real quadratic closure of* Q, denoted Q_{R}, is the union of all real quadratic extensions of Q.

1.6. Definition. Let F be a real quadratic extension of Q. Then *rootrank*(F) is 0 if $F=Q$ and *rootrank*(F) is $h>0$ if

(i) rootrank(F) is not less than h, and

(ii) there exists a real quadratic extension F_1 of Q with rootrank$(F_1)=h-1$ and $F=F_1(\sqrt{\alpha_1}, \sqrt{\alpha_2}, \ldots, \sqrt{\alpha_t})$ where, for $k=1,\ldots,t$, $\alpha_k>0$ and $\alpha_k \in F_1$. Let $\alpha \in Q_{\mathrm{R}}$. Then *rootrank*$(\alpha)=\min\{h$: $\alpha \in F$ and rootrank$(F)=h\}$.

In the following section we will construct nets that can 'realize' certain nonnegative real numbers. It will be shown that if two such numbers can be realized by two nets, then there is a uniform way to construct nets which can realize their sum, product, or quotient. It would be extremely useful to also be able to realize their *difference* in a uniform way. It is easy to see that this is not possible as otherwise we could build a net which could realize the difference of the larger from the smaller and thus build a net which could realize a negative number. However, this is not a serious problem for quadratic extensions of the rationals, because differences can be rewritten using products, sums, and quotients.

For example,

$$\sqrt{3}-\sqrt{2}=1/(\sqrt{3}+\sqrt{2}).$$

The following result generalizes this example to any element of Q_{R}:

1.7. Lemma. *Let $r>0\in Q_{\mathrm{R}}$ be such that rootrank$(r)=n$. Then r can be expressed as the sum, product, and quotient of positive elements of Q_{R}, each of which has rootrank $\leq n$.*

Proof. If rootrank$(r)=0$, the result is trivial. For $n\geq 1$, there exists a real quadratic extension F_1 of Q such that rootrank$(F_1)=n-1$ and $r\in F=F_1(\sqrt{s_1},\ldots,\sqrt{s_n})$ where $s_1,\ldots,s_n$ are positive elements of F_1. Moreover, we may assume $[F:F_j]=2^n$. Enumerate the subsets of $\{\sqrt{s_1},\ldots,\sqrt{s_n}\}$, $V_1,\ldots,V_m$ where $m=2^n$. Let $V_1=\emptyset$. Let $\pi_1=1$, and for $i=2,\ldots,m$ let π_i = the product of the elements of V_i. Then $\{\pi_i: i=1,\ldots,m\}$ forms a basis for F over F_1. Suppose $r=a_{i1},\pi_{i1}+\cdots+a_{it}\pi_{it}$ where $a_{ij}\neq 0$, $j=1,\ldots,t$. Let $W=\bigcup_{j=1,\ldots,t}V_{ij}$. W is contained in $\{\sqrt{s_1},\ldots,\sqrt{s_n}\}$. The proof will proceed by induction on n and the order of W.

Suppose $W=\emptyset$. Then $r=a_1\in F_1$, rootrank$(r)\leq n-1$ and the result follows by induction. Now suppose that $|W|=k>0$ and the theorem has been established for $|W|<k$. Let $\sqrt{s_u}\in W$ and suppose, for $j=1,\ldots,x$, $\sqrt{s_u}\in V_{ij}$ and, for $j=x+1,\ldots,t$, $\sqrt{s_u}\notin V_{ij}$.

Let

$$r_1=\sqrt{s_u}(a_{i1}\beta_{i1}+\cdots+a_{ix}\beta_{ix})$$

where $\beta_{ij}=\pi_{ij}/\sqrt{s_u}$ for $j=1,\ldots,x$ and

$$r_2=a_{ix+1}\pi_{ix+1}+\cdots+a_{it}\pi_{it}.$$

Then $r=r_1+r_2$. For $j=1,\ldots,x$, let U_{ij} be the subset of $\{\sqrt{s_1},\ldots,\sqrt{s_n}\}$ such that the product of the elements in U_{ij} equals β_{ij}.

Let

$$W_1=\bigcup_{j=1,\ldots,x}U_{ij}$$

and

$$W_2=\bigcup_{j=x+1,\ldots,t}V_{ij}.$$

Then $|W_1|$ and $|W_2|$ are each less than k.

Let $y=|r_1|$ and $z=|r_2|$. By induction on k the lemma holds for $y/\sqrt{s_u}$ and hence for y, and the result holds for z and so for $y+z$.

Since neither y^2 nor z^2 contains a term involving $\sqrt{s_u}$, if $y>z$, then by induction on k the result holds for $y-z=(y^2-z^2)/(y+z)$. (Similarly, the result holds if $z>y$.) □

2. The network model

In some models of neural networks (such as perceptrons [6]), it is required that neurons fire synchronously. In others (such as Boltzmann machines [4]) the firing is of necessity asynchronous.

Here the networks will possess both asynchronous units and blocks of synchronized units. These synchronous components are included to allow for generality in functional capability. (More on this below.) There is one important restriction on synchronized blocks: the outputs from all neurons in a synchronous block all go to the same neuron which is not a member of a synchronous block (including the outputting block). The synchronous block acts as a kind of preprocessor for this neuron.

In the illustrations, a block of synchronized neurons will be enclosed in a rectangle. The connections between neurons, the axons, will be represented as lines from one neuron to another which terminate in solid dots (excitatory input) or open dots (inhibitory input) and can branch at their termination point to represent weighted input. Certain external axons will carry input to the net from the outside world. For an example, see Fig. 1.

For an individual neuron, the rule for firing is the usual one; each neuron readjusts its firing state randomly in time (but with the same mean attempt rate) outputting a pulse (output = 1) if at that time the difference between the excitatory and inhibitory input exceeds its threshold. Otherwise the output is 0. Neurons in a synchronized block individually fire by the same rule, however, they readjust their states at the same time, and input pulses to more than one neuron of the block from a neuron outside the block (or from external input) along a branching axon arrive at each of these neurons simultaneously. Moreover, output pulses leaving the block, simultaneously arrive at their destination. A synchronous block of this type leading to a neuron can compute the logical EXCLUSIVE OR function of two inputs which could not be computed without synchronicity.

Consider first the case of a single neuron, n. By way of example, suppose n has threshold 2, and two excitatory input axons with frequencies f_1, f_2 respectively. (Frequency here and throughout represents the probability of firing at a given time.) This situation is shown in Fig. 2. Then n will fire only in the case when both input axons send a pulse to n at the same time. Therefore the output frequency of n is $f_1 f_2$.

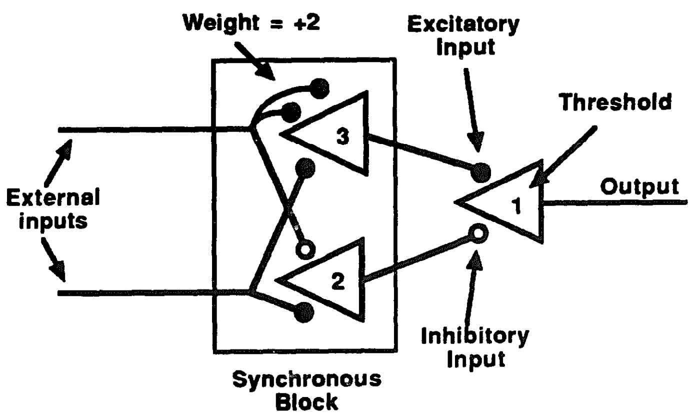


Fig. 1.

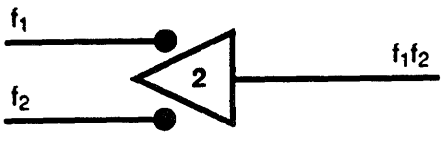


Fig. 2.

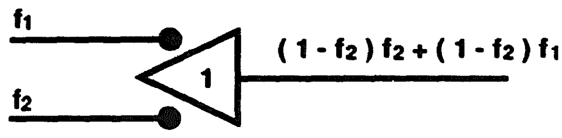


Fig. 3.

For the example in Fig. 3, n will fire when one or the other of the input axons send a pulse to n. Therefore, the output frequency is $(1-f_1)f_2+f_1(1-f_2)+f_1f_2$. In general the output from n will be a sum over products of input frequencies f_i or the difference between 1 and these frequencies, $1-f_i$.

As in [7], we define the following two mappings. The first, π_1, maps the vertices of $\mathbf{I}^k$ to Boolean expressions in the formal variables $F_1, \ldots, F_k$ whereby each vertex is mapped to the conjunction $\hat{F}_1\hat{F}_2 \cdots \hat{F}_k$ where $\hat{F}_i$ is F_i if the ith coordinate of the vertex is 1 and $\hat{F}_i$ is $!F_i$, the negation of F_i, if the ith coordinate is 0. The second, π_2, maps Boolean expressions in $F_1, \ldots, F_k$ to real expressions in the variables $f_1, \ldots, f_k$ by replacing conjunctions with products, negations with subtraction from 1, and each F_i with f_i.

For example the vertex $(1,0,0,1)$ is mapped by π_1 to the Boolean expression $F_1!F_2!F_3F_4$ which is mapped in turn by π_2 to the real expression $f_1(1-f_2)(1-f_3)f_4$.

Suppose that the neuron n receives input from k axons with firing frequencies $f_1, f_2, \ldots, f_k$ respectively. For each vertex b in $\mathbf{I}^k$, let the ith coordinate of b represent the firing status of the ith input axon. Let B be the subset of vertices which represent an input which will cause the neuron to fire. Then the firing frequency of the neuron is

$$\alpha = \sum_{b \in B} \pi_2(\pi_1(b)). \tag{1}$$

Now consider the case where a neuron receives input from one or more neurons in a block of synchronous neurons. An example is given in Fig. 4. The output from n_1 is $\alpha_1 = f_1f_2$ and from n_2 is $\alpha_2 = f_2f_3$. If n_1 and n_2 were not synchronized, then the output from n would be $\alpha = f_1f_2f_2f_3$. Since they are synchronized the pulses of the f_2 input required to fire n_1 and n_2 arrive at the neurons at the same time.

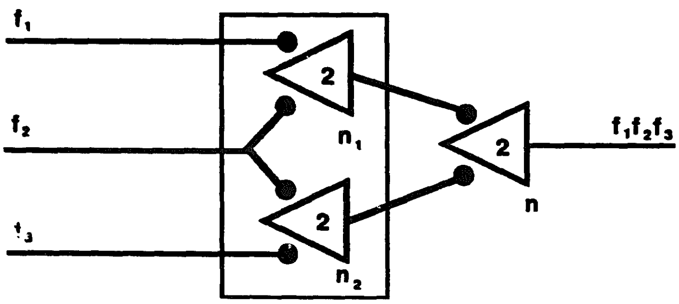


Fig. 4.

Therefore, $\alpha = f_1 f_2 f_3$. In the general case of two neurons, suppose n_1 receives inputs $f_1, \dots, f_k$ and n_2 receives inputs from perhaps some of the same neurons as n_1 and in addition receives input from $f_{k+1}, \dots, f_{k+m}$. Let b be the vertices of $\mathbf{I}^{k+m}$ (where the first k coordinates of b represent the firing status of the inputs to n_1, and the last m coordinates represent the firing status of the additional inputs to n_2). Let B_1 (B_2) be the set of vertices which represent inputs which cause n_1 (n_2) to fire. Then the output frequencies, α_1, α_2, and α satisfy

$$\alpha_1 = \sum_{b \in B_1} \pi_2(\pi_1(b)), \qquad \alpha_2 = \sum_{b \in B_2} \pi_2(\pi_1(b))$$

and

$$\alpha = \sum_{b \in B_1 \cap B_2} \pi_2(\pi_1(b)).$$

Clearly, this can be generalized to the case where n receives any number of inputs from the block or inhibitory inputs from the block or inputs from other blocks or other neurons. The troublesome (but not intractable) case where one synchronous block receives inputs from another has been eliminated by the restriction on synchronous blocks mentioned in the introductory description of the model. The important point is that α can be written as a sum of products of the input frequencies to the blocks feeding n and the input frequencies to n itself (or the difference between 1 and these frequencies).

By keeping the firing frequencies of the input axons fixed, it is hoped that the network will eventually settle down into a *steady state*, i.e. the firing pattern of each individual neuron will converge to a fixed frequency. (A similar problem was studied in [4] and [5] where the connections between neurons were symmetric.) In general, equilibrium is not assured. But in any event, for a given set of input frequencies, there are only a finite number of possible values to which the output frequency of a given neuron *could* converge, and these possible convergent values are all roots of a polynomial determined by the net and the input frequencies.

2.1. Theorem. *Let N be a neural net. Suppose there are k input axons to N and they are firing at the frequencies $f_1, \dots, f_k$ respectively. And suppose that in time the firing frequencies of each of the neurons of N converges. Then for each neuron n of N there exists an integer m and polynomials $g_0(x_1, x_2, \dots, x_k), \dots, g_m(x_1, x_2, \dots, x_k)$ over Q such that the firing frequency of n is a root of the polynomial $g_0(f_1, f_2, \dots, f_k) + g_1(f_1, f_2, \dots, f_k)x + \dots + g_m(f_1, f_2, \dots, f_k)x^m$.*

The proof of 2.1 is greatly aided by the following:

2.2. Lemma. *There exist polynomials $q_0(x_0, \dots, x_r, y_0, \dots, y_t), \dots, q_{rt}(x_0, \dots, x_r, y_0, \dots, y_t)$ over Q such that if α_1 is a root of a polynomial $g_0 + g_1 x + \dots + g_r x^r$ over Q $(g_r \neq 0)$ and α_2 is a root of a polynomial $h_0 + h_1 y + \dots + h_t y^t$ over Q $(h_0, h_t \neq 0)$, then $\alpha_1 \alpha_2$ is a root of*

$$q_0(g_0, \dots, g_r, h_0, \dots, h_t) + q_1(g_0, \dots, g_r, h_0, \dots, h_t)x + \dots + q_{rt}(g_0, \dots, g_r, h_0, \dots, h_t)x^{rt}.$$

A similar result can be obtained for $\alpha_1+\alpha_2$, $\alpha_1-\alpha_2$, *and* $\alpha_1 \div \alpha_2$.

Proof. The proof is a standard one from abstract algebra. The significance of the proof is the uniformity with which the polynomials $q_0, \dots, q_{rt}$ can be constructed. Let $\beta=\alpha_1\alpha_2$. (The sum and difference of roots are handled similarly. The quotient is a slightly different case which is handled below.) Now

$$\alpha_1^r=-g_0/g_r-\cdots-g_{r-1}/g_r\alpha_1^{r-1}$$

and a similar equation exists for α_2^t. Let $x_0, \dots, x_r$, $y_0, \dots, y_t$, x, and y be real variables and let

$$X=-x_0/x_r-x_1/x_rx-\cdots-x_{r-1}/x_rx^{r-1},$$
$$Y=-y_0/y_t-y_1/y_ty-\cdots-y_{t-1}/y_ty^{t-1}.$$

Note that if g_i (h_i) is substituted for x_i (y_i) for each i and α_1 (α_2) is substituted for x (y), then we obtain the equation for α_1^r (α_2^t) above.

Consider the set of products $P=\{x^iy^j: i=0,\dots,r-1, j=0,\dots,t-1\}$ (Fix an ordering on this set, $1, x, \dots, x^{r-1}y^{t-1}$.) Consider the products of xy with each element in P. By substituting the expressions for X and Y for each occurrence of x^r or y^t respectively whenever they occur in these products, we see that each of these products can be written as a linear combination of the elements of P with coefficients being rational functions of $x_0, \dots, x_r, y_0, \dots, y_t$. Let M be the $rt\times rt$ matrix where M_{ij} = the coefficient of the ith element in P in the expansion of the product of the jth element of P with xy. Let v be the rt-dimensional row vector $[1\alpha_1 \dots \alpha_1^{r-1}\alpha_2^{t-1}]$. And let M_1 be the matrix formed by substituting $g_0, \dots, g_r, h_0, \dots, h_t$ for $x_0, \dots, x_r$, $y_0, \dots, y_t$ in M.

Then $\beta v=vM_1$, and so β is a root of the characteristic polynomial of M_1. Since this polynomial can be determined by substituting the g_i's and h_i's for the x_i's and y_j's in the characteristic polynomial of M, the result follows.

In order to construct polynomials for the quotient, we first note that $1/\alpha_2$ is the root of $h_0y^t+h_1y^{t-1}+\cdots+h_t$ and then use the method above with the product $\alpha_1 * 1/\alpha_2$. $\square$

Proof of 2.1. The proof is by induction on the number of internal axons in N. Suppose first that there are no internal axons. Then N consists of a collection of individual neurons with no connections between them. From the discussion preceding Theorem 2.2, the output from each neuron can be expressed as in (1) and so the output frequency is a root of a linear polynomial of the type desired.

Now suppose the network has $m>0$ internal axons and the theorem has been proven for less than m. Let n be a neuron of N and suppose first that there is an internal axon from n to a neuron n_1 (possibly itself). Construct a new network N' by removing this axon and creating a new external axon leading to n_1. Let the firing frequency of this new input be f. By induction, there exist an integer P and polynomials $g_0(x_1,\ldots,x_k,y),\ldots,g_p(x_1,\ldots,x_k,y)$ such that the output from n in N at equilibrium is a root of

$$g_0(f_1,\ldots,f_k,f)+\cdots+g_p(f_1,\ldots,f_k,f)x^p.$$

Then the output α from n in N satisfies $0=g_0(f_1,\ldots,f_k,\alpha)+\cdots+g_p(f_1,\ldots,f_k,\alpha)\alpha^p$ and the result follows.

Now suppose there are no axons leading from n to another neuron. There are two cases to consider. First, suppose there are no synchronous blocks of neurons with output axons leading directly to n. Let $n_1,\ldots,n_t$ be the neurons which have output axons leading to n. By the argument in the preceding paragraph the output frequencies from each of the n_j's satisfy the theorem. But the output frequency from n can be written as a sum similar to (1) with the outputs from the n_j's included with the f_i's as inputs to n, i.e. the output from n is the sum and difference of products of frequencies satisfying the theorem. The result follows immediately from Lemma 2.

Finally suppose that there are synchronous blocks with outputs to n. The neurons which feed each of these blocks satisfy the theorem (since there is an internal axon of the network coming out of each of them, a case considered above) and so as in the discussion of synchronous blocks preceding 2.1, the output from n is again determined by sums of products of frequencies satisfying the theorem and so by 2.2 the result follows. □

At this point the focus of discussion will change from the *frequency* of firing to the firing *ratio* i.e. the ratio of on pulses to off pulses. The change may seem insignificant since if f is a frequency and r is the corresponding ratio, then

$$r=\frac{f}{1-r} \quad \text{and} \quad f=\frac{r}{1+r}. \tag{2}$$

But there are some important advantages and disadvantages to this change. The major disadvantage is that when considering frequencies, it is a simple matter to construct a net which yields the *product* of two frequencies, since the product results from the logical AND function. (See Fig. 2.) With ratios it will still be possible to compute the product of ratios in a uniform way, but not quite as easily. (More on this below.) The other disadvantage, a caution actually, is that a frequency of 1 yields an undefined ratio. The advantages of considering ratios are:

(i) It will be shown that sums can be computed uniformly. Since frequencies must be in the range from 0 to 1, there can be no way to compute sums uniformly because if there were, then the method could be used to compute the sum of two frequencies each greater than $\frac{1}{2}$ yielding a result greater than 1.

(ii) With ratios, the quotient is a simple matter. In fact, negation, which changes a frequency f into $1-f$, changes a ratio r to $1/r$. Quotients will be very handy when considering the minimal extension field of Q containing the possible values that a network can produce.

In Theorem 2.1, a polynomial was constructed whose roots were the possible output frequencies from a neuron at equilibrium. The following defines the corresponding polynomial for ratios in place of frequencies. The polynomials are produced by substituting (2) into the polynomials of 2.1.

2.3. Definition. Let N be a neural net with k input axons. Let n be a neuron in N. Then the *equilibrium polynomial for* N at n, denoted by $E_n^N(x)$, is the monic polynomial of minimal degree with coefficients in $Q(r_1, \ldots, r_k)$ such that if N is receiving input *ratios* $r_1, \ldots, r_k$, each ≥ 0, then the steady state output ratio from n is a root of the polynomial. (Where N is understood, $E_n^N(x)$ will be abbreviated $E_n(x)$).

2.4. Definition. Given N, n, and E_n^N as above, the *structural equilibrium polynomial for* N at n, denoted by $S_n^N(x)$ (or $S_n(x)$ when N is understood) is derived from $E_n^N(x)$ by assuming all input ratios (or frequencies) are 0.

The main results of this section concern the roots of E_n^N and S_n^N under restrictions on the amount of feedback in N. The degree of feedback in the networks will be measured by considering the circular feedback level of the underlying digraph of the network.

2.5. Definition. Given a neural net N, the *underlying directed graph of* N, is the digraph $G(N)$ with vertices $V = \{n$: n is a synchronized block of neurons or n is a neuron of N not belonging to any synchronized block of neurons$\}$ and $E = \{(n_1, n_2)$: there exists an axon from n_1 to $n_2\}$.

2.6. Definition. The *class of rank k networks with circular feedback*, denoted by CFN_k, is the collection of networks N such that $\mathrm{CF}(G(N)) = k$.

CFN_0

Rank 0 networks have no feedback whatsoever. Input pulses enter certain neurons and pass through the network in one direction.

2.7. Theorem. *Let N be a network such that* $\mathrm{CF}(G(N)) = 0$. *Let n be a neuron of N. Then* $E_n(x)$ *is linear and* $S_n(x) = x$ *or is undefined.*

Proof. The proof is by induction on h = height $G(N)$. If $h = 0$, then the network consists of a single neuron or a collection of individual neurons with no axons between them. The output from a neuron n can be expressed as in (1) and the result for E_n follows immediately. With inputs of all 0's the output frequency from n is either 0 or 1. In the former case, $S_n(x) = x$ and in the latter, it is undefined. For $h > 0$, a neuron n at the end of a path of length h in secgraph $G(N)$ receives inputs from external axons or from neurons at the end of paths of length $< h$. By induction, the equilibrium polynomial at these neurons is linear so that the output from each neuron (that is, the input into n) is a rational function of the input ratios and so using (1) again, $E_n(x)$ is linear. $S_n(x)$ will be defined only if the outputs from each of these neurons is 0 in which case the result holds. □

Though level 0 nets do not possess much computational power, there is one net which does compute the reciprocal function of an input ratio. (This is the negation function of the input frequency.) This net is shown in Fig. 5. If the input frequency is f, then the output frequency is $g=(1-f)$. Let $f=r/(1+r)$ and $s=g/(1-g)$, then $s=1/r$ and so $E_n(x)=x-1/r$.

CFN_1

At rank 1, since each section of $G(N)$ consists of a simple closed path of neurons (or blocks), there is a limited amount of feedback in the net. It turns out that $E_n^N(x)$ and $S_n^N(x)$ are still linear, but that $S_n^N(x)$ is sufficiently general so that every non-negative rational number can be the root of S_N for some N and n.

Moreover, level 1 nets have the capacity to compute the sum and product functions of input ratios.

2.8. Theorem. *Let N be a network such that* $\mathrm{CF}(G(N))=1$. *Let n be a neuron of N. Then $E_n(x)$ and $S_n(x)$ are linear.*

Proof. Consider a section S of $G(N)$. Each vertex of S represents either a neuron or a block of synchronous neurons of N. Since blocks cannot lead to each other or themselves there must be at least one vertex which represents a single neuron. Let M be the subnetwork of N which is represented by S and let m be one such neuron. Consider the axons coming from other neurons in N into M as external inputs to M, and along with the actual external inputs to M suppose there is a total of k inputs with frequencies $f_1, \ldots, f_k$. A new network M' can be formed by removing the axons that exit m and go to the next neuron (or block) in S and replacing them with new external inputs. (If the next vertex after m in the closed path represents a single neuron, then there will be only one axon to remove. If the next vertex is a block, there may be more than one axon, but all pulses on these axons arrive at the block simultaneously.) Then M' is a simple linear path and the new input(s) which was created enters it at only its first neuron (or block). But then the products in (1) when

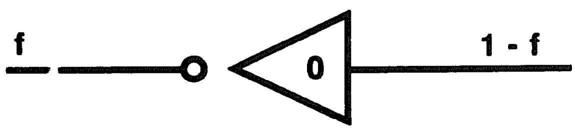


Fig. 5.

computed for each neuron (or block) in the path in turn will contain at most one factor of the new input. Therefore, the output from m in M' can be expressed as a linear function of this new input, that is, if f is this new input and β is the output from m in M', then $\beta = af + b$ where a and $b \in Q[f_1, \ldots, f_k]$. Then if α is the output from m in M, then $\alpha = a\alpha + b$. Substituting $\alpha = r/(1+r)$ shows that E_m^M is linear. Once the output from m is known, the output from each of the other neurons in M can be computed in turn by a series of computations like (1). Therefore, the result follows for each neuron in M. Moreover, since the output from each section in $G(N)$ can be viewed as input to subsequent sections in secgraph $G(N)$, the result holds for all of N. □

2.9. Theorem. *There exist rank* 1 *networks* N_1 *and* N_2 *such that for all* r_1, r_2, *if* r_1 *and* r_2 *are used as input ratios, then* N_1 *produces* $r_1 + r_2$ *and* N_2 *produces* $r_1 r_2$ *as output ratios.*

Proof. The networks N_1 and N_2 are shown in Figs. 6 and 7. In N_1,

$$h = f_1(1-f_2) + f_2(1-f_1) + f_1 f_2 h.$$

Then

$$h = \frac{f_1(1-f_2) + f_2(1-f_1)}{1 - f_1 f_2}$$

and so

$$\begin{aligned} h/(1-h) &= \frac{f_1(1-f_2) + f_2(1-f_1)}{1 - f_1 f_2 - (f_1(1-f_2) + f_2(1-f_1))} \\ &= \frac{f_1(1-f_2) + f_2(1-f_1)}{(1-f_1)(1-f_2)} \\ &= r_1 + r_2. \end{aligned}$$

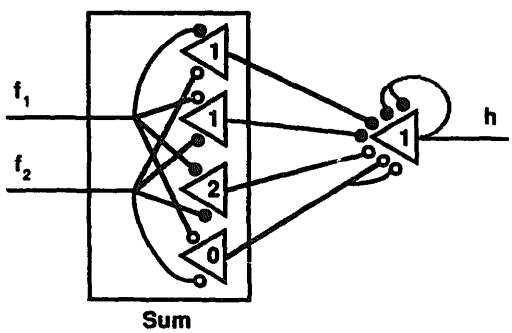


Fig. 6.

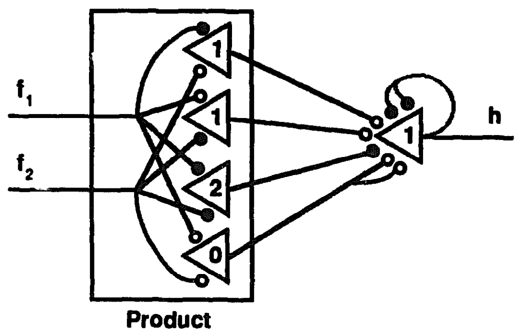


Fig. 7.

In N_2,

$$h=f_1f_2+h(f_1(1-f_2)+f_2(1-f_1))$$
$$=\frac{f_1f_2}{1-(f_1(1-f_2)+f_2(1-f_1))}$$

and so

$$\frac{h}{1-h}=\frac{f_1f_2}{(1-f_1)(1-f_2)}=r_1r_2. \qquad \square$$

2.10. Corollary. *For every non-negative rational number α there exists a rank* 1 *network N containing a neuron n such that $S_n^N(x)$ has α as a root.*

Proof. The network in Fig. 8 produces 1 as its output ratio. Then using Theorem 2.9 repeatedly and the negation function of Fig. 5, any positive integer, any reciprocal positive integer, and any positive rational can be produced. $\square$

CFN_2

At rank 2 the possible ratios will be extended from the rationals to include real quadratic extensions of the rationals. It will be shown that if $\alpha \geq 0 \in Q_R$, then there exists a rank 2 net which produces α as an output ratio. In particular, there exists a level 2 net which can compute the square root of every input ratio. This net will be exhibited first.

2.11. Theorem. *There exists a* CFN_2 *net N with a single input ratio r and a single neuron n such that $E_n^N(x)=x^2-r$. That is, for every non-negative real number r which is used as an input to N,*

$$R=\sqrt{r} \text{ is an output ratio from } N.$$

Proof. Consider the net in Fig. 9. The neuron n will fire if and only if there is no pulse on the inhibiting feedback axon along with a pulse on either the input axon or the other feedback axon or when there is a pulse on the inhibiting feedback axon and on both the other axons. (It is important to note here that pulses on the two feedback axons arrive independently in time of each other; they are not synchronized.)

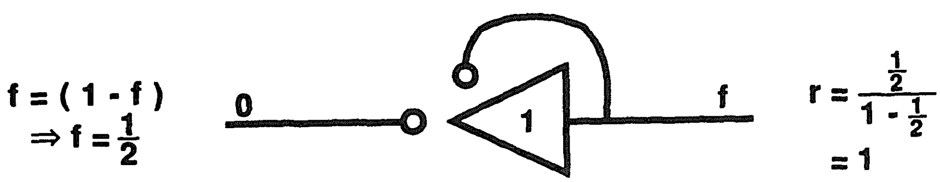
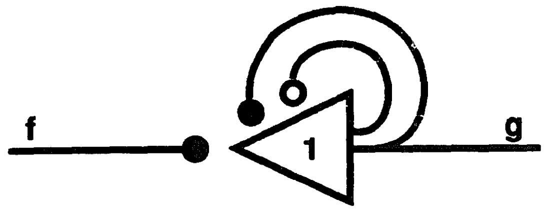


Fig. 8.



Square Root

Fig. 9.

Therefore, the output frequency from n is $g=(1-g)[g+f-fg]+g[fg]$, where f is the input frequency. If r is the input ratio, i.e., $f=r/(1+r)$ and s is the output ratio from n, i.e., $g=s/(1+s)$, then

$$\frac{s}{1+s}=\frac{1}{1+s}\left(\frac{s}{1+s}+\frac{r}{1+r}-\frac{sr}{(1+s)(1+r)}\right)+\frac{s}{1+s}\left(\frac{sr}{(1+s)(1+r)}\right)$$

and so

$$s(1+s)(1+r)=s(1+r)+r(1+s)-sr+ssr$$

which reduces to $s^2=r$. □

Using 2.11 it is now possible to construct a network which receives an input *frequency* f and obtains output frequency g where $g=\sqrt{f}$. Let r be the ratio corresponding to f. Using sums, products, and the square root, it is possible to construct a net with output ratio $s=\sqrt{r(1+r)}+r$. But then the output frequency $g=s/(1+s)=\sqrt{f}$.

Let us now turn to the question of what ratios are possible as outputs of level 2 nets. Using 2.9 and the negation function, it is possible to produce the sum, product, or quotient of ratios. Using 2.11, it is possible to produce square roots. It is not possible to uniformly construct a net which produces the difference of two ratios because otherwise it would be possible to realize a negative difference. However, by 1.7, differences can be rewritten using sums, products and quotients. This leads to the following:

2.12. Theorem. *If $\alpha \geq 0 \in Q_{\mathrm{R}}$, then α can be produced as an output ratio of a* CFN$_2$ *net.*

Proof. *The result follows immediately from* 1.7, 2.9, *and* 2.11. □

To show that output ratios of level 2 nets necessarily belong to Q_{R}, we will need the following lemma which takes a closer look at level 1 nets:

2.13. Lemma. *Let N be a network consisting of a simple closed path of neurons (or blocks). Let $f=f_1, f_2, \ldots, f_k$ be the external input frequencies to N. Let n be a neuron* *of N. Let* $r=f/(1-f)$ *and let the output frequency from n be* $g=s/(1+s)$. *Then*

$$s=\frac{Ar+B}{Cr+D}$$

where $A, B, C,$ *and* D *are polynomials in* $f_2, \ldots, f_k$ *and* $A, B, C, D \geq 0$.

Proof. The result follows from a closer inspection of the sum of products in (1). Since these sums represent frequencies, it should be noted that $0 \leq$ such sums ≤ 1. Suppose the neuron following n in the circle has output α. Write α as a sum of products, and separate the terms of this sum which involve g from those involving $1-g$. Then

$$\alpha=ag+b(1-g)$$

where a and b are sums like (1) in $f_1, f_2, \ldots, f_k$ and are linear in f and $(1-f)$.

Now suppose α is fed into another neuron along with other inputs and the output is β. Then

$$\begin{aligned}\beta&=c\alpha+d(1-\alpha) \quad (0 \leq c, d \leq 1)\\&=c(ag+b(1-g))+d(1-(ag+b(1-g)))\\&=acg+bc(1-g)-adg-bd(1-g)+d-dg+dg\\&=eg+h(1-g)\end{aligned}$$

where $e=a(c-d)+d$ and $h=b(c-d)+d$.

If $c-d \leq 0$, then since $0 \leq a$, we have $e \leq d \leq 1$, and if $c-d>0$, then since $a \leq 1$, we have $e \leq c \leq 1$. Since $e=ac+d(1-a)$, $0 \leq e$. Similarly, $0 \leq h \leq 1$. Moreover, since a and b are linear in f and $(1-f)$ and c and d involve only external inputs, it follows that e and h are linear in f and $(1-f)$. Continuing in this way around the simple closed path yields

$$\begin{aligned}g&=(if+j(1-f))g+(kf+m(1-f))(1-g)\\&=\frac{kf+m(1-f)}{1-(if+j(1-f)+(kf+m(1-f)))}\end{aligned}$$

where i, j, k, m are sums of products involving $f_2, \ldots, f_k$ (and their differences with 1) and $0 \leq i, j, k, m \leq 1$. Then

$$s(1-(if+j(1-f))+(kf+m(1-f)))=(kf+m(1-f))(s+1)$$

and so

$$\begin{aligned}s&=\frac{kf+m(1-f)}{1-(if+j(1-f))}\\&=\frac{kr+m}{r+1-(ir+j)}\\&=\frac{kr+m}{(1-i)r+(1-j)}\end{aligned}$$

and the result follows. □

2.14. Theorem. *Let N be a* CFN_2 *network and let S be a section of $G(N)$ such that* $\mathrm{CF}(S)=2$. *Let M be the subnetwork of N represented by S (with inputs to M being from external iputs or neurons not in M). Let m be a neuron in M. Then $E_m^M(x)$ has degree at most* 2.

Proof. S consists entirely of disjoint components, each of which is either a CFN_0 component (i.e. representing a single neuron (or block)) or a CFN_1 component (i.e. representing a simple closed path of neurons (or blocks)), and edges connecting these components in a simple closed path. (See Fig. 10.) At least one of the components is a CFN_1 component, S. The edge from S to the next (possibly itself) component in the closed path in S must represent an axon from an individual neuron m and not a block. Otherwise, output from the block would go to both the neuron in the simple closed path in S and to a neuron outside S and by definition, output from a block feeds a single neuron only. Form a new network M' by removing the axon represented by this edge and replacing it with a new external input axon with input ratio r. Let the output from m in M' be s. M' is then a linear chain or CFN_0 and CFN_1 nets. Suppose the components in the chain in order are $S_1, \ldots, S_k=S$. Let n_1 be the neuron in S_1 whose output leads to S_2. Let s_1 be he output from n_1. Then by Lemma 2.13, $s_1=(A_1r+B_1)/(C_1r+D_1)$ where A_1, B_1, C_1, D_1 are polynomials in the other inputs to M'. s_1 along with external inputs are then the input into S_2 and then by 2.13 again, the output from a neuron n_2 in S_2 is

$$s_2=\frac{(A_2s_1B_2)}{(C_2s_1+D_2)}$$

$$=\frac{[(A_2A_1+B_2C_1)r+(A_2B_1+B_2D_1)]}{[(C_2A_1+D_2C_1)r+(C_2B_1+D_2D_1)]}.$$

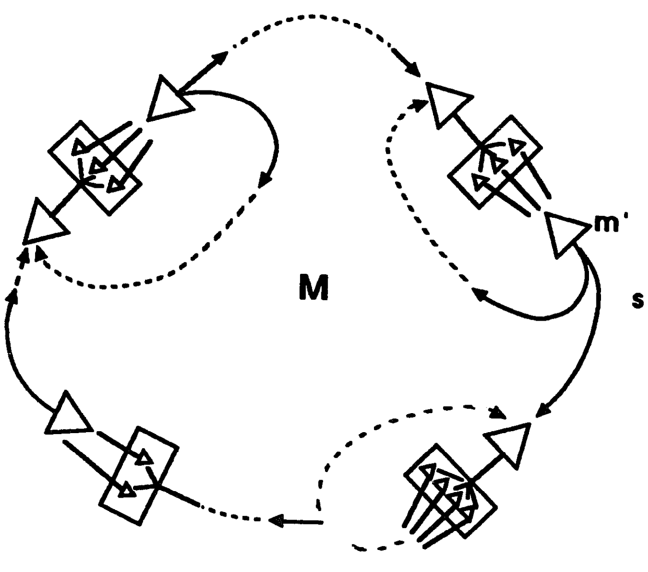


Fig. 10.

Continuing in this way yields $s=(Ar+B)/(Cr+D)$ where A, B, C, and D are the desired polynomials, and setting s equal to r yields the result for $E_m^M(x)$. Fixing r as a root of this polynomial and using 2.13 again, yields the result for each neuron of S_1 and continuing in this way into each component of M yields the result for all neurons of M. □

2.15. Corollary. *If α is an output of a level 2 net, then $\alpha \in Q_R$.*

Proof. Level 2 nets consist of acyclic networks of sections which are of level 2 or less. By 2.14, the output ratios of each section are roots of quadratic polynomials of the input ratios to the section. The result follows by induction on the number of sections along each path in the network. □

Rank 3 and above

At rank 3 we are able to extend the set of possible ratios a great deal. In particular, the ratios need not be roots of polynomials which are solvable by radicals.

2.16. Example. Consider the polynomial $p(x)=x^5-2x^3-2x-2$. By Eisenstein's criteria, $p(x)$ is irreducible over Q. Since $p(-1)>0$ and $p(0)<0$ and $p'(x)=5x^4-6x^2-2$ has only two real roots, $p(x)$ has exactly 3 real roots. Therefore, the Galois group of the splitting field for $p(x)$ over Q contains a 5-cycle (since the degree of $p=5$) and a transposition (complex conjugation) and so is the full symmetric group S_5 and is unsolvable. Moreover, $S_n^N(x)=p(x)$ for the CPN_3 net illustrated in Fig. 11. In the illustration, ratios are written above the axons.

It is left as an open question whether or not level 3 nets (or any given level for that matter) are sufficiently general to produce every positive real number which is algebraic over the rationals as the output ratio of some net. The author conjectures that this is not the case. The final result here shows that roots of polynomials of degree k can easily be realized at level $2k$.

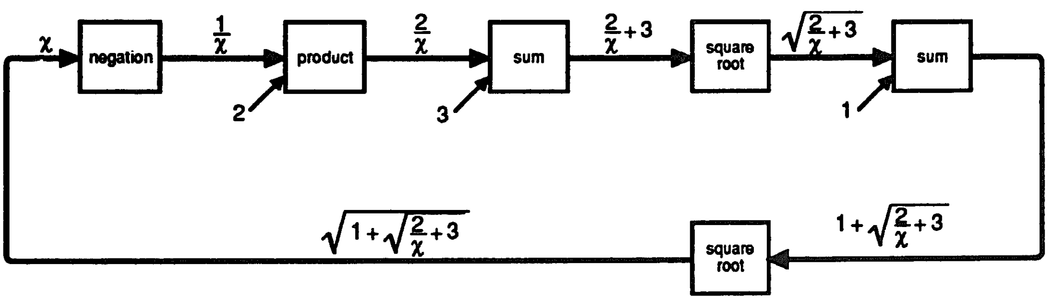


Fig. 11. $\chi=\sqrt{1+\sqrt{\left(\frac{2}{\chi}\right)+3}} \Rightarrow (\chi^2+1)^2=(2/\chi)+3+4\chi^2 \Rightarrow \chi^5-2\chi^3-2\chi-2=0.$

2.17. Theorem. *Let $\alpha > 0$ be a root of a polynomial $p(x)$ of degree k with rational coefficients. Then there exists a net $N \in \mathrm{CFN}_m$ with $m \le 2k$ containing a neuron n such that α is a root of $S_n^N(x)$.*

Proof. If α is rational (i.e. $k=1$), then the result follows from 2.10. Assume α is not rational. Factor out the highest power of x common to all terms of $p(x)$ and multiply by -1 if necessary, so that the constant term is positive. Let the resulting polynomial be $p_1(x)$. Then $p_1(x) = q(x) - r(x) + a_0$ where $q(x)$ and $r(x)$ have no constant terms, all coefficients of $q(x)$ and $r(x)$ are non-negative, and a_0 is a positive rational number. Moreover, since α is positive, $r(x) \neq 0$. Then α is a solution to the equation $x = s(x)/t(x)$ where $s(x) = q(x) + a_0$ and $t(x) = r(x)/x$. Suppose that $s(x) = a_i x^i + \cdots + a_0$. Consider the CFN_1 net in Fig. 12(a). The net uses the level 1 nets from 2.9 to produce products and sums so that with input r, the output is $s(r)$. A similar net can be constructed to produce $t(r)$ and as in Fig. 5, with one extra neuron, $1/t(r)$. The net in Fig. 12(b) combines these two nets in the product $s(r) \cdot 1/t(r)$ with the result becoming the new value for r. It remains to be shown that the CF level of this net is at most $2k$.

Consider first the subnet consisting of the neuron n and the path to itself on the horizontal lines through each of the upper row of boxes shown. This is a CFN_2 net since each box is a CFN_1 subnet. Each of the axons leading from n to a product box adds at most one to the level of the net. This follows immediately from the definition of circular feedback because removing each axon in turn reduces the net to a single component at a lower level. The path from n through n_1 and horizontally through the lower row of boxes again increases the level by (at most) one (since the upper half of the net is a single component and the chain through the lower half

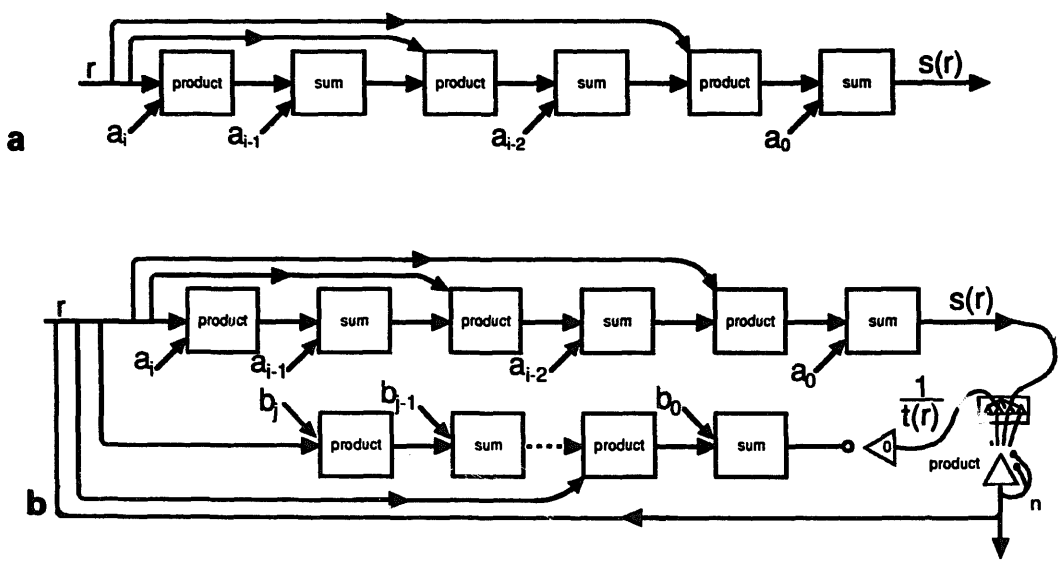


Fig. 12.

produces a simple closed path from this component through CFN_0 and CFN_1 subnets and back to the upper component again). As before, each of the axons from n to the product boxes in the lower row adds one to the level. Since the degrees of $s(r)$ and of $t(r)$ are $\leq k$ with one of them being less than k, the net has rank at most $2k$. $\square$

References

[1] L.C. Eggan, Transition graphs and the star height of regular events, Michigan Math. J. 10 (1963) 385–397.
[2] G. Eisman, Synergy in machines, J. Pure Appl. Algebra 28 (1983) 155–177.
[3] G.B. Ermentrout, Period doublings and possible chaos in neural models. SIAM J. Appl. Math. 44 (1) (1984) 80–95.
[4] G.E. Hinton, T.J. Sejnowski and D.H. Ackley, Boltzmann machines: Constraint satisfaction networks that learn, Technical Report CMU-CS-84-119, Carnegie-Mellon University, Pittsburg, PA, 1984.
[5] J.J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, Proc. Nat. Acad. Sci. U.S.A. 79 (1982) 2554–2558.
[6] M. Minsky and S. Papert, Perceptrons (MIT Press, Cambridge, MA, 1968).
[7] R.J. Williams, The logic of activation functions, in: D.E. Rumelhart and J.L. McClelland, eds., Parallel Distributed Processing (MIT Press, Cambridge, MA, 1986).