* RELATÓRIO TÉCNICO *

IMPLEMENTAÇÃO DE REDES NEURONAIS

Moacyr H Cruz de Azevedo

NCE 03/91

Março/91

Universidade Federal do Rio de Janeiro
Núcleo de Computação Eletrônica
Caixa Postal 2324
20001 - Rio de Janeiro - RJ
BRASIL

Moacyr Henrique Cruz de Azevedo

IMPLEMENTAÇÕES DE REDES NEURONAIS

RESUMO

As primeiras redes neuronais tinham seu funcionamento simulado em computadores seqüenciais. Como redes neuronais com algum significado prático precisam de centenas de neurônios, os pesquisadores logo se depararam com uma barreira intransponível, o tempo de simulação.

Para contornar o problema da simulação, alguns pesquisadores passaram a utilizar os processadores paralelos existentes. Outra linha de pesquisadores optou por desenvolver hardware especializado, os chamados **NEURO-COMPUTADORES**, que podem ser eletrônicos ou eletro-óticos. As implementações eletrônicas utilizam circuitos integrados comerciais e/ou circuitos especialmente projetados, os chamados **NEURO-CHIPS**. As versões eletrônicas podem ainda ser analógicas ou digitais. As implementações eletro-óticas de neuro-computadores criam os chamados **NEURO-COMPUTADORES óTICOS**.

Esse trabalho descreve quatro implementações feitas para redes neuronais: neuro-computador analógico, neuro-computador digital, neuro-computador ótico, e simulação em computador paralelo.

IMPLEMENTATIONS OF NEURAL NETWORKS

ABSTRACT

The first neural networks have been simulated on sequential computers. As neural networks with some practical importance have hundreds of neurons, the simulation time has become a difficult barrier for the researchers.

To avoid this problem, some researchers began to simulate on existing parallel computers. Other researchers chose to develop a specialized hardware, the **NEURO-COMPUTERS**, that can be eletronic or eletro-optic. The eletronic implementation can use commercial or special chips, the **NEURAL-CHIPS**, and have digital or analog implementation. The eletro-optic implementation is called **OPTICAL NEURAL-COMPUTERS**.

This work discribes four implementations for neural networks: analog neural-computer, digital neural-computer, optical neural-computer, and simulation on a parallel computer.

IMPLEMENTAÇÕES DE REDES NEURONAIS

Moacyr Henrique Cruz de Azevedo

ÍNDICE

IMPLEMENTAÇÕES DE REDES NEURONAIS

Moacyr Henrique Cruz de Azevedo

1 - INTRODUÇÃO

Com o aumento do interesse em redes neuronais os pesquisadores iniciaram a simulação do funcionamento em computadores seqüenciais. Como redes neuronais com algum significado prático precisam de centenas de neurônios, os pesquisadores logo se depararam com uma barreira intransponível, o tempo de simulação.

Redes neuronais têm como característica o paralelismo maciço. Para contornar o problema do tempo de simulação alguns pesquisadores passaram a utilizar os processadores paralelos existentes. Essa opção tinha como maior vantagem a implementação imediata da rede neuronal após a conversão do algoritmo seqüencial em paralelo, o que representava, na maioria dos casos, menor esforço de pesquisa e resultados mais rápidos. A simulação de redes neuronais em processadores paralelos existentes tornou-se, e ainda é, bastante utilizada.

Outra linha de pesquisadores optou por desenvolver hardware especial para redes neuronais, os chamados **NEURO-COMPUTADORES**. A implementação por hardware é mais difícil e demorada, necessitando de maior esforço e tempo de pesquisa. As implementações atualmente utilizadas podem ser eletrônicas ou eletro-óticas.

As implementações eletrônicas utilizam circuitos integrados comerciais e/ou circuitos especialmente projetados, os chamados **NEURO-CHIPS**. As versões eletrônicas podem ainda ser analógicas ou digitais. Na maioria dessas implementações um computador digital, ou computador hospedeiro, é necessário para estabelecer a arquitetura das conexão, os ganhos sinápticos e a seqüência de aprendizado.

As implementações eletro-óticas de neuro-computadores criam os chamados **NEURO-COMPUTADORES ÓTICOS**. Nessas implementacões os fios são substituídos por feixes luminosos e parte da computação também é feita oticamente.

2 - TIPOS DE IMPLEMENTAÇÃO

2.1 - Implementação em Processadores Paralelos Existentes

Essa implementação requer programas que simulem a operação da rede que se deseja testar. Os tempos de aprendizado, estabilidade e resposta da rede neuronal dependem do tempo de processamento do

programa emulador, variando com a implementação e com o computador utilizado. Por outro lado as redes podem ser facilmente alteradas pela mudança do programa emulador.

Uma das principais características das redes neuronais é o paralelismo maciço. Com a simulação parte desse paralelismo se perde, mesmo em computadores com múltiplos processadores, devido ao carater seqüencial na execução das instruções. Isso pode ser minimizado, e experiências comprovam, com a utilização de processadores paralelos operando com granularidade fina.

Hardwares especiais usando tecnologia VLSI, ou neuro-chips, são tremendamente mais rápidos do que simulações. Entretanto esses hardwares são deficientes na flexibilidade de explorar diversos padrões de conectividade e algoritmos de aprendizado diferentes.

2.2 - Implementação em Hardware Analógico

O cérebro de animais superiores é capaz de computar em velocidades equivalentes a mais de 10**18 operações de ponto flutuante por segundo. Em comparação, os computadores digitais mais rápidos chegam apenas a 10**10 operações por segundo. Cérebros biológicos obtêm essa eficiência devido ao seu modo de operação analógico, que permite processamento paralelo em tempo real em grande escala e evita os procedimentos iterativos dos computadores digitais. Como conseqüência, as implementações digitais de redes neuronais são bastante lentas.

As implementações analógicas necessitam de resistores de alta precisão e são facilmente afetadas por ruídos elétricos. Assim, a fabricação de redes neuronais analógicas grandes é difícil, principalmente quando feita em um, ou poucos, chips.

2.3 - Implementação em Hardware Digital

Uma das desvantagens dos circuitos digitais é que um neurônio digital requer mais transistores do que um neurônio analógico. Uma tecnologia que reduz bastante esse problema é a "Wafer Scale Integration" (WSI). Essa tecnologia interliga vários circuitos integrados existentes em um wafer formando um super-circuito.

Em um wafer são colocados centenas de circuitos integrados básicos que implementam parte de uma rede neuronal, um neurônio por exemplo. Após sua confecção o wafer é testado para verificar quais os "módulos neuronais" que estão funcionando. Como a complexidade desses módulos é pequena, devido à simplicidade eletrônica de um neurônio, a quantidade de circuitos defeituosos não é muito grande. Os módulos em funcionamento são interligados pela colocação de camadas metalizadas sobre o wafer como se fossem fios, formando assim uma rede com centenas de neurônios.

Os problemas com dimensionamento (um wafer tem aproximadamente

10 cm de diâmetro) e estabilidade são bastante reduzidos. Porém surgem outros problemas, sendo o mais crítico a emissão térmica. Se apenas um circuito integrado encapsulado individualmente se aquece, principalmente em altas freqüências, imagine a quantidade de calor gerado por um wafer que contém centenas de circuitos em operação simultânea e separados por apenas alguns microns!

Esse grave problema térmico tornou o encapsulamento desses wafers uma tarefa quase impossível e sua comercialização inviável até o momento. De qualquer forma, a tecnologia WSI ainda está sendo utilizada para implementações eletrônicas de redes neuronais, porém para funcionamento apenas em laboratórios.

2.4 - Implementação Ótica

O treinamento e uso de redes neuronais requer, basicamente, dois tipos de operações: computação e comunicação. Circuitos integrados existentes operam em nano-segundos, têm dimensões expressas em microns, e custam milésimos de centavos de dolar por gate fabricado. Isso faz da operação computação uma tarefa satisfatória em sistemas eletrônicos.

Sinais eletrônicos precisam de condutores para levá-los de um gate ao outro de um circuito integrado. Embora a espessura desses condutores seja de poucos microns, o espaço para eles necessário (e também para isolá-los) pode se tornar tão grande que a área de silício restante pode ser insuficiente para acomodar os circuitos que fazem a computação.

Nos neuro-computadores óticos os neurônios são conectados através de feixes luminosos ao invés de condutores metálicos. Os caminhos óticos não precisam ser isolados e podem ser feitos em três dimensões, enquanto circuitos integrados são essencialmente planares com alguma flexibilidade obtida pela técnica de múltiplas camadas. Uma característica interessante da Ótica é que feixes luminosos podem se cruzar sem provocar interferências. A densidade dos caminhos óticos fica limitada apenas pelo espaçamento entre os foto-emissores e foto-detetores, e pela dispersão (espalhamento ótico) dos foto-emissores. Essas distâncias normalmente são de poucos microns.

Finalmente, todos os caminhos óticos podem estar operando simultaneamente, aumentando a taxa de transmissão da rede. Consegue-se assim um sistema capaz de obter conectividade total operando na velocidade da luz.

Além das vantagens já apresentadas, os neuro-computadores óticos podem armazenar os pesos das conexões sinápticas em hologramas de alta densidade. Estimativas teóricas limitam em 10**12 bits por cm3 essa capacidade de armazenamento. Embora essa capacidade ainda não tenha sido obtida na prática, a potencialidade extremamente alta de armazenagem não pode ser ignorada.

Dadas todas essas vantagens, acrescidas do potencial computacional desenvolvido na Ótica, poder-se-ia perguntar: Por que se fariam redes de outras formas? Infelizmente, implementações óticas apresentam diversos problemas práticos. Dispositivos óticos têm suas características próprias, algumas vezes relacionadas com comunicações em fibras óticas, que normalmente não coincidem com as necessidades das redes neuronais. Outros motivos são custo elevado, grande tamanho físico e alinhamento crítico.

Embora redes neuronais óticas sejam mais adequadas em processamento de imagem, os resultados até agora obtidos são bastante fracos. Independentememte dos problemas, o potencial de sistemas óticos continua a motivar os esforços de pesquisas nessa área. As configurações atualmente em estudo de redes neuronais óticas podem ser divididas em duas categorias: **MULTIPLICADORES VETOR-MATRICIAIS** e **CORRELATORES HOLOGRÁFICOS.**

Esse trabalho requer conhecimentos básicos na área que podem ser consequidos em [5].

3 - SIMULANDO UMA REDE NEURONAL NO COMPUTADOR PARALELO AAP-2

3.1 - Introdução

É descrita uma implementação paralela do algoritmo de aprendizado por **back-propagation** no computador AAP-2 [1].

O AAP-2, desenvolvido pela NTT, é um processador vetorial celular maciçamente paralelo baseado na tecnologia VLSI. Ele consiste de 65536 processadores de um bit, proporcionando mecanismos de transferência de dados em altas velocidades e diversas operações modificáveis no processador.

3.2 - Arquitetura do AAP-2

O AAP-2 é formado por 65536 processadores elementares (PEs) e pode operar como processador auxiliar de um computador hospedeiro. A figura 1 descreve a configuração do sistema AAP-2, que se divide em quatro componentes principais: um vetor de processadores elementares (PEs), uma unidade de controle vetorial, uma memória para buffer de dados, e uma unidade de interface com o computador hospedeiro.

O diagrama de blocos de um PE é visto na figura 2. Todos os processadores são controlados por uma seqüência de instruções comuns (SIMD) enviada a todos pela unidade de controle vetorial, e são conectados aos seus vizinhos PEs mais próximos em estrutura matricial toroidal (torus). Nessa estrutura os elementos da primeira e última linha são vizinhos, isso é se comunicam, o mesmo ocorrendo com os elementos da primeira e última coluna.

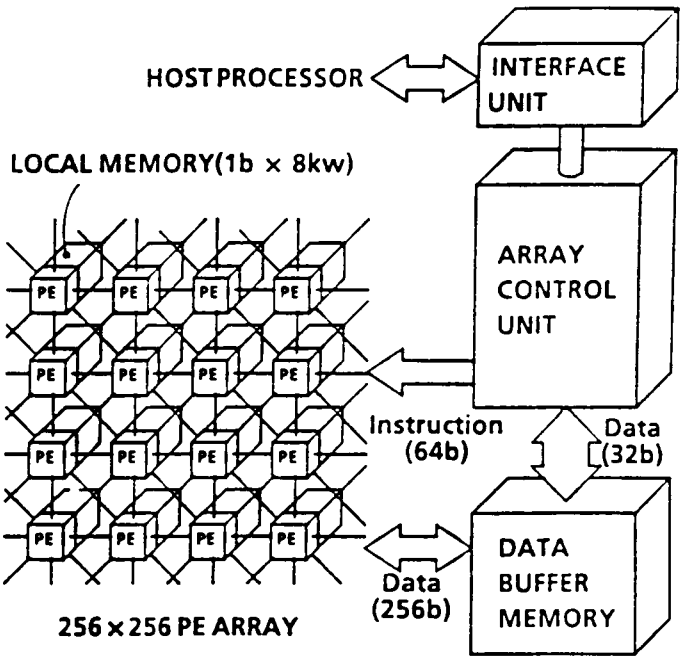


Figura 1 : Configuração do Sistema do AAP-2

Em cada PE as operações aritméticas e lógicas, as saídas dos multiplexadores, e a direção de transferência dos dados podem ser alterados através de um registro de controle. A estrutura dos caminhos de dados consiste de um barramento direcionável de oito e outro de quatro vias de dados de/para os vizinhos.

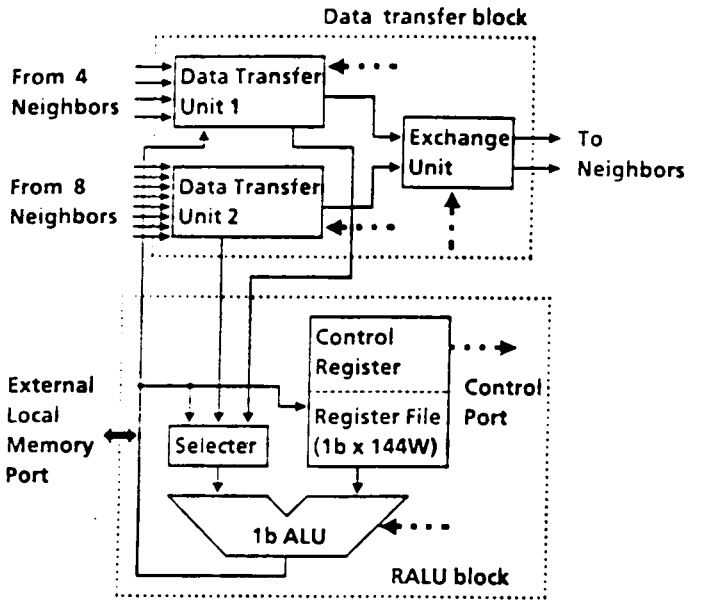


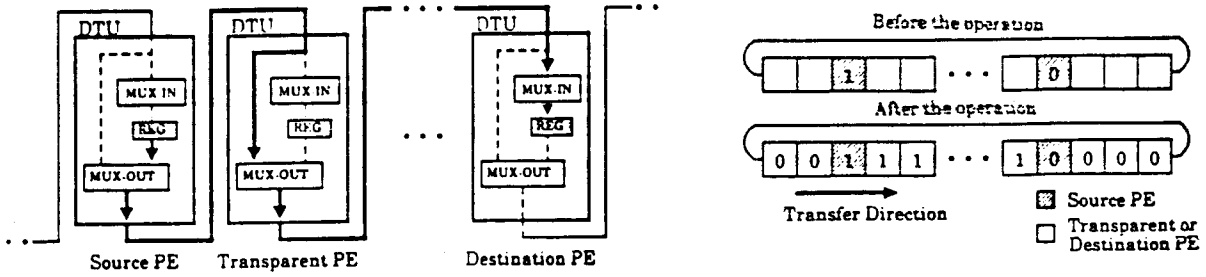
Figura 2 : Diagrama em Blocos de um Processador Elementar (PE)

O número de processadores, ou tamanho do vetor, do AAP-2 pode ser expandido de 64K até 1M processadores. Características do AAP-2 importantes para computação neuronal serão vistas a seguir.

3.3 - Mecanismos de Transferência em Alta Velocidade

Esse mecanismo pode ser dividido em duas partes:

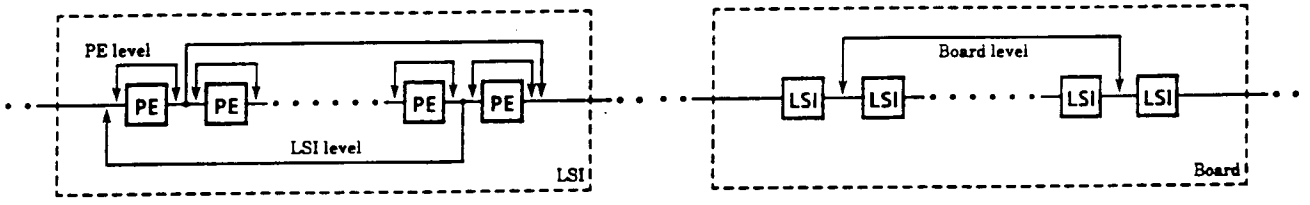
a) **Transferência por propagação:** essa operação, vista na figura 3, é adicional à operação normal de deslocamento (shift). O dado no PE origem é transmitido para um PE distante tão rápido quanto o sinal possa se propagar, assincronamente em cada período de clock, através dos PEs intermediários (transparentes). Essa operação é repetida de acordo com a distância da propagação. São necessários 20 ciclos de máquina, no pior caso, para transferir um dado através de 256 PEs, o que é 13 vezes mais rápido do que se fossem usadas operações de deslocamento.



(a) (b)

Figura 3 : Transferência por Propagação
(a) Diferenças funcionais entre PEs
(b) Antes e depois da operação

b) **Transferência por desvios (bypass) hierárquicos:** os PEs estão conectados em estrutura torus permitindo desvios a níveis de PEs, chip (LSI) e placa (PCB). Cada chip (LSI) é formado por 7 PEs, e cada placa é formada por 6 chips. Os desvios a nível LSI e PCB podem ocorrem por linha ou coluna de PEs, como visto na figura 4, enquanto o desvio a nível PE corresponde à transferência por propagação.



(a) (b)

Figura 4 : Conexão por Desvios Hierárquicos
(a) Desvio nos LSIs **(b)** Desvio nas PCBs

3.4 - Comunicação Entre Processadores

Em uma máquina maciçamente paralela, a comunicação entre processadores influencia bastante a eficiência do processamento paralelo. Existem três possibilidades de comunicação entre processadores usando os mecanismos de transmissão descritos:

a) **Conexão com vizinhos mais próximos: (1)** Com uma operação de deslocamento (shift) todos os dados em um vetor de PEs são transferidos para seu vizinho mais próximo. **(2)** Com operações de transferência por propagação dados podem ser transmitidos entre PEs distantes através de PEs intermediários, onde os dados não precisam ficar armazenados.

b) **Conexão por caminho programável:** Isso permite que qualquer PE se comunique com qualquer PE distante através de uma conexão direta. Dependendo do caminho entre PEs origem e destino, que é determinado por um roteador em dois níveis controlado por software, dados de comando são colocados no registro de controle de direção das transferências, e então os dados são transferidos através do caminho roteado. Um exemplo é visto na figura 5.

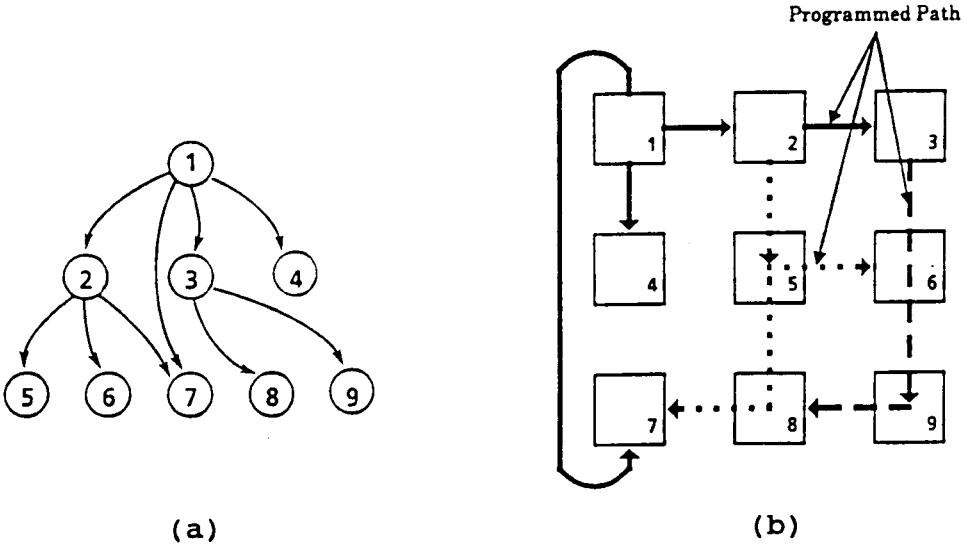


Figura 5 : Exemplo de Conexão com Caminho Programado
(a) Conexão entre nós **(b)** Caminho programado

c) **Conexão por pacote de dados:** essa transferência é controlada por software.

3.5 - Outras Operações Úteis

Outras operações existentes no AAP-2, e que foram úteis na implementação da simulação de uma rede neuronal, são:

(1) **Soma por Propagação:** é uma operação de soma rápida entre PEs

de uma linha ou coluna, utilizando a transferência por propagação. São necessários 30 ciclos de máquina por bit para somar variáveis em 256 processadores.

(2) **Broadcast para Todos os Processadores:** essa operação pode ser feita em apenas um ciclo por bit, utilizando-se uma operação da ALU.

(3) **Broacast pela Linha ou Coluna:** apenas 5 ciclos de máquina por bit são necessários para enviar uma mensagem para toda uma linha ou coluna de 256 processadores.

(4) **OR Global:** a operação de OR entre todos os processadores pode ser feita em apenas 3 ciclos de máquina.

3.6 - Algoritmo de Back-Propagation

O algoritmo de back-propagation é um procedimento de correção do erro na aprendizagem. Os passos desse algoritmo podem ser resumidos como:

(1) Propagação Forward:
   a) distribuir as ativações/valores de entrada das unidades para seus pesos respectivos;
   b) multiplicar as ativações pelos valores dos pesos;
   c) somar os valores calculados para a camada seguinte de unidades;
   d) aplicar a função de ativação a essa soma.

(2) Propagação Backward:
   a) distribuir os valores dos erros das unidades para seus pesos respectivos;
   b) multiplicar os erros pelo valor dos pesos;
   c) somar os valores calculados para a camada anterior de unidades;
   d) calcular a derivada da função de ativação;
   e) atualizar os pesos com o gradiente dos erros.

Embora esse algoritmo deva ser executado seqüencialmente, cada passo pode ser calculado em paralelo alocando-se um processador para cada peso sináptico.

3.7 - Implementando o Algoritmo de Back-Propagation

Para se obter processamento eficiente em máquinas paralelas deve-se equalizar e minimizar o volume de comunicação entre os processadores. Na implementação no AAP-2 foi alocado um processador para cada peso sináptico em cada uma das camadas. Assim, cada PE simula uma conexão entre nós de cada camada.

Para ilustrar assume-se que a rede é totalmente conectada, com o número de unidades nas camadas de entrada, escondida e saída

iguais à figura 6(a). A alocação dos pesos entre os PEs é dada na figura 6(b). Essa alocação permite transferências de dados em paralelo sem conflitos nos vetores de PEs.

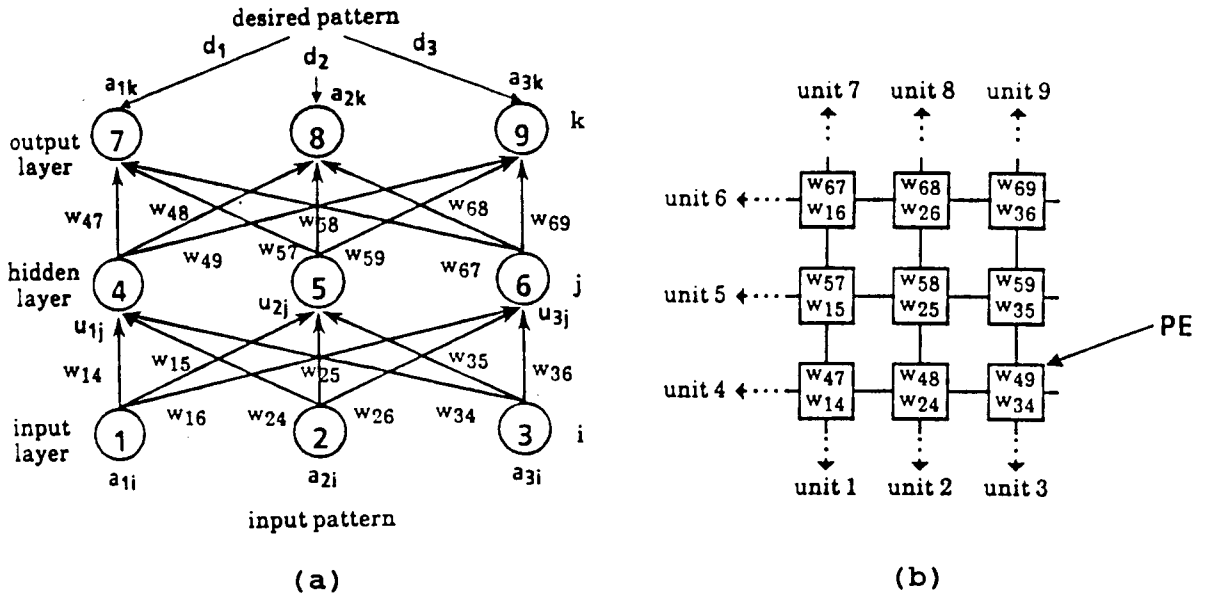


Figura 6 : Alocação de Processadores
(a) Rede de 2 camadas **(b)** Alocação de pesos

Com esta alocação a soma ponderada pelos pesos pode ser feita em toda uma linha/coluna de PEs trocando-se a direção da transferência de dados para horizontal (linha) e vertical (coluna) alternadamente em cada camada.

A única operação nesta implementação é o cálculo dos valores de ativação, o que é feito segundo o método da tabela de verificação (table-look-up), ao invés de operações de ponto flutuante.

Os passos da implementação são:

(1) Propagação Forward (figura 7(a))

a) os valores iniciais dos pesos, as tabelas da função sigmóide (x, f(x), y, f(y)), e os dados de entrada são enviados por broadcast pelo computador hospedeiro em pedaços de 256 bits de largura, utilizando transferência por propagação e por desvios;

b) os valores de ativação são multiplicados pelos pesos em cada PE, e os resultados são somados ao longo de cada linha (coluna) usando-se soma por propagação;

c) os resultados das somas, que são os valores de entrada para as unidades da próxima camada, Ui(1), são enviados por broadcast ao longo de cada linha (coluna) através de transferência por propagação;

d) os resultados, aplicação da função sigmóide ao valor de Ui(l), podem ser obtidos calculando-se (Ui(l) - x) para cada PE. Finalmente os resultados são enviados por broacast ao longo de cada linha (coluna) como sendo os valores de ativação para a próxima camada.

(2) Propagação Backward (figura 7(b))

a) Os valores dos erros em cada unidade são calculados em cada PE;

b) os valores dos erros são multiplicados em cada PE pelos pesos das conexões com a camada anterior, e os resultados são somados ao longo de cada linha (coluna) usando-se soma por propagação;

c) os resultados das somas são enviados por broadcast ao longo de cada linha (coluna) através de transferência por propagação;

d) calcula-se a derivada da função sigmóide e os pesos são atualizados.

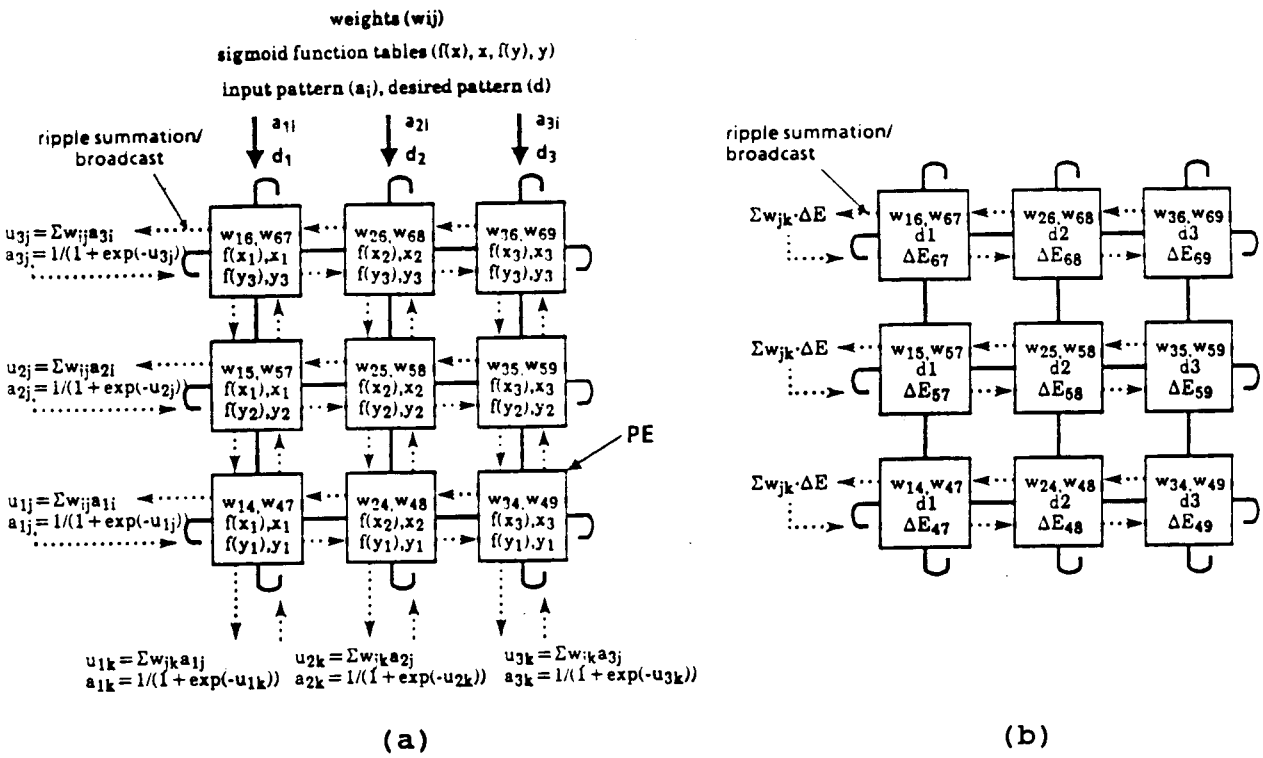


Figura 7 : Implementação Paralela no AAP-2
(a) Propagação forward **(b)** Propagação backward

As figuras 6 e 7 mostram uma implementação para uma rede com três camadas. As direções das transmissões por broadcast e das operações de soma são alternadamente trocadas para cada camada.

3.8 - Resultados

A implementação de back-propagation foi utilizada para reconhecimento de caracteres. O programa foi escrito em AAPL, que é uma linguagem similar a APL.

A rede usada na implementação possui três camadas (entrada, escondida e saída), cada uma delas formada por 256 unidades. As unidades de cada camada são totalmente interconectadas, tanto para a camada anterior como para a camada seguinte, o que significa 131072 conexões entre as camadas de entrada e saída.

Nessa implementação o tempo de execução para uma rodada de aprendizado foi de 7.33 mili-segundos, correspondendo a 18 milhões de conexões por segundo (MCPS). Isso inclui as propagações forward e backward, leituras dos padrões de entrada e de saída desejada, cálculo dos erros e dos novos pesos.

O tempo de execução é dividido, por operações, nas seguintes proporções:

(1) Aritméticas: 70%
incluindo multiplicações, somas e subtrações

(2) Deslocamento de bits (shift): 5%
feito após multiplicações

(3) Comunicações entre PEs: 20%
incluindo broadcast para todos os processadores, broadcast ao longo de linhas (colunas) de processadores, e somas por propagação;

(4) Outros: 5%
na maior parte substituições simples.

A maior parte do tempo é consumida com operações internas aos processadores. Isso mostra que a implementação no AAP-2 obteve boa performance pois mantém os processadores ocupados a maior parte do tempo.

MÁQUINA	MCPS	VELOCIDADE RELATIVA
SUN-3/60	0.03	1
IBM 3090	0.40	13
16k CM-1	2.6	87
M380 com VP-100	4.7	157
Warp	17	567
AAP-2	18	600

Tabela 1 : Comparação no Tempo de Execução

A tabela 1 compara o tempo de execução do algoritmo de back-propagation em várias máquinas.

4 - UM NEURO-COMPUTADOR ANALÓGICO DE USO GERAL

4.1 - Introdução

Essa máquina [2] destina-se a processamentos em tempo real de dados do mundo real, tais como visão, acústica ou robótica, e o desenvolvimento de redes neuronais especiais. Atualmente os chips dessa máquina estão sendo fabricados e testados.

O computador é configurável e composto de módulos interconectados com vetores de neurônios, sinapses e chaveadores modificáveis. Ele executa totalmente em modo analógico, mas a arquitetura das conexões, os ganhos sinápticos, as constantes de tempo, e os parâmetros dos neurônios são estabelecidos de forma digital.

Cada neurônio tem um número limitado de entradas, podendo se conectar a quaisquer, porém não a todos, neurônios.

4.2 - Arquitetura

A arquitetura do computador analógico pode ser vista na figura 8.

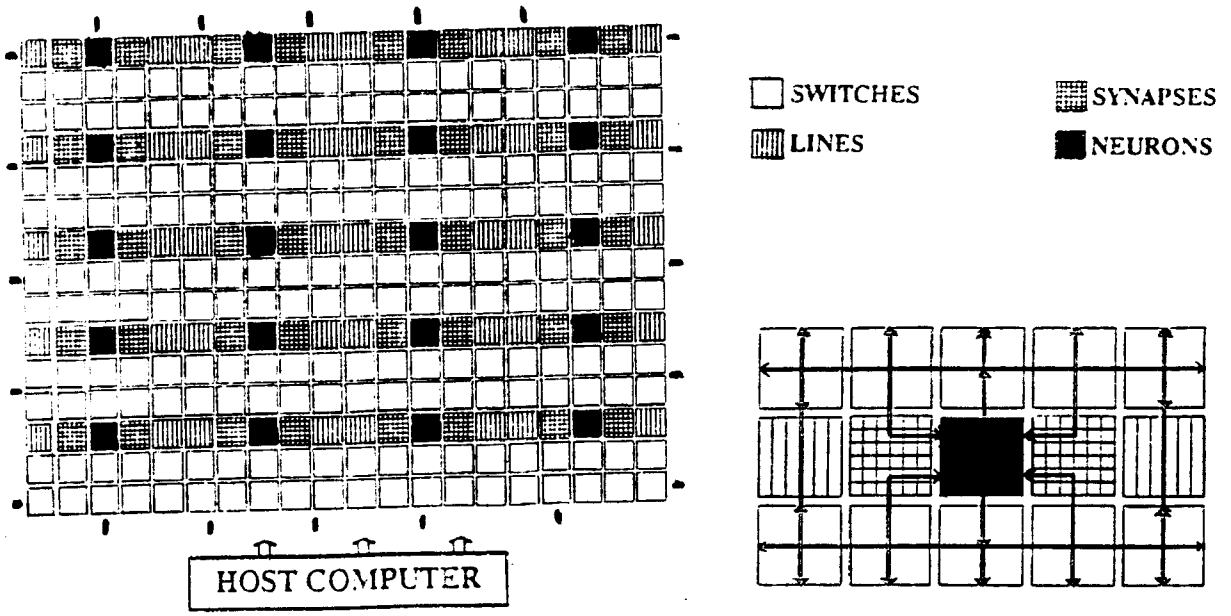


Figura 8 : Arquitetura do Computador
(a) Layout e arquitetura geral
(b) Direção dos dados pelos módulos

A máquina é formada por grande número dos seguintes elementos: **neurônios**, **sinapses**, **chaveadores de roteamento**, e **linhas de**

conexão.

O desenho modular permite expandir a rede para qualquer tamanho. Estima-se, para redes entre 1000 e 100000 neurônios, que sua velocidade de operação será maior que a de computadores digitais disponíveis.

Vetores desses elementos são fabricados com tecnologia VLSI. Os módulos são arrumados planarmente e conectados diretamente a seus vizinhos.

Os vetores de neurônios estão arrumados em linhas e colunas, figura 8(a), tendo em volta vetores de sinapses e axiomas. O protótipo que está sendo desenvolvido terá 50 módulos de neurônios, totalizando 800 neurônios cada um com 64 sinapses.

As saídas de cada neurônio estão acima e abaixo, sendo roteadas através dos módulos chaveadores, que estão à esquerda e direita, para os módulos sinápticos acima e abaixo. As entradas para os neurônios, através das sinapses, estão à direita e esquerda. Alimentação e linhas digitais de controle correm de cima para baixo.

4.3 - O Módulo de Neurônios

Cada chip de neurônios contém 16 neurônios, um multiplexador analógico, e lógicas de controle (figuras 9 e 10). Cada unidade tem threshold (alimentação) ajustável, valor mínimo de saída no threshold ajustável, e saída máxima (figura 11).

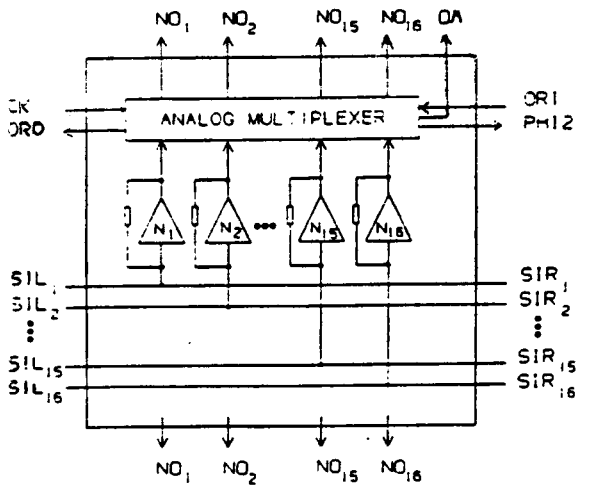


Figura 9 : Diagrama em Blocos do Chip com 16 Neurônios

As entradas de cada neurônio chegam de chips de sinapses localizados à direita e esquerda (SIR, SIL). As saídas (NO) vão para chips de chaveadores acima e abaixo.

Cada neurônio tem uma segunda entrada que estabelece o valor mínimo de saída no threshold, comum aos 16 neurônios do chip, selecionada através de uma linha de sinapse separada. O threshold é estabelecido a partir de uma sinapse conectada a uma tensão fixa.

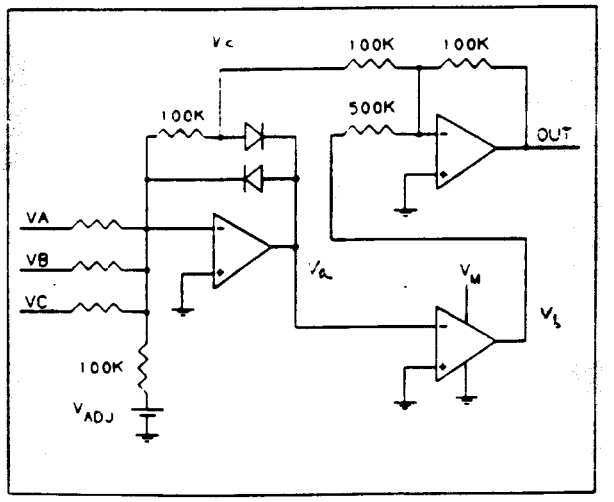


Figura 10 : Circuito de um Neurônio

Durante o aprendizado, os ganhos sinápticos e os parâmetros do neurônio são calculados digitalmente, de acordo com os valores de saída dos neurônios. Para que isso seja possível, um multiplexador analógico envia a saída do neurônio para uma linha comum (OM), que está ligada a um conversor A/D rápido cuja saída é armazenada na memória. Passando-se pulsos de controle (ORO e ORI) de chip para chip, todos os neurônios são lidos seqüencialmente a cada 2 mili-segundos.

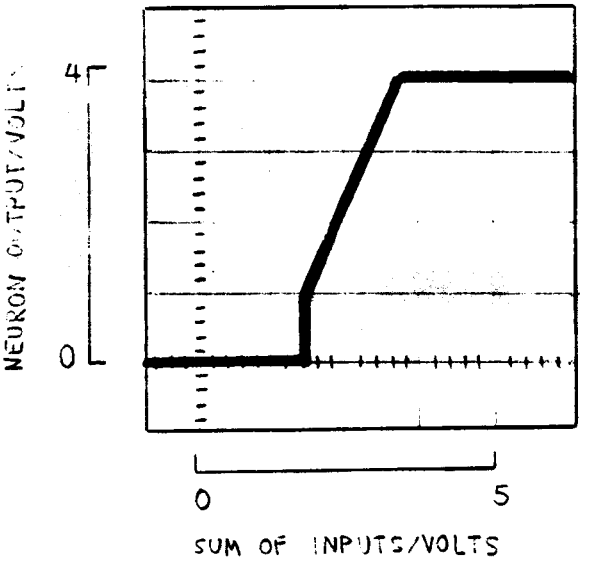


Figura 11 : Função de Transferência do Neurônio da Figura 10

4.4 - O Módulo de Sinapses

Cada chip de sinapses contém 32 x 16 vetores de sinapses. O ganho sináptico de cada sinapse é estabelecido através de entrada serial vinda do computador, que é armazenado em cada sinapse. O ganho sináptico pode variar entre 0 e 10 com uma resolução de 5 bits, havendo um sexto bit para o sinal. Os valores dos ganhos sinápticos variam logaritmicamente.

O diagrama em blocos desse módulo é visto na figura 12. Ele é formado por um vetor de 32 por 16 sinapses e outro vetor similar com memória de 6 bits.

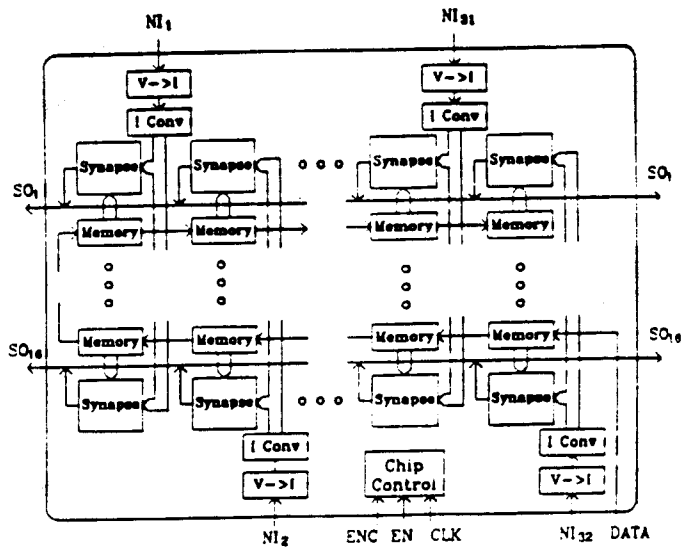


Figura 12 : Diagrama em Blocos do Chip de Sinapses

O chip tem 32 linhas de entrada (NIj) vindas das saídas dos neurônios roteadas pelos chips chaveadores. Os valores das entradas, que podem variar entre 0 e 4 volts, são transformados em corrente por um conversor V-I.

Existem 16 linhas de saída (SOj) que contêm a soma das correntes na saída dos 32 conversores V-I da coluna i de sinapses. Essas saídas são conectadas às 16 entradas correspondentes dos neurônios vizinhos, como foi mostrado na figura 8.

Outras 16 entradas adicionais (EI), permitem aumentar o número de sinapses na entrada de um neurônio de 32 para 64 ou mais, cascateando-se chips de sinapses.

Embora a resolução de cada sinapse seja limitada a 5 bits, várias sinapses alimentadas por um neurônio podem ser combinadas, através dos chaveadores, permitindo grande resolução e intervalos variados.

4.5 - O Módulo de Chaveadores

Os módulos de chaves servem para rotear os sinais entre os neurônios, e para permitir alterações na arquitetura da rede. Cada módulo contém um vetor de cruzamento pontual de 32 x 32 chaves analógicas configuráveis por entrada digital serial.

A figura 13(a) mostra o diagrama em blocos com as principais partes do chip.

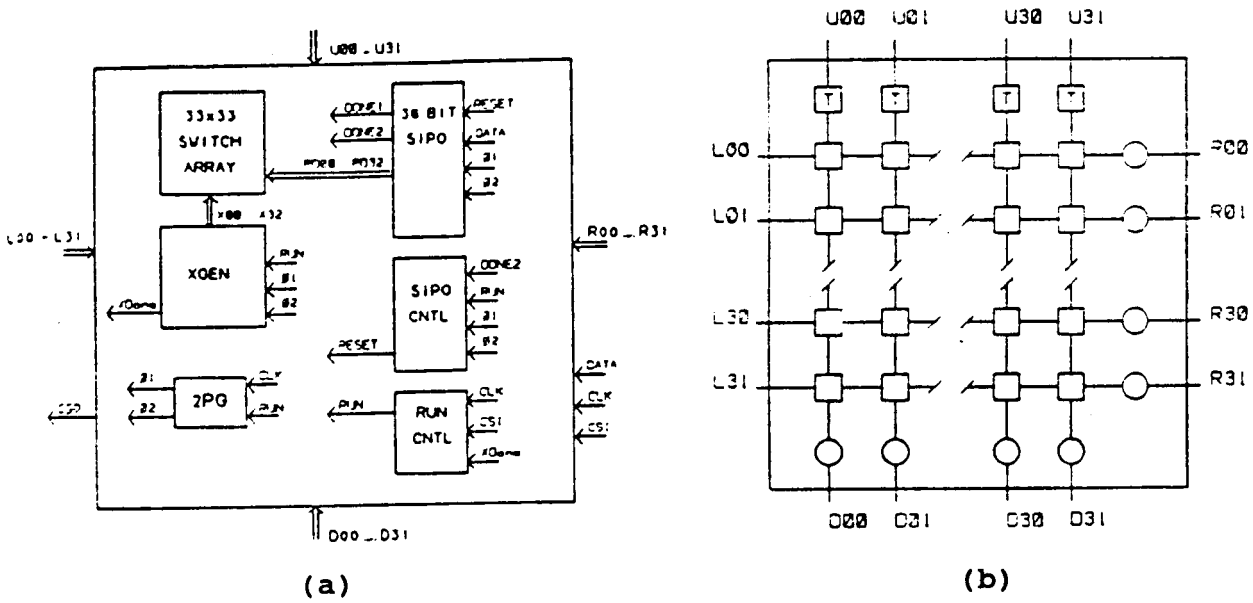


Figura 13 : Diagrama em Blocos do Chip de Chaveadores
(a) Principais partes **(b)** Diagrama dos chaveadores

Ele é formado de chaveadores, lógica de controle, registro de deslocamento com entrada serial e saída paralela (SIPO) e controle, gerador de pulso de escrita (XGen) e gerador de relógio de duas fases (2PG).

A figura 13(b) mostra o diagrama em blocos dos chaveadores. Cada quadrado representa uma chave analógica com memória de controle de chaveamento com um bit. Os dados de controle podem conectar qualquer linha horizontal a qualquer outra vertical, bastando escrever "1" na célula de memória apropriada.

Os círculos ao longo dos lados direito e inferior também representam chaves e memória. Essas chaves, em série com as linhas horizontais e verticais, permitem que o micro-processador corte uma linha horizontal ou vertical no chip. Isso permite que os barramentos de ligação possam ser particionados em vários segmentos, aumentando o número de conexões que se pode obter.

Estes módulos também contêm circuitos para controlar as constantes de tempo da função de transferência das sinapses, indicado por um quadrado com a letra "T" na figura 13(b).

4.6 - Ajuste das Constantes de Tempo das Sinapses

Para analisar ou gerar padrões temporais como eles ocorrem no movimento ou na voz, deve ser possível ajustar as constantes de tempo da função de transferência sináptica.

Para isso, o sinal de entrada de uma sinapse deve passar por um filtro passa baixa. Um controle de 4 bits na constante de tempo, permitindo intervalo entre 5 e 500 mili-segundos, é suficiente para tratar dados do mundo real.

Como nem todas as sinapses precisam dessa característica, o circuito será colocado apenas em um número limitado de linhas do chip de chaveadores.

5 - UMA REDE NEURONAL USANDO CIRCUITOS DIGITAIS WSI

5.1 - Introdução

Nessa rede neuronal WSI , três novas tecnologias [3] foram usadas: **barramento digital compartilhado no tempo, utilização eficiente da armazenagem dos pesos**, e **circuitos de controle de aprendizagem redundantes.** As duas primeiras permitem mais de 500 neurônios e 30000 sinapses em apenas um wafer de 8 centímetros de diâmetro. A terceira torna possível a implementação em apenas um wafer de silício.

Implementações eletrônicas de redes neuronais são bastante tolerantes a falhas. A compensação para neurônios com falha é possível quando não existem falhas nos circuitos de controle do aprendizado. Esses circuitos são usados para reescrever os pesos sinápticos de acordo com o algoritmo de aprendizado. Quando existe uma falha nesses circuitos, ocorrem aprendizados incorretos, e a rede não opera corretamente. O wafer funciona incorretamente. Nessa rede foram criados circuitos redundantes de controle do aprendizado, de forma a evitar este problema.

Um resumo das principais características dessa nova rede neuronal WSI é dado a seguir:

(1) Circuito: Completamente Digital

(2) Arquitetura: Barramento digital compartilhado no tempo; utilização eficiente da armazenagem dos pesos; e circuitos de controle de aprendizagem redundantes

(3) Complexidade: 540 neurônios e 34560 sinapses

(4) Saída do Neurônio: 9 bits

(5) Peso Sináptico: 8 bits

(6) Processo: 0,8 microns CMOS

(7) Metodologia de Projeto: Gate Array

(8) Tamanho do Wafer: 5 polegadas de diâmetro

5.2 - Barramento Digital Compartilhado no Tempo

Para conectar completamente uma rede de **N** neurônios, como na rede de Hopfield, são necessárias **N x N** sinapses. Para uma rede com 100 neurônios são necessárias 10000 sinapses. O hardware que está sendo descrito não é capaz de conter esse número de sinapses.

De forma a contornar esse problema, as conexões entre os neurônios são multiplexadas no tempo usando-se um barramento digital, como mostra a figura 14.

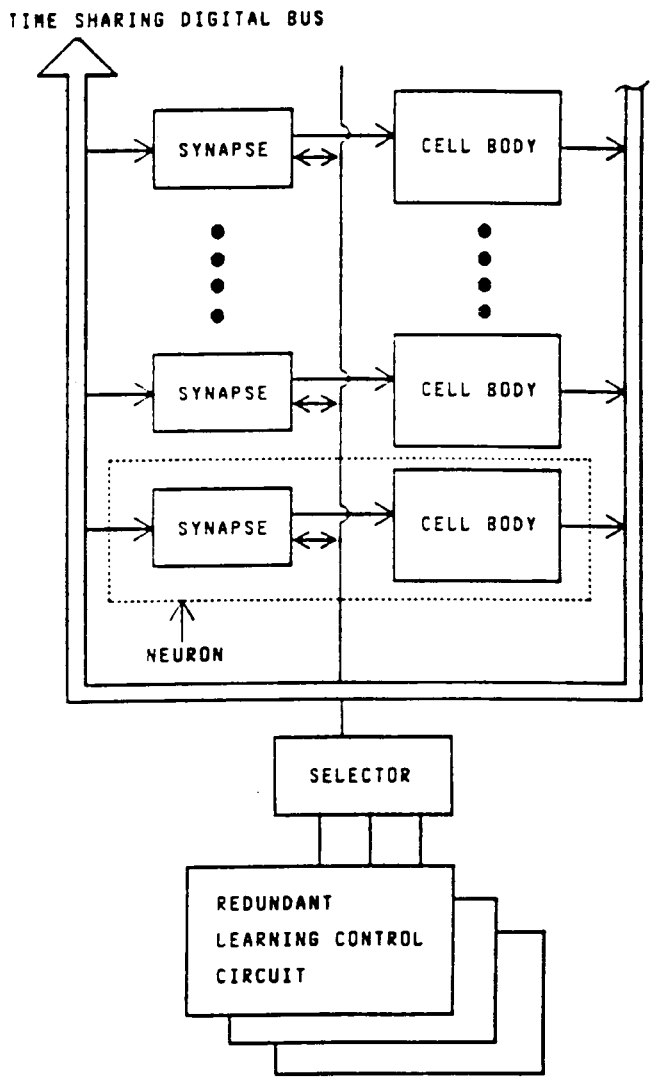


Figura 14 : Arquitetura da Rede Neuronal WSI

Nessa arquitetura cada neurônio necessita de apenas uma sinapse. Conseqüentemente, apenas **N** sinapses são necessárias para conectar totalmente cada um dos **N** neurônios.

Cada neurônio tem seu próprio endereço. O neurônio transmissor, selecionado pelo sinal de endereço, envia sua saída para outros neurônios, por broadcast, usando o barramento digital. O neurônio receptor lê esse valor de saída no barramento com o endereço do transmissor. O produto entre cada valor de saída com o peso armazenado na sinapse é acumulado na célula.

Todos os neurônios estarão completamente conectados com todos os demais depois que o endereço variar do primeiro ao último.

5.3 - Utilização Eficiente da Armazenagem dos Pesos

A tecnologia descrita no ítem anterior permite reduzir drasticamente o número de multiplicadores, pois cada neurônio precisa de apenas uma sinapse. Por outro lado, cada neurônio precisa de grande quantidade de circuitos para armazenar os pesos sinápticos para mais de 500 neurônios.

Para reduzir o número de circuitos de armazenamento, cada neurônio nesse circuito WSI armazena apenas o peso sináptico para os 64 neurônios mais próximos. Uma estrutura em árvore, mostrada na figura 15, é usada para aumentar o número de sinapses por neurônio. Nessa figura realiza-se um neurônio com 190 sinapses.

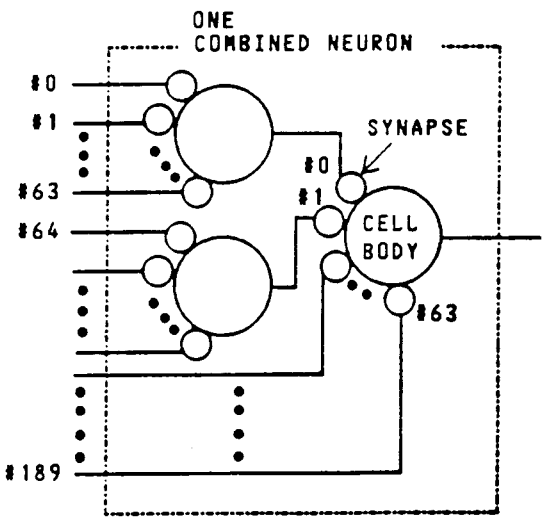


Figura 15 : Estrutura em Árvore de Neurônios

5.4 - Circuitos de Controle de Aprendizagem Redundantes

Redes neuronais podem compensar pelo mau funcionamento de neurônios usando um mecanismo organizado, desde que os circuitos de controle de aprendizagem estejam corretos.

Existe uma média de 10 falhas quando se usa um wafer de silício de 5 polegadas. Conseqüentemente, o número de neurônios a compensar é de 10 em 500 neurônios. Uma rede não será afetada pela não utilização de apenas 2% dos neurônios.

Entretanto, redes não são tolerantes a falhas quando elas existem nos circuitos de controle do aprendizado, usados para reescrever os pesos sinápticos. Para evitar esse problema, foram usados três circuitos de controle do aprendizado de forma redundante. Um dos circuitos de controle sem falha é selecionado e conectado à rede.

Fios e drivers de barramentos e circuitos periféricos, como controladores de barramento e entrada e saída, também foram desenhados de forma a ter redundâncias, tornando esses circuitos também tolerantes a falhas.

5.5 - Configuração do WSI

O desenho do WSI pode ser visto na figura 16. Um wafer é composto por 60 blocos de neurônios, 9 blocos de barramento, e um bloco com circuitos redundantes de controle de aprendizagem e circuitos periféricos. Os 9 blocos centrais arrumados longitudinalmente são blocos de barramento. Cada bloco de neurônios é formado por 9 neurônios. Assim, um wafer é composto de 540 neurônios.

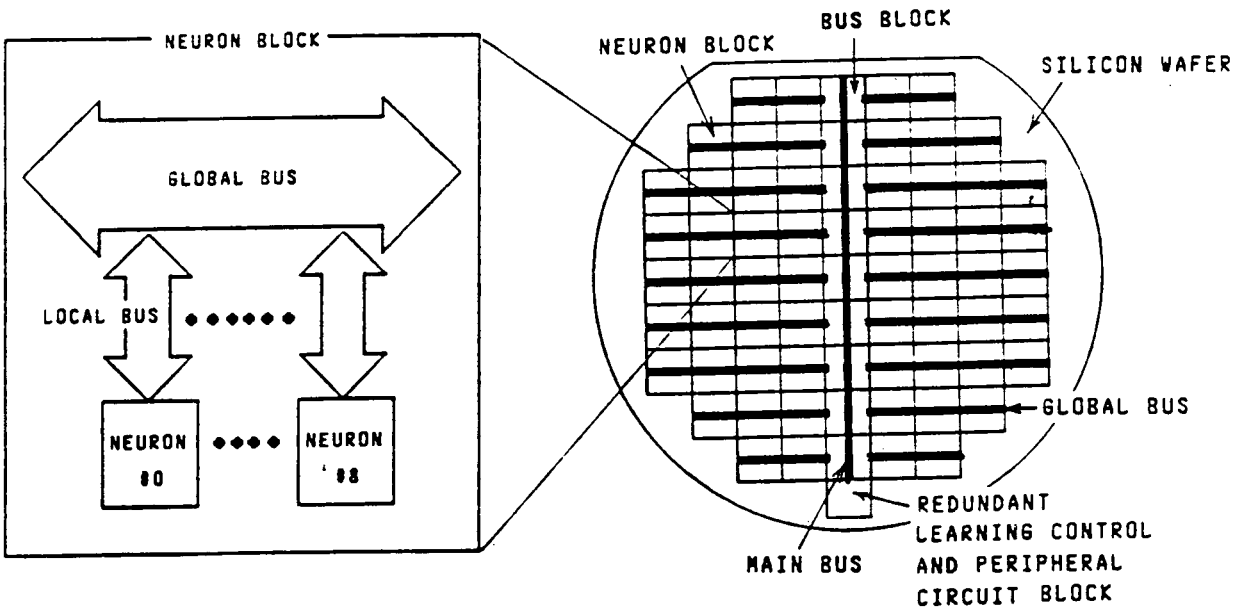


Figura 16 : Estrutura da Rede Neuronal WSI

É usada uma estrutura hierárquica para os barramentos. O barramento principal está colocado verticalmente no centro do wafer, através de blocos de barramento. Os barramentos globais são colocados horizontalmente em cada linha de blocos de neurônios. Cada neurônio se conecta ao barramento global mais próximo através do barramento local.

Tabelas da função sigmóide, desenhadas para terem redundâncias em cada bloco de barramento, são conectadas ao barramento principal. Transformações sigmoidais são realizadas no barramento principal.

5.6 - Neurônio

O diagrama do circuito de um neurônio é visto na figura 17. Uma sinapse é formada por um conjunto de registros e por um multiplicador. Os endereços dos neurônios e seus pesos sinápticos estão armazenados nesse conjunto de registros. Os produtos dos pesos com o valor de saída dos outros neurônios são calculados por esse multiplicador.

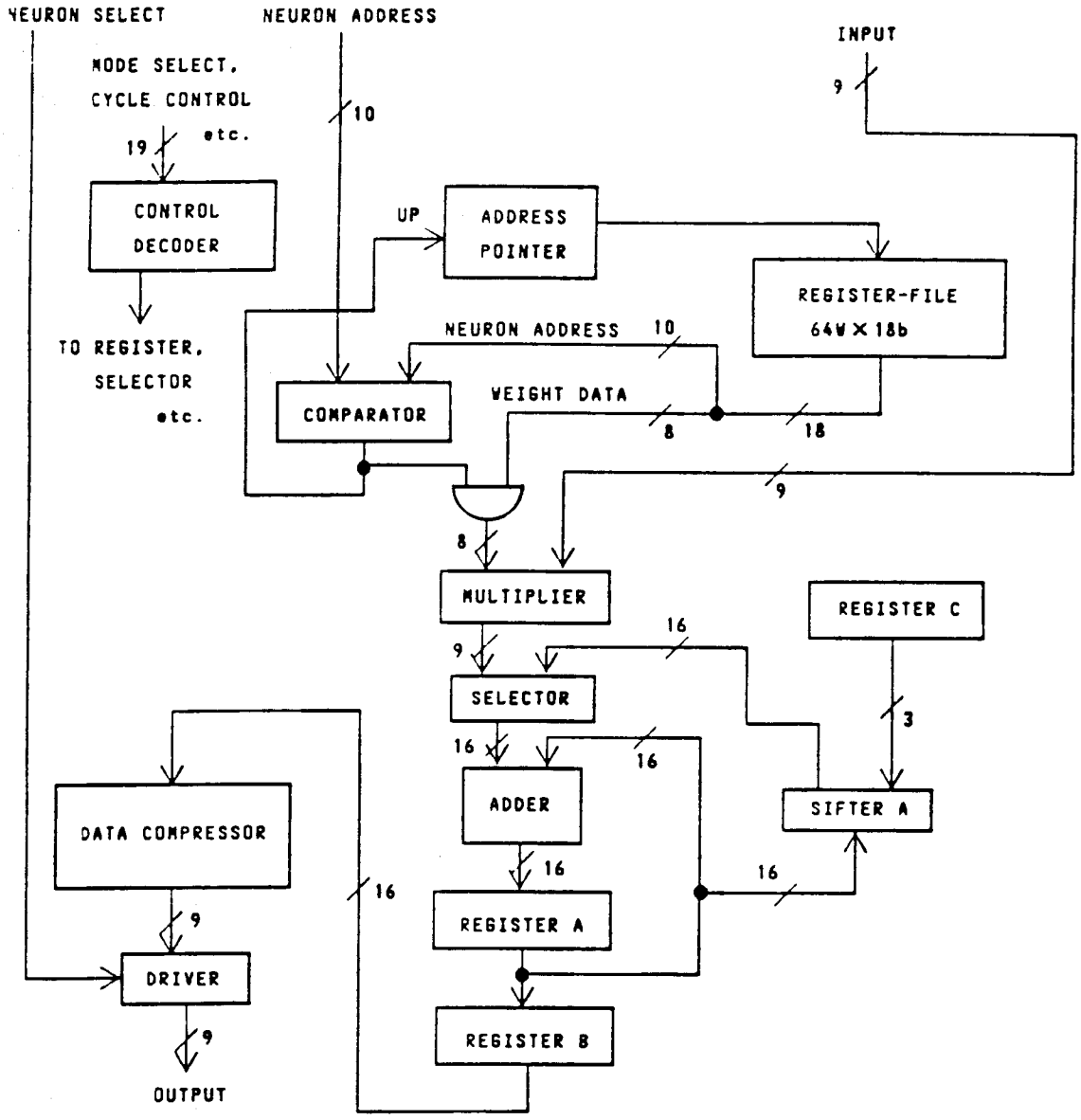


Figura 17 : Diagrama do Circuito de um Neurônio

A saída do multiplicador, isso é, a saída da sinapse, é acumulado usando-se o somador e o registro **A**. Depois que o endereço dos neurônios varia do primeiro ao último, o produto escalar entre o vetor de pesos e o vetor com as saídas dos neurônios estará armazenado no registro **A**.

O produto escalar do ciclo anterior fica armazenado no registro **B**. O valor no registro **B** é transmitido, por broadcast pelo barramento, aos demais neurônios quando ele é selecionado no ciclo.

Uma constante de realimentação é armazenada no registro **C**. Realimentação em um neurônio é tratada pelo registro **C** e pelo deslocador (shifter) **A**.

Nessa arquitetura a saída do neurônio é formada por 9 bits. Entretanto, o somador e os registros são de 16 bits. Os 9 bits mais significativos são escolhidos entre os 16 bits. Essa compressão de dados é tratada no circuito compressor de dados, que é formado por um registro e um deslocador.

O conjunto de registros armazena 64 pesos, que são lidos seqüencialmente. Para realizar acesso rápido ao dados, um ponteiro indica o dado que será lido em seguida. Essa estrutura é vista na figura 18.

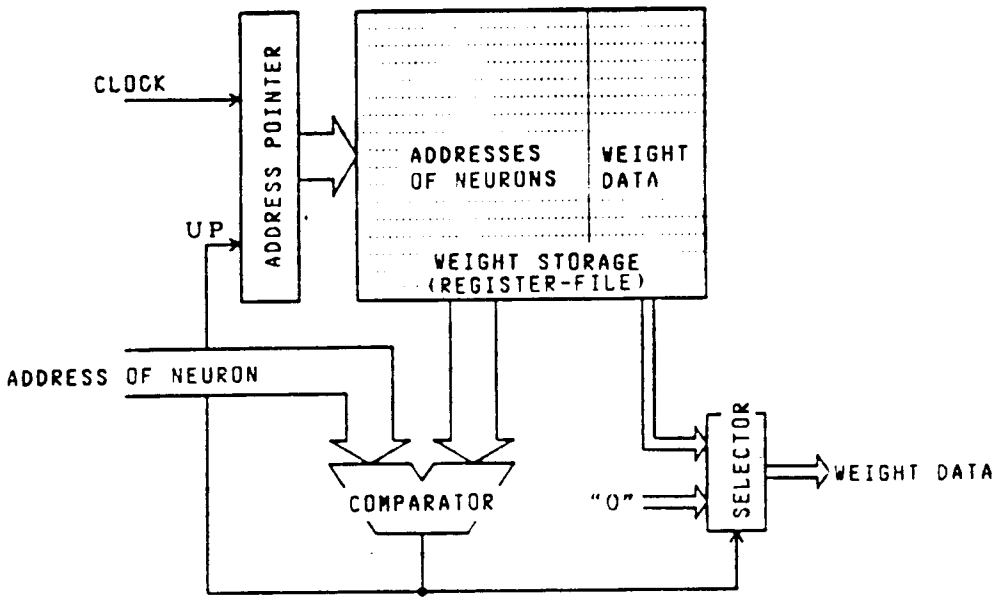


Figura 18 : Estrutura de Acesso Rápido aos Pesos

6 - NEURO-COMPUTADORES COM MULTIPLICADORES VETOR-MATRICIAIS

6.1 - Introdução

As operações na maioria das redes neuronais artificiais podem ser

resumidas matematicamente como uma série de multiplicações entre um vetor e uma matriz, uma para cada camada da rede.

Para calcular a saída de uma camada, um vetor de entrada é multiplicado pela matriz de pesos para produzir o vetor de saída. Esse vetor é, então, operado pela função de ativação para produzir o vetor de saída da camada.

Em redes neuronais biológicas essa operação é feita por um grande número de neurônios que operam simultaneamente. O sistema responde rapidamente independente da morosidade de cada neurônio.

Quando redes neuronais artificiais são simuladas em computadores de uso geral, o paralelismo natural dessa computação se perde. Independente da rapidez do computador onde é feita a simulação, o número de operações necessárias para fazer uma multiplicação matricial é proporcional ao quadrado da dimensão do vetor de entrada, tornando o tempo de computação extremamente longo para redes com alguma finalidade prática.

6.2 - Multiplicador Matricial Eletro-Ótico

Redes neuronais eletro-óticas executam a multiplicação matricial totalmente em paralelo [4]. O tempo de operação dessas redes é limitado apenas pelos dispositivos eletro-óticos disponíveis, situando-se na faixa de sub-nano-segundos.

A figura 19 mostra um sistema capaz de multiplicar um vetor de seis elementos por uma matriz de 6x5 elementos, gerando um vetor de cinco elementos como saída.

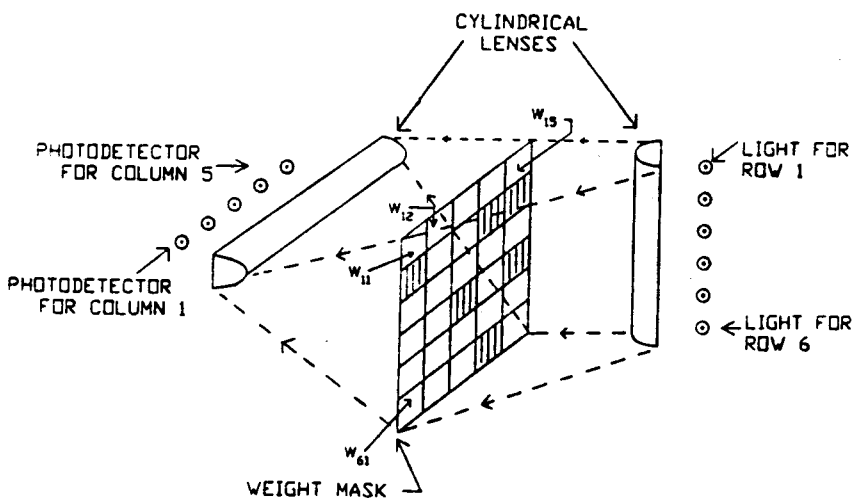


Figura 19 : Multiplicador Vetor-Matricial Eletro-Ótico

Na direita, a luz gerada por uma coluna de foto-emissores passa por uma lente cilíndrica. Essa lente torna uniforme a luz que ilumina apenas uma das linhas da máscara de pesos. Assim, o

emissor 1 ilumina as áreas W-11, W-12, ... W-15. A máscara de pesos pode ser um filme fotográfico onde a transmitância de cada quadrado (quantidade de luz que atravessa o filme) é proporcional ao peso.

Do lado esquerdo, uma segunda lente cilíndrica focaliza a luz que atravessa cada coluna da máscara no foto-detetor correspondente. Dessa forma, a luz que incide no detetor 1 é a soma dos produtos das intensidades luminosas multiplicadas pelas transmitâncias da coluna 1.

A saída de cada foto-detetor representa o produto entre o vetor de entrada e uma coluna na matriz de pesos. Dessa forma, o conjunto de saídas é igual ao produto entre o vetor de entrada e a matriz de pesos.

A multiplicação matricial é feita totalmente em paralelo, sendo o tempo total dessa multiplicação inferior a um nano-segundo. Esse tempo é totalmente independente do tamanho da matriz, fazendo com que a rede possa ser aumentada sem alterar o tempo da operação.

Nesse exemplo a matriz de pesos é fixa. Um método promissor utiliza uma máscara de cristal líquido em lugar do filme fotográfico. Isso permite que os pesos sejam alterados eletrônicamente em décimos de micro-segundos. Até o momento as máscaras de cristal líquido só são viáveis para pesos binários.

6.3 - Memória Associativa Bidirecional Eletro-Ótica

Se a saída do foto-detetor for utilizada para realimentar o foto-emissor correspondente produz-se uma rede de Hopfield eletro-ótica. Para isso, basta que uma função de ativação (threshold) seja incluída. Atualmente, essa função de ativação é melhor implementada por circuitos eletrônicos após cada foto-detetor.

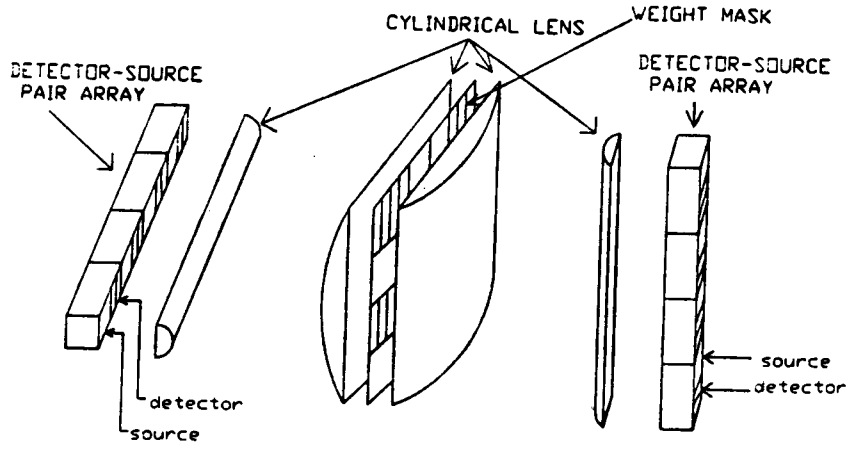


Figura 20 : Memória Associativa Bidirecional Eletro-Ótica

Se dois sistemas da figura 19 forem colocados em cascata, com a saída do segundo realimentando a entrada do primeiro, obtém-se uma memória associativa bidirecional eletro-ótica. Para garantir a estabilidade do sistema, a segunda matriz de pesos deve ser igual a transposta da primeira matriz.

A figura 20 descreve a implementação proposta por Kosko em 1987, que utiliza apenas um sistema ótico.

Nesse sistema, cada foto-emissor e foto-detetor é substituido por um par foto-emissor-detetor. A operação é similar à descrita na multiplicação vetor-matriz, exceto que cada foto-detetor alimenta seu foto-emissor adjacente.

Em operação, a luz de cada emissor da direita passa por uma lente cilíndrica e ilumina a linha correspondente na máscara de pesos. A segunda lente funciona como espalhador luminoso na direção horizontal, tornando-a uniforme (colimada) na direção vertical.

No lado esquerdo, cada detetor recebe toda a luz que ilumina uma coluna da máscara de pesos. Uma eletrônica implementa a função de ativação (threshold) cuja saída alimenta o emissor adjacente (lado esquerdo). Essa luz passa pelo sistema ótico e ilumina a mesma coluna.

Observe que o sistema ótico é atravessado por raios luminosos nos dois sentidos, ou seja, da direita para a esquerda e da esquerda para a direita. Como raios luminosos não interagem, isso não causa problemas.

No lado direito, cada detetor recebe toda a luz que ilumina uma linha da máscara de pesos. Uma eletrônica implementa a função de ativação cuja saída alimenta o emissor adjacente (lado direito). Completa-se dessa forma o caminho com realimentação.

Observe que a estabilidade do sistema é conseguida mesmo que a matriz não seja simétrica nem tenha sua diagonal principal igual a zero.

6.4 - Vetores Lineares de Modulação

O modulador linear é um dispositivo ainda em desenvolvimento, cuja grande vantagem será simplificar consideravelmente as estruturas das redes eletro-óticas.

Como pode ser visto na figura 21, ele é formado por uma placa fina composta de tiras alternadas de material sensível à luz (foto-detetores) e de moduladores (de intensidade) óticos. A permissividade de cada tira moduladora ótica pode ser alterada eletrônicamente.

A figura 22 mostra o esquema simplificado de um multiplicador vetor-matricial utilizando o dispositivo linear de modulação. As

tiras horizontais de modulação ótica são controladas eletrônicamente, tendo cada tira a transmitância proporcional a um dos elementos do vetor de entrada.

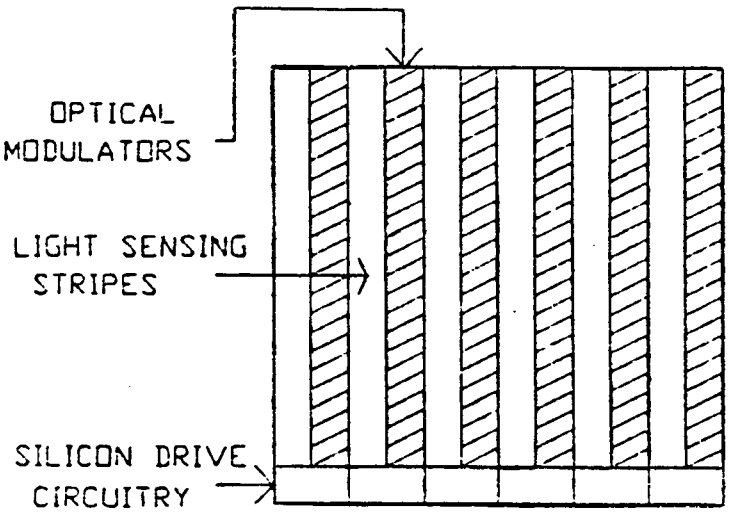


Figura 21 : Vetor Linear de Modulação Luminosa

Esse sistema não tem uma fonte luminosa independente para cada linha. Apenas uma fonte de luz uniforme (colimizada) ilumina o sistema (pela direita). A luz atravessa todas as tiras de moduladores e ilumina a máscara de pesos. No lado esquerdo tiras de sensores óticos captam a luz que ilumina uma coluna da máscara de pesos.

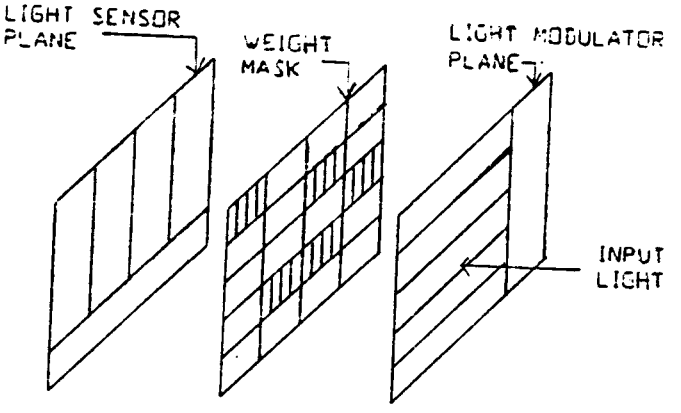


Figura 22 : Multiplicador Matricial Usando Moduladores Lineares

Devido à utilização de luz uniforme as lentes cilíndricas não são necessárias. A vantagem desse sistema compacto e oticamente simples pode ser desfeita pela relativa lentidão na sua operação. A tecnologia atual precisa de dezenas de micro-segundos para chavear os moduladores óticos.

6.5 - Memória Associativa Usando Moduladores Lineares

É similar ao multiplicador descrito no item anterior, exceto que cada tira de sensores óticos no lado esquerdo alimenta um circuito de ativação, que por sua vez controla a transmitância da tira moduladora adjacente (figura 23).

Uma segunda fonte de luz uniforme é necessária para iluminar os moduladores do lado esquerdo e todas as colunas da máscara de pesos.

A saída de cada tira de sensores óticos no lado direito alimenta um circuito de ativação, que por sua vez controla a transmitância da tira moduladora adjacente.

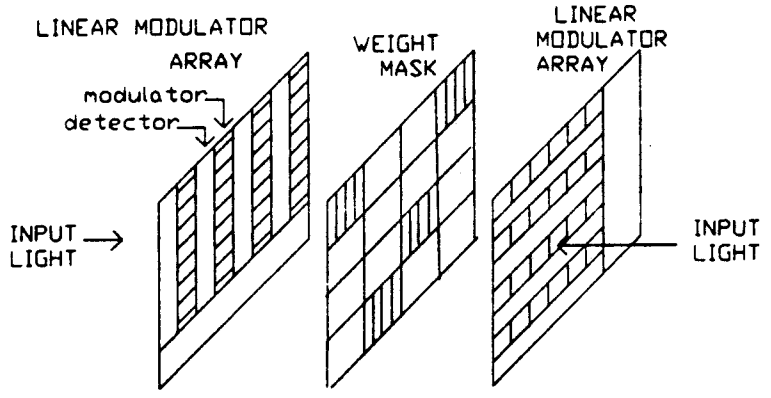


Figura 23 : Memória Associativa Usando Moduladores Lineares

7 - NEURO-COMPUTADORES COM CORRELATORES HOLOGRÁFICOS

7.1 - Introdução

Correlatores são sistemas que procuram quais dos objetos por ele armazenados (padrões) mais se assemelham a um objeto dado para comparação.

Embora existam vários tipos de correlatores holográficos, seus princípios fundamentais são os mesmos. Eles armazenam imagens padrões em hologramas finos ou volumosos, recuperando esses padrões através de um loop luminoso realimentado.

Uma imagem, que pode estar distorcida ou incompleta, é aplicada ao sistema e simultaneamente correlacionada oticamente com todos os padrões armazenados. Essas correlações são aplicadas a uma função de ativação e realimentadas para a entrada.

A correlação mais forte reforça, e possivelmente corrige ou completa, a imagem de entrada. A imagem melhorada passa repetidas vezes pelo loop ótico e se aproxima, a cada loop, do padrão

armazenado de que mais se assemelha. O sistema se estabiliza na imagem padrão desejada.

O termo imagem é utilizado para descrever os padrões que serão reconhecidos. Embora esse processo seja mais adequado para aplicações de reconhecimento de imagens, a entrada pode ser considerada como um vetor genérico e o sistema se transforma em uma memória associativa de propósito geral.

7.2 - Exemplo de Correlator Holográfico

A figura 24 mostra um sistema ótico usado para reconhecimento de imagems [4].

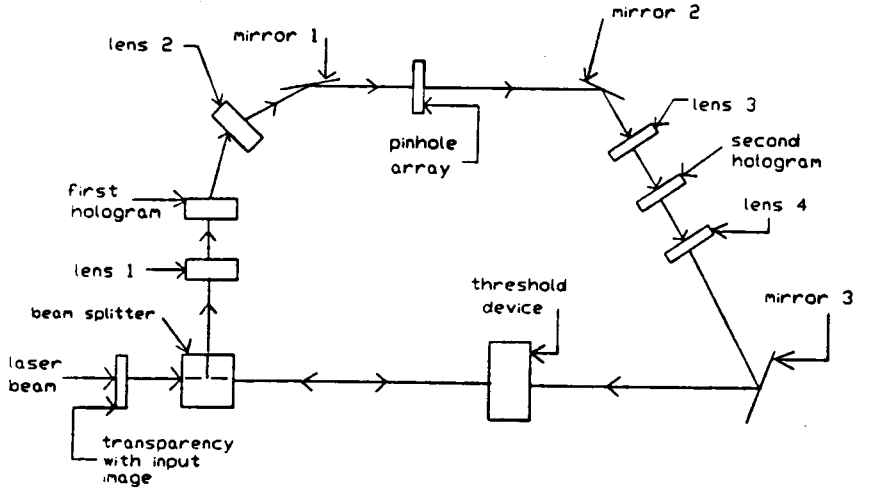


Figura 24 : Sistema Ótico de Reconhecimento de Imagens

A entrada do sistema é formada por uma transparência, onde está a imagem de entrada, que é ilumidada por um feixe de laser.

A imagem de entrada é aplicada ao "beam splitter" que a passa ao "threshold device", cuja finalidade será descrita mais adiante. A imagem é refletida no "threshold device", passa de volta pelo "beam splitter" e depois pela lente 1 que a focaliza no primeiro holograma.

No primeiro holograma estão armazenadas as diversas imagens para comparação. A imagem de entrada é correlacionada com cada uma dessas imagens no holograma produzindo padrões luminosos. O brilho desses padrões varia com o grau de correlação, que é a medida de similaridade entre duas imagens.

A lente 2 e o espelho 1 projetam esses padrões misturados no "pinhole array" onde eles são separados espacialmente (cada padrão em uma área diferente da nova imagem).

Os vários padrões luminosos passam pelo espelho 2 e lente 3 que os focalizam no segundo holograma, cujas imagens armazenadas são as mesmas do primeiro holograma.

Os padrões que formam a imagem que sai do segundo holograma terão maior reforço quanto melhor for sua correlação com a imagem de entrada. A lente 4 e o espelho 3 projetam essas múltiplas correlações na superfície traseira do "threshold device".

O "threshold device" é a chave para o funcionamento desse sistema. A sua superfície frontal reflete mais intensamente o padrão mais brilhante que está na sua suferfície traseira, na qual estão projetadas as várias correlações entre a imagem de entrada e as diversas imagens armazenadas.

O padrão mais brilhante será aquele resultante da correlação da imagem de entrada com a imagem armazenada à qual ela mais se assemelha. A imagem reforçada e refletida é projetada no "beam splitter" onde ela torna a entrar no loop, podendo ser reforçada nos próximos loops.

Eventualmente o sistema converge para a imagem armazenada que mais se assemelha à imagem de entrada. Nesse ponto a imagem de entrada pode ser removida que o padrão armazenado continuará circulando no sistema até que ele seja reinicializado.

Apesar de seu grande potencial para correlacionar imagens, a qualidade das imagens obtidas em sistemas existentes é fraca, sua complexidade e custo são elevados, além de serem grandes e difíceis de alinhar. O seu grande potencial em aplicações militares e industriais deverão motivar melhorias nesses sistemas.

7.3 - Hologramas Volumosos

Alguns cristais podem curvar por reflexão um feixe de luz incidente. O ângulo dessa curvatura e a intensidade da luz refletida podem ser alterados por um laser.

Se neurônios são designados para receber e transmitir luz, esses cristais podem ser utilizados para interconectar grandes redes. A capacidade de interconexão nesses sistemas foi estimada entre 10**8 e 10**10 interconexões por cm3 de cristal.

A intensidade e direção de um feixe luminoso incidente em um cristal são determinados por um holograma gravado no interior desse cristal por um laser.

Em redes neuronais esses cristais podem ser utilizados para fazer as interconexões entre os neurônios e para armazenar a matriz de pesos sinápticos. Esses cristais seriam gerados durante a fase de treinamento, e poderiam ser alterados por algum algoritmo de aprendizado.

8 - CONCLUSÕES

Foram apresentadas cinco maneiras diferentes de se implementar uma rede neuronal: uma através de simulação em computadores paralelos; duas através do desenvolvimento de hardware específico, uma analógica e outra digital; e duas através de sistemas óticos, uma com multiplicadores vetor-matriciais e outra com correlatores holográficos.

Entre essas formas de implementar, a simulação em computadores paralelos disponíveis é a mais utilizada, além de ser aquela que propicia resultados em tempos menores. Um grande problema a enfrentar, nesse caso, é a utilização de toda, ou grande parte, das potencialidades da arquitetura onde é feita a simulação. Casos como a implementação no computador AAP-2 (capítulo 3) nem sempre têm tão grande sucesso em "encaixar", de forma tão perfeita, as facilidades oferecidas pelo computador com as características das redes neuronais.

O desenvolvimento de hardware especializado tende a apresentar melhores resultados do que as simulações, principalmente para redes com algum significado prático (milhares de neurônios), onde o tempo de simulação se torna inviável.

Neuro-computadores analógicos, como o descrito no capítulo 4 e do qual ainda não existem dados práticos, têm a vantagem de serem mais rápidos e implementar um neurônio com menos transistores do que em circuitos digitais. Uma desvantagem é sua maior sensibilidade a ruídos elétricos, o que se torna uma limitação para a fabricação de redes em apenas um chip, como na tecnologia WSI. Outra desvantagem é a menor flexibilidade em se alterar a arquitetura da rede, mas que pode ser contornada com a utilização de controles digitais, como é o caso da rede apresentada. Para o processamento de dados do mundo real, como voz e movimento de robôs, as redes analógicas o fazem de forma mais eficiente, pois a inclusão de constantes de tempo torna as equações dos neurônio mais complexas.

A maior parte dos neuro-computadores digitais, como o do capítulo 5, tentam implementar toda uma rede neuronal, com centenas de neurônios, em apenas um chip. Apesar das dificuldades térmicas da tecnologia WSI, ela tem sido usada em alguns projetos digitais. Algumas das desvantagens dos sistemas digitais em relação aos analógicos podem ser resolvidas com a aplicação de novas técnicas de projeto, como a multiplexação de sinapses.

Redes neuronais óticas propiciam grandes vantagens em velocidade e densidade de interconexão em relação às demais implementações. Suas maiores aplicações são o reconhecimento de padrões e o processamento de imagens. As limitações atuais em dispositivos eletro-óticos criaram problemas sérios que devem ser solucionados antes que as redes neuronais óticas possam ser amplamente utilizadas. Considerando-se a grande quantidade de pesquisas que

estão sendo feitas na área, e o grande interesse militar, espera-se um rápido progresso nessa área.

9 - BIBLIOGRAFIA

1 - **Takumi Watanabe et al.** "Neural Network Simulation on a Massively Parallel Cellular Array Processor: AAP-2". International Joint Conference on Neural Networks, 1989, Washington, vol. II, p.155-161.

2 - **Paul Mueller et al.** "A General Purpose Analog Neural Computer". International Joint Conference on Neural Networks, 1989, Washington, vol. II, p.177-182.

3 - **Moritoshi Yasunaga et all.** "A Wafer Scale Integration Neural Network Utilizing Completely Digital Circuits". International Joint Conference on Neural Networks, 1989, Washington, vol. II, p.213-217.

4 - **Philip D. Wasserman.** "Neural Computing - Theory and Practic". ANZA Research Inc, Van Nostrand Reinhold, 1989, New York, 230p.

5 - **Luis A. V. Carvalho et al.** "Redes Neuronais Artificiais: A Volta do Cérebro Eletrônico?". Revista Ciência Hoje, vol. 12, nº 70, jan/fev. 1991, p.12-21.