

DCA

Universidade Estadual de Campinas
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE COMPUTAÇÃO E AUTOMAÇÃO



SENSOR: Um Sistema Sensorial Inteligente Distribuído

Este exemplar corresponde à redação final da tese
defendida por Adriane Beatriz de Souza
Serapião e aprovada pela Comissão
 julgadora em 29 / 02 / 96.

Orientador

Por: Adriane Beatriz de Souza Serapião
Orientador: Prof^o Dr. Armando Freitas da Rocha

Dissertação submetida à Faculdade de Engenharia Elétrica da Universidade Estadual de Campinas, para preenchimento dos pré-requisitos parciais para obtenção do Título de Mestre em Engenharia Elétrica.

29 de fevereiro de 1996



CIDADE: BC
CHAMADA: T/UNICAMP
SE 65s
27445
667/96
R\$ 11,00
25/04/96
CNPJ

CM-00086979-1

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

Se65s Serapião, Adriane Beatriz de Souza
Sensor: um sistema sensorial inteligente distribuído /
Adriane Beatriz de Souza Serapião.--Campinas, SP: [s.n.],
1996.

Orientador: Armando Freitas da Rocha.
Dissertação (mestrado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica.

1. Redes neurais (Computação). 2. Reconhecimento de
padrões. 3. Visão por computador. 4. Linguagens
formais. 5. Inteligência artificial. I. Rocha, Armando
Freitas da. II. Universidade Estadual de Campinas.
Faculdade de Engenharia Elétrica. III. Título.

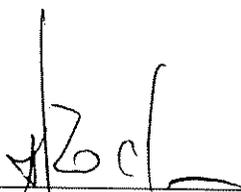
Tese de Mestrado

Título da tese: **SENSOR: Um Sistema Sensorial Inteligente Distribuído**

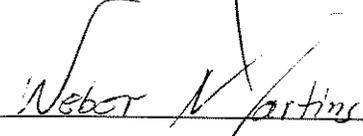
Autora: **Adriane Beatriz de Souza Serapião**

Orientador: **Prof^o Dr. Armando Freitas da Rocha**

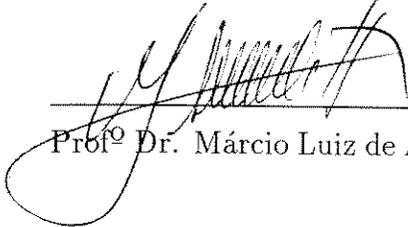
Tese defendida e aprovada, em 29 de fevereiro de 1996, pela banca examinadora constituída pelos professores:



Prof^o Dr. Armando Freitas da Rocha (Presidente)



Prof^o Dr. Weber Martins — EEE/UFG



Prof^o Dr. Márcio Luiz de Andrade Netto — FEE/UNICAMP

Aos meus pais, Armando e Regina,
com amor e carinho.

Agradecimentos

A Deus, por ter me dado condições de saúde e energia para lutar e procurar com tenacidade, vencer todas as etapas de minha vida, por pequenas que possam parecer.

Ao Prof^o Armando Freitas da Rocha, pela orientação.

Ao amigo Rodrigo Pinto Lemos, pela acolhida na minha chegada à Campinas.

Aos amigos Rosângela de Paula Amorim, Tomás A. C. Badan e Reinaldo Martinez Palhares, pela amizade fortalecida com a convivência.

Aos amigos Sérgio L. C. de Oliveira e Marcelo Stehling de Castro, pelo crescimento proporcionado com a descoberta destas novas amizades.

Aos funcionários do DEP/FEM, pela solicitude com que sempre me atenderam.

Ao amigo Marcos Paulo F. Rebello, o Caco, pelo auxílio e sugestões no desenvolvimento computacional deste trabalho.

À Prof^a Maria Salette Mayer de Aquino Giuliano, do CEL-UNICAMP, pelas oportunidades e pelas lições diárias de que a vida do profissional acadêmico não é feita apenas de livros e ciência, mas também de humildade e humanidade.

Aos meus familiares, e aos amigos de Goiânia, por entenderem minha ausência e apoiarem-me em todos os sentidos, em todos os momentos.

Aos meus grandes amigos, Andréa e Ivan Rizzo Guilherme, pelo carinho e pela força nas horas difíceis, e pela alegria compartilhada com os sucessos. E em especial, ao Ivan, pelas discussões e grande contribuição neste trabalho. São pessoas para as quais qualquer palavra é insuficiente para demonstrar o meu apreço, consideração e agradecimento.

E por fim, à CAPES, por financiar parte desta pesquisa.

“Sou um idealista. Não sei para onde
estou indo. Mas estou a caminho.”

Carl Sandburg

Resumo

O objetivo deste trabalho é utilizar um sistema distribuído, chamado **SENSOR**, para simular o sistema visual natural e verificar a consistência de algumas hipóteses sobre sua fisiologia. Visto que o processamento da informação visual ocorre em estágios distintos que requerem a especialização e a distribuição de células no cérebro, usamos esta analogia para criar em um sistema computacional distribuído, agentes capazes de realizar as mesmas funções que seus análogos biológicos. Para tanto, estes agentes foram modelados à partir do neurônio proposto por Rocha [Roc92], que possui capacidades computacionais numéricas e simbólicas, e cujo processamento é suportado pela teoria de linguagens formais nebulosas. Utilizou-se a abordagem sintática como ferramenta básica para o reconhecimento de padrões e análise de cenas, e para avaliar a potencialidade do sistema **SENSOR**. Propôs-se aqui um algoritmo para o reconhecimento e aprendizagem de padrões, chamado *algoritmo de aprendizagem por moldes*. Nesta abordagem, os padrões são contornos de imagem extraídos pelo sistema visual artificial que obedecem uma gramática nebulosa definida e cenas são compostas por objetos que partilham relações suportadas por gramáticas nebulosas aprendidas ou definidas *ad hoc*.

Abstract

This work describes an artificial distributed intelligent system, called **SENSOR**, aimed to simulate the natural visual system and to verify the consistency of some hypothesis in the literature about its physiology. The processing of the visual information occurs in different levels, requiring the specialization and distribution of the neuron. This paradigm was used to create the artificial system, where agents able to perform the same functions of their biological analogous. These agents were modeled using the formal neuron supported by fuzzy formal language theory. The syntactic approach in the theory of artificial visual systems was chosen as the basic tool to handle pattern recognition and scene analysis. The patterns are extracted from the image as contours, complying a defined fuzzy grammar and scenes are composed by objects, sharing relations supported by learned fuzzy grammars or *ad hoc*.

Conteúdo

1	Introdução	1
2	Linguagens Formais Nebulosas	5
2.1	Introdução	5
2.2	Gramáticas	6
2.3	Linguagens Nebulosas	7
2.4	Ambigüidade	9
2.5	Semântica da Linguagem	13
2.6	Dobra de Cadeias	14
3	Sistemas Genéticos	17
3.1	Introdução	17
3.2	Mecanismos Moleculares de Controle Celular	18
3.3	O Modelo Formal de uma Célula	19
3.4	Linguagens Químicas	22
3.5	Processamento Distribuído	30
3.6	Processamento Delimitado	33

4	Neurônio: a unidade fundamental	35
4.1	Introdução	35
4.2	O Neurônio Biológico	36
4.2.1	A Membrana Celular	36
4.2.2	Estrutura e Fisiologia do Neurônio	38
4.2.3	O Processamento Elétrico das Sinapses	39
4.2.4	O Processamento Químico das Sinapses	40
4.2.5	A Codificação Sináptica	42
4.2.6	Aprendizagem	42
4.3	O Neurônio Artificial	43
4.3.1	O Modelo Formal do Neurônio	45
4.3.2	Redes Neurais Artificiais	46
5	O Sistema Sensorial Inteligente Distribuído	47
5.1	Introdução	47
5.2	O Sistema Visual Humano	47
5.3	O Sistema Visual Computacional	55
5.3.1	A Abordagem Sintática	56
5.3.2	Modelo Clássico de um Sistema de Reconhecimento Sintático de Padrões	58
5.4	O Sistema SENSOR	60
6	O Reconhecimento Sintático de Padrões	65

6.1	Introdução	65
6.2	Modelo usado por SENSOR para o Reconhecimento Sintático de Padrões .	65
6.3	Pré-processamento	67
6.3.1	Segmentação de Imagens	74
6.4	Representação do Padrão	76
6.5	Análise de Sintaxe	81
6.5.1	Algoritmo de Acoplamento Nebuloso	81
6.5.2	Algoritmo de Aprendizagem	82
<hr/>		
7	Resultados	85
7.1	Introdução	85
7.2	Extração de Características Básicas	85
7.3	Aprendizagem	88
7.4	Reconhecimento do Padrão	90
7.5	Análise de Cenas	92
8	Conclusão	99
	Bibliografia	103

Lista de Figuras

2.1	Exemplo de duas partições nebulosas em um mesmo universo de discurso. . .	14
4.1	Modelo do mosaico fluido.	36
4.2	Estrutura do neurônio e anatomia fisiológica da sinapse.	38
5.1	Corte horizontal pelo olho (esquema).	48
5.2	Organização neural da retina.	49
5.3	Mapa citoarquitetônico de Brodmann.	51
5.4	Conexões entre diferentes áreas do córtex visual.	52
5.5	Conexões no córtex visual primário.	53
5.6	Exemplo de primitivas de uma cena.	56
5.7	Árvores de derivação.	57
5.8	Sistema para o reconhecimento sintático de padrões.	58
5.9	Exemplo de uma rede usada pelo SENSOR.	63
6.1	Rede para o reconhecimento de padrões pelo sistema SENSOR.	66
6.2	Rede neural com agentes distribuídos para visão.	68
6.3	Função do agente Retina.	70

6.4	Função de codificação dos graus de luminância.	71
6.5	Função dos Agentes Linha Horizontal e Linha Vertical.	71
6.6	Função do Agente Borda.	72
6.7	Focos dentro do campo visual.	72
6.8	Atuação do agente Fóvea.	73
6.9	Áreas de cálculo das densidades da fóvea e blob.	73
6.10	Verificação da continuidade de um segmento.	74
6.11	Segmentação de uma imagem.	75
6.12	Agente primário “Direção”.	76
6.13	Direções possíveis.	78
6.14	Agente “Pontos Significativos”.	80
7.1	Um exemplo de extração de características básicas.	89
7.2	Alguns exemplos de padrões de treinamento	90
7.3	Exemplo de uma cena representativa.	92
7.4	Exemplos de uma árvore de derivação.	94
7.5	Possíveis localizações entre dois elementos de uma cena.	95
7.6	Exemplo de uma cena não-representativa.	96
7.7	Exemplo de uma cena de conflito.	97
7.8	Exemplos de uma árvore de derivação mais complexa.	98
7.9	Exemplo de uma cena permitida.	98

Lista de Tabelas

7.1 Reconhecimento do padrão carro.	91
---	----

Capítulo 1

Introdução

Recentemente, várias publicações [Adl94], [McSh95] vêm surgindo na comunidade científica internacional, buscando inspiração nos sistemas biológicos para modelar sistemas artificiais. A finalidade é ajudar a compreender melhor como tais sistemas biológicos funcionam e para auxiliar no desenvolvimento de sistemas artificiais baseados no comportamento de seus análogos biológicos.

Seguindo esta linha, o objetivo deste trabalho é construir um sistema sensorial distribuído computacional, chamado **SENSOR**, para modelar o sistema visual, a fim de averiguar a consistência das informações que são conhecidas à respeito deste sistema biológico, já que existe uma grande quantidade de dados publicados sobre ele [Sin95], [Mau95]. Esta implementação permitirá também a avaliação da potencialidade do sistema proposto.

O sistema visual realiza o seu processamento através da integração das propriedades e das respostas de vários neurônios funcionalmente especializados, em vários níveis corticais. A retina recebe a imagem visual, envia-a para o núcleo geniculado lateral (NGL), este retransmite-a para o córtex visual primário e áreas visuais secundárias ou de associação.

À nível de retina e NGL, os neurônios extraem as características mais primitivas da imagem, tais como composição RGB, contrastes, etc., enquanto as células corticais se encarregam de analisar relações entre estas características, gerando descrições complexas, tais como linhas, contornos, etc. Outras áreas do córtex visual secundário utilizam estas informações em análises mais complexas da imagem, como reconhecimento de formas, características de movimento, etc.

A unidade de processamento no córtex é a coluna cortical, definida como um conjunto de neurônios conectados de acordo com um padrão básico, realizando um processamento específico, como por exemplo, o reconhecimento da existência de linhas em determinadas direções, a identificação de sentido de movimento e outros. Na análise de uma imagem, inúmeras colunas em uma mesma área (como no córtex visual primário) operam um processamento paralelo, enquanto que uma hierarquia organiza as colunas das diversas áreas em sistemas funcionais complexos. O processamento visual no cérebro é, portanto, um processamento concorrente, em que agentes especializados são distribuídos espacialmente em grupos que estão organizados serialmente por uma hierarquia definida. Do ponto de vista computacional, o sistema visual é um sistema distribuído, porém hierarquizado, onde a hierarquia não só possibilita um controle dos conflitos entre agentes, como suporta um processamento recursivo, no qual cenas complexas são definidas e identificadas como relações estáveis entre objetos, e estes são identificados por relações estáveis entre características mais elementares da imagem.

O sistema **SENSOR** é dito distribuído pois utiliza um conjunto de agentes especializados em realizar tarefas específicas. Através da ação coordenada de conjunto de agentes, inteligentes ou não, capazes de resolver partes deste problema, tal sistema se habilita a encontrar a solução geral de um problema, tal como no sistema visual.

Os agentes são compostos por um único ou por conjuntos de neurônios artificiais, modelados segundo proposto por Rocha [Roc92]. Estes neurônios possuem uma capacidade computacional tanto simbólica quanto numérica, sendo implementadas segundo as transações sinápticas químicas e elétricas do neurônio biológico.

O neurônio pode ser visto, então, como um agente que troca mensagens químicas com outros neurônios para tentar resolver um problema e usa sinais elétricos apenas para acelerar esta troca de mensagens. Os neurônios possuem transmissores que são armazenados em vesículas nos terminais pré-sinápticos, cuja liberação ocorre com a chegada de trens de pulsos elétricos. Ao serem liberados na fenda sináptica, os transmissores se acoplam aos receptores pós-sinápticos, podendo alterar a condutância iônica da membrana ou ativar segundos mensageiros que, por sua vez, podem modificar o metabolismo celular, atuando em locais específicos da célula ou para controlar a leitura dos gens a nível celular. O controle gênico permite a diferenciação dos neurônios para diferentes atividades através do fornecimento de diferentes transmissores, receptores, etc., envolvidos em diferentes processamentos bioquímicos. Em outras palavras, este mecanismo permite que os neurônios possam ser funcionalmente especializados em diferentes tarefas.

Para suportar o formalismo do processamento químico interno e das transações sinápticas entre neurônios, propomos o uso da teoria de linguagens formais nebulosas, pois parece ser uma ferramenta mais adequada para se trabalhar com processamentos simbólicos.

Aproveitando a implementação do sistema visual e da teoria de linguagens formais nebulosas, aplicamos o **SENSOR** ao reconhecimento sintático de padrões, pois dentre as abordagens computacionais existentes para a solução deste problema, a estatística e a sintática, esta última é suportada pelo uso de uma linguagem formal. Além do mais, as etapas de processamento do modelo de reconhecimento sintático podem ser sub-divididas e distribuídas entre diversos agentes.

Como uma extensão deste modelo de reconhecimento de padrões, o problema de análise de cena pode ser focado e tratado com os mesmos procedimentos utilizados no problema de reconhecimento de padrões, com um nível de complexidade um pouco maior.

O capítulo 2 fornece um embasamento teórico sobre os principais conceitos de “linguagens formais nebulosas”. Diversos aspectos desta teoria são abordados, tais como gramáticas nebulosas, semântica, ambigüidade e dobras de cadeias (que é a introdução de uma não linearidade no tratamento de cadeias), voltados para o suporte da teoria de computação molecular.

O capítulo 3 mostra como a teoria de linguagens formais nebulosas pode ser usada para suportar a modelagem de uma célula e do controle celular.

No capítulo 4, que é dividido em duas partes, são apresentados, primeiramente, alguns conceitos biológicos sobre o neurônio, de forma resumida, justificando e abordando pontos relevantes para o desenvolvimento de um modelo artificial de neurônio. Tal modelo é desenvolvido na segunda parte do capítulo, tendo como suporte a teoria de linguagens formais nebulosas.

O capítulo 5 introduz conceitos importantes relativos ao sistema visual humano que são usados pelo **SENSOR** na implementação do sistema visual artificial. Depois, apresenta o modelo clássico do reconhecimento sintático de padrões (RSP), que servirá também como inspiração para estruturar o desenvolvimento do modelo utilizado pelo **SENSOR**. Por último, descreve a estrutura, organização e premissas básicas do sistema **SENSOR**.

O modelo do **SENSOR** para o reconhecimento sintático de padrões é desenvolvido no capítulo 6. Em seguida, para as etapas do processo de RSP (pré-processamento,

representação do padrão e análise de sintaxe) foram desenvolvidos os modelos de agentes neurais que constituem-nas. O sistema visual artificial foi usado como um mecanismo para demonstrar como a integração de vários agentes pode solucionar um problema de busca de contornos em uma imagem e, assim, funcionar como a etapa de *pré-processamento*, sendo, inclusive, utilizado para a segmentação de imagens. Para a *representação do padrão* mostrou-se os agentes que compõem esta etapa e o processamento químico que eles realizam a nível celular. Na *análise de sintaxe*, introduziu-se um novo algoritmo para a aprendizagem e o reconhecimento de padrões, baseado em linguagens formais nebulosas.

O capítulo 7 traz uma amostra dos resultados obtidos pelo modelo de reconhecimento sintático de padrões e pela análise de cenas, suportados pelo **SENSOR**. Foi demonstrada, ainda, a utilização de gramáticas nebulosas para a representação dos padrões, e de como foram utilizadas as informações simbólicas e numéricas fornecidas pelos agentes em cada etapa desta aplicação.

Por fim, o capítulo 8 apresenta as conclusões gerais sobre o trabalho e traz sugestões para trabalhos a serem desenvolvidos em etapas futuras.

Capítulo 2

Linguagens Formais Nebulosas

2.1 Introdução

Tradicionalmente, as Ciências Exatas (Física, Química, Engenharia, etc.) colocam a compreensão de um fenômeno como a habilidade de analisá-lo em termos quantitativos, governados por equações, integrais ou derivadas. Todavia, técnicas quantitativas convencionais de análise de sistemas são incapazes de descrever sistemas biológicos e humanísticos, tais como economia, sociologia e psicologia, e muitas vezes são insuficientes para descrever certos fenômenos estudados pelas Ciências Exatas devido à sua complexidade, ou à sua semelhança com o comportamento de sistemas biológicos ou humanísticos. Como uma alternativa, estes fenômenos ou sistemas podem ser descritos por palavras ou sentenças no lugar de números, ou sejam, podem ter uma representação lingüística (ou simbólica) em substituição aos métodos clássicos, mantendo sua análise em termos quantitativos através de quantificadores lingüísticos e descrevendo seu comportamento em termos de fatos e regras. Assim, tais sistemas passam a ser regidos por classes nebulosas, conceitos vagos e imprecisos, ao invés de equações matemáticas.

A nebulosidade também desempenha um importante papel na cognição humana, pois muitas classes (e conceitos) encontradas no mundo real são nebulosas, pois os elementos-chaves do pensamento humano não são números, mas conjuntos nebulosos, nos quais classes de objetos ou conceitos têm uma transição gradual de sua pertinência a um conjunto para a não-pertinência (como uma temperatura passar de baixa para alta, ou algum lugar estar perto ou longe). Assim, tais conceitos são representados por uma linguagem natural,

cujo significado torna-se genericamente compreensível.

As linguagens formais foram justamente criadas para tentar-se compreender melhor as propriedades das linguagens naturais, sendo depois aplicadas na investigação de sistemas computacionais teóricos e em linguagens de programação. O método mais poderoso encontrado para descrever uma linguagem foi através da formalização de uma gramática estruturada por frases e regidas por um conjunto de regras de reescrita. Assim, um sistema pode ser descrito por sentenças compostas através de uma gramática, de uma dada linguagem, que representam características próprias de tal sistema. Todavia, a linguagem carrega, às vezes, uma certa ambigüidade na descrição de algum fenômeno. Por isso, desenvolveu-se uma teoria de linguagens formais nebulosas [DuPr80], buscando dar maior flexibilidade no tratamento e compreensão da linguagem.

2.2 Gramáticas

Uma gramática pode ser vista como um conjunto de regras que geram os elementos de um conjunto. Formalmente, pode ser definida como a estrutura

$$G = \{V_s, V_n, V_t, \mathcal{P}, \eta\} \quad (2.1)$$

onde:

V_s é o conjunto de símbolos iniciais;

V_n é o conjunto de símbolos não-terminais;

V_t é o conjunto de símbolos terminais;

η é o elemento nulo (vazio) e

\mathcal{P} é o conjunto de regras de reescrita definidas como:

$$p : \alpha s_i \beta \rightarrow \alpha s_j \beta \quad (2.2)$$

$$\alpha, \beta, s_j \in V_s \cup V_n \cup V_t \cup \eta \text{ e } s_i \in V_s \cup V_n$$

Uma regra $p \in \mathcal{P}$ reescreve a cadeia s_i como a cadeia s_j . s_i é definida como um cadeia de símbolos de $V_s \cup V_n$ e s_j é definida como uma cadeia de símbolos de $V_s \cup V_n \cup V_t \cup \eta$. Por simplicidade,

$$V^+ = V_s \cup V_n \text{ e } V^* = V_s \cup V_n \cup V_t \quad (2.3)$$

Segundo a classificação de Chomsky [Cho57], distinguem-se quatro tipos de gramáticas:

Gramática Regular (GR): permite produções do tipo

$$p : s_i \rightarrow \alpha s_j \quad (2.4)$$

$$\alpha \in V_t \cup \eta, \text{ e } s_j \in V^+$$

Gramática Livre de Contexto (GLC): as produções possíveis são

$$p : s_i \rightarrow \alpha s_j \quad (2.5)$$

$$\alpha, s_j \in V^* \cup \eta$$

Gramática Sensível a Contexto (GSC): as produções permitidas estão na forma

$$p : \alpha s_i \beta \rightarrow \alpha s_j \beta \quad (2.6)$$

$$\alpha, \beta, s_j \in V^* \cup \eta$$

onde, o comprimento de $\alpha s_i \beta$ não é maior que o comprimento de $\alpha s_j \beta$.

Gramática Irrestrita (GI): sem restrições ao comprimento da cadeia.

Portanto, temos que: $GR \supset GLC \supset GSC \supset GI$.

A cadeia de derivação $d(s_o, s_j)$ da cadeia $s_j \in V^*$ de G é o conjunto ordenado das produções necessárias para transformar o símbolo inicial $s_o \in V_s$ em s_j . Em outras palavras,

$$d(s_o, s_j) = \alpha s_o \beta \rightarrow \alpha s_1 \beta \cdots \alpha s_i \beta \rightarrow \alpha s_j \beta \quad (2.7)$$

2.3 Linguagens Nebulosas

Uma linguagem formal L é definida como um sub-conjunto de cadeias geradas pela gramática de suporte G . As cadeias geradas por G e aceitas como pertencentes à L são chamadas *fórmulas bem formadas (fbf)* de G segundo L . Uma cadeia s_j produzida por G é uma *fbf* se ela pertence a V_t . Ou seja, as cadeias s_j aceitas pela linguagem $L(G)$ e suportadas por G são aquelas *fbf*(s_o, s_j) obtidas como:

$$fbf(s_o, s_j) = d(s_o, s_j) = s_o \rightarrow \alpha s_1 \beta \cdots \alpha s_i \beta \rightarrow s_j, \quad s_j \in V_t \quad (2.8)$$

Deste modo, a sentença $\alpha s_o \beta$ é aceita como uma sentença de L , se existir pelo menos uma $fbf(\alpha s_o \beta, \alpha s_j \beta)$ para reescrevê-la em $\alpha s_j \beta$, pertencente a V_t .

O processamento de qualquer sentença $\alpha s_o \beta$ de L é um conjunto ordenado seqüencialmente de operações de reescrita, com cada uma delas envolvendo os seguintes passos:

1. acoplamento: o lado esquerdo de uma regra de reescrita é comparado aos símbolos da cadeia s_j a ser processada. Em seguida, se a comparação for bem sucedida, tem-se
2. reescrita: a sub-cadeia que combina com s_j é, então, substituída pelo lado direito da regra de reescrita aceita. Finalmente,
3. aceitação: a pertinência da cadeia final s_k produzida pela cadeia de derivação $d(s_o, s_k)$ em V_t é calculada no intervalo fechado $[0, 1]$.

De acordo com esta abordagem, podemos definir:

- a) o grau de similaridade $\mu(s_i, s_j)$ entre duas cadeias s_i e s_j é uma medida em $(V^*)^l$, onde l é o comprimento da maior das cadeias, tal que

$$\mu : (V^*)^l \rightarrow [0, 1] \quad (2.9)$$

$$\mu(s_i, s_j) \rightarrow \begin{cases} 1, & \text{se } s_i \text{ tende a ser igual a } s_j \\ 0, & \text{caso contrário} \end{cases}$$

- b) o grau de acoplamento $\mu(s_k, V_t)$ de s_k como uma fbf de L é calculado como o grau máximo de similaridade $\mu(s_k, s_t)$ de s_k com as cadeias $s_t \in V_t$. Ou seja,

$$\mu(s_k, V_t) = \vee_{V_t} \mu(s_k, s_t) \quad (2.10)$$

Estas duas condições definem L como uma linguagem nebulosa.

Se

- a) $\mu(s_i, s_j)$ em (2.9) se comporta como uma função booleana, assumindo o valor 1 ou 0, isto é

$$\mu : (V^*)^l \rightarrow (0, 1) \quad (2.11)$$

$$\mu(s_i, s_j) = \begin{cases} 1, & \text{se } s_i \text{ é igual a } s_j \\ 0, & \text{caso contrário} \end{cases}$$

e

b) $\mu(s_k, V_t)$ em (2.10) assume um comportamento de uma função identidade, ou seja

$$\mu(s_k, V_t) = \begin{cases} 1, & \text{se } s_k \in V_t \\ 0, & \text{caso contrário} \end{cases} \quad (2.12)$$

então, L é uma linguagem clássica.

2.4 Ambigüidade

As linguagens nebulosas exibem propriedades marcantes em relação às linguagens clássicas, pois, dado $s_i \in V^*$, podem existir várias $s_k \in V^*$ para as quais $\mu(s_i, s_k) > 0$. Isto significa que muitas cadeias de derivação podem reescrever s_i em diferentes $s_k \in V^*$. Visto que a ambigüidade de uma sentença s_o de L depende de quantas cadeias de derivação $d(s_o, s_t)$ possuem $s_t \in V_t$, as linguagens nebulosas são, desta forma, naturalmente ambíguas. O processamento de linguagens nebulosas requer também, em geral, operações concorrentes utilizando um processo distribuído espacialmente.

Esta ambigüidade pode ser mantida em níveis aceitáveis por meio das restrições:

1. *comprimento da cadeia terminal*: o comprimento $l(s_t)$ da cadeia $s_t \in V_t$ pode ser diminuído gradualmente, a fim de reduzir o poder combinatorial das linguagens nebulosas, através da redução do número de passos de derivação de qualquer *fbf* de L . Em outras palavras,

$$l(s_t) < \alpha \quad (2.13)$$

o valor atual α é um dos parâmetros determinantes do grau $a(L)$ de ambigüidade de L ;

2. *cardinalidade de V_t* : é um outro importante parâmetro para a determinação do valor atual de $a(L)$, por causa da possibilidade de qualquer cadeia s_i vir a ser uma *fbf* de L , dependendo da similaridade $\mu(s_i, s_j)$ de s_i , para qualquer $s_t \in V_t$. Assim, o número $A(L)$ de possíveis *fbf* de L é uma função da cardinalidade $C(V_t)$

$$A(L) = f(C(V_t)) \quad (2.14)$$

e f não pode ser a função identidade se uma fbf de L pode ser definida, mesmo que $\mu(s_i, s_j) < 0, s_t \in V_t$. $A(L) = C(V_t)$ somente no caso de $\mu(s_i, s_j) = 1, s_t \in V_t$ para toda fbf de L ;

3. *número total de cópias $A(s_i)$ disponíveis de $s_i \in V^*$* : pelo menos uma cópia de $s_i \in V^*$ tem que estar disponível para disparar cada possível cadeia de derivação $d(s_i, s_j)$ suportada por uma regra de reescrita $\alpha s_k \beta \rightarrow \alpha s_j \beta$, para que

$$\mu(s_i, s_k) > 0$$

4. *número total de cópias $A(s_k)$ de $s_k \in V^*$* : pelo menos uma cópia da cadeia que inclui $\alpha s_k \beta$ tem que estar disponível para permitir que a regra $\alpha s_k \beta \rightarrow \alpha s_j \beta, \mu(s_i, s_k) > 0$ seja usada;

5. *número atual $a(\alpha s_i \beta)$ de cópias ativadas da sentença $\alpha s_i \beta$ a ser processada*: pelo menos uma cópia da cadeia $\alpha s_i \beta$ tem que ser selecionada (ativada) para poder ser reescrita.

Estas restrições implicam que não apenas a qualidade é levada em consideração no processamento de uma linguagem nebulosa, mas que as suas quantidades também desempenham um importante papel na determinação da reescrita das sentenças nebulosas. Isto é um contraste bem claro em comparação com as linguagens clássicas, nas quais nenhuma atenção é dada ao processo de cópia de uma cadeia.

6. *Prioridade de comparação*: sempre que não houver cópias suficientes de $s_i \in V^*$ para disparar todas as suas cadeias de derivação $d(s_i, s_j)$, a prioridade deve ser dada para aquelas regras $\alpha s_k \beta \rightarrow \alpha s_j \beta$ que possuem os maiores valores de $\mu(s_i, s_k)$. A prioridade também deve ser dada para aquelas regras suportadas pelas $s_k \in V^*$ que têm o maior número $A(s_k)$ de cópias disponíveis.

Deste modo, a possibilidade $\rho(d(s_i, s_j))$ de se usar a cadeia de derivação $d(s_i, s_j)$, suportada por $\alpha s_k \beta \rightarrow \alpha s_j \beta, \mu(s_i, s_k) > 0, k = 1, \dots, n$, para reescrever s_j deve depender tanto de $A(s_j)$ e $A(s_k)$ quanto de $\mu(s_i, s_k)$, isto é,

$$\rho(d(s_i, s_j)) = \max_{k=1}^n (f(A(s_i), A(s_k), \mu(s_i, s_k))) \quad (2.15)$$

$$f : A(s_i) \times A(s_k) \times \mu(s_i, s_k) \rightarrow [0, 1]$$

O número $a(\alpha s_j \beta)$ de cópias de $s_j \in V^*$ produzidas (ativadas) por $s_i \in V^*$ é dependente de $a(\alpha s_i \beta)$ e de $d(s_i, s_j)$, ou seja,

$$a(\alpha s_j \beta) = \partial(a(\alpha s_i \beta), \rho(d(s_i, s_j))) \quad (2.16)$$

ou

$$a(\alpha s_j \beta) = \phi_{k=1}^n (\partial((a(\alpha s_i \beta), \rho(d(s_k, s_j)))) \quad (2.17)$$

onde, ϕ , ∂ são normas ou co-normas \mathcal{T} [Lee90].

Uma linguagem nebulosa primitiva (LNP) é definida se as únicas restrições usadas para controlar a ambigüidade são derivadas das regras 1 e 2 anteriores. Uma linguagem nebulosa complexa (LNC) é definida sempre que o controle da ambigüidade é suportado por mecanismos derivados de outras regras, como as regras de 3 a 6, acima. Enquanto as linguagens clássicas são principalmente LNPs suportadas por um raciocínio monotônico, as LNCs podem ser usadas para implementar diferentes tipos de lógicas não-monotônicas, tais como:

a) *lógica default*: pois $\mu(s_i, s_j) < \mu(s_i, s_k)$ suporta cadeias de derivação do tipo

$$\begin{aligned} a(s_j) > 0, \quad \mu(s_i, s_j) \ll 1 &\Rightarrow \alpha s_i \beta \rightarrow \alpha s_j \beta, \text{ a não ser que,} \\ a(s_k) > 0, \quad \mu(s_i, s_k) \rightarrow 1 &\Rightarrow \alpha s_i \beta \rightarrow \alpha s_k \beta \end{aligned} \quad (2.18)$$

isto é, dado $\mu(s_i, s_j) < \mu(s_i, s_k)$, s_j reescreve s_i a menos que s_k esteja disponível para reescrever s_i ;

b) *lógica de recursos finitos*: ela pode assumir que o número de cópias disponíveis $A(s_i)$ de $s_i \in V^*$ seja decrementado por uma quantidade proporcional à quantidade $a(\alpha s_i \beta)$ das sentenças copiadas $\alpha s_i \beta$ nas equações (2.16) e (2.17), sempre que s_j reescrever s_i , tal que

$$\alpha s_i \beta \rightarrow \alpha s_j \beta \text{ enquanto } A(s_i) > 0 \quad (2.19)$$

c) *lógica de limiares*: pois pode assumir que a sentença $\alpha s_i \beta$ possa ser reescrita somente se seu número de cópias disponíveis $a(\alpha s_i \beta)$ estiver acima de um mínimo dado θ , isto é,

$$a(\alpha s_i \beta) > \theta \quad (2.20)$$

As lógicas acima podem ser consideradas como casos especiais em uma família muito mais ampla de lógicas, conhecida como:

d) *lógica nebulosa*: porque ela pode assumir que uma sentença $\alpha s_i \beta$ seja reescrita como

$$Q(\rho(d(s_i, s_j))) \Rightarrow \alpha s_i \beta \rightarrow \alpha s_j \beta \vee \dots \wedge \rho(d(s_i, s_k)) \Rightarrow \alpha s_i \beta \rightarrow \alpha s_j \beta \quad (2.21)$$

$$A(\alpha s_j \beta) = f(\rho(d(s_i, s_j)) \dots \rho(d(s_i, s_k))) \quad (2.22)$$

onde, Q é um quantificador lógico e \Rightarrow significa o suporte; \vee e \wedge são respectivamente, normas e co-normas \mathcal{T} ; $\rho(d(s_i, s_j))$ é uma função de $A(s_i)$, $A(s_k)$, e $\mu(s_i, s_k)$; e f pode ser definida como uma função de filtragem. A leitura da equação (2.21) é:

A cadeia $\alpha s_i \beta$ é reescrita por Q (das regras relevantes $\alpha s_k \beta \rightarrow \alpha s_j \beta \vee \dots \wedge \dots$) na cadeia $\alpha s_j \beta$,

onde, $\rho(d(s_i, s_k))$ mede a relevância (ou suporte) da regra $\alpha s_k \beta \rightarrow \alpha s_j \beta$.

As quantidades disponíveis $A(s_i)$, $A(s_k)$, de $s_i, s_k \in V^*$ em $\alpha s_k \beta \rightarrow \alpha s_j \beta$ que reescrevem s_i em s_j em $d(s_o, \dots, s_i, s_k, \dots, s_t)$, são também dependentes do sucesso das regras de reescrita em contribuição para o processamento de uma *fbf* de L , dependendo do valor atual de $\mu(s_t, V_t)$. Deste modo,

- a) *recompensa*: incrementa $A(s_i)$, $A(s_k)$ sempre que $d(s_o, \dots, s_i, s_k, \dots, s_t)$ suceder a ativação de s_t para a qual $\mu(s_t, V_t) \rightarrow 1$. Isto pode ser considerado uma consequência da ativação de uma cópia de regras de produção do tipo

$$\rho s_i \rightarrow s_i, \rho s_k \rightarrow s_k \quad (2.23)$$

no contexto de recompensa ρ .

- b) *punição*: decrementa $A(s_i)$, $A(s_k)$ quando $d(s_o, \dots, s_i, s_k, \dots, s_t)$ ativa s_t para a qual $\mu(s_t, V_t) \rightarrow 0,5$. Isto pode ser considerado uma consequência da ativação de uma cópia de regras de produção do tipo

$$\pi s_i \rightarrow \eta, \pi s_k \rightarrow \eta \quad (2.24)$$

no contexto de punição π .

- c) *reconsideração*: decrementa $\mu(s_i, s_k)$ sempre que $d(s_o, \dots, s_i, s_k, \dots, s_t)$ ativar s_t para a qual $\mu(s_t, V_t) \rightarrow 0$. Isto pode ser visto como uma consequência da ativação de uma cópia de regras de produção do tipo

$$\lambda s_i \rightarrow s'_i, \lambda s_k \rightarrow s'_k, \mu(s_i, s_k) > \mu(s'_i, s'_k) \quad (2.25)$$

no contexto λ , que modifica a estrutura interna das cadeias s_i e s_k para torná-las cadeias menos similares.

d) *reforço*: incrementa $\mu(s_i, s_k)$ quando $d(s_o, \dots, s_i, s_k, \dots, s_t)$ ativa s_t para a qual $\mu(s_t, V_t) \rightarrow 0,5, \mu(s_t, V_t) > 0,5$. Isto pode ser encarado como uma consequência da ativação de uma cópia de regras de produção do tipo

$$vs_i \rightarrow s'_i, vs_k \rightarrow s'_k, \mu(s_i, s_k) < \mu(s'_i, s'_k) \quad (2.26)$$

no contexto v , que muda a estrutura interna das cadeias s_i e s_k para torná-las cadeias mais similares.

Deste modo, tanto a aprendizagem quantitativa (a,b) quanto a qualitativa (c,d) podem ser usadas para reduzir a ambigüidade das linguagens nebulosas, pois reconsideração e reforço implicam na mudança da estrutura dos símbolos, enquanto recompensa e punição estão associadas a mudanças do número de cópias disponíveis dos símbolos. Todos os processos acima implicam na modificação da possibilidade $\rho(d(s_o, \dots, s_i, s_k, \dots, s_t))$ de disparo da cadeia de derivação $d(s_o, \dots, s_i, s_k, \dots, s_t)$, e podem ser usadas para controlar a ambigüidade. Reconsideração e reforço são conceitos formais que podem ser relacionados aos conceitos de mutação e recombinação, enquanto que recompensa e punição podem ser consideradas ferramentas para implementação da seleção natural.

2.5 Semântica da Linguagem

Os conjuntos de símbolos iniciais V_s , terminais V_t e não-terminais V_n representam as informações semânticas que expressam os atributos de classes. Estes atributos podem possuir uma representação qualitativa e uma quantitativa (valor). Assim, uma sentença ou cadeia é um conjunto de símbolos que reúnem os atributos de uma classe.

Na representação qualitativa, símbolos são designados para identificar classes, ou melhor, as classes, ou seus elementos, recebem rótulos que os diferenciam. Já na representação quantitativa, um valor que representa um atributo deve ser transformado em símbolo. Esta operação é feita pelo uso de variáveis lingüísticas.

Uma variável lingüística [Zad73], geralmente, contém um conjunto de termos, onde cada termo está definido no mesmo universo de discurso, a fim de representá-lo como um todo. A quantidade de termos em um conjunto de termos é conhecida como *partição nebulosa*, e determina a *granularidade* do espaço nebuloso (e a quantidade de símbolos).

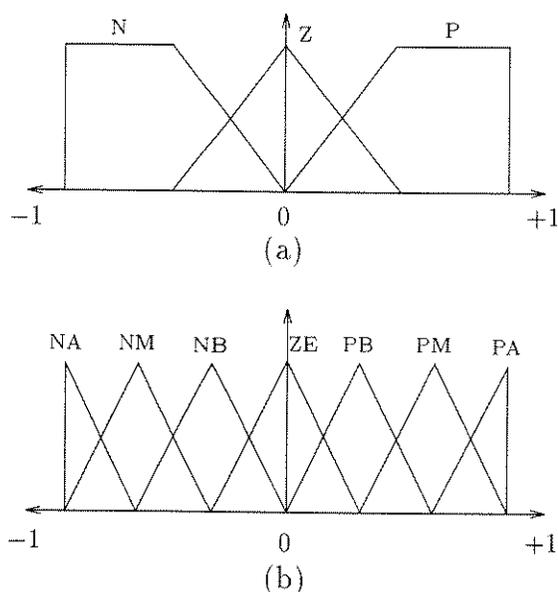


Figura 2.1: Exemplo de duas partições nebulosas em um mesmo universo de discurso.

Por exemplo, na fig. 2.1 temos uma representação de duas partições nebulosas para um mesmo universo de discurso definido em $[-1, +1]$. A primeira partição (a), partição grossa, é composta por três termos: negativo (N), zero (Z) e positivo (P). Na partição fina (b) existem sete termos: negativo alto (NA), negativo médio (NM), negativo baixo (NB), zero (ZE), positivo baixo (PB), positivo médio (PM) e positivo alto (PA). Estas diferentes partições correspondem a diferentes graus de refinamento que se pode dar a uma variável lingüística, que depende da precisão desejada.

2.6 Dobra de Cadeias

Seja considerado o caso onde $s_i, s_j \in V^*$ são cadeias do tipo

$$s_i = \alpha s_p \{ \dots \gamma \dots \}_k s_q \beta, \quad s_j = \alpha s_r \{ \dots \lambda \dots \}_l s_s \beta \quad (2.27)$$

onde $\{ \dots \}_n$ significa uma cadeia qualquer de tamanho n . Um acoplamento não-linear $\mu(s_i, s_j)$ entre $s_i, s_k \in V^*$ é definido no caso

$$\mu(s_i, s_j) = f'(\mu(s_p, s_q), \mu(s_r, s_s), \mu(k/l), \mu(\gamma, \lambda)) \quad (2.28)$$

isto é, o acoplamento entre s_i e s_j é uma função dos acoplamentos individuais $\mu(s_p, s_r)$ e $\mu(s_q, s_s)$, da similaridade do comprimento de $\{ \dots \gamma \dots \}_k$ e de $\{ \dots \lambda \dots \}_l$, e da similaridade $\mu(\gamma, \lambda)$ de γ e de λ .

Neste contexto, as cadeias

$$\{\dots\gamma\dots\}_k \text{ e } \{\dots\lambda\dots\}_l \quad (2.29)$$

são chamadas de dobras $\Gamma(s_i)$ e $\Gamma(s_j)$ de s_i e s_j , respectivamente, pois assume-se que as cadeias $\alpha s_i \beta$, $\alpha s_j \beta$ podem ser dobradas nas cadeias $\{\dots\gamma\dots\}_k$, $\{\dots\lambda\dots\}_l$, que vêm a ser funcionalmente similares às cadeias lineares $\alpha s_p s_q \beta$ e $\alpha s_r s_s \beta$. A possibilidade de dobra $\rho(\Gamma(s_i), \Gamma(s_j))$ é definida como uma função de $\mu(k, l)$ e de $\mu(\gamma, \lambda)$, ou seja,

$$\rho(\Gamma(s_i), \Gamma(s_j)) = f^n(\mu(k, l), \mu(\gamma, \lambda)) \quad (2.30)$$

$$\mu(s_i, s_j) = f(\mu(s_p, s_q), \mu(s_r, s_s) \rho(\Gamma(s_i), \Gamma(s_j))) \quad (2.31)$$

isto é, o acoplamento entre s_i e s_j depende das similaridades $\mu(s_p, s_q)$ e $\mu(s_r, s_s)$ entre s_p e s_q , s_r e s_s e da possibilidade de dobra $\rho(\Gamma(s_i), \Gamma(s_j))$.

Nesta condição:

- a) $\mu(s_p, s_q)$, $\mu(s_r, s_s)$ suportam a *codificação primária* da informação em s_i e s_j ;
- b) $\mu(k/l)$ é responsável pela *codificação secundária* da informação em s_i e s_j e
- c) $\mu(\gamma, \lambda)$ suporta a *codificação terciária* da informação em s_i e s_j .

Uma linguagem L composta por cadeias do tipo descrito em (2.27) é chamada *Linguagem Não-Linear Nebulosa* L_n . A gramática G_n que suporta uma L_n tem uma complexidade maior que a de uma gramática G que suporta uma linguagem linear L , pois inclui um conjunto específico F de regras associadas ao processo de dobra das cadeias de L , ou seja,

$$G_n = \{V_s, V_n, V_t, P, F, \eta\} \quad (2.32)$$

onde, V_s , V_n , V_t , P e η são definidos como em (2.8) e F é definido como um conjunto de regras do tipo:

a) *dobramento*:

$$\alpha s_i \dots \gamma \dots s_j \beta \rightarrow \alpha s_i \{\dots\gamma\dots\}_s s_j \beta \quad (2.33)$$

define $\{\dots\gamma\dots\}_s$ como a dobra de $\alpha s_i \dots \gamma \dots s_j \beta$, isto é, $\Gamma(\alpha s_i \dots \gamma \dots s_j \beta) = \{\dots\gamma\dots\}_s$.

b) *corte*:

$$\alpha s_i \{ \dots \gamma \dots \}_s s_j \beta \rightarrow \alpha s_i s_j \beta, \dots \gamma \dots \quad (2.34)$$

desconcatena $\alpha s_i s_j \beta$ de sua dobra $\{ \dots \gamma \dots \}_s$.

c) *inserção*:

$$\alpha s_i s_j \beta, \dots \gamma \dots \rightarrow \alpha s_i \dots \gamma \dots s_j \beta \quad (2.35)$$

insere a cadeia $\dots \gamma \dots$ dentro da cadeia $\alpha s_i s_j \beta$ em uma possível dobra.

d) *reordenação*:

$$\alpha s_i \dots \gamma \dots s_j s_k \beta \rightarrow \alpha s_i s_j \dots \gamma \dots s_k \beta \quad (2.36)$$

O dobramento, o corte e a inserção podem ser usados para reordenar os símbolos de uma cadeia $\alpha s_i \dots \gamma \dots s_j s_k \beta$.

Capítulo 3

Sistemas Genéticos

3.1 Introdução

Os genes codificam proteínas, os atuadores reais do processo de computação molecular a nível celular, e são compostos por duas sub-cadeias, a sub-cadeia de controle, que contém as especificações das condições de leitura gênica, e a sub-cadeia de código, que contém o código protéico propriamente dito. A produção protéica é ordenada hierarquicamente por reações bioquímicas encadeadas, ou seja, a atividade bioquímica celular é organizada em uma seqüência de eventos ordenados através de redes genéticas, onde estas podem ser ativadas em paralelo. As redes genéticas são caracterizadas pelo conjunto de genes que codificam uma família de proteínas e pelo conjunto de vias ou canais de comunicação que relacionam o local de produção destas proteínas com os sítios de sua atuação. A função desta especificidade do endereçamento bioquímico é permitir que várias redes genéticas possam operar em paralelo no interior do espaço celular, sem sobrecarregar a célula com seus processos.

Todavia, os sistemas genéticos são sistemas complexos, e para qualquer organismo torna-se necessário encontrar um formalismo para representar e analisar estes sistemas, a fim de compreender a sua funcionalidade. Assim, propomos a teoria de linguagens formais nebulosas para codificar e desenvolver uma “linguagem genética” que represente o conhecimento e o processamento inerente a uma rede genética.

3.2 Mecanismos Moleculares de Controle Celular

Basicamente, existem dois métodos diferentes para o controle das atividades bioquímicas da célula: a *regulação genética*, responsável pelo controle das atividades dos próprios genes, e nosso foco de atenção, e a *regulação enzimática*, responsável pelo controle do nível de atividade das enzimas na célula.

O funcionamento celular é controlado pelos genes, através da leitura do DNA, quando estes determinam quais substâncias serão sintetizadas pelas células - que estruturas, quais enzimas, quais compostos químicos.

O DNA é composto por duas cadeias de polinucleotídeos (*fitas de DNA*) enroladas uma em torno da outra, formando uma estrutura denominada *dupla hélice*. Cada cadeia é uma seqüência de nucleotídeos, onde cada nucleotídeo é composto por apenas uma dentre quatro bases diferentes: adenina (A), guanina (G), citosina (C) e timina (T). As fitas de DNA contêm uma seqüência ordenada de bases, de modo que uma fita seja complementar à outra, devido ao pareamento das bases destas duas cadeias, sendo A pareada com T, e C com G. A seqüência destas bases ao longo das cadeias de polinucleotídeos de cada hélice fornece um código que especifica a seqüência dos aminoácidos que codificam uma proteína. Cada três nucleotídeos produz um aminoácido dentre os 20 tipos conhecidos.

O DNA celular localiza-se no núcleo da célula, ao passo que a síntese protéica, em sua maior parte, ocorre no citoplasma. A leitura do código de DNA resulta na síntese do mRNA, seguida da translação da informação para o tRNA, e dele para a síntese protéica no citoplasma pelos ribossomos. As condições para a leitura do gene são também especificadas por meio de tuplas de nucleotídeos. O gene é, desta forma, composto por duas diferentes cadeias:

- a) a *cadeia de código* é composta por seqüências significativas de nucleotídeos chamadas *exons* misturadas com seqüências não significativas chamadas *introns*. Os introns são retirados do mRNA depois da leitura do DNA pela ação de moléculas específicas que são capazes de reconhecer os delimitadores das suas seqüências. As cadeias remanescentes de exons contêm o código da proteína. Por este motivo, a codificação da leitura do DNA é considerada um processo não-linear;
- b) a *cadeia de controle* é composta por várias sub-cadeias:
 - b.1) *promotor (TATA box)*: é uma seqüência de nucleotídeos composta prin-

cipalmente por timina e adenina, que deve ser ativada para habilitar a leitura do código;

b.2) *indutor*: a ativação desta sub-cadeia de nucleotídeos acelera a leitura do código, aumentando o número de cópias disponíveis de seu mRNA associado. e

b.3) *repressor*: a ativação desta sub-cadeia de nucleotídeos reprime a leitura do código, reduzindo o número de cópias disponíveis de seu mRNA associado.

A organização da informação genética nos cromossomos está baseada sobre uma hierarquia complexa, onde alguns genes (chamados genes prematuros) codificam as proteínas promotoras, indutoras ou repressoras que controlam a leitura em outros genes (chamados genes reguladores).

A linguagem genética é muito parecida, em princípio, a uma linguagem escrita que consiste de um conjunto de símbolos formando as letras do alfabeto. As letras estão ordenadas em seqüências específicas para formar palavras e as palavras estão dispostas em seqüências para formar frases. A linguagem genética contém apenas quatro letras, que correspondem às quatro bases A, G, C e T. As palavras são seqüências de três bases que especificam determinados aminoácidos. Estas palavras estão ordenadas numa seqüência linear ao longo das cadeias de DNA, e a seqüência de palavras especificando a estrutura de uma única proteína corresponde a um gene; assim, um gene é equivalente a uma frase.

3.3 O Modelo Formal de uma Célula

De acordo com Rocha [RSR96], os conjuntos de proteínas estão organizados em conjuntos ordenados de interações químicas, que definem cadeias bioquímicas ou caminhos, que são, por sua vez, organizados em *Sistemas Químicos Funcionais (SQF)* envolvidos no metabolismo celular, excitabilidade, reprodução, etc. Alguns sinais químicos atuam como disparadores ou iniciadores destas cadeias bioquímicas, enquanto outros sub-conjuntos de proteínas são ativados em seus passos intermediários ou como produtos do *SQF*. Formalmente, um *SQF* é definido como:

$$SQF = \{P_i, P_n, P_i, C\} \quad (3.1)$$

onde:

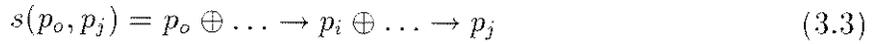
a) P_i é o conjunto de sinais químicos iniciais;

- b) P_n é o conjunto de produtos químicos intermediários;
 c) P_t é o conjunto de produtos químicos finais, e
 d) C é o conjunto de reações químicas do tipo:



denotando que a interação entre os produtos $p_i \dots p_j$ produz (ou ativa) p_k .

Uma cadeia bioquímica $s(p_o, p_j)$ é caracterizada como o conjunto ordenado de reações necessárias para produzir (ou ativar) $p_j \in P_t$, onde $p_o \in P_i$ está disponível, isto é,



Um *Sistema Funcional Biológico* (*SFB*) é definido como uma família de n *SQFs* codificada por um dado conjunto de genes G . Em outras palavras:

$$SFB = \{P_i, P_n, P_t, C, G\} \quad (3.4)$$

onde G é a família de genes que definem os conjuntos P^j de produtos ($P^j = P_i^j \cup P_n^j \cup P_t^j$) de cada *SQF* _{j} pertencente ao sistema. Se G é uma codificação hierárquica da família $P = \{P^j\}_{j=1,n}$ de n destes P^j , então ela organiza o $\{SQF_j\}_{j=1,n}$ em uma família estruturada de cadeias bioquímicas, aqui chamada *Rede Genética* (*RG*). Deste modo, qualquer *SFB* pode ser também definido como uma família de k *RGs* que interagem:

$$SFB = \{RG_j\}_{j=1,k} \quad (3.5)$$

Um tipo particular de produto $p_r \in P_t$ é obtido pelo seguinte tipo de cadeia bioquímica:



isto é, p_r é uma associação repetida de n cadeias $(p_i \oplus \dots \oplus p_j)_m$ de tamanho m . Seja p_r denominado de *produto regular*. Se os símbolos $p_i \dots p_j$ de $(p_i \oplus \dots \oplus p_j)_m$ são iguais ($p_i = \dots = p_j$), então p_r é chamado de *produto regular simples*, caso contrário, é chamado *produto regular complexo*. Um exemplo típico de p_r simples é o polímero. As membranas celulares são exemplos de p_r s complexos.

Um *SFB circular* (*SBC*) é definido como:

$$SBC = \{P_i, P_n, P_t, M, C, G\} \quad (3.7)$$

onde M é um conjunto de $p_r \in P_t$ complexos. Em outras palavras, M é um conjunto de membranas definido por G . A célula é um exemplo típico de um SBC . Uma importante propriedade da delimitação é o fechamento parcial que ele fornece para o seu SBC , o qual permite que um $p_o \in P_i$ não seja codificado por G , mas seja obtido no ambiente circundante do SBC . Em outras palavras, as definições básicas em (3.1) e (3.2) podem ser colocadas no caso de um SBC como:

- a) P_i é o conjunto de produtos iniciais protéicos (P_i^p) ou não-protéicos (P_i^n). P_i é também particionado em um conjunto de produtos contidos fora P_i^o e dentro P_i^i do limite do SBC ;
- b) P_n é o conjunto de produtos intermediários protéicos (P_n^p) ou não-protéicos (P_n^n);
- c) P_t é o conjunto de produtos finais protéicos (P_t^p) ou não-protéicos (P_t^n);
- d) C é o conjunto de cadeias bioquímicas $s(p_o, p_j)$;
- e) $M \subset P_t$, e
- f) G é o conjunto de genes que codificam o conjunto P^p de proteínas contidas em $P_i \cup P_n \cup P_t$.

Dois sub-conjuntos especiais dos elementos de M são:

- g) *o conjunto R de receptores da membrana*: composto por aqueles elementos p_1 nas cadeias bioquímicas $s(p_o, p_1, \dots, p_j)$ ativados por $p_o \in P_i^o$, o que significa dizer que são disparados por produtos iniciais fora do SBC circundado (delimitado), e
- h) *o conjunto T de carreadores da membrana*: composto por aqueles elementos p_1 nas cadeias bioquímicas $s(p_o, p_1, p_o)$ que movem $p_o \in P_i^o$ do lado de fora para o interior do limite do SBC .

Dois SBC_i , SBC_j diferentes são funcionalmente associados se eles compartilham de um sub-conjunto de produtos P_c que são produtos finais do SBC_i e produtos iniciais do SBC_j , ou seja,

- $P_i \cap P_c \neq \emptyset$ no SBC_i , e
- $P_t \cap P_c \neq \emptyset$ no SBC_j ,

onde, \neq significa diferente e \emptyset denota o conjunto vazio. Por causa dos elementos de P_c poderem ser promotores, indutores ou repressores dos genes de G no SBC_j , a interação funcional entre dois diferentes SBC_i, SBC_j pode resultar na especialização do SBC_j para produzir (ativar) um definido sub-conjunto P_s de seu produto final P_t . A especialização do SBC_j para produzir P_s é acompanhada pela seleção de um conjunto G_s de G como genes que podem ser lidos neste mesmo SBC_j . Um *Sistema Biológico Circular Múltiplo Distribuído (SCM)* é uma família de $SBCs$ especializados, compartilhando a mesma genética G . Um organismo multicelular qualquer é o melhor exemplo de um SCM .

Cada SBC tem uma membrana que é uma interface que coleta a informação disponível no meio externo e utiliza-a em reações bioquímicas de acordo com a sua genética (gramática), produzindo uma ou mais cadeias, onde os símbolos pertencentes a estas cadeias são adquiridos de um dicionário, e onde os produtos das cadeias podem ser utilizados por outros $SBCs$, se forem transportados para o meio extra-celular.

Desta forma, os $SBCs$ podem ser organizados em redes celulares $SCMs$ para obter-se um raciocínio do tipo:

$$Q(\rho(d(s_i, s_j)) \Rightarrow \alpha s_i \beta \rightarrow \alpha s_j \beta \vee \dots \wedge \rho(d(s_i, s_k)) \Rightarrow \alpha s_i \beta \rightarrow \alpha s_j \beta) \quad (3.8)$$

$$A(\alpha s_j \beta) = f(\rho(d(s_i, s_j)) \dots \rho(d(s_i, s_k))) \quad (3.9)$$

onde, Q é um quantificador lógico (por exemplo, *a maioria das regras, x de n regras, pelo menos*, etc.); \vee e \wedge são respectivamente, normas e co-normas T ; $\rho(d(s_i, s_j))$ é uma função de $A(s_i), A(s_k), \mu(s_i, s_k)$; e f pode ser definida como uma função de filtragem.

3.4 Linguagens Químicas

Seja dado o seguinte dicionário primitivo de símbolos:

a) os conjuntos complementares N^-, N^+ de nucleotídeos

$$N^- = \{V, W, X, Y, Z\}, \quad N^+ = \{v, w, x, y, z\} \quad (3.10)$$

onde, V = adenina, W = timina, X = citosina, Y = guanina e Z =uracil;

b) os conjuntos complementares A^-, A^+ de aminoácidos

$$A^- = \{A, \dots, T\}, \quad A^+ = \{a, \dots, t\} \quad (3.11)$$

onde cada letra é atribuída a um dos 20 aminoácidos que compõem todas as proteínas conhecidas;

c) os conjuntos complementares I^- , I^+ de compostos químicos tais como íons, produtos metabólicos, etc.

$$I^- = \{\underline{A}, \dots, \underline{T}\}, \quad I^+ = \{\underline{a}, \dots, \underline{t}\} \quad (3.12)$$

onde cada letra sublinhada representa um composto químico diferente.

Os conjuntos complementares serão usados para representar os estados inativo (-) e ativo (+) dos mesmos símbolos.

Sejam as propriedades de acoplamento entre os dicionários S_i , S_j descritas por meio de matrizes bidimensionais M do tipo:

$$M(S_i, S_j) = \{\{m_{ij}\}_{i=1,k}\}_{j=1,k} \quad (3.13)$$

onde, m_{ij} é o grau de acoplamento entre os símbolos $s_i \in S_i$ e $s_j \in S_j$ medidos no intervalo fechado $[0, 1]$, tal que m_{ij} tende a 1 se s_i se acopla perfeitamente à s_j ; caso contrário m_{ij} tende a 0. A matriz $M(N, N)$ para N em (3.10), (3.11) e (3.12) pode ser uma matriz ordinária ideal:

$$\begin{array}{ccccc} & V & W & X & Y & W \\ V & 0 & 1 & 0 & 0 & 0 \\ W & 1 & 0 & 0 & 0 & 0 \\ X & 0 & 0 & 0 & 1 & 1 \\ Y & 0 & 0 & 1 & 0 & 0 \\ Z & 0 & 0 & 1 & 0 & 0 \end{array} \quad (3.14)$$

significando que, a adenina (timina) acopla-se perfeitamente (ou é traduzida em) com timina (adenina); citosina (guanina ou uracil) acopla-se perfeitamente com guanina ou uracil (citosina); e assim por diante.

Ela pode ser também uma matriz nebulosa:

$$\begin{array}{ccccc} & V & W & X & Y & W \\ V & m_{11} & m_{12} & m_{13} & m_{14} & m_{15} \\ W & m_{21} & m_{22} & m_{23} & m_{24} & m_{25} \\ X & m_{31} & m_{32} & m_{33} & m_{34} & m_{35} \\ Y & m_{41} & m_{42} & m_{43} & m_{44} & m_{45} \\ Z & m_{51} & m_{52} & m_{53} & m_{54} & m_{55} \end{array} \quad (3.15)$$

onde, $m_{ii} = 0$ e $0 < m_{ij} \leq 1$.

A matriz de aminoácidos $M(A, A)$ pode ser caracterizada da mesma forma, para descrever interações entre proteínas. As correlações entre aminoácidos e nucleotídeos são descritas por meio de duas diferentes matrizes $M(N, A)$ e $M(A, N)$, que relatam a codificação genética de proteínas e o controle da leitura do gene, respectivamente.

O conjunto de genes G é considerado como um conjunto de cadeias compostas por tuplas de nucleotídeos $n_i \in N = N^+ \cup N^-$. No caso da cadeia de código, cada tripla de nucleotídeos codifica um aminoácido específico ou sinal de parada. Na cadeia de controle, um acoplamento nebuloso define as relações entre moléculas e tuplas de nucleotídeos de tamanho t (por exemplo, a TATA box). Deste modo, um gene $g_i \in G$ em (3.7), é uma cadeia complexa do tipo:

$$g_i = \{(n_i \dots n_j)_t \dots\}_c \{(n_k \dots n_l)_3 \dots\}_n \quad (3.16)$$

composta de c tuplas $(n_i \dots n_j)_t$ e n triplas $(n_m \dots n_n)_3$ de nucleotídeos, correspondendo, respectivamente, às cadeias de controle g_i^c e de código g_i^e . Em outras palavras,

$$g_i = g_i^c \oplus g_i^e = \{(n_i \dots n_j)_t \dots\}_c \oplus \{(n_k \dots n_l)_3 \dots\}_n \quad (3.17)$$

Estas cadeias genéticas podem exibir um dos dois estados:

- a) *gene ativo* g_i : se os símbolos n_i que compõem g_i pertencerem à N^+ , ou seja, $n_i \in N^+$,
ou
- b) *gene inativo* \underline{g}_i : se os símbolos n_i que compõem g_i pertencerem à N^- , ou seja, $n_i \in N^-$.

O estado atual das cadeias \underline{g}_i^c e \underline{g}_i^e são dependentes da relação entre as condições do ambiente atual e as condições de controle codificadas por \underline{g}_i^c . Quando os promotores e indutores predominam sobre os repressores, o gene é ativado; caso contrário, será considerado um gene inativo. Deste modo, os genes de G codificam regras de reescrita do tipo:

$$\begin{aligned} \underline{g}_i^c &\rightarrow g_i^c \\ \underline{g}_i^e &\rightarrow g_i^e \end{aligned} \quad (3.18)$$

isto quer dizer que, g_i^e é lido toda vez que g_i^c é ativado, pois seus promotores e indutores predominam sobre os repressores. Nestas condições, a leitura do g_i^e é controlada pelos sinais químicos no ambiente do núcleo e vizinhança.

As cadeias de controle e código podem ser consideradas como compostas por outras sub-cadeias. No caso de g_i^c , estas sub-cadeias são:

- a) g_i^p : a cadeia do promotor que é composta principalmente por timina e adenina (TATA box), que deve ser ativada para habilitar a leitura do gene;
- b) g_i^i : as cadeias de indutores ou intensificadores, que são cadeias cuja ativação acelera a leitura do gene, ou seja, aumenta o grau de ativação de g_i , e
- c) g_i^r : as cadeias de repressores, que são cadeias cuja ativação inibe a leitura do gene, ou seja, decrementa o grau de ativação de g_i .

Em outras palavras, a ativação é um processo de três etapas:

- a) primeiro, uma proteína externa chamada promotor p , deve fixar-se à cadeia do promotor g_i^p para habilitá-lo:

$$p \oplus \underline{g}_i^p \rightarrow g_i^p \quad (3.19)$$

- b) na seqüência, outros sinais químicos chamados indutores ou intensificadores, devem ligar-se às cadeias de indutores para ativar g_i^i :

$$i \oplus \underline{g}_i^i \rightarrow g_i^i \quad (3.20)$$

- c) e finalmente, os sinais químicos repressores r devem ligar-se à cadeia de repressor g_i^r para desativar g_i :

$$r \oplus \underline{g}_i^r \rightarrow g_i^r \quad (3.21)$$

Deste modo, o grau de ativação $\mu(g_i)$ de g_i recebe uma função f das cadeias ativas g_i^p e g_i^i , e uma função inversa f^{-1} das cadeias de repressores ativadas g_i^r , isto é,

$$\mu(g_i) = f(g_i^p, g_i^i \dots) \otimes f^{-1}(g_i^r \dots) \quad (3.22)$$

onde, \otimes é uma norma \mathcal{T} .

Por simplicidade, os co-fatores necessários para ligar os intensificadores ou os repressores ao promotor, não serão representados como cadeias separadas de g_i^i e g_i^r , respectivamente.

A sub-cadeia de código g_i^e é uma cadeia do tipo:

$$s_i\{\dots\gamma\dots\}_s s_j \quad (3.23)$$

onde, s_i e s_j são os exons de \underline{g}_i^e e $\{\dots\gamma\dots\}_s$ é a dobra ou o intron de \underline{g}_i^e . Os símbolos que compõem s_i , s_j e γ são uma das 64 possíveis triplas t , definidas sob o dicionário N em (3.10). A ativação de g_i^e resulta na síntese de um RNA mensageiro (mRNA) como uma cópia completa da cadeia de codificação, e no corte de seus introns. Em outras palavras,

a) inicialmente, g_i^e é copiado em uma cadeia de mRNA:

$$g_i^c \oplus \underline{g}_i^e \rightarrow g_i^e \rightarrow s_i\{\dots\gamma\dots\}_s s_j \quad (3.24)$$

b) que é então, dobrada e cortada em uma cadeia de triplas $\{t_i\dots t_j\}_l$, que codifica uma seqüência de aminoácidos de uma dada proteína p :

$$s_i\{\dots\gamma\dots\}_s s_j \rightarrow s_i s_j = \{t_i\dots t_j\}_l \quad (3.25)$$

Qualquer proteína $p \in P = P_i \cup P_n \cup P_t$ em (3.7), é uma cadeia de tamanho l de símbolos $a_i \in A = A^+ \cup A^-$ equivalente à sub-cadeia do exon de código do gene $\{t_i\dots t_j\}_l$, ou seja,

$$p_i = \{a_i\dots a_k\}_l \cong \{t_i\dots t_j\}_l \quad (3.26)$$

que é produzido ou está disponível em um estado inativo em algum sítio celular específico, com vistas a exercer uma ação no mesmo ou diferente endereço, toda vez que for ativado por algum sinal recebido, representado por outra cadeia química suportada pelos dicionários em (3.11) e/ou (3.12).

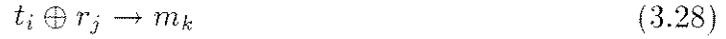
Os seguintes conjuntos de proteínas podem ser definidos, dado o dicionário $D = A \cup I$ e a genética G :

$$\begin{aligned} G &: (D)^m \rightarrow T \\ G &: (D)^n \rightarrow R \\ G &: (D)^p \rightarrow M \end{aligned} \quad (3.27)$$

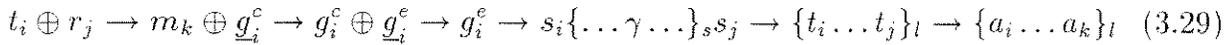
onde, m , n e p são, respectivamente, os tamanhos das cadeias em T , R e M , ou seja, os conjuntos de transmissores moleculares (neurotransmissores, hormônios, etc.), receptores da membrana ou carreadores, e segundos mensageiros, respectivamente. O conjunto $D^+ = A^+ \cup I^+$ será considerado como o conjunto de compostos químicos ativos e o conjunto $D^- =$

$A^- \cup I^-$ será o conjunto de compostos químicos inativos. Por exemplo, $ABCDEF$ pode ser considerada como um receptor inativo $r \in R$, abc pode ser declarado um transmissor ativo $t \in T$ e $DEFMPR$ pode ser considerado um controlador $m \in M$.

Os transmissores t_i (mensageiros primários) ligam-se aos receptores r_j da membrana para ativar segundos mensageiros m_k :



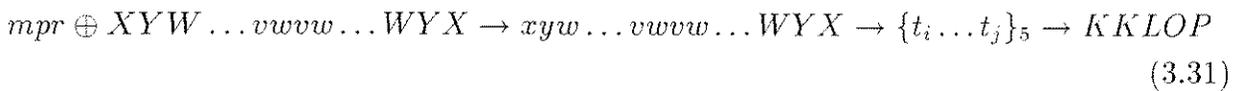
cuja ação principal é servir como um disparador de sinais que controlam a ativação de outras cadeias químicas. Alguns destes mensageiros controlarão as cadeias metabólicas que envolvem o ATP cíclico, GMP, etc. Outros controladores podem ser dirigidos para o núcleo para atuar como promotores, indutores ou repressores do gene. Desta maneira, os transmissores t_i podem ser considerados como símbolos iniciais de uma gramática química, que usa os mensageiros m_k como cadeias não-terminais para controlar a produção de produtos finais codificados por uma genética G . Em outras palavras,



Por exemplo, a ligação do transmissor ativo abc ao sítio complementar ABC do receptor $ABCDEF$ pode ativar seu sítio DEF ; que pode, por sua vez, acoplar-se ao controlador $DEFMPR$, a fim de ativar e liberar a cadeia MPR . Ou seja,

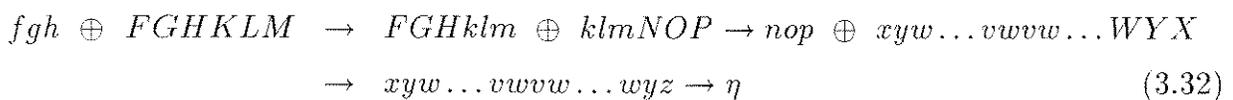


A cadeia liberada mpr pode ser direcionada para o núcleo para ativar um gene g_i , com vistas a mudar a especificação da produção da proteína $KKLOP$:



onde, $\dots vvvw \dots$ é a sub-cadeia de promotor ativa; WYX é a cadeia de repressor inativa; WYZ e wyz são as cadeias de indutor inativas e ativas controladas por mpr , e $\{t_i \dots t_j\}_5$ é a cadeia de exon do gene g_i que codifica a proteína $KKLOP$.

Agora, a ligação de um transmissor ativo fgh ao sítio complementar FGH de outro receptor $FGHKLM$ pode ativar seu sítio KLM que, por sua vez, acopla-se ao controlador $KLMNOP$, a fim de ativar e liberar a cadeia NOP , que é um repressor deste mesmo gene g_i . Desta forma, a síntese de $KKLOP$ pode ser interrompida. Em outras palavras,



onde, η é o símbolo nulo.

Por simplicidade, o acoplamento entre aminoácidos nos exemplos anteriores, foi descrito por meio da similaridade entre letras maiúsculas e minúsculas. Como cada letra (maiúscula ou minúscula) codifica um aminoácido (3.11), as regras acima podem ser mal interpretadas, pois tanto da cadeia de transmissor (controlador) quanto da de receptor requerem, em sua composição, a mesma seqüência de aminoácidos. Em uma notação mais restrita, ambas as cadeias devem ser codificadas por diferentes letras, e a matriz de acoplamento tem que ser definida. Por exemplo, o transmissor *abc* pode perfeitamente ligar-se ao receptor *PRMKLO* se a seguinte matriz de acoplamento for fornecida:

$$\begin{array}{cccccc}
 & \dots & K & L & \dots & O & \dots \\
 a & & & 1 & & & \\
 b & & & & 1 & \dots & \\
 c & & & & & & 1
 \end{array} \tag{3.33}$$

Deste modo, as propriedades de codificação de $M(A, N)$ podem ser especificadas para descrever as interações entre promotores, indutores, e/ou receptores, e dos genes por eles controlados.

Seja

$$A^* = A \cup \{-\} \tag{3.34}$$

onde, A é o dicionário de aminoácidos em (3.11), e $-$ é o sinal de parada. Por causa da equivalência das triplas de nucleotídeos t_i com os símbolos de A^* , tem-se a seguir, uma notação mais apropriada para o gene:

$$REP_PRO_IND_MPOST \dots \tag{3.35}$$

onde:

- a proteína ativa **rep** é o repressor que liga-se à posição *REP* do gene composto pelos correspondentes nucleotídeos complementares aos aminoácidos *r*, *e* e *p*;
- a proteína ativa **pro** é o promotor que liga-se à posição *PRO* do gene composto pelos correspondentes nucleotídeos complementares aos aminoácidos *p*, *r* e *o*;
- a proteína ativa **ind** é o indutor que liga-se à posição *REP* do gene composto pelos correspondentes nucleotídeos complementares dos aminoácidos *i*, *n* e *d*;

- a proteína inativa $MPOST \dots$ é codificada pela cadeia de código

$$g_i^e = MPOST \dots \quad (3.36)$$

Finalmente, alguns aminoácidos ou combinação deles, são conhecidos por operarem sobre outras proteínas, para dobrar ou cortá-las, para inserir um novo elemento dentro delas, ou para reordenar seus elementos, etc. Estas são as operações definidas em F (2.32), no caso de gramáticas nebulosas não-lineares. Em outras palavras, as seguintes operações podem ser atribuídas a uma proteína, de acordo com a composição dos aminoácidos:

a) *dobra*:

$$abc!fgh! + ABC\{\dots\}FGH \rightarrow ABCFGH + \{\dots\} \quad (3.37)$$

isto é, a proteína ativa $abc!fgh!$ dobra a proteína $ABC\{\dots\}FGH$ entre ABC e FGH , independentemente da atual composição da cadeia $\{\dots\}$. Uma alternativa semântica para o operador $!$ é:

$$abc!???? + ABC\{\dots\}FGH \rightarrow ABCFGH + \{\dots\} + abc!???? \quad (3.38)$$

ou seja, a proteína ativa $abc!$ faz a dobra do tamanho de quatro aminoácidos da proteína $ABC\{\dots\}FGH$ após o sítio ABC , independentemente da atual composição da cadeia $\{\dots\}$.

b) *Troca*:

$$abc\$DEF + ABC_KLM_ \rightarrow ABC_DEF_ + abc\$DEF \quad (3.39)$$

ou seja, a proteína ativa $abc\$DEF$ liga-se ao sítio ABC de $ABC_KLM_$ e muda KLM para DEF . O operador $\$$ é útil para mudar o estado de ativação de um sítio:

$$abc\$def + ABC_DEF_ \rightarrow ABC_def_ + abc\$def \quad (3.40)$$

c) *Deleção*:

$$\%abc\$def + ABC_DEF_ \rightarrow ABC_def_ \quad (3.41)$$

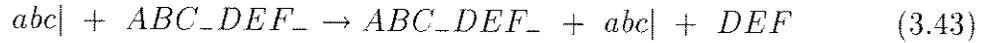
isto é, a proteína ativada $\%abc\$def$ é destruída após exercer sua ação.

d) *Cópia e inclusão*:

$$abc@ + ABC_DEF_ \rightarrow ABC_DEF_ + abc@DEF \quad (3.42)$$

isto é, a proteína ativa $abc@$ copia a cadeia DEF e anexa-a em si mesma.

e) *Cópia e liberação:*



isto é, a proteína ativa $abc|$ copia a cadeia DEF e libera-a no ambiente.

f) *União:*



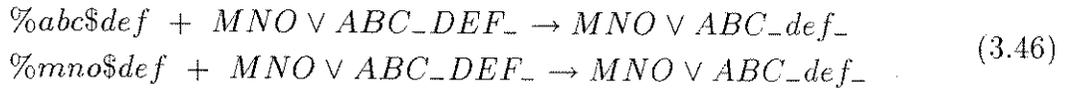
ou seja, a proteína ativa $abc\&def$ combina as cadeias químicas ABC e $DEF_$ em uma nova molécula $ABCDEF_$.

Sejam os seguintes conectivos também definidos:

a) **E** (\wedge): para denotar que a conjunção da cadeia é necessária em qualquer das operações anteriores. Por exemplo,



b) **OU** (\vee): para denotar que a cadeia de disjunção suporta qualquer das operações anteriores: Por exemplo:



Qualquer outra operação com cadeia pode ser definida de um modo similar, mas deve-se ter em mente que, as transações das cadeias anteriores são dependentes da quantidade de cadeias disponíveis e nos graus de ativação de algumas cadeias, como mostradas pelas equações (2.15) e (2.16).

Assim, uma proteína é composta pelas cadeias de acoplamento (por exemplo, $\%abc$, abc , etc.) e de operação (tais como, $\$def$, $\$DEF$, $@$, etc.). Um gene é composto por cadeias associadas por meio de sinais de paradas ou conectivos.

3.5 Processamento Distribuído

Uma característica marcante do processamento celular bioquímico é que ele é distribuído em locais (como membranas, citoplasma, etc.) e compartimentos diferentes (núcleo,

ribossomo, complexo de Golgi, etc.), com o propósito de evitar procedimentos de separação exaustivos em um espaço químico bem homogêneo, necessário para isolar produtos finais definidos. As barreiras de difusão e seqüestro químico também são ferramentas para organizar conjuntos de reações químicas em redes bioquímicas coerentes e objetivas, que trabalham para uma fisiologia celular eficiente. Esta organização é necessária, e até mesmo o vírus precisa que o ambiente celular do hospedeiro seja organizado para poder se reproduzir. A evolução forneceu este processamento químico distribuído dos procariotos aos organismos pluricelulares, pela especialização de organelas, células, sistemas e órgãos. O seqüenciamento de eventos também é uma outra propriedade básica da vida, suportada por um controle genético hierárquico da bioquímica celular. Os genes reguladores são usados para ativar a leitura dos genes estruturais, que contêm a informação necessária para dirigir a reprodução celular, desenvolvimento embriogênico, etc. O sistema nervoso, um sistema celular completo, foi criado pela natureza para auxiliar a organização espacial e a seqüencial do processamento da informação fornecida por redes genéticas mais complexas de animais superiores.

Uma rede genética pode ser considerada, em sua definição mais primitiva, como um conjunto de genes que codificam uma família de proteínas e o conjunto das vias de sinais, que mostram o local onde as proteínas são produzidas e onde atuam. A redução da ambigüidade podem ser feita por meio de uma distribuição não-homogênea de símbolos e/ou cadeias em um dado espaço de processamento. Deste modo, as cadeias (símbolos) são produzidos (ou ativados) em locais definidos para serem usados em algum outro local específico ou de difusão. Isto implica em mover cadeias dentro de um espaço definido por meio de sistemas carreadores mais ou menos específicos, que são suportados por sistemas de correio ou difusão, como canais de comunicação. Funções de custo são associadas à transmissão de mensagens através de qualquer canal de informação.

Assim, uma rede genética RG_k , suportada por uma genética G , é formalmente definida como:

$$RG_k = \{P_i, P_n, P_t, B, G, N\} \quad (3.47)$$

onde, P_i , P_n , P_t , B e G são definidos como em (3.7), e N é uma rede definida por:

- a) o conjunto O de elementos discretos o_k , chamados operadores (ou organela, ou agente), onde cada um deles tem um endereço lógico ou espacial definido a_k , e é especializado no processamento de um conjunto B^k de uma ou mais regras de

reescrita (ou reações químicas) $c_j \in B$. Assim,

$$o_k = \{P_i^k \subset P_i, P_n^k \subset P_n, P_t^k \subset P_t, B^k \subset a_k\} \quad (3.48)$$

b) o conjunto I de canais de informação (correio ou difusão) que suporta uma troca de mensagem (cadeia) entre elementos de O .

Cada canal de informação $i_m \in I$ é especificado por:

c) um conjunto A_i de endereços de entrada a_i , que refere-se aos elementos $o_n \in O$, produzindo as cadeias (mensagens) a serem liberadas em o_k ;

d) um conjunto A_o de endereços de saída a_o , que refere-se aos elementos $o_r \in O$ que recebem (aceitam) as cadeias produzidas por o_k , e por

e) uma função de custo δ , que descreve as restrições temporais e espaciais das mensagens veiculadas através do canal.

O endereço pode ser uma direção física ou lógica. No primeiro caso, o sistema de correio (como axônio, transporte intracelular, etc.) libera as mensagens em locais específicos do espaço de comunicação. No segundo caso, as mensagens são amplamente distribuídas (por exemplo, difusão, corrente sanguínea, etc.) para ligarem-se aos receptores específicos dos agentes que estão aptos a recebê-las.

Assim,

$$i_m = \{A_s, A_r, \delta(A_s, A_r, t)\} \quad (3.49)$$

O conjunto O de agentes é particionado nos seguintes conjuntos, de acordo com seus produtos de entrada (cadeias):

f) O_i : é o conjunto de agentes de entrada que usam símbolos iniciais para produzir cadeias não-terminais. No caso destes agentes, $P_t^k = \emptyset$;

g) O_n : o conjunto de agentes intermediários que usam símbolos não-terminais para produzir cadeias não-terminais. No caso destes agentes, $P_i^k = \emptyset$ e $P_t^k = \emptyset$;

h) O_t : o conjunto de agentes de saída que usam símbolos não-terminais para produzir cadeias terminais. No caso destes agentes, $P_i^k = \emptyset$.

Diferentes redes genéticas RG_k suportadas por uma gramática comum G podem sofrer uma ativação paralela, mas a hierarquia de G pode organizar estas atividades paralelas em um processamento concorrente, pela imposição de uma ordem seqüencial, pela ativação de redes distribuídas em diferentes localizações espaciais.



3.6 Processamento Delimitado

As membranas são usadas para definir espaços de processamento específicos, tais como os espaços intra e extracelular, e no interior da célula, os espaços do núcleo, do complexo de Golgi, dos ribossomos, etc. Os sistemas carreadores definidos estão organizados tanto em membranas quanto dentro do espaço intracelular, a fim de mover moléculas de um lugar para outro, quando necessário para o processamento celular das informações intra e extracelulares. Toda vez que as membranas são usadas para delimitar um dado processamento, elas fornecem, entre outras, as seguintes funções principais:

- a) restringir processamentos definidos com limites específicos, a fim de diminuir a ambigüidade das linguagens químicas nebulosas;
- b) organizar a interface de comunicação entre os espaços interno e externo, como definidos, com o propósito de reconhecer símbolos ou mover cadeias iniciais do ambiente externo e liberar os produtos finais para este mesmo espaço externo, e
- c) definir o espaço métrico usado para especificar os endereços usados pelos canais de comunicação.

E ainda,

- d) os sistemas de transporte intracelular (por exemplo, transportes axônicos retrógrado e anterógrado) podem ser usados para mover substâncias químicas entre locais definidos, limpando um espaço de processamento definido.

Outro processo usado para organizar o processo de linguagem nebulosa é:

- e) seqüestrar substâncias químicas específicas (tais como íons cálcio) em algum sítio específico, para liberá-los em um momento apropriado, a fim de atuar em um processamento específico.

A combinação dos processos acima é amplamente usada pelas células, como no caso dos transmissores neurais, cujos precursores são produzidos no corpo das células, movidos para os terminais pré-sinápticos por meio do transporte axônico, onde a síntese final ocorre, usando compostos químicos importados das células pós-sinápticas. Assim, a ativação dos receptores dos dendritos podem disparar sinais axônicos elétricos (trens), cujo objetivo é liberar estes transmissores para atuarem sob as células pós-sinápticas. Deste modo, os neurônios podem ser considerados como agentes especializados na troca de mensagens,

em que estas mensagens são suportadas por uma linguagem nebulosa definida por sua genética, e onde os trens são usados para acelerar o processo de troca de mensagens, ou seja, reduzir o custo de transmissão da mensagem.

Desta maneira, a célula C_i suportada por uma genética G pode ser considerada como uma família de n redes genéticas RG_k , distribuídas sob um conjunto de espaços de computação S_m , organizados pelo conjunto M de suas membranas:

$$C_i = \{P_i, P_n, P_t, M, B, G, N\} = (\{RG_k\}_{k=1,n}, M) \quad (3.50)$$

Daí, as células podem ser organizadas em redes de ordens mais altas NN (por exemplo, no caso do sistema nervoso) aumentando o poder computacional das linguagens nebulosas definidas por sua genética comum G :

$$NN = \{C_i, C_n, C_t, G, N\} \quad (3.51)$$

onde:

- a) C_i é o conjunto de células de entrada (sensoriais) que reconhecem o conjunto de símbolos iniciais $P_i \in G$;
- b) C_n é o conjunto de células intermediárias (associativas) que produzem o conjunto de símbolos não-terminais $P_n \in G$;
- c) C_t é o conjunto de células de saída (motoras) associadas ao conjunto de símbolos terminais $P_t \in G$;
- d) N é a matriz de comunicação que suporta a troca de mensagens entre estas células.

Capítulo 4

Neurônio: a unidade fundamental

4.1 Introdução

O sistema nervoso é uma rede organizada de células nervosas ou neurônios. O neurônio é a unidade fundamental de processamento do cérebro e também do nosso modelo artificial distribuído. As funções do sistema nervoso são determinadas por neurônios organizados em uma grande variedade de redes neurais, cada uma desempenhando um determinado papel. Assim, funções críticas do cérebro, como o processamento de informações sensoriais, a programação motora, as respostas emocionais, a aprendizagem e memória são produzidas por conjuntos de neurônios interconectados que trocam informações entre si.

A capacidade do cérebro em mudar seu funcionamento diante da repetição de situações semelhantes e da ocorrência de situações novas é uma de suas mais importantes características. A atividade neural, portanto, tem a mesma propriedade. Esta habilidade que o sistema nervoso tem de alterar o seu modo de funcionamento chama-se *plasticidade neural* e é um dos motivos que nos inspiraram em nossa metodologia. Para melhor se entender o porquê de sua utilização, fazemos uma breve revisão dos conceitos biológicos em que nos baseamos e apresentamos o nosso modelo artificial.

4.2 O Neurônio Biológico

4.2.1 A Membrana Celular

O neurônio é a unidade celular formadora do sistema nervoso, e como todas as células, é constituído por uma membrana celular e um citoplasma. A membrana celular é composta por lipídios e proteínas, e separa o citoplasma do líquido extracelular (solução aquosa diluída). O citoplasma, por sua vez, é formado por água, eletrólitos, proteínas, lipídios e carboidratos. A composição do líquido extracelular e do líquido intracelular (citoplasma) não é a mesma. Enquanto o primeiro contém grandes quantidades de sódio e cloro, o último contém grandes quantidades de potássio, fosfatos e proteínas. As membranas servem como barreiras de permeabilidade para manter a composição citoplasmática diferente do líquido extracelular.

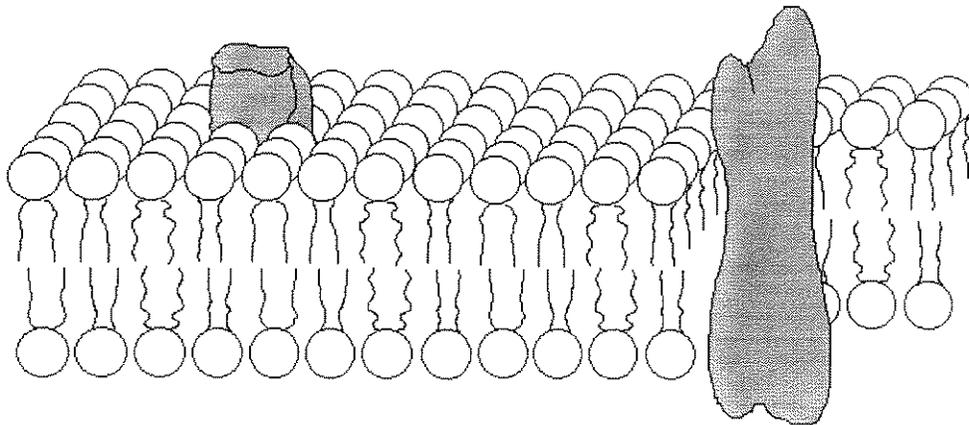


Figura 4.1: Modelo do mosaico fluido.

A membrana celular tem como estrutura básica uma camada bilipídica, que é uma película delgada de lipídios, com espessura de duas moléculas. A bicamada é formada por uma parte *hidrofílica* (solúvel em água), que está voltada para a superfície, e uma outra parte *hidrofóbica* (solúvel em gordura), que ocupa a região central da membrana, como mostrado na fig. 4.1. Esta estrutura é conhecida como *modelo do mosaico fluido*.

A membrana possui proteínas tubulares chamadas canais que atravessam-na e que permitem, seletivamente, a passagem de água e de íons do interior para o exterior da célula e vice-versa, por meio de diversos tipos de transporte, como difusão, osmose, trans-

portes facilitado e ativo. Essa permeabilidade seletiva é responsável pelo fluxo iônico existente na célula e causa uma diferença de potencial chamada de *potencial de repouso da membrana*. O trânsito de moléculas através da membrana é necessário para que a composição citoplasmática da célula seja mantida constante, por mais variável que seja o meio extracelular, condição esta essencial para a sua sobrevivência.

Por curiosidade, o potencial de repouso da membrana (E) pode ser calculado segundo a *equação de Goldman*, que tem a forma:

$$E = \frac{RT}{zF} \ln \frac{P_K[K^+]_e + P_{Na}[Na^+]_e + P_{Cl}[Cl^-]_e}{P_K[K^+]_i + P_{Na}[Na^+]_i + P_{Cl}[Cl^-]_i} \quad (4.1)$$

onde: R é a constante universal dos gases;
 T é a temperatura absoluta;
 z é a valência do íon;
 F é a constante de Faraday;
 P_i é o coeficiente de permeabilidade do íon i ;
 $[i]_e$ é a concentração extracelular do íon i e
 $[i]_i$ é a concentração intracelular do íon i .

Quando por algum motivo a concentração intra ou extracelular é mudada, o potencial de repouso da membrana é alterado, como por exemplo, se ocorre aumento ou diminuição da permeabilidade da membrana aos íons sódio ou potássio, de modo que haja uma elevação inicial suficiente da membrana acima de um limiar, é produzido um *potencial de ação*, resultante da abertura de vários canais de sódio; caso contrário, nenhuma resposta é produzida. É o chamado *potencial de tudo ou nada*. Esse potencial pode ser gerado em qualquer parte da membrana e é propagado nas regiões adjacentes ao longo da mesma, criando um *impulso*, ou seja, a membrana foi excitada.

Por não ser objetivo deste trabalho, uma análise mais profunda sobre potenciais de ação pode ser encontrada em [KSJ91],[Guy92]. O que é de interesse é saber que eles são gerados à partir de fluxos iônicos através da membrana celular, devido à condutância dos canais de sódio, principalmente, e de potássio, ou por um outro agente capaz de modificar transitoriamente o potencial de repouso da membrana.

4.2.2 Estrutura e Fisiologia do Neurônio

Uma célula nervosa típica, como visto na fig. 4.2, pode ser dividida morfológicamente em quatro componentes distintos: o corpo celular, os dendritos, o axônio e os terminais pré-sinápticos. O *corpo celular* é o centro metabólico do neurônio, ou seja, o local onde ocorre execução dos processos decorrentes dos dendritos e dos enviados ao axônio. O neurônio, geralmente, tem vários *dendritos*, que servem como o principal meio de recepção de sinais de outras células nervosas, e apenas um *axônio*, que é a sua principal via de condução. O axônio, por sua vez, fornece várias ramificações para outras partes do cérebro ou periferia do corpo. Ele origina-se em uma região especializada do corpo celular chamada *cone do axônio*, onde um potencial de ação é iniciado, desde que um limiar crítico seja atingido, conforme será visto mais adiante.

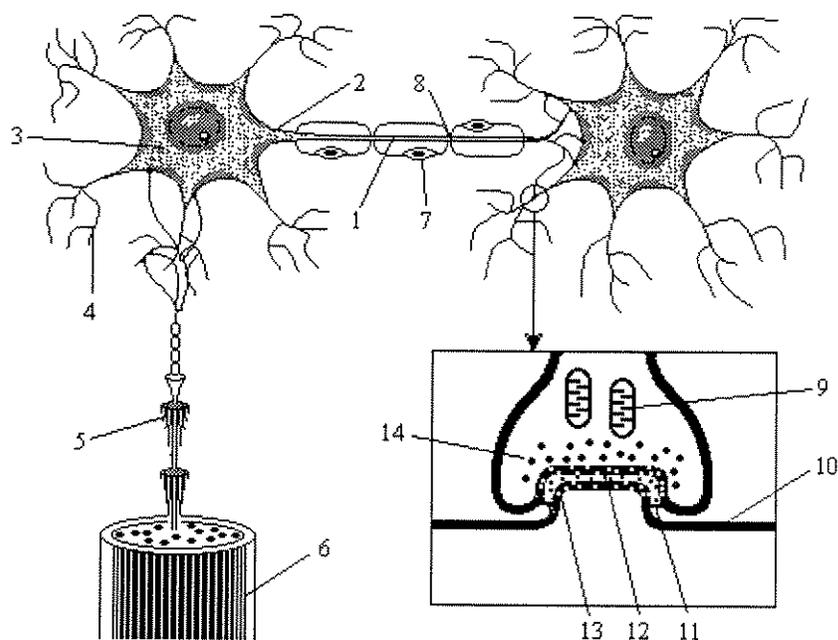


Figura 4.2: Estrutura do neurônio e anatomia fisiológica da sinapse.

No detalhe, vista do acoplamento entre o terminal pré-sináptico e o neurônio pós-sináptico.
 1. Axônio, 2. cone do axônio, 3. corpo celular, 4. dendritos, 5. fibra nervosa, 6. nervo espinhal, 7. célula de Schwann, 8. nodo de Ranvier, 9. mitocôndria, 10. membrana pós-sináptica, 11. fenda sináptica, 12. neurotransmissor, 13. proteínas receptoras e 14. vesículas de transmissores.

Próximo ao seu fim, o axônio se divide em finos ramos que tem expansões especializadas chamadas *terminais pré-sinápticos*. É por meio dos seus terminais que um neurônio transmite informações sobre sua atividade para as superfícies receptoras dos outros neurônios

(que podem ser os dendritos ou o corpo celular). Este ponto de junção é denominado *sinapse*. A célula que envia a informação é chamada célula pré-sináptica e a célula que recebe a informação é a célula pós-sináptica. O espaço que separa as duas células é conhecido como *fenda sináptica* e ela comunica-se livremente com o espaço extracelular.

Há dois tipos fundamentais de sinapse, de acordo com a morfologia e o mecanismo de transmissão: a sinapse elétrica e a sinapse química.

No cérebro, as sinapses elétricas são menos comuns que as sinapses químicas, e são caracterizadas pela rápida velocidade de transmissão e pela definição da sua função. Por exemplo, as sinapses elétricas geralmente não permitem ações inibitórias de longa duração que mudem a sua efetividade. Já as sinapses químicas mediam tanto ações inibitórias quanto excitatórias e estão envolvidas em comportamentos mais complexos que os das sinapses elétricas. Portanto, as sinapses químicas são mais flexíveis e capazes de realizar mudanças mais duradouras na sua efetividade. Esta plasticidade é importante para várias funções superiores do cérebro, como aprendizagem e memória. Outro aspecto importante, é que as sinapses químicas possuem a capacidade de amplificar os sinais neurais.

4.2.3 O Processamento Elétrico das Sinapses

Quando as correntes originadas na terminação axônica pela propagação do potencial de ação ao longo do axônio chegam à região da fenda sináptica, encontram uma brecha estreitíssima entre a membrana pré-sináptica e a pós-sináptica, é a chamada *zônula de oclusão*. Como a resistência nesta região é muito alta, as correntes encontram mais facilidade para atravessar diretamente a membrana pós-sináptica, agindo como estímulos elétricos, e, promovendo apenas correntes elétricas através dela ou desencadeando impulsos completos.

As sinapses elétricas permitem a condução destes estímulos em ambas as direções e são úteis particularmente nas vias reflexas, pois elas precisam de transmissão rápida entre as células ou a resposta síncrona de alguns neurônios, tais como as células miocárdicas, as musculares lisas intestinais e os hepatócitos.

4.2.4 O Processamento Químico das Sinapses

O terminal pré-sináptico é separado do corpo neuronal pela *fenda sináptica*, e ele contém duas estruturas internas importantes para o tipo de função exercida pela sinapse: as *vesículas sinápticas* e as *mitocôndrias*. A vesícula sináptica tem como conteúdo uma solução de moléculas de uma substância específica conhecida como *mediador químico da transmissão* ou *neurotransmissor*. As mitocôndrias, por sua vez, fornecem a energia necessária para a síntese de mais substância transmissora. Essa energia é provida pelo trifosfato de adenosina ou ATP.

A transmissão química se inicia quando um potencial de ação se propaga por um terminal pré-sináptico, causando o esvaziamento de um pequeno número de vesículas no interior da fenda sináptica. O (neuro)transmissor liberado modifica a condutância da membrana plasmática pós-sináptica para um ou mais íons. Essa alteração na condutância, caso o íon não esteja em equilíbrio, promove o fluxo desse íon, causando a variação do potencial da membrana da célula pós-sináptica, que leva à excitação ou inibição do neurônio, de acordo com as características de seus receptores. A quantidade de (neuro)transmissores liberados na fenda sináptica está fortemente relacionada ao número de íons cálcio que entram no terminal pré-sináptico. Acredita-se que este mecanismo funcione da seguinte maneira. Ao entrarem no terminal, os íons cálcio se ligam a moléculas protéicas na superfície interna da membrana, em locais conhecidos como *sítios de liberação*. A ligação faz com que as vesículas transmissoras localizadas na vizinhança do local também se liguem à membrana, fundindo-se a ela e, por último, abram-se, por *exocitose*, para o exterior. Após a liberação do (neuro)transmissor, ele é destruído ou removido de algum modo, para evitar que sua ação continue para sempre e para que após alguns milissegundos possa haver transferência de informação ou não. Os mecanismos responsáveis pela remoção dos transmissores são os seguintes:

- por difusão do transmissor da fenda para os líquidos em volta;
- por destruição enzimática dentro da fenda;
- por transporte ativo de volta ao terminal pré-sináptico que o liberou e reutilização. Este processo é chamado *recaptação do transmissor*; e
- por ligação a uma molécula da membrana pré-sináptica.

Esta última possibilidade é a que nos interessa. Ela causa a alteração conformacional

da estrutura da membrana, e conseqüente inibição ou excitação da célula pós-sináptica, ocasionada pelo acoplamento entre o transmissor e o receptor. Inibem, se a membrana neuronal contiver receptores inibitórios, e excitam, se ela contiver receptores excitatórios.

O acoplamento do transmissor ao receptor, que depende da afinidade química entre eles, ativa algumas moléculas pós-sinápticas, denominadas *atuadores*, que exercem uma ação tanto sobre os neurônios pré e pós-sináptico quanto sobre células vizinhas. A ação exercida pelos atuadores depende de sua estrutura e função, e algumas das ações possíveis são:

- a) modificar a permeabilidade pós-sináptica, promovendo a modificação do potencial da membrana (canais iônicos);
- b) atuar como uma enzima cuja atividade é controlar alguma cadeia metabólica, alterando a quantidade de energia disponível na membrana, onde, em geral, esta energia é usada para modificar os limiares da membrana (neurônio modulador);
- c) agir como uma molécula de controle ou regulatória, responsável por uma das seguintes ações (neurônio controlador):
 - c.1) a modulação da ligação entre transmissor e receptor;
 - c.2) a especificação da leitura do DNA;
 - c.3) a ativação da leitura do DNA ou
 - c.4) a especificação final da síntese protéica de certas moléculas.

As proteínas produzidas pela célula pré-sináptica são chamadas *precursores*, e as liberadas pela célula pós-sináptica são chamadas *controladores*. Um mesmo precursor pode ser utilizado para produzir diferentes tipos de transmissores em diferentes terminais axônicos, pois a especificação final do transmissor depende dos controladores produzidos pela célula pós-sináptica. Assim, a especificação genética do transmissor depende dos neurônios pré e pós-sinápticos, mas a ativação da leitura de genes específicos é dirigida pela ativação dos receptores pós-sinápticos. Esse controle da leitura é executado por moléculas que são liberadas dentro da fenda sináptica juntas com o transmissor através da vesícula sináptica e conhecidas como *sinalizadores*. Esta plasticidade da codificação química é um fator importante para a diferenciação dos neurônios durante o crescimento embriogênico.

Uma outra característica importante das sinapses químicas é que elas permitem apenas

a condução unidirecional dos estímulos; isso possibilita que os sinais sejam dirigidos para pontos específicos e causem ações específicas.

4.2.5 A Codificação Sináptica

A codificação (ou transdução) sináptica é a transformação da informação mediada pelo transmissor em uma alteração do potencial da membrana pós-sináptica. Nas sinapses elétricas, o fluxo de correntes pré-sinápticas através da membrana pós-sináptica muda imediatamente o seu potencial, promovendo uma *hiperpolarização* ou *hipopolarização*, dependendo do sentido do fluxo; portanto, a transdução é direta. Já nas sinapses químicas, a ligação transmissor-receptor é que promove as variações de permeabilidade da membrana e determina a sua excitação ou inibição, determinando qual o tipo de mensageiro liberado.

4.2.6 Aprendizagem

A aprendizagem pode ser vista como a construção de uma representação de um conhecimento observado no mundo exterior pelo cérebro, segundo a interpretação e reconhecimento dos sinais e mensagens trocadas pelos neurônios, que expressam relações ou associações entre as informações recebidas pelos diversos mecanismos sensoriais.

Os neurônios do cérebro produzem moduladores que estão envolvidos no controle da aprendizagem, de acordo com as necessidades do ser vivo. O processo de modulação segue os seguintes passos:

- a ativação sináptica é um resultado dos sinais químicos que servem como marcadores dos caminhos neurais ativados durante o processamento de informação;
- os resultados das etapas do processamento são calculados de acordo com alguns objetivos de algumas áreas específicas do cérebro, conhecidas como *áreas de controle da aprendizagem*. Dependendo deste cálculo:
- as sinapses ativadas podem ser *recompensadas* ou *punidas* através da liberação de moduladores adequados para as áreas de controle da aprendizagem.

Estas atividades são realizadas por um complexo processamento químico. Assim, a aprendizagem no cérebro é também um processo complexo que depende:

1. da especificação dos conjuntos de transmissores e sinalizadores nos terminais pré-sinápticos;
2. da especificação dos receptores e controladores pós-sinápticos;
3. da regulação da quantidade de transmissores, sinalizadores, receptores e controladores especificados;
4. do ajuste da função de decodificação pré-sináptico que descreve a ativação axônica e a liberação do transmissor.

4.3 O Neurônio Artificial

O modelo artificial do neurônio usado nesta dissertação foi proposto por Rocha [Roc92], e utiliza os conhecimentos da fisiologia da sinapse, o que o torna um processador complexo, combinando os processamentos sinápticos elétrico e químico, e permitindo tanto o processamento numérico quanto o simbólico.

A transição da informação do neurônio pré-sináptico para o neurônio pós-sináptico envolve a liberação de mensageiros químicos pré-sinápticos denominados transmissores (t) que se ligam a certas moléculas pós-sinápticas chamadas receptores (r). A ligação t/r ativa, por sua vez, outras moléculas pós-sinápticas chamadas, genericamente, controladores (c), cujo propósito é exercer uma ação sobre a atividade neural.

Esta transação sináptica pode ser descrita, formalmente, como:

$$t + r \mapsto c \rightarrow \text{ação} \quad (4.2)$$

A quantidade $a(i, c)$ de controladores ativados na i -ésima sinapse do neurônio n_j , depende:

- a) da quantidade $a(i, r)$ do receptor r disponível no sítio pós-sináptico;
- b) da quantidade $a(i, t)$ do transmissor t disponível no terminal pré-sináptico, e
- c) da afinidade química $\mu(t, r)$ entre t e r .

Seja

$A(i, t)$ a quantidade total do transmissor t armazenada na célula pré-sináptica.

$A(i, r)$ a quantidade total do receptor r armazenada na célula pós-sináptica.

m o número de fontes s_i que fornecem informações de entrada para um neurônio n_j , com relevância w_i cada uma. A saída s_j é obtida como:

$$a = \sum_{i=1}^m s_i * w_i \quad (4.3)$$

$$s_j = f(a) \quad (4.4)$$

isto é, s_j é a função $f(a)$ da média ponderada a , das entradas s_i . f é, em geral, uma função booleana tal como:

$$f(a) = \begin{cases} 1, & \text{se } a > \alpha \\ 0, & \text{caso contrário} \end{cases} \quad (4.5)$$

mas qualquer outra função de filtragem pode ser definida.

A força sináptica w_i é calculada como:

$$w_i = A(i, t)\rho A(i, r)\lambda\mu(t, r) \quad (4.6)$$

onde, ρ e λ são, em geral, normas triangulares.

Assim, a quantidade total $a(i, t)$ do transmissor liberado na i -ésima sinapse pode ser calculado como:

$$a(i, t) = s_i * w_i \quad (4.7)$$

onde, $*$ é, geralmente, uma norma triangular.

A quantidade $a(i, c)$ do controlador ativado pelo acoplamento t/r é:

$$a(i, c) = g(a(i, t)) \quad (4.8)$$

onde, g é uma função qualquer.

A ação do controlador c pode ser de dois tipos diferentes:

- a) ativar o axônio: neste caso, a quantidade total da ativação axônica pode ser obtida pela equação (4.3), que é o neurônio de McCulloch-Pitts, e a transação sináptica definida pela equação (4.2) é um processamento numérico;

- b) modificar o tipo e a quantidade de moléculas que existem nos neurônios pré e/ou pós-sinápticos. Neste caso, a transação sináptica definida pela equação (4.2) é uma regra de produção e o processamento é simbólico.

Finalmente, diferentes transmissores t podem ser liberados na mesma célula, dependendo do seu grau de ativação axônico, pois diferentes funções de filtragem podem ser usadas para ativar diferentes terminais pré-sinápticos de uma mesma célula que armazena diferentes tipos de transmissores.

4.3.1 O Modelo Formal do Neurônio

Um neurônio N pode ser definido como a seguinte estrutura:

$$N = \{\{W_p\}, W_o, T, R, C, \Theta, \{\alpha, g\}, L(G)\} \quad (4.9)$$

onde

- $W_p (p = 1, \dots, n)$ é o conjunto de entradas pré-sinápticas que atuam em N por todos os seus n axônios pré-sinápticos.
- W_o é o código de saída de N .
- T é o conjunto de transmissores usados por N para trocar mensagens com outros neurônios;
- R é o conjunto de receptores que se ligam aos transmissores liberados pelos neurônios pré-sinápticos. A força s_i das sinapses com o i -ésimo neurônio pré-sináptico é:

$$s_i = A(i, t) \wedge A(i, r) * \mu(t, r) \odot v_o$$

e a atividade pós-sináptica v_i é calculada como:

$$v_i = w_i \circ s_i$$

onde, $A(i, t)$ é a quantidade do transmissor t no neurônio pós-sináptico n_i ; $A(i, r)$ é a quantidade do receptor r disponível para o acoplamento com t ; $\mu(t, r)$ é a afinidade da ligação t/r , v_o é a variação da energia utilizada para esta ligação; e $\wedge, *, \odot$ e \circ são normas e co-normas triangulares.

- C é o conjunto de controladores que podem ser ativados pelo acoplamento (t/r) , onde o elemento c_j é dado pela relação:

$$t_i + r_j \mapsto c_j$$

para $r_j \in R, t_i \in T_p$ e $c_j \in C$.

- Θ é a função usada para agregar a atividade pós-sináptica atual

$$a_n = \Theta_{i=1}^n w_i * s_i \quad (4.10)$$

- α representa o conjunto de limiares e g a função de codificação, para uma saída codificada w dada por:

$$w = \begin{cases} w_i, & \text{se } a_n < \alpha_1 \\ w_u, & \text{se } a_n \geq \alpha \\ g(a_n), & \text{caso contrário} \end{cases} \quad (4.11)$$

- $L(G)$ é uma linguagem L associada á uma gramática G , que define o acoplamento entre transmissores e receptores, e a ativação de controladores.

4.3.2 Redes Neurais Artificiais

Através da associação de neurônios, hierarquicamente organizados, é possível construir uma rede neural capaz de realizar uma tarefa que possa exigir tanto um processamento numérico, quanto simbólico, ou ambos. Cada neurônio da rede neural é definido como o modelo da equação (4.9), possuindo uma família particular de transmissores, receptores e controladores, e ainda, tendo uma função de filtragem axônica e limiares específicos, de acordo com sua habilidade. A configuração de uma rede neural e a especificação de cada neurônio dependem do objetivo ou tarefa que a rede deve desempenhar.

Capítulo 5

O Sistema Sensorial Inteligente Distribuído

5.1 Introdução

Este capítulo descreve sumariamente os fundamentos biológicos do sistema visual humano e os fundamentos computacionais que foram usados no desenvolvimento do sistema sensorial inteligente distribuído, o **SENSOR**, para a implementação do sistema visual artificial. O enfoque computacional é voltado ao reconhecimento de padrões, através da abordagem sintática. Aqui também são apresentadas a estrutura básica e a organização do **SENSOR**.

5.2 O Sistema Visual Humano

A nossa percepção visual ocorre em estágios distintos. A luz que atravessa a córnea é projetada no fundo do olho, onde é convertida em um sinal elétrico para um órgão sensorial específico, a retina (fig. 5.1). A conversão da luz em sinais elétricos é realizada por neurônios retiniais especializados, os fotorreceptores. Existem dois tipos de fotorreceptores na retina: os cones e os bastonetes. Os cones são as células responsáveis pelas diferenças de cor e intensidade luminosa (luminância). Na retina, há três tipos de cones intercalados ao acaso, sensíveis às cores vermelha, verde ou azul, que devido à ativação

ou não deles, simultaneamente, podem ser recebidas todas as outras sensações de cores. Já os bastonetes, em maior número na retina, mas acromáticos, são os responsáveis pela percepção de diferenças entre claro e escuro.

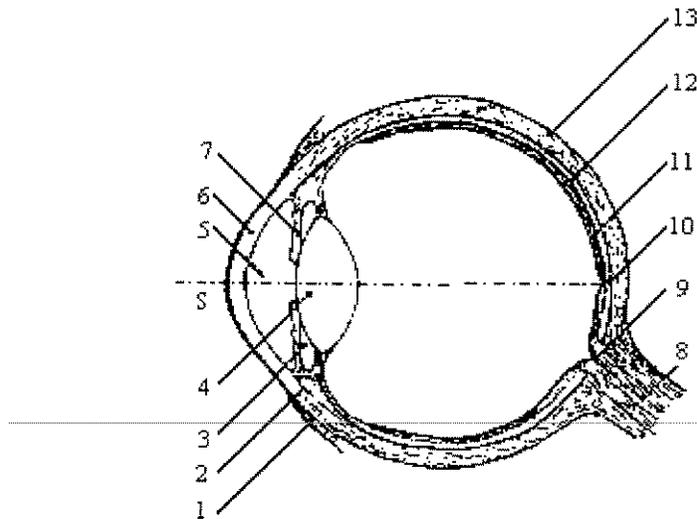


Figura 5.1: Corte horizontal pelo olho (esquema).

Visto de topo sobre a superfície de corte do olho. 1.conjuntiva; 2.corpo ciliar com músculos e ligamentos para o cristalino; 3.câmara posterior; 4.cristalino; 5.câmara anterior; 6.córnea; 7.íris; 8.nervo óptico; 9.ponto cego; 10.fóvea; 11.retina; 12.coróide; 13.esclerótica; S = eixo visual (linha de ligamento entre a fóvea e o ponto central da córnea, a pupila).

A parte mais importante da retina para a visão, a fóvea, contém apenas cones, portanto, é uma área de visão mais acurada, de maior resolução e menos distorcida, e é responsável por gerar informações sobre cor e detalhes. Por isso, para reconhecer um objeto claramente é preciso que a imagem do objeto de interesse incida sobre a fóvea.

A retina possui cinco classes de neurônios: fotorreceptores, células bipolares, células horizontais, células amácrinas e células ganglionares. As *células bipolares* ligam os fotorreceptores às células ganglionares. A modulação das informações do receptor para as células ganglionares é realizada por dois tipo de interneurônios: as *células horizontais*, que mediam as interações laterais entre receptores e células bipolares, e que estão relacionadas com a análise espacial do objeto; e, as *células amácrinas* que também interligam as vias em linha da retina, entre células bipolares e ganglionares, e atuam na detecção de movimento na imagem visual, influenciando na ativação das células ganglionares. As

células ganglionares levam a informação para o nervo óptico, e este para o cérebro, para ser processada (fig. 5.2).

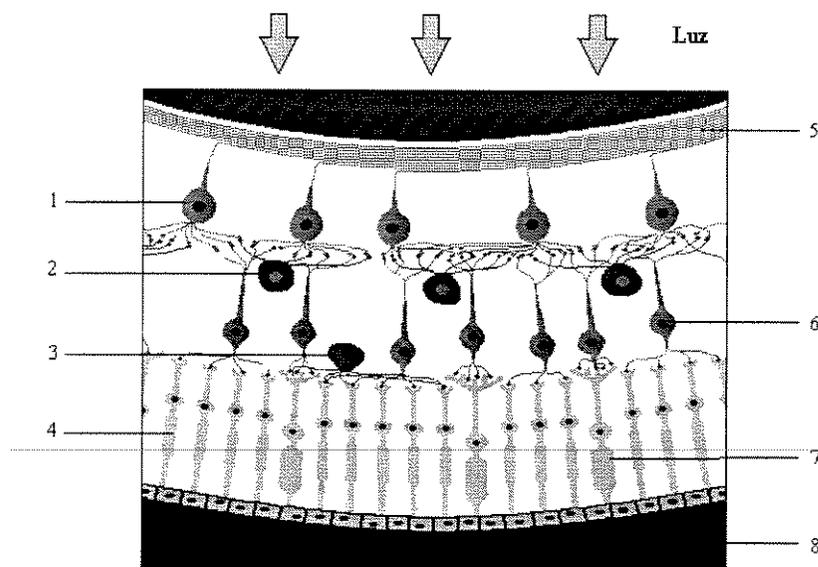


Figura 5.2: Organização neural da retina.

Vista do corte da retina. 1.célula ganglionar; 2.célula amácrina; 3.célula horizontal; 4.bastonete; 5.membrana limitante interna; 6.célula bipolar; 7.cone; 8.camada pigmentar.

Devido à modulação da informação visual luminosa pelas células horizontais, as células ganglionares podem ter propriedades mais complexas de resposta aos estímulos, como por exemplo, detectar bordas, independentemente do nível de iluminação. A detecção é feita através de um fenômeno chamado *antagonismo centro-circundante*. A via visual proveniente da área de incidência luminosa mais central é excitada, enquanto a área ao redor é inibida, ou vice-versa. Isso permite uma alta acuidade visual na transmissão das bordas de contraste na imagem visual [KSJ91].

As interações antagônicas entre áreas vizinhas da retina são mediadas pelas células horizontais. Os cones liberam um transmissor que despolarizam as células horizontais, que, por sua vez, liberam um transmissor que hiperpolariza cones vizinhos. Desta forma, a hiperpolarização dos mesmos cones pela luz leva à despolarização dos cones vizinhos por meio dos interneurônios horizontais. Assim, as células horizontais tornam ainda mais escuros os cones que estão mais escuros do que os seus cones vizinhos iluminados apenas

fracamente, ou que não estão iluminados no todo.

A ação de células horizontais na retina é o primeiro passo em uma cadeia de eventos que é repetida sucessivamente em níveis mais altos dos sistema visual para aumentar o contraste entre claro e escuro, a fim de facilitar a distinção de contornos.

As células ganglionares transmitem, então, as informações visuais para o *núcleo geniculado lateral* (NGL) e este as retransmite para o córtex visual.

O córtex cerebral é uma região que recebe informações de entradas sensoriais (olhos, ouvidos, tato, etc.), processa informações e também projeta saídas motoras, mas acredita-se que esteja principalmente relacionado a eventos mentais, associação e integração de todo tipo de informação.

O córtex opera de forma modular, com seus neurônios organizados em colunas de cerca de 400 μm de diâmetro e 4000 elementos, onde aproximadamente 2000 são células piramidais. Os elementos de cada um destes módulos estão dispostos em seis camadas arranjadas, principalmente, verticalmente.

Uma característica fundamental do córtex cerebral é a similaridade de sua organização entre diferentes áreas, o que leva a crer que o córtex realiza operações computacionais de natureza geral, todavia estas áreas exibem diferenças locais de estruturas que suportam funções tão diferentes como percepção, programação motora, memória, planejamento, processamento de linguagem e raciocínio. Uma divisão funcional destas áreas foi proposta por Brodmann (fig. 5.3), por exemplo, as áreas 17 e 18 possuem neurônios especializados no tratamento dos sinais visuais. Podemos então considerar cada uma destas áreas como módulos que integram a atividade de diferentes agentes especializados.

O córtex visual está localizado nos lobos occipitais do cérebro e é dividido em *córtex visual primário* (V1) e *áreas visuais secundárias* (fig. 5.4).

O córtex visual primário recebe os sinais provenientes da retina e tem a função de decompor o mundo visual. O córtex é dividido em colunas verticais de neurônios, representando unidades funcionais diversas, e os sinais visuais aferentes quando chegam até aí, passam sucessivamente pelas camadas, e os segmentos da informação visual vão sendo decifrados.

As áreas visuais secundárias ou de associação dissecam e analisam os vários aspectos da imagem visual para, por fim, identificar o que é esta imagem, tendo como bases as

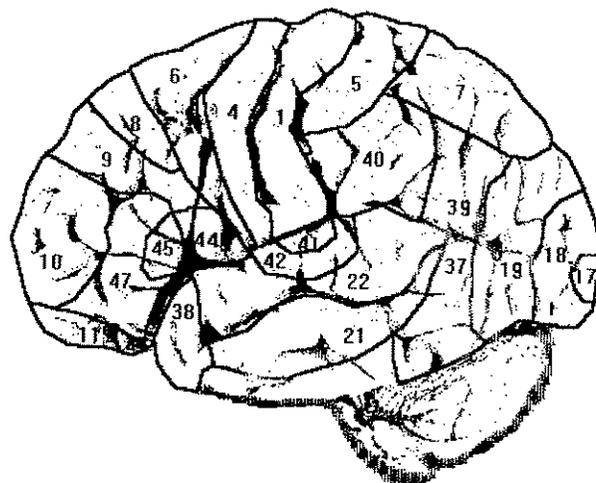


Figura 5.3: Mapa citoarquitetônico de Brodmann.

construções ou modelos cognitivos internos do cérebro.

Os neurônios do córtex visual dos mamíferos (V1) calculam relações espaciais e temporais particulares entre as respostas das células ganglionares da retina e representam estas relações de acordo com as respostas das características específicas. Alguns dos tipos de *características básicas* extraídas são a localização, orientação e polaridade dos gradientes de luminância; suas direções de movimento; suas composições espectrais; e sua disparidade interocular, que indica a distância do que se vê. Para que as características básicas possam ser extraídas, os sinais das células ganglionares retiniais devem ser correlacionados com um outro sinal; segundo Hubel e Wiesel [HuWi62], isto parece ser conseguido pela recombinação seletiva de entradas. Assim, para detectar e representar as descargas concomitantes das células ganglionares que respondem aos traços verticais de um objeto, as entradas das linhas verticalmente orientadas das células ganglionares convergem seletivamente em células corticais individuais (fig. 5.5) [Sin95]. Esta estratégia de recombinação de entradas é também usada para calcular outras relações que são analisadas à nível de V1, e as mesmas operações básicas estão integradas nas áreas corticais pré-estriadas. De acordo com a divergência substancial das projeções posteriores à V1 e pela especialização funcional de neurônios em áreas pré-estriadas, muitas destas operações parecem ser realizadas em paralelo, onde cada área calcula um sub-conjunto particular de relações de ordens mais altas no espaço de características.

Todavia, esta estratégia de recombinação de entradas e geração de células com propriedades de respostas seletivas não é indicada para descrições exaustivas, tanto de ca-

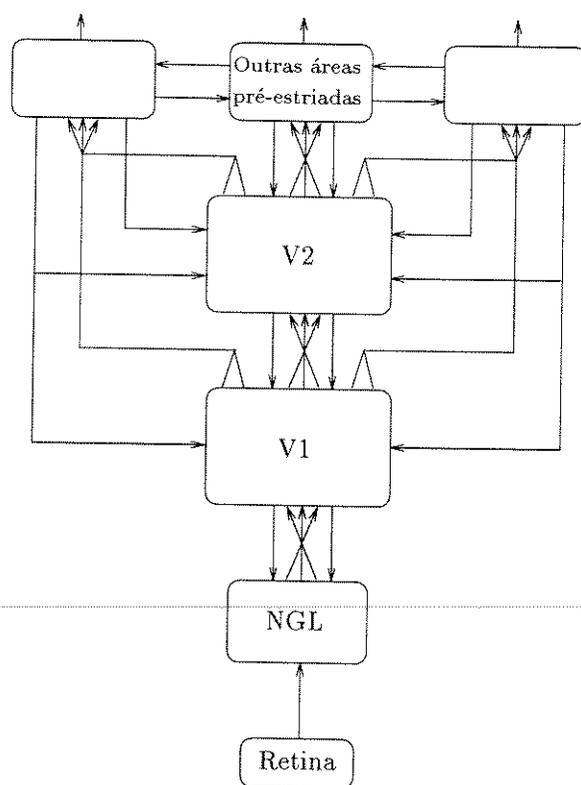


Figura 5.4: Conexões entre diferentes áreas do córtex visual.

racterísticas elementares representadas em V1, quanto dos conjuntos de características básicas de objetos naturais muito mais complexos. Em todos os estágios de processamento, os neurônios ficam bastante sensíveis às variações dos parâmetros dos estímulos ao longo das características de diferentes dimensões. As respostas das células individuais são ambíguas, e a descrição completa de uma característica básica particular é obtida apenas pelo cálculo conjunto das respostas graduadas de uma população de neurônios.

Enquanto os neurônios na área visual V1 respondem bem a bordas ou faixas de luz, os neurônios dos estágios de processamento posteriores representam aspectos cada vez mais complexos da imagem retinal. Estes neurônios nos estágios posteriores podem ser extremamente seletivos, respondendo somente a formas específicas, complexas, ou padrões de movimento, por exemplo. Deste modo, a visão é suportada por níveis de processamento cortical que cobrem, coletivamente, uma faixa de características do estímulo, de simples a complexo.

A criação de representações da imagem da retina é, entretanto, apenas um componente da visão. A visão é um processo ativo que seleciona uma parte limitada da imagem visual

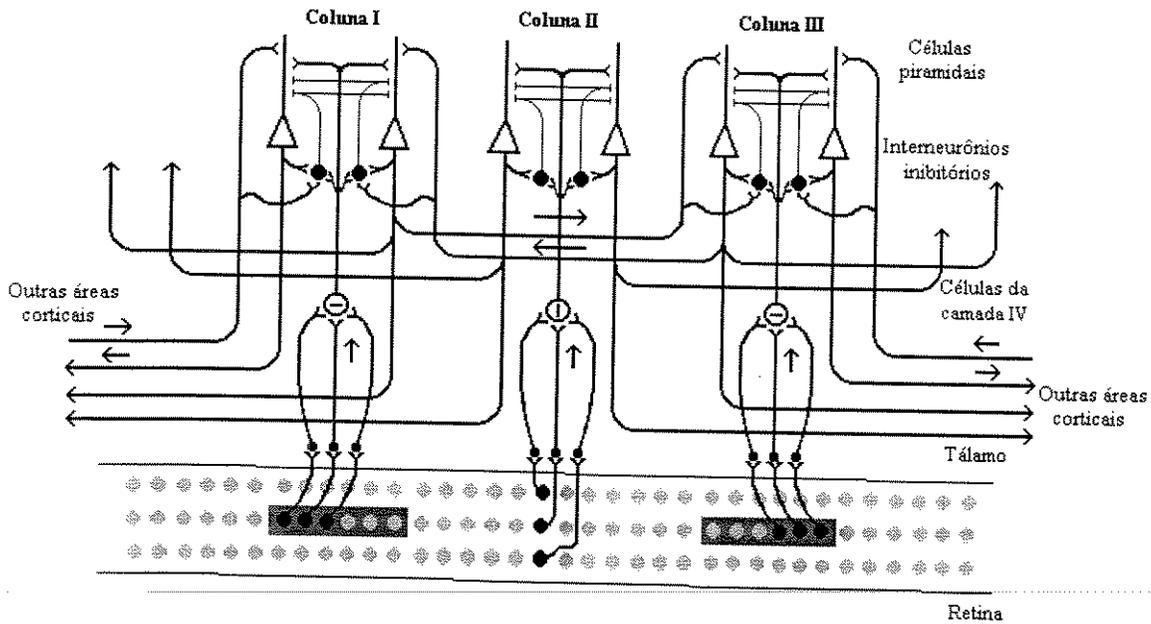


Figura 5.5: Conexões no córtex visual primário.

Uma suposição é que a especialização dos neurônios corticais para características particulares resulta da combinação específica de conexões progressivas convergentes, como exemplificado aqui por células sensíveis a contornos orientados horizontalmente (colunas I e III) e orientados verticalmente (coluna II). A saída das células ganglionares retiniais (círculos pretos na retina) que estão alinhadas em linhas horizontais (coluna I e III) ou linhas verticais (coluna II), respectivamente, converge após a substituição no tálamo por células corticais na camada IV. Por causa desta combinação específica de entradas, as células da camada IV adquirem campos receptivos seletivos à orientação, ajustados para as orientações vertical e horizontal. A saída das células da camada IV seletivas à orientação é, então, substituída por células piramidais em outras camadas verticais, e estas, por sua vez, projetam conexões progressivas para as áreas visuais pré-estriadas.

para a atenção concentrada. Embora porções não-selecionadas da imagem não sejam perdidas para a percepção, em um momento, podemos dar atenção completa somente a uma quantidade bastante limitada da informação visual. Uma vez que este sub-conjunto de sinais é selecionado, ele deve, então, ser interpretado. Assim, os eventos que guiam a atenção visual incluem um processo de edição substancial que não enfatiza as informações irrelevantes, mas acrescenta interpretações e inferências ao significado da informação em questão (foco de atenção).

Estudos recentes em macacos [Mau95] mostraram que esta edição dos sinais visuais começa nos estágios iniciais do processamento no córtex cerebral. O que o observador está

tentando ver e o que o observador conhece da cena visual têm considerável impacto no que é representado pelo córtex visual. Estes estudos mostram que a maior parte das áreas no córtex visual, mesmo aquelas em estágios relativamente iniciais do processamento, não dão um mesmo peso a todas as partes da imagem da retina. Ao contrário, elas representam preferencialmente aqueles elementos para os quais o observador está prestando atenção. Como muitos estudos examinaram a influência da atenção nas representações corticais, ela pode ser considerada como um fator cada vez mais importante na determinação de padrões de atividade no córtex visual.

Estas descobertas estão mudando o modo de ver o córtex visual. A opinião estabelecida de que as áreas corticais existem para extrair e representar várias características básicas sensoriais da imagem da retina não percebe um aspecto essencial das representações corticais. As diferenças entre os estágios iniciais e finais do processamento visual incluem não apenas as mudanças na complexidade das características dos estímulos que eles representam, mas também uma transição de representações reais da imagem visual para representações que realçam o objeto de interesse atual do observador.

O córtex é dividido em duas correntes de processamento sensorial, *parietal* e *temporal*, cada uma delas abrangendo áreas que representam diferentes tipos de informações sensoriais, contribuindo para diferentes tipos de comportamentos visuais. As áreas da corrente parietal analisam movimento e relações espaciais, ao passo que as áreas da corrente temporal estão mais envolvidas no reconhecimento e identificação visual.

Os neurônios da via parietal codificam também a posição e o movimento de alvos no espaço quando estes desaparecem de vista, ou seja, criam representações de estímulos não visíveis. Isto acontece quando o observador pode lembrar-se de estímulos visuais semelhantes remanescentes.

Os neurônios da via temporal são especializados em representar características básicas dos objetos visuais, tais como cor, textura e forma. Assim como a via parietal, a temporal contém neurônios que codificam informações sobre estímulos visuais que são lembrados de experiências anteriores (memória).

Estas funções das correntes (ou vias) parietal e temporal podem ser, então, conceitualizadas, respectivamente, como processos para determinação de “*onde*” e “*o que*” a visão percebe.

Em outras palavras, a visão está centrada em duas tarefas: detectar e localizar, e

discriminar (identificar). A detecção é considerada uma tarefa mais simples, pois o observador requer menos informações, enquanto que a discriminação acarreta alguns cálculos adicionais. Segundo Julesz [SaJu85], a detecção e a localização podem ser determinadas por um processo *paralelo*, enquanto os cálculos adicionais da discriminação são realizados por um processo *serial* que requer *atenção focal*.

Assim, a percepção visual envolve a descoberta e codificação de várias características na imagem, por vários grupos neuronais, atuando paralela e hierarquicamente.

5.3 O Sistema Visual Computacional

Os padrões são os meios pelos quais nós interpretamos o mundo. Ou seja, o que o nosso sistema visual observa no mundo externo é analisado e classificado, segundo nossas representações internas. Este processo, na área de computação, é conhecido como *reconhecimento de padrões*.

As diversas técnicas computacionais utilizadas para solucionar problemas de reconhecimento de padrão podem ser divididas em duas categorias: *estatística* e *sintática*. Na abordagem estatística, um conjunto de medidas básicas é extraído de uma amostra e o reconhecimento (classificação) é feito mediante o particionamento do seu espaço, em busca de características que representem padrões [DuHa73]. Este é um procedimento baseado na teoria de probabilidade e em estatísticas. Todavia, em alguns problemas de reconhecimento de padrão, o número de características necessárias para caracterizar um padrão pode ser muito grande, dependendo da complexidade dele, e torna-se impraticável reunir todas essas características em uma classe. Assim, uma alternativa para contornar esta dificuldade é descrever este padrão em uma composição hierarquizada de sub-padrões, segundo uma certa sintaxe, definida por uma gramática em uma linguagem. Esta é a abordagem sintática.

No reconhecimento sintático de padrões, um conjunto básico de padrões primitivos é selecionado para formar o conjunto de símbolos terminais da gramática. As produções da gramática são uma lista de relações permitidas entre as primitivas. As classes de padrões são um conjunto de cadeias geradas por uma gramática de padrões. Como um padrão pode conter uma descrição um pouco vaga, dotada de uma certa imprecisão e as linguagens formais possuem estruturas rígidas, adotamos uma linguagem nebulosa para representar

e classificar um padrão. Desta forma, um padrão particular descrito pela gramática nebulosa possui um certo grau de pertinência às classes de padrões estabelecidas, que é calculado por uma composição nebulosa, com vistas a identificar a quais classes ele pertence. É esta técnica que utilizaremos neste trabalho.

5.3.1 A Abordagem Sintática

A idéia da abordagem sintática é que dado um certo padrão, ele possa ser dividido em partes menores, cuja representação possa ser feita através de uma estrutura hierárquica, como uma estrutura tipo árvore, por exemplo.

Para representar uma informação hierárquica sobre um padrão, este deve ser descrito em termos de padrões mais simples, ou seja, em sub-padrões, e cada sub-padrão deve ser descrito em termos de sub-padrões mais simples ainda, e assim por diante, até que se encontre o sub-padrão mais simples possível, que é conhecido como *primitiva*. Esta abordagem esboça uma analogia entre uma estrutura hierarquizada de padrões com a sintaxe das linguagens naturais. Segundo este ponto de vista, os padrões seriam considerados *frases* ou *sentenças* construídas à partir da concatenação de *palavras*, e estas, por sua vez, seriam construídas à partir da concatenação de *letras* ou *caracteres*.

Façamos uma analogia entre uma sentença em linguagem natural e uma descrição sintática do padrão pela construção de módulos de interpretação para ambos.

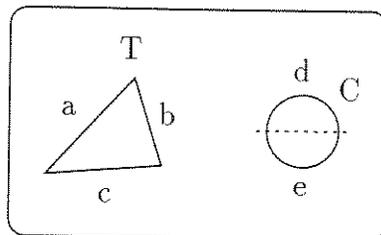


Figura 5.6: Exemplo de primitivas de uma cena.

Seja a sentença: “O bebê dormiu bastante” e a figura 5.6. Ambas podem ser interpretadas em estruturas da forma da fig. 5.7, chamadas *árvores de derivação*.

As regras sintáticas que reescrevem estas árvores de derivação são:

$$\langle S \rangle \rightarrow \langle SN \rangle \langle SV \rangle$$

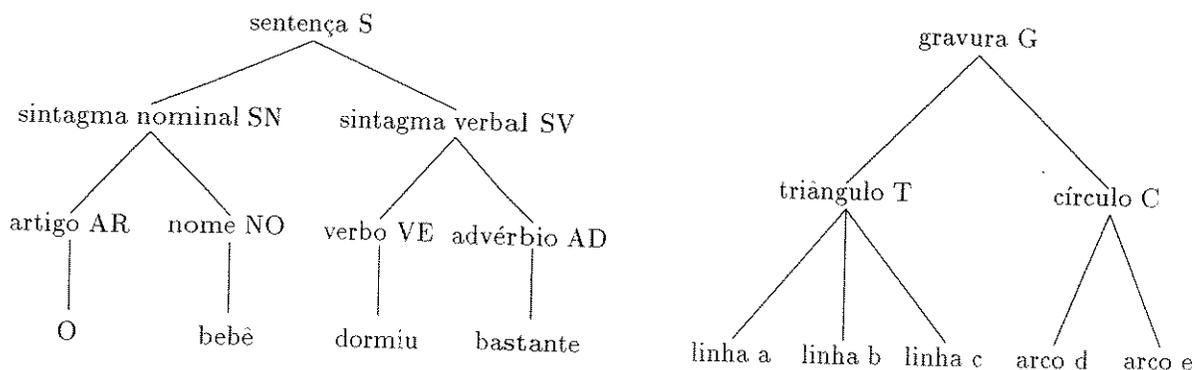


Figura 5.7: Árvores de derivação.

$\langle \text{SN} \rangle \rightarrow \langle \text{AR} \rangle \langle \text{NO} \rangle$

$\langle \text{SV} \rangle \rightarrow \langle \text{VE} \rangle \langle \text{AD} \rangle$

$\langle \text{AR} \rangle \rightarrow \text{O}$

$\langle \text{NO} \rangle \rightarrow \text{bebê}$

$\langle \text{VE} \rangle \rightarrow \text{dormiu}$

$\langle \text{AD} \rangle \rightarrow \text{bastante}$

$\langle \text{G} \rangle \rightarrow \langle \text{T} \rangle \langle \text{C} \rangle$

$\langle \text{T} \rangle \rightarrow \langle \text{a} \rangle \langle \text{b} \rangle \langle \text{c} \rangle$

$\langle \text{C} \rangle \rightarrow \langle \text{d} \rangle \langle \text{e} \rangle$

$\langle \text{a} \rangle \rightarrow \text{segmento de linha a}$

$\langle \text{b} \rangle \rightarrow \text{segmento de linha b}$

$\langle \text{c} \rangle \rightarrow \text{segmento de linha c}$

$\langle \text{d} \rangle \rightarrow \text{arco d}$

$\langle \text{e} \rangle \rightarrow \text{arco e}$

onde, o símbolo \rightarrow significa “pode ser escrito como”.

Desta forma, a linguagem que fornece uma descrição estrutural dos padrões segundo um conjunto de padrões primitivos, assim como suas operações de composição, é chamada de *linguagem de descrição de padrão*; as regras que regem a composição de padrões em sub-padrões, e destes em primitivas, são definidas por uma *gramática* da linguagem de descrição de padrão.

Após cada primitiva de um dado padrão ser identificada, o processo de reconhecimento é realizado através da análise da sintaxe da sentença que descreve este padrão, se ele é ou

não sintaticamente correto em relação a uma determinada gramática.

A abordagem sintática para o reconhecimento de padrões fornece uma capacidade para descrever um grande conjunto de padrões complexos através do uso de pequenos conjuntos de primitivas de padrões simples e regras gramaticais.

Um dos aspectos mais favoráveis à utilização da abordagem sintática é a capacidade recursiva de uma gramática. Uma regra de reescrita da gramática pode ser aplicada várias vezes, até que possa expressar de um modo bem compactado, algumas características estruturais básicas de um conjunto infinito de sentenças.

5.3.2 Modelo Clássico de um Sistema de Reconhecimento Sintático de Padrões

Um sistema de reconhecimento sintático de padrões, segundo Fu [Fu74] consiste de três partes, *pré-processamento*, *representação ou descrição do padrão* e *análise de sintaxe*.

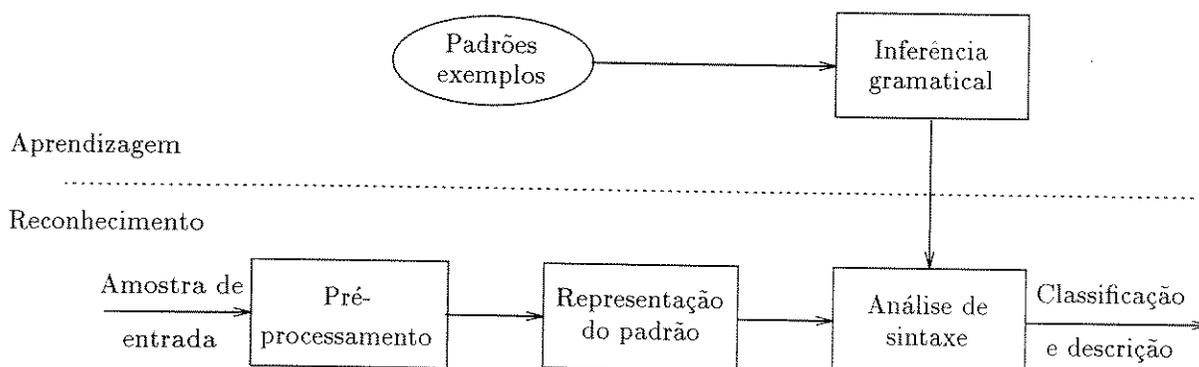


Figura 5.8: Sistema para o reconhecimento sintático de padrões.

As funções que fazem parte do pré-processamento são (i) a codificação e aproximação do padrão; e (ii) filtragem e restauração. Estas funções são as responsáveis pelo tratamento da amostra, de forma a deixá-la disponível para o processamento. A codificação e a aproximação trabalham a informação de entrada para torná-la mais compacta (compressão de dados) e as funções de filtragem e restauração possibilitam a recuperação de uma amostra degradada (por exemplo, com ruídos) ou que tenha perdido qualidade nas operações anteriores.

Em seguida, cada amostra pré-processada é então representada por uma linguagem es-

triturada, em uma cadeia. Este processo, chamado de representação do padrão, consiste da (i) segmentação da amostra, e (ii) extração de características básicas (primitivas). Para representar um padrão em termos de seus sub-padrões, o padrão deve ser segmentado e suas primitivas identificadas. Em outras palavras, cada padrão pré-processado é segmentado em sub-padrões e padrões primitivos baseados em uma sintaxe pré-especificada, dependendo do tipo de descrição desejada, e onde, por sua vez, cada sub-padrão pode ser identificado com um dado conjunto de primitivas. Neste ponto, cada padrão está representado por um conjunto de primitivas reunidos por operações sintáticas específicas.

Assim, o primeiro passo na formulação de um modelo sintático para a descrição do padrão é a determinação de um conjunto de primitivas, ou características básicas. Esta determinação é fortemente influenciada pela natureza dos dados, pelo tipo de aplicação e pela tecnologia disponível para a implantação do sistema.

O próximo passo é a construção de uma gramática (ou gramáticas) capaz de gerar uma linguagem (ou linguagens) para descrever os padrões em estudo. O ideal seria ter-se uma máquina de inferência gramatical que inferisse uma gramática de um conjunto de cadeias-exemplo dadas que descrevessem os padrões estudados. Contudo, muitas vezes, os projetistas constroem a gramática baseada em um conhecimento *a priori* disponível e na sua experiência, devido à complexidade que tal máquina poderia exigir.

O passo seguinte é projetar um analisador (reconhecedor) que reconhecerá os padrões (representados por cadeias) gerados pela gramática. Existem vários meios para se implementar o analisador. Uma abordagem mais rígida é construir uma gramática para cada classe de padrão. O analisador projetado para uma gramática particular é capaz de reconhecer apenas os padrões correspondentes a esta gramática. Assim, dado um conjunto de padrões e um padrão desconhecido, uma gramática pode reconhecer ou não este padrão não-conhecido. Uma abordagem mais flexível é construir-se um analisador que, dado um padrão de entrada, ele compara-o com outras cadeias prototípicas (ou de referência) de cada classe, por meio de suas primitivas. O padrão de entrada é reconhecido como pertencente à mesma classe da cadeia de referência que obtiver o melhor acoplamento. A solução para este problema de reconhecimento é chamada *análise de sintaxe*.

Segundo esta última possibilidade, as gramáticas ou regras de sintaxes são empregadas para descrever a sintaxe de linguagens ou as relações estruturais de padrões. Além da descrição estrutural, uma gramática pode ser usada para caracterizar uma fonte sintática que gera todas as sentenças (finitas ou não) em uma linguagem, ou dos padrões pertencentes

a uma classe particular. Para modelar uma linguagem ou descrever uma classe de padrões ou estrutura de dados mais realisticamente, uma gramática precisa ser diretamente inferida de um conjunto de sentenças-exemplo ou de um conjunto de padrões-exemplo. Este problema da aprendizagem de uma gramática baseada em um conjunto de sentenças-exemplo é chamado *inferência gramatical*. A principal dificuldade desta inferência está centrada sobretudo nos procedimentos utilizados para inferir as regras sintáticas de uma gramática desconhecida G , baseada em um conjunto finito de sentenças ou cadeias dadas de $L(G)$ e para reconhecer outras cadeias que sejam da mesma natureza que o conjunto dado.

A medida da qualidade da gramática inferida pode ser arbitrariamente definida, de modo que ela produza um resultado significativo.

Pelo que foi visto, este modelo possui etapas com funções bem definidas e pode ser implementado através de agentes distribuídos que executam as diversas tarefas dos módulos descritos aqui. A ferramenta que será usada para esta implementação é o **SENSOR**.

5.4 O Sistema **SENSOR**

SENSOR é um sistema sensorial inteligente distribuído que utiliza uma coleção de agentes especializados, inteligentes ou não, desenvolvidos para descrever, extrair características básicas, classificar dados sensoriais, e tomar decisões, de acordo, com análises efetuadas em cima de um dado domínio de conhecimento [RoSe94]. O **SENSOR** foi implementado dentro do ambiente KARDS© [ReRo95].

Um neurônio é um *agente primário* e a associação destes neurônios formando uma rede neural capaz de solucionar um dado problema, chamamos *agente complexo*. Esta é uma definição recursiva, assim um agente complexo pode ser considerado um agente primário por um outro agente mais complexo, e assim por diante. Em outras palavras, um agente é um neurônio ou uma coletânea de neurônios que se reúnem, e mantêm suas habilidades particulares, a fim de, juntos, tornarem-se aptos a realizar uma tarefa.

As informações que os agentes possuem, e trocam entre si, devem ser codificadas de maneira que, pelo menos, alguns outros agentes entendam o seu significado. Para explicar melhor como codificar um conjunto de informações neste sistema proposto, suponhamos um sistema qualquer, com duas variáveis correlacionáveis sendo observadas no tempo.

Sejam x e y estas duas variáveis, e P um conjunto de observações de x, y em diferentes instantes $i = 1, 2, \dots, n$:

$$P = [x_i, y_i \mid i = 1, n] \quad (5.1)$$

Seja P a descrição da trajetória do ponto \mathbf{p} no espaço XY . Muitas características básicas específicas de P podem ser descobertas pela análise das relações estabelecidas pelas posições do ponto no espaço. Estas características básicas fornecem as informações qualitativas sobre a trajetória do ponto no espaço.

Os agentes são usados para descobrir algumas características básicas destas trajetórias, tais como direção relativa, ângulo, e distância entre pontos vizinhos, tendência, periodicidade, etc., recodificando P em uma cadeia p de símbolos. Sempre que uma cadeia p contiver uma descrição simbólica de eventos em P , um vetor v pode ser criado contendo as coordenadas x, y associadas a cada elemento de p . As diferentes descrições simbólicas e vetoriais p, v de P serão geradas por diferentes agentes especializados:

$$D(P) = \{(p_1, v_1), (p_2, v_2), \dots, (p_n, v_n)\} \quad (5.2)$$

Seja

$$D(P_k) = \{(p'_1, v'_1), (p'_2, v'_2), \dots, (p'_n, v'_n)\} \quad (5.3)$$

a descrição de uma trajetória particular prototípica P_k usada na classificação, onde a forma da trajetória atual P é uma característica discriminante, onde p'_i é a cadeia fornecida por um agente especializado, descrevendo algumas características básicas dos pontos no espaço XY . Por exemplo, p'_1 poderia ser a descrição das tendências da trajetória descrita pelos pontos; p'_2 poderia ser a cadeia que descreve em quais pontos estas tendências são mudadas; \dots ; p'_n poderia fornecer informações sobre os ângulos associados aos pontos em p'_2 ; etc.

Resumindo, o conhecimento de um agente sobre parte de um problema ou o procedimento realizado por vários agentes na solução global deste problema pode ser representado através de uma descrição simbólica, numérica, ou ambas, no seu domínio de conhecimento.

Este domínio é codificado e interpretado por uma rede (ou grafo) ou conjunto de redes inter-relacionadas, cujos nodos são especializados em ações específicas para o tratamento do problema enfocado.

Para construir-se uma rede, o primeiro passo é ordenar os nodos pela *importância* ou pela *seqüência temporal* em cada informação deve ser obtida ou cada ação deve ser

realizada. Esta ordenação produz dois tipos distintos de redes, que expressam a solução do problema tratado: a ordenação por importância codifica uma estrutura *declarativa* e a ordenação temporal fornece uma estrutura *procedural* [Roc92].

Relembrando o modelo artificial do neurônio proposto por Rocha [Roc92], tem-se que no nível sináptico, um terminal pré-sináptico libera mensagens químicas chamadas transmissores que se ligam a moléculas pós-sinápticas denominadas receptores, e que por sua vez, ativam uma terceira molécula chamada controlador, capaz de modificar a atividade do neurônio pós-sináptico ou de células vizinhas. Esta abordagem permite que cada unidade neural possa realizar tanto processamento numérico quanto simbólico.

O neurônio capaz de executar estes dois tipos de processamento pode ser dividido em três partes com diferentes propriedades ou funções:

- *dendritos (entradas)* - são responsáveis pela recepção de mensagens provenientes de outros neurônios, para os quais são especializados;
- *corpo celular (computação)* - realiza o processamento de todas as mensagens que chegam ao neurônio;
- *axônio (saída)* - recodifica os resultados dos cálculos executados pelo corpo celular em mensagens liberadas para neurônios definidos (*sistema de correio*) ou transmitidas para uma população de neurônios que possuem os receptores adequados das mensagens (*sistema de difusão*).

As diferentes funções que os neurônios podem realizar são devidas ao controle gênico que equipa-os com diferentes tipos de transmissores, receptores, etc., envolvidos em diferentes processamentos bioquímicos.

Um nodo pode conter um neurônio, um agente ou uma outra rede. Isto significa que, dependendo da complexidade do problema, a complexidade da solução aumenta e precisa-se, cada vez mais, de níveis mais complexos. Assim, como tínhamos agentes primitivos e complexos, podemos ter agora redes primitivas e complexas.

Os nodos das redes são programados para realizar tanto cálculos numéricos quanto raciocínio simbólico, de acordo com as tarefas desejadas. Os tipos de tarefas são diferenciadas em quatro níveis gerais, nos quais para cada agente é atribuído métodos ou funções adequadas [Ser95]. Os níveis pré-definidos de uma rede de conhecimento qualquer, como mostrado na fig. 5.9, são:

- *nível de entrada* - os nodos recebem os dados a serem processados, é um nível sensorial;
- *nível associativo* - os nodos processam e integram as informações recebidas pelas entradas;
- *nível de decisão* - os nodos avaliam os resultados e tomam decisões; e
- *nível de saída* - os nodos executam ações determinadas ou sinalizam resultados.

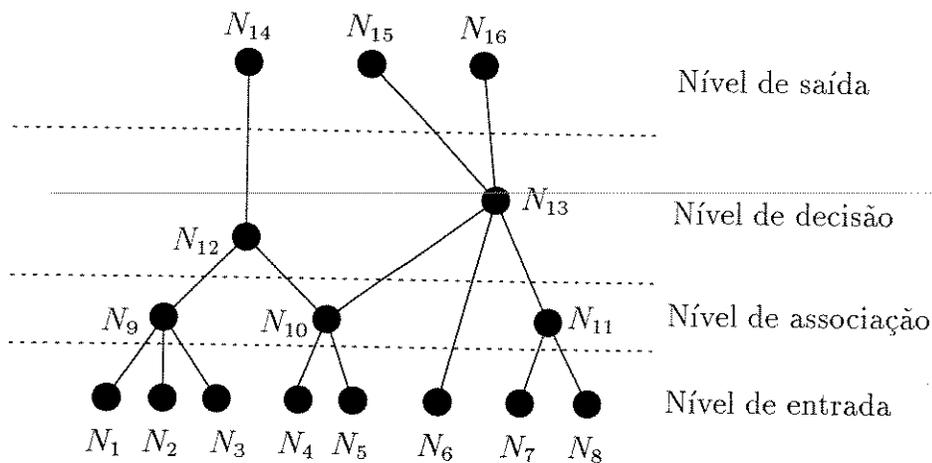


Figura 5.9: Exemplo de uma rede usada pelo SENSOR.

Cada nodo da rede contém, na verdade, as três mesmas propriedades definidas para o neurônio, ou seja, *entrada*, *computação* e *saída*. Novamente, cada uma destas propriedades pode estar associada a um agente, a uma rede, ou ser simplesmente um nodo funcionalmente especializado em alguma operação qualquer.

À nível cerebral, isto corresponde a várias redes neurais conectadas, formando níveis aptos a processar hierarquicamente uma informação. Exemplificando, quando o sistema visual recebe uma imagem, progressivamente ele a envia para as áreas superiores, as áreas associativas I, II, ..., VI, à medida em que vai sendo processada, até encontrar um significado para ela, ou então incluí-la como um novo padrão aprendido.

Esta rede permite o processamento de linguagens formais, visto que o neurônio é um agente que processa um sub-conjunto de processos definidos por uma genética, e que possui vias de condução que determinam o local de sua atuação. A rede organiza estes neurônios em unidades de complexidade crescente para processamento de cadeias de derivação.

Além disso, a rede definida por estes nodos é uma rede hierárquica que permite um raciocínio lógico formado por um conhecimento codificado em um conjunto de regras do tipo:

$$\begin{aligned} &\text{Se } X_1 \text{ é } A_1 \text{ e } Y_1 \text{ é } B_1 \text{ e } \dots \text{ e } W_1 \text{ é } C_1 \text{ então } Z_1 \text{ é } D_1 \text{ senão} \\ &\text{Se } X_2 \text{ é } A_2 \text{ e } Y_2 \text{ é } B_2 \text{ e } \dots \text{ e } W_2 \text{ é } C_2 \text{ então } Z_2 \text{ é } D_2 \text{ senão} \\ &\quad \vdots \\ &\text{Se } X_n \text{ é } A_n \text{ e } Y_n \text{ é } B_n \text{ e } \dots \text{ e } W_n \text{ é } C_n \text{ então } Z_n \text{ é } D_n. \end{aligned}$$

A navegação de uma é feita da esquerda para a direita e de baixo para cima. Primeiro, todos os nodos do nível de entrada que estão ligados a um mesmo nodo em um nível superior, formando um agrupamento (cluster), são processados, seguidos deste nodo. Sucessivamente, os clusters de ordem mais alta ainda são processados da mesma forma, desde que todos os clusters inferiores ligados a estes tenham sido percorridos. O nível de decisão fornece os resultados do processamento quando um nodo de saída é ativado. Como exemplo, tomemos a rede de conhecimento da figura 5.9. A seqüência da sua navegação é a seguinte: N_1 - N_2 - N_3 e N_9 (cluster), N_4 - N_5 e N_{10} (cluster), N_{12} (cluster). Dependendo da decisão de N_{12} , o próximo nodo é o N_{14} e é o fim do processamento, senão N_{13} deve ser processado. Mas N_{13} é um cluster também, então, antes dele, os nodos inferiores à ele ligados devem ser percorridos. A seqüência continua em N_6 , N_7 - N_8 e N_{11} (cluster), e por fim, N_{13} . O nodo N_{13} decide então, entre os nodos N_{15} e N_{16} e a navegação termina.

Capítulo 6

O Reconhecimento Sintático de Padrões

6.1 Introdução

Apresentamos, neste capítulo, o modelo que o **SENSOR** aplica no reconhecimento sintático de padrões, sendo que as funções das etapas de processamento foram inspirados nos neurônios do córtex cerebral para a implementação dos agentes.

Para que um computador possa identificar a forma de uma imagem que chega até ele, sugerimos que agentes especializados em diversas atividades sejam interligados, de modo que haja troca de informação entre eles, e a tarefas de reconhecimento de padrões e análise de cenas possam ser realizadas através da divisão destas tarefas em etapas distintas e por partes.

6.2 Modelo usado por **SENSOR** para o Reconhecimento Sintático de Padrões

SENSOR utiliza o mesmo esquema representado pela fig. 5.8 para o reconhecimento sintático de padrões, no entanto, nem todas as etapas descritas anteriormente possuem as mesmas funções. Aqui, apresentamos sumariamente os principais pontos considerados

na implementação do mesmo.

A diferença mais importante entre os dois modelos é que no modelo clássico tenta-se, à partir do padrão como um todo, encontrar as primitivas do padrão, e assim, realizar o reconhecimento ou classificação. No **SENSOR**, tal processo é o inverso. À partir das primitivas do padrão, reconstruímo-lo e buscamos efetuar o reconhecimento. Em outras palavras, segundo a fig. 5.7, o modelo clássico percorria a árvore de cima para baixo, já o **SENSOR** percorreria-a de baixo para cima.

A etapa de pré-processamento consiste em receber uma imagem no formato *bitmap* e extrair dela o contorno, segundo o procedimento descrito na próxima seção, de modo a obtermos um conjunto de pares ordenados no espaço bidimensional *XY*. Em seguida, estes pontos passam por uma etapa de compressão de dados, onde extrai-se os pontos significativos, ou seja, os pontos que são relevantes para descrever o contorno.

O padrão descrito pelos pontos significativos é, então, reescrito em uma cadeia de símbolos que representam a direção entre dois pontos consecutivos, que são as primitivas consideradas no processo de reconhecimento.

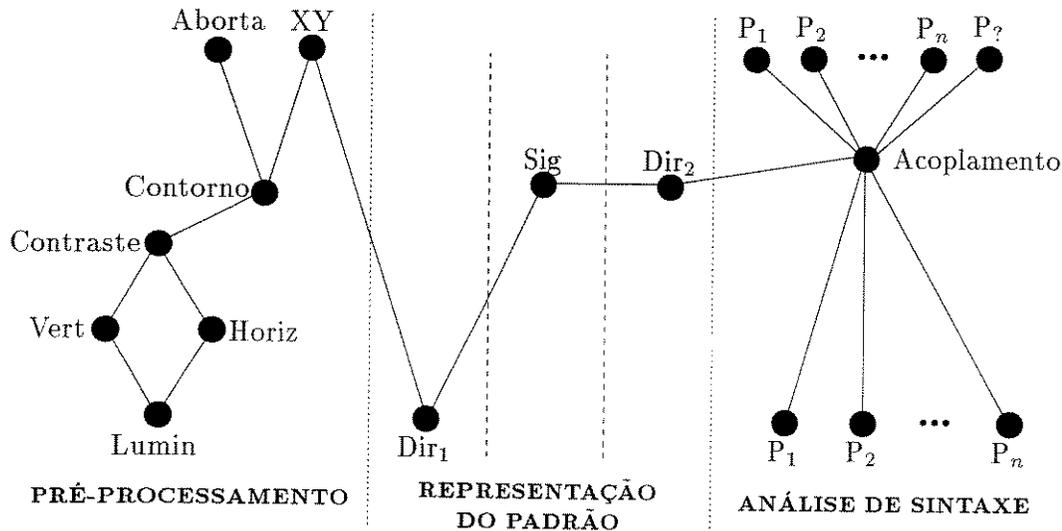


Figura 6.1: Rede para o reconhecimento de padrões pelo sistema **SENSOR**.

A cadeia de símbolos das direções é então comparada com um conjunto de cadeias prototípicas da mesma natureza, que foram previamente aprendidas de um conjunto de exemplos. A atribuição de uma classe ao padrão em observação é feita de acordo com o padrão aprendido que apresentar o maior grau de acoplamento ou dos padrões que apresentarem um grau de acoplamento maior que um dado limiar, ou seja, um padrão

observado pode ser referido à uma ou várias classes distintas.

A figura 6.1 mostra este procedimento descrito na forma de uma rede, que pode ser dividida nos três módulos: pré-processamento, representação do padrão e análise de sintaxe. A sua navegação começa com o pré-processamento, e é a seguinte: o nodo **Lumin** recebe a imagem, trata-a adequadamente e retira as informações sobre os graus de luminância. Os nodos **Vert** e **Horiz** encontram, respectivamente, as transições dos graus de luminância do claro para o escuro, e vice-versa, respectivamente, na vertical e na horizontal. O nodo **Contraste** agrega estas informações e libera-as para o nodo **Contorno** encontrar o contorno da imagem. Se ele não consegue realizar esta operação, o processamento termina (nodo **Aborta**). Caso contrário, disponibiliza estas informações para a etapa de representação de padrão, nodo **XY**, no formato de um conjunto de pares de pontos (x, y) . Nesta fase, o nodo **Dir₁** calcula a direção entre dois pontos consecutivos e **Sig** extrai os pontos significativos. Este, por sua vez, envia-os ao nodo **Dir₂**. Este nodo transforma os pontos significativos em uma outra cadeia de direções simbólicas, que será usada na etapa de análise de sintaxe. Já nesta etapa, o nodo **Acoplamento** recebe as cadeias de conjuntos de padrões previamente aprendidos (nodos **P₁**, **P₂**, ..., **P_n**), armazenados em uma base de conhecimento, e compara-os com o padrão fornecido por **Dir₂**. O nodo **Acoplamento** classifica-o, então, como pertencente a uma ou mais classes dos padrões de referência, e os nodos correspondentes a estas referidas classes são ativados no nível de saída, produzindo uma resposta ou ação. Se o padrão não for reconhecido, o nodo **P_?** é ativado, sinalizando o fato.

Como foi visto anteriormente, uma rede pode ser vista como uma rede neural, cujos nodos são os neurônios, e onde cada neurônio é composto por dendritos, corpo celular e axônio, de modo que cada um é funcionalmente especializado, podendo cada uma destas estruturas serem, por sua vez, também compostas por agentes especializados na execução de tarefas específicas.

6.3 Pré-processamento

A primeira etapa que o um sistema computacional deve realizar no reconhecimento de padrões é transformar a imagem que ele recebe em uma informação que possa ser trabalhada pelos módulos seguintes.

A rede neural da fig. 6.2 mostra como um problema de visão computacional, a extração de contornos de uma imagem, que é um tipo de pré-processamento da imagem, pode ser resolvido através da integração de vários agentes que simulam o comportamento visual biológico.

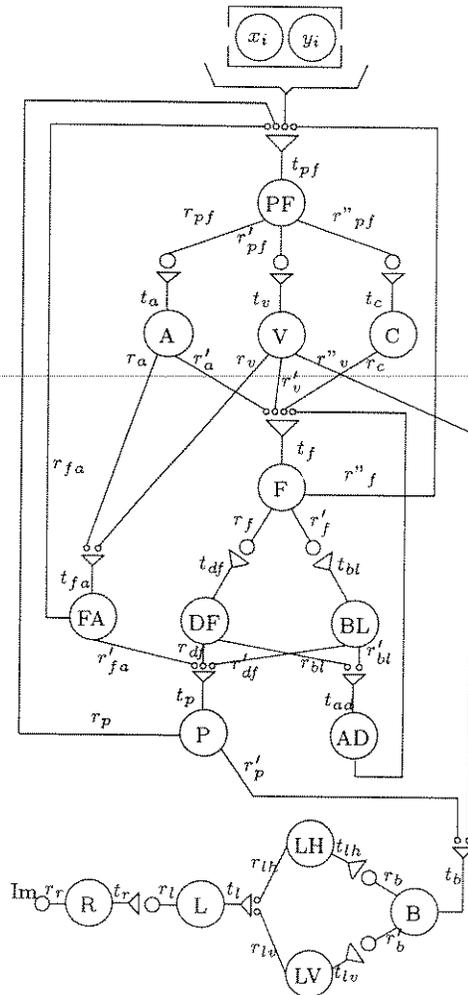


Figura 6.2: Rede neural com agentes distribuídos para visão.

A rede neural para este problema recebe uma imagem e divide-a em grades horizontais e verticais, formando uma matriz. Para cada posição desta matriz calcula-se a cor média e a luminância. Daí, procura-se em toda a matriz, os pontos consecutivos considerados como contraste, ou seja, os pontos em que a transição dos graus de luminância é abrupta. A quantidade de pontos obtida é influenciada pelas dimensões das grades escolhidas. Quanto menor a grade, maior o número de pontos. A imagem é, então, redesenhada, mantendo-se apenas estes pontos de contraste.

À partir do ponto mais extremo (foco) da parte superior da matriz, inicia-se a busca da seqüência dos pontos que determinam a forma da figura. Em torno deste ponto, a uma distância definida, procura-se um outro ponto (próximo foco) que dê continuidade ao seqüenciamento. Esta procura é feita da seguinte forma, a região de busca (fóvea) é dividida em 4 partes. Em cada uma destas partes, verifica-se qual possui o maior número de pontos (densidade). No entorno destas sub-regiões, também pesquisa-se a quantidade de pontos (blobs). Esta pesquisa é feita para tentar garantir que um dado segmento tenha um prolongamento na região seguinte. Estas duas informações possibilitam a escolha da próxima direção a ser seguida, dentre as 4 possíveis.

Depois da escolha da direção, verifica-se se os pontos pertencentes à região escolhida definem um seguimento contínuo, ou seja, sem interrupção. Caso esta condição seja satisfeita, escolhe-se o próximo foco como sendo o ponto mais distante do foco atual, e este procedimento se repete até que um foco seja igual ou esteja na vizinhança do primeiro foco escolhido.

Se alguma condição não é cumprida, os mesmos processos de verificação são realizados para as demais regiões, sempre em ordem decrescente, com base nas suas densidades de pontos. As sub-regiões da fóvea e o entorno devem possuir, pelo menos, um ponto.

Se ainda não for possível determinar o próximo foco após a verificação de todas as sub-regiões, aumenta-se, progressivamente, enquanto necessário, e até um certo limite, o tamanho da área da fóvea (ou campo visual), em um processo chamado *adaptação*, como tentativa para prosseguir com o seqüenciamento.

Como última tentativa, o próximo foco tenta ser encontrado, em combinações que desconsideram apenas uma eventual quebra de continuidade. Caso não seja suficiente, o sistema não é capaz de encontrar a seqüência de pontos.

Vale ressaltar, que o sistema tem o cuidado de verificar se uma possível direção de prosseguimento não é a última escolhida. Assim, tenta-se evitar que o seqüenciamento retorne pelo mesmo caminho por onde veio anteriormente.

Outro cuidado foi o de verificar se os focos entram em um encadeamento repetitivo dos pontos, o que impediria a continuação do processo de busca da seqüência dos limites da figura.

A seguir, apresentamos uma breve descrição dos agentes que compõem o sistema de

extração de contornos.

Primeira fase - “Onde”

- *Agente Retina* (R) - capta uma imagem em bitmap, divide-a em uma grade (reticulado), uma matriz $m \times n$, onde, para cada posição da matriz (célula), determina-se a cor média incidente, através da equação:

$$cor_{ij} = \frac{\sum_{k=1}^p pixel_k}{n} \quad (6.1)$$

onde, *pixel* é a cor de cada pixel pertencente à célula ij e p é o número de pixels existentes na célula.

A seguir, para a cor média de cada célula extrai-se os graus de pigmentação (quantidade) vermelho, verde ou azul.

Este agente funciona como um filtro para as três cores básicas.

A figura 6.3 exemplifica este processo.

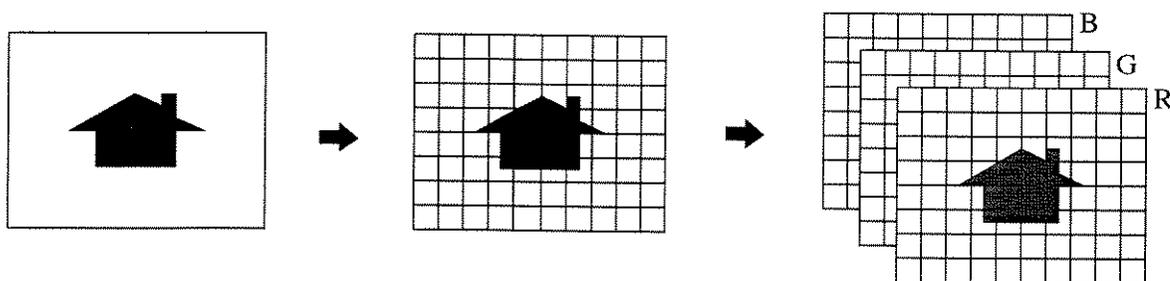


Figura 6.3: Função do agente Retina.

- *Agente Luminância* (L) - transforma as informações sobre cores em graus de luminância (Y), onde os graus são símbolos nebulosos que significam se o ponto é branco (b), claro (c), médio (m), escuro (e) ou preto (p). Esta transformação é feita segundo a *Terceira Lei de Groissmann*, dada a seguir:

$$Y = 0,30 \times R + 0,59 \times G + 0,11 \times B \quad (6.2)$$

onde: Y é um número real no intervalo $[0, 1]$, e R , G e B são, respectivamente, as quantidades de vermelho, verde e azul.

A função que codifica os graus de luminância em símbolos nebulosos é a função representada pela figura 6.4.

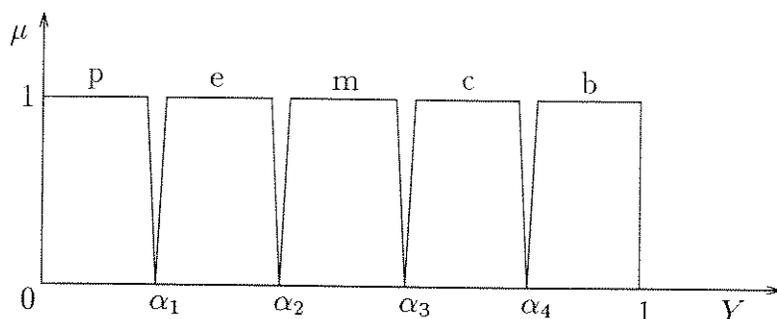


Figura 6.4: Função de codificação dos graus de luminância.

- *Agentes Linha Horizontal (LH) e Linha Vertical (LV)* - detectam, respectivamente, os contrastes existentes nas linhas horizontais e verticais da imagem segmentada. Dado um dicionário D que contém as combinações entre símbolos (seqüências) consideradas como contraste, como por exemplo, $D = \{pb, bp, ce, \dots, eb\}$, definem os pontos que determinam o contorno da imagem, como mostrado pela fig. 6.5.

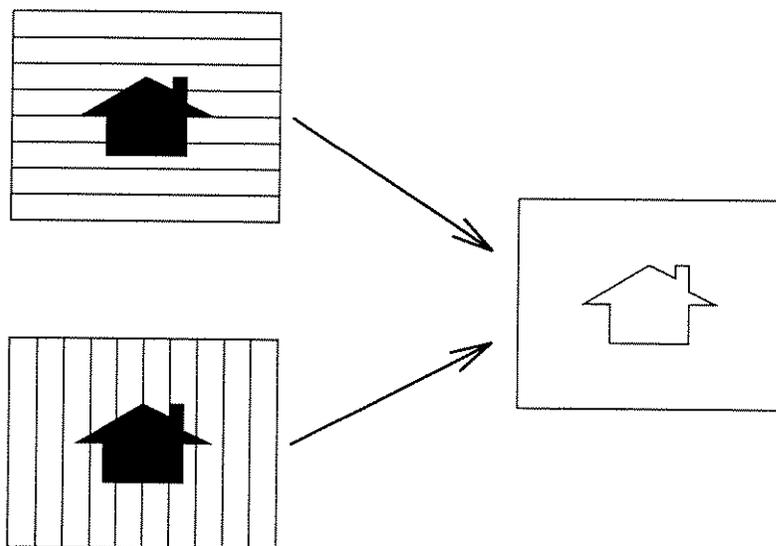


Figura 6.5: Função dos Agentes Linha Horizontal e Linha Vertical.

- *Agente Borda (B)* - reescreve a imagem em uma grade igual ao do agente Retina, considerando apenas as informações (pontos) referentes aos contrastes da imagem, ou seja, refaz (redesenha) o mapa da imagem, considerando apenas as posições que marcam o contraste.

Até aqui, o sistema visual sabe apenas que vê alguma coisa (detecção), mas ainda não discriminou o que ele vê. Este problema de identificar a forma do objeto em uma cena é

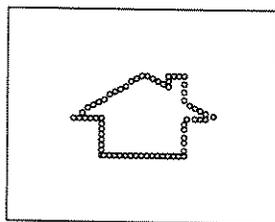


Figura 6.6: Função do Agente Borda.

resolvido pela movimentação dos olhos ao seu redor, focalizando a atenção em diferentes partes do objeto em diferentes tempos, seqüencialmente.

Segunda fase - “O que”

- *Agente Foco (P)* - fixa um ponto para iniciar ou seguir o seqüenciamento do contorno da imagem, ou seja, marca uma posição como referência.
- *Agente Foco Anterior (FA)* - guarda a informação de qual foi o último foco (ponto) escolhido, para comparações futuras. Atua como um dispositivo de memória.

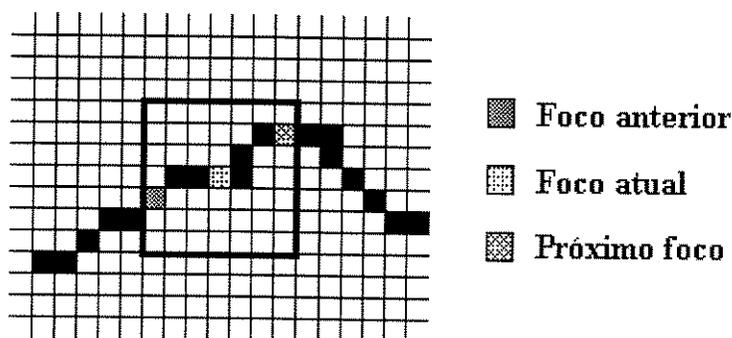


Figura 6.7: Focos dentro do campo visual.

- *Agente Fóvea (F)* - dado um foco, analisa qual a melhor direção a prosseguir, dentro de um campo visual (distância máxima de observação em torno do foco).

O campo visual é dividido em quatro regiões distintas, conforme fig. 6.8. Os agentes *Densidade da Fóvea* e *Blob*, dentre outros, fornecem informações sobre cada uma destas quatro regiões ao agente *Fóvea*, a fim de que ele possa reunir dados para determinar em qual destas regiões se encontra o próximo foco.

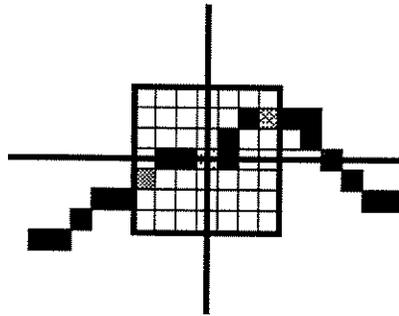


Figura 6.8: Atuação do agente Fóvea.

Os agentes *Anterior*, *Vizinhança* e *Continuidade* são os outros agentes que fornecem as outras informações para o *Fóvea*.

- *Agente Densidade da Fóvea* (DF) - calcula a densidade de pontos existentes na região da fóvea.

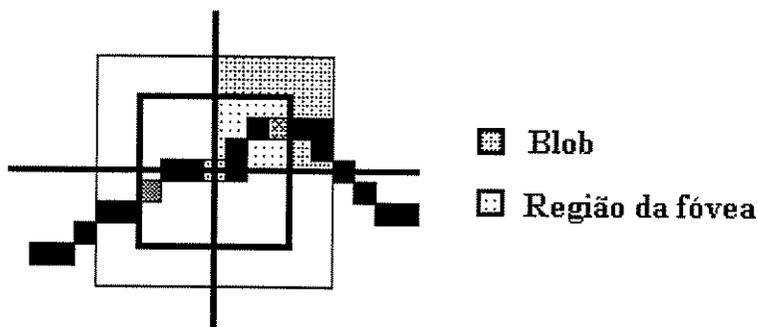


Figura 6.9: Áreas de cálculo das densidades da fóvea e blob.

- *Agente Blob* (BL) - calcula a densidade de pontos existentes na região da vizinhança da fóvea.
- *Agente Anterior* (A) - verifica se a observação está sendo feita na área que contém o foco anterior, e se não está entrando em um loop.
- *Agente Vizinhança* (V) - verifica se a observação está sendo feita em uma área vizinha à área que contém o foco anterior.
- *Agente Continuidade* (C) - verifica se, dentro da área inspecionada, existe quebra de continuidade entre os pontos destacados.

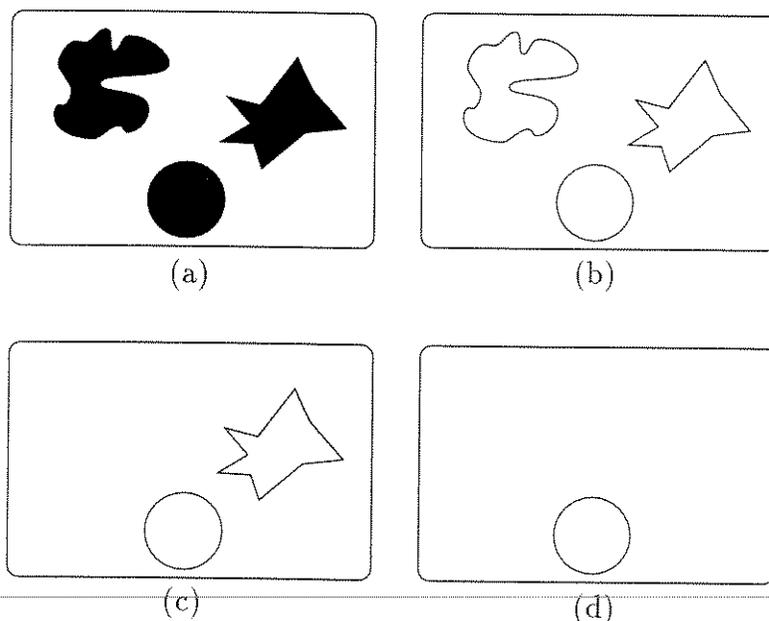


Figura 6.11: Segmentação de uma imagem.

- repetir os passos 2 e 3 até que todos os componentes sejam extraídos ou até que não seja mais possível.

Este processo, contudo, apresenta alguns inconvenientes. Como foi visto na seção anterior, o número de pontos que definem um contraste depende das dimensões do reticulado escolhido. Quanto maior o número de linhas e colunas definidos pelo agente Retina, maior vai ser o número de pontos de contraste na imagem. Se os elementos de uma imagem forem muito complexos ou se os tamanhos dos elementos forem bem desiguais, estes elementos podem ser fortemente influenciados pelas dimensões do reticulado, e uma boa grade para obter-se o contorno de um elemento pode não ser suficiente para extrair-se o contorno de um outro, e o processamento é interrompido. Além disso, a imagem deve ser o mais limpa possível, pois se houver algum ponto considerado como contraste, que não faz parte de nenhum elemento, dependendo da sua posição, ele inviabilizará a segmentação completa da imagem.

Apesar disso, este procedimento traz vantagens no que diz respeito ao tratamento de cenas, pois além de fornecer os contornos, fornece a posição espacial que um elemento tem em relação ao outro, caso seja necessário utilizar-se esta relação.

6.4 Representação do Padrão

Após os dados referentes à imagem estarem disponíveis, eles devem ser reescritos sintaticamente para ser reconhecido. Diversos tipos de características básicas podem ser representadas para este fim, mas utilizamos aqui a transformação dos pontos do contorno em uma cadeia de símbolos que define a direção entre dois pontos consecutivos.

A seguir, mostramos um agente capaz de detectar a direção (fig. 6.12) entre dois pontos no espaço (agente “Direção”), formado à partir de uma rede neural artificial simbólica. Este é um chamado agente primário, pois através de sua interação com outros neurônios ou agentes, gera agentes mais complexos. O tipo de troca de mensagens entre os neurônios usada para a interação entre os agentes, *correio*, é implementada, por exemplo, pelo agente complexo “Pontos Significativos”, que é formado à partir deste agente primário.

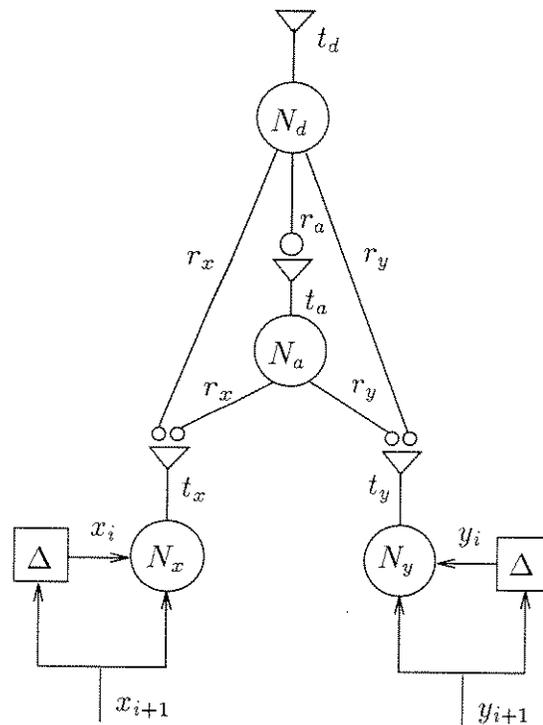


Figura 6.12: Agente primário “Direção”.

A função do agente “Direção” é determinar a posição do ponto (x_{i+1}, y_{i+1}) em relação ao ponto anterior (x_i, y_i) , de acordo com uma das 16 posições possíveis da fig. 6.13. A rede neural que implementa o agente “Direção” é uma rede composta por dois neurônios sensoriais N_x e N_y , um neurônio N_a e um neurônio de saída N_d . Os neurônios sensoriais

recebem x_{i+1} e y_{i+1} como entradas, cujos limiares são fornecidos pelos elementos de atraso Δ . Estes elementos retêm as coordenadas anteriores x_i e y_i (os limiares). Os axônios dos neurônios sensoriais realizam uma codificação dos sinais de entrada, de acordo com os seus limiares, e então liberam uma certa quantidade m_x e m_y do transmissor correspondente t_u ($u = x$ ou y), segundo a função de filtragem axônica a seguir:

$$t_u = \begin{cases} n, & \text{se } u_{i+1} < u_i \\ p, & \text{caso contrário} \end{cases} \quad (6.3)$$

As quantidades de transmissores m_x e m_y são sempre iguais às quantidades totais de receptores do neurônio pós-sináptico N_a . Quando os transmissores t_x e t_y se acoplam aos receptores r_x e r_y , respectivamente, um transmissor t_a é ativado, de acordo com a seguinte função:

$$t_a = \begin{cases} a, & \text{se } m_y/m_x < \alpha_1 \\ b, & \text{se } \alpha_1 \leq m_y/m_x < \alpha_2 \\ c, & \text{se } \alpha_2 \leq m_y/m_x < \alpha_3 \\ d, & \text{se } \alpha_3 \leq m_y/m_x < \alpha_4 \\ e, & \text{se } m_y/m_x \geq \alpha_4 \end{cases} \quad (6.4)$$

O neurônio N_d possui receptores para os transmissores dos neurônios N_x , N_y e N_a . Os receptores ativados com o acoplamento dos transmissores t_x , t_y e t_a liberam, por sua vez, o transmissor t_d , que corresponde à direção.

A posição de (x_{i+1}, y_{i+1}) em relação a (x_i, y_i) , é mapeada em uma dentre as 16 possíveis direções, conforme a fig. 6.13. A gramática usada para escolher a direção é a seguinte:

$$G_{N_d} = \{V_s, V_n, V_t, \mathcal{P}, \eta\}$$

$$V_n = \{p, n, a, b, c, d, e\}$$

$$V_t = \{A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P\}$$

$$P = \left\{ \begin{array}{cccc} p * a \rightarrow P & ppb \rightarrow A & ppc \rightarrow B & ppd \rightarrow C \\ *pe \rightarrow D & npd \rightarrow E & npc \rightarrow F & npb \rightarrow G \\ n * a \rightarrow H & nnb \rightarrow I & nnc \rightarrow J & nnd \rightarrow K \\ *ne \rightarrow L & pnd \rightarrow M & pnc \rightarrow N & pnb \rightarrow O \end{array} \right\} \quad (6.5)$$

onde $*$ é o símbolo de não-relevante.

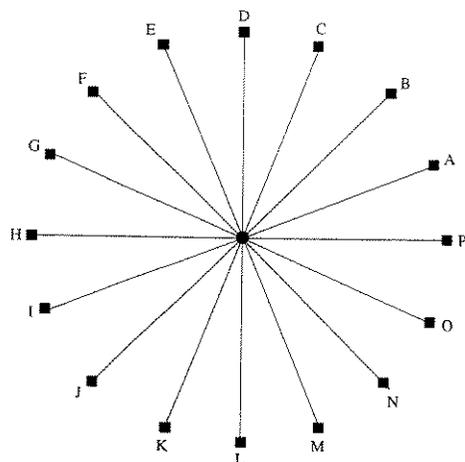


Figura 6.13: Direções possíveis.

À partir do agente primário “Direção” podemos construir agentes mais complexos para realizarem diferentes tarefas. Os agentes complexos são formados através da comunicação deste com diversos outros agentes.

Um exemplo de agente complexo formado à partir do agente “Direção” é o agente “Pontos Significativos” [RvZR93]. A tarefa dele é comprimir um conjunto de dados em um espaço bidimensional. Esta compressão considera que, em um conjunto de pontos bidimensionais, os pontos que são relevantes neste conjunto são somente aqueles que determinam uma mudança de direção na trajetória descrita por eles. Além do agente “Direção”, ele é formado também pelo agente “Extraí Significativos”. A estrutura do agente “Pontos Significativos” é hierárquica, ou seja, quando um agente termina uma tarefa ele deve enviar sua mensagem para um outro determinado agente. Portanto, o protocolo de comunicação inter-agentes é o sistema de correio.

De acordo com a fig. 6.14, o neurônio N_d do agente “Direção” libera o seu transmissor para o agente “Extraí Significativos”. Este agente é composto pelos neurônios: N_{vd} , N_c , ($j = 1, 2, 3$) e por cinco pares de neurônios (N_{x_l}, N_{y_l}) ($l = i - 3, i - 2, \dots, i + 1$).

O papel do neurônio N_{vd} é apenas calcular a variação entre duas direções e liberar para o agente seguinte um transmissor.

Quando o transmissor t_d se acopla ao receptor r_{vd} do neurônio N_{vd} , ele ativa um controlador c_{vd} que atua sobre o próprio neurônio, fazendo-o liberar um determinado transmissor t_{vd} . O receptor r_{vd} tem armazenado como informação a última direção d_i

enviada pelo transmissor t_d . Este acoplamento é representado da seguinte forma:

$$t_d + r_{vd} \mapsto c_{vd} \Rightarrow t_{vd} \quad (6.6)$$

$$t_{vd} = \begin{cases} z, & \text{se } t_d = r_{vd} \\ n, & \text{se } \| t_d - r_{vd} \| \geq 8 \\ p, & \text{caso contrário} \end{cases} \quad (6.7)$$

onde:

z → sem variação de direção;

n → variação horária;

p → variação anti-horária e

$\| \|$ é o módulo da diferença entre as posições das direções no espaço.

O transmissor t_{vd} se liga, sincronamente, aos receptores N_{c_j} , com um acoplamento apenas à cada evento. O primeiro acoplamento é realizado junto ao receptor r_{c_1} , depois ao receptor r_{c_2} , e por fim, ao receptor r_{c_3} . Em cada evento, um controlador c_{c_1} é ativado para transferir o transmissor de N_{c_2} para N_{c_3} e de N_{c_1} para N_{c_2} , ou seja atuar sobre neurônios vizinhos. Este processo codifica as três últimas variações de direção nos receptores r_{c_1} , r_{c_2} e r_{c_3} . Cada neurônio N_{c_j} é especializado no reconhecimento de uma classe dos possíveis padrões de mudança de direção (e que determinam um ponto significativo), de modo que apenas um pode estar ativo a cada tempo. Quando nenhum neurônio N_{c_j} estiver ativo, significa que não existe ponto significativo. Se um neurônio N_{c_j} está ativo, então o transmissor t_{c_j} liberado é de um tipo, caso contrário, ele libera um outro transmissor. Esta codificação do tipo de transmissor liberado é feita pela seqüência dos transmissores liberados pelo neurônios N_{vd} , na ordem de eventos t_1, t_2, t_3 . Assim, as funções de ativação dos neurônios N_{c_j} são:

$$\begin{aligned} t_{c_1} &= \begin{cases} S, & \text{se } t_k \text{'s} = zpp, npp, znn \text{ ou } pnn \\ N, & \text{caso contrário} \end{cases} \\ t_{c_2} &= \begin{cases} S, & \text{se } t_k \text{'s} = zpz \text{ ou } znz \\ N, & \text{caso contrário} \end{cases} \\ t_{c_3} &= \begin{cases} S, & \text{se } t_k \text{'s} = pzp, nzn, zzp \text{ ou } zzn \\ N, & \text{caso contrário} \end{cases} \end{aligned} \quad (6.8)$$

onde, S significa que o neurônio está ativo e N que está inativo.

A tarefa dos cinco pares de neurônios (N_{x_l}, N_{y_l}) é atuar como um nível de memória, armazenando os últimos cinco pares de entrada (x_l, y_l) .

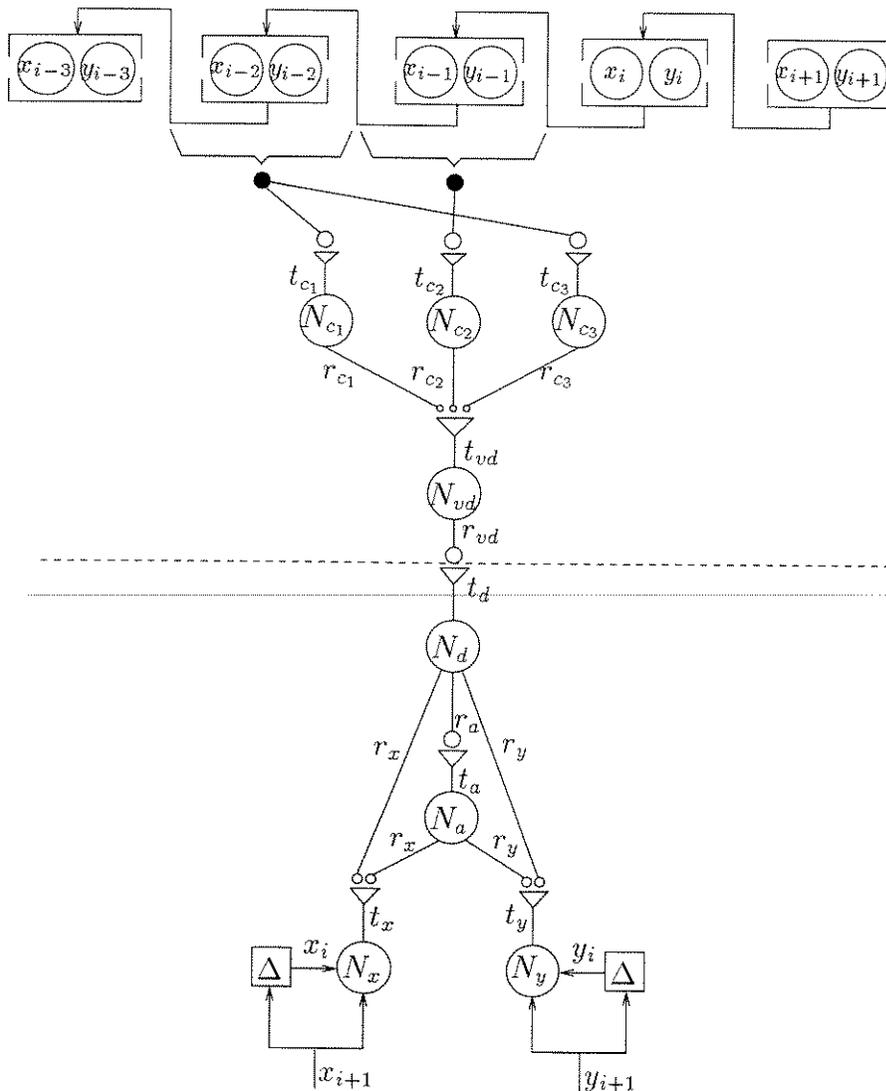


Figura 6.14: Agente “Pontos Significativos”.

As conexões entre os neurônios N_{c_j} e (N_{x_l}, N_{y_l}) determinam qual é a posição do ponto significativo associado a cada classe de padrões estabelecidas pelos neurônios N_{c_j} . De acordo com as conexões, se o neurônio N_{c_1} ou N_{c_3} estiver ativo, então o ponto significativo é o ponto (x_{i-2}, y_{i-2}) . Caso o neurônio ativo seja N_{c_2} , o ponto significativo é (x_{i-1}, y_{i-1}) .

Este processo de codificação baseia-se na definição do grau de afinidade $\mu(t, r)$ do acoplamento (t/r) entre os transmissores dos neurônios pré-sinápticos N_{c_j} com os receptores dos neurônios pós-sinápticos (N_{x_l}, N_{y_l}) . Por definição, temos:

$$\mu(t_{c_1}, r_{x_l}) = \mu(t_{c_1}, r_{y_l}) = \begin{cases} 1, & \text{se } l = i - 2 \\ 0, & \text{caso contrário} \end{cases}$$

$$\mu(t_{c_2}, r_{x_l}) = \mu(t_{c_2}, r_{y_l}) = \begin{cases} 1, & \text{se } l = i - 1 \\ 0, & \text{caso contrário} \end{cases} \quad (6.9)$$

$$\mu(t_{c_3}, r_{x_l}) = \mu(t_{c_3}, r_{y_l}) = \begin{cases} 1, & \text{se } l = i - 2 \\ 0, & \text{caso contrário} \end{cases}$$

6.5 Análise de Sintaxe

Como última etapa, a cadeia que contém a descrição de um padrão deve ser comparada com as cadeias de padrões prototípicos, aprendidos pela extração de características básicas comuns à vários exemplos correspondentes à esta classe, e então, ela é classificada segundo a sua similaridade com estas cadeias de padrões prototípicos.

Apresentamos a seguir, os algoritmos de comparação de cadeias (*algoritmo de acoplamento nebuloso*) e de aprendizagem dos padrões (*algoritmo de aprendizagem por molde*).

6.5.1 Algoritmo de Acoplamento Nebuloso

O algoritmo de acoplamento nebuloso é um algoritmo que permite que um padrão seja comparado com outro, no qual um grau de similaridade entre eles é atribuído, e onde cada padrão é definido como uma cadeia ordenada de símbolos [SRRP96]. Este algoritmo foi implementado segundo os operadores das linguagens formais nebulosas e é descrito a seguir:

- a) tomar uma cadeia inicial s_j e uma cadeia de comparação s_k ;
- b) definir uma vizinhança nebulosa N_c para cada caracter c_c de ordem c de s_j , com o seu intervalo (l_i, l_s) , tal que qualquer caracter de $c_{c'}$ de s_k , dentro do intervalo $l_i < c' < l_s$ pode ser comparado ao caracter c_c ;
- c) definir o conjunto ordenado M_c de caracteres de acoplamento c_m considerado equivalente à c_c ;
- d) encontrar para cada caracter c_c de s_j o melhor acoplamento do caracter $c_{c'}$ de s_k restringido pelos itens b e c anteriores, e calculado como:

$$\mu(c_c, c_{c'}) = \max_{c'=l_i}^{l_s} (\mu(c_m, c_{c'}) \times f(c')) \quad (6.10)$$

onde, $f(c')$ é a função de pertinência que define N_c ;

- e) o grau de acoplamento atual $\mu(s_j, s_k)$ entre as cadeias s_j e s_k deve ser calculado da seguinte maneira:

$$\mu(s_j, s_k) = \frac{\sum_{c=1}^n \mu(c_c, c_{c'})}{n} \quad (6.11)$$

ou seja, é a média do grau de acoplamento calculado para todos os n caracteres de s_j .

Um acoplamento parcial é definido se algum caracter c_c de s_j não for encontrado em s_k . Neste caso, $\mu(s_j, s_k)$ é calculado como a média dos m caracteres c_c de s_j para os quais $\mu(c_c, c_{c'}) > 0$, e o acoplamento parcial é sinalizado pela atribuição de um valor negativo a $\mu(s_j, s_k)$.

6.5.2 Algoritmo de Aprendizagem

A aprendizagem de um conjunto de cadeias prototípicas p_1 a p_n é feita pela descoberta de símbolos comuns que descrevem um conjunto de exemplos E de formas a serem reconhecidas. Existem diferentes estratégias para aprender estas características básicas comuns. Neste trabalho, utilizamos um processo que toma uma cadeia inicial e_1 de E como molde e muda sua estrutura dependendo da sua similaridade com as outras cadeias de E . Esta estratégia é chamada *algoritmo de aprendizagem por molde*. Este algoritmo também foi implementado segundo a teoria das linguagens formais nebulosas.

O algoritmo de aprendizagem por molde é definido como segue:

- a) tomar um dicionário

$$D = \{c_1, \dots, c_n\} \quad (6.12)$$

de caracteres que compõem as cadeias de exemplo a serem alfabeticamente ordenadas, tais que c_i precede c_j se $i < j$;

- b) fornecer o intervalo (l_a, l_z) para restringir a aprendizagem de N_c , tal que seu intervalo aprendido (l_i, l_s) seja restrito a $l_i > l_a$ e $l_s < l_z$;
- c) fornecer o intervalo (l_t, l_u) para restringir a aprendizagem de M_c , tal que seus caracteres aprendidos c_m sejam restringidos a $l_t < c - m < l_u$. Deste modo, cada M_c aprendido é composto por símbolos no intervalo (m_i, m_s) ;

- d) tomar as primeiras duas cadeias de E como a cadeia de molde s_t e a cadeia de modificação s_m ;
- e) encontrar para cada caracter c_c de s_t o melhor acoplamento do caracter $c_{c'}$ de s_m restringido pelos ítems b e c anteriores, e
- f) dado que o caracter atual $c_{c'}$ foi acoplado na posição p , ajustar os limites atuais, (l_i, l_s) e (m_i, m_s) de acordo com

$$\begin{aligned} l_i &= \min_{j=1}^n(\max(p_j, l_a)) \\ l_s &= \max_{j=1}^n(\min(p_j, l_z)) \end{aligned} \quad (6.13)$$

$$\begin{aligned} m_i &= \min_{j=1}^n(\max(m_j, l_t)) \\ m_s &= \max_{j=1}^n(\min(m_j, l_u)) \end{aligned} \quad (6.14)$$

onde n é o número de caracteres acoplados em cada um dos intervalos definidos.

- g) substituir s_t pela cadeia composta por todos os caracteres c_c de s_t para os quais $\mu(c_c, c_{c'}) > 0$;
- h) tomar a próxima cadeia em E como a nova s_m e repetir os passos f, g e h, e
- i) considerar a cadeia final s_t como a cadeia prototípica que codifica a solução do problema.

Aprender uma cadeia de padrão prototípico p_j não é apenas aprender os seus caracteres significativos, mas também o conjunto associado M_c e a vizinhança N_c .

Este algoritmo é altamente dependente da ordem em que o conjunto de exemplos é apresentado. Se porventura, existirem padrões que possuam uma similaridade bastante reduzida entre si em um conjunto de exemplos, a cadeia final aprendida poderá ser um padrão que não traduz corretamente as características básicas comuns entre os padrões. Quanto antes estes padrões aparecerem na lista de encadeamento, pior será o resultado final. Portanto, várias estratégias podem ser adotadas para buscar-se um melhor desempenho do algoritmo. Por exemplo, dada a cadeia inicial e_1 e a lista de exemplos s_j ($j = 1, n$), pode-se calcular, individualmente, o grau de acoplamento entre a cadeia inicial e um padrão, e em seguida, um especialista decide, baseado na comparação destes

dados, quais os padrões que formarão o conjunto de treinamento, bem como a ordem de apresentação dos mesmos. Um outro procedimento possível é que, ao serem realizadas as comparações individuais, o conjunto escolhido seja formado automaticamente pelos padrões que possuem um grau de acoplamento maior que um dado limiar, isto é

$$\mu(e_1, s_j) > \alpha \quad (6.15)$$

Estes são alternativas que constituem uma *aprendizagem supervisionada*.

Capítulo 7

Resultados

7.1 Introdução

Este capítulo apresenta exemplos de como o sistema **SENSOR** retira, organiza e utiliza as informações que recebe para a aprendizagem e reconhecimento de padrões.

Para o reconhecimento de padrões, o **SENSOR** usa a descrição do padrão como uma cadeia de símbolos que codifica a direção entre dois pontos consecutivos do contorno de uma imagem, todavia, o método aqui descrito pode ser utilizado para qualquer outro tipo de característica básica que possa ser escrita na forma de uma cadeia de símbolos que represente os padrões. Por fim, o mesmo tratamento dado ao problema de reconhecimento de padrões é aplicado à análise de cenas.

7.2 Extração de Características Básicas

De acordo com a rede mostrada na fig. 6.1, o primeiro passo neste processo é obter a descrição da borda da imagem *Im*. Este trabalho é feito pelo nodo **Contorno**, que utiliza a informação fornecida pelos nodos **Vert** e **Horiz**.

Dado o dicionário:

$$D_1 = \{p = \text{preto}, e = \text{escuro}, m = \text{médio}, c = \text{claro}, b = \text{branco}\} \quad (7.1)$$

o nodo **Lumin** reescreve uma imagem no formato *bitmap* (fig. 7.1.a) em uma cadeia s_1 de símbolos que codificam a luminância, que é calculada por:

$$Y = 0,30 \times R + 0,59 \times G + 0,11 \times B \quad (7.2)$$

segundo os graus de vermelho, verde e azul presentes na imagem, conforme foi apresentado pelos *agentes Retina e Luminância* (cap. 6), e transforma-a nas variáveis lingüísticas *preto, escuro, médio, claro e branco*.

O nodo **Contraste** gera o vetor v_1 que contém a localização das sub-cadeias de contraste $D_2 = \{bp, pb, eb, be, pc, cp\}$ no reticulado que representa o bitmap (fig. 7.1b). A borda da figura é obtida como uma sub-cadeia fechada $\{b\}_k$ de s_1 associada às k células vizinhas no bitmap que possuem os símbolos em D_2 . Uma sub-cadeia fechada (contorno) é definida como tendo seus caracteres inicial e final localizados na mesma célula (mesma posição) ou na vizinhança desta. Assim, (s_2, v_2) que foram produzidos pelo nodo **Contorno** são compostos pelas sub-cadeias que definem o contorno e os vetores associados.

Seja dado

$$\Delta x = x_i - x_{i-1}, \Delta y = y_i - y_{i-1} \quad (7.3)$$

como as variações de coordenadas em v_2 das células associadas aos símbolos consecutivos (i e $i - 1$) em s_2 , que produzem as cadeias s_3^x e s_3^y definidas pelo dicionário:

$$D_3 = \{p = \text{positivo}, n = \text{negativo}, z = \text{zero}\}$$

$$D_4 = \{a = \text{nulo}, b = \text{pequeno}, c = \text{médio}, d = \text{grande}, e = \text{enorme}\}$$

$$D_5 = D_3 \cup D_4 \quad (7.4)$$

tal que

$$\text{negativo} \rightarrow \Delta x \text{ ou } \Delta y < 0$$

$$\text{positivo} \rightarrow \Delta x \text{ ou } \Delta y > 0$$

$$\text{zero} \rightarrow \Delta x \text{ ou } \Delta y = 0$$

$$\text{nulo} \rightarrow \Delta y / \Delta x < \alpha_1 \quad (7.5)$$

$$\text{pequeno} \rightarrow \alpha_1 \leq \Delta y / \Delta x < \alpha_2$$

$$\text{médio} \rightarrow \alpha_2 \leq \Delta y / \Delta x < \alpha_3$$

$$\text{grande} \rightarrow \alpha_3 \leq \Delta y / \Delta x < \alpha_4$$

$$\text{enorme} \rightarrow \Delta y / \Delta x \geq \alpha_4$$

O nodo **Dir**₁ combina as cadeias s_3^x e s_3^y em uma outra cadeia s_4 de símbolos pertencentes ao seguinte dicionário:

$$D_6 = \{A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P\} \quad (7.6)$$

e que representam as direções permitidas no espaço XY , tal como na figura 6.13, e de acordo com a seguinte gramática:

$$\begin{aligned} G_1 &= \{V_{s_1}, V_{n_1}, V_{t_1}, P_1, \eta\} \\ V_{s_1} &= \{s_2, v_2\} \\ V_{n_1} &= \{z, n, p, a, b, c, d, e, s_3^x, s_3^y\} \\ V_{t_1} &= D_6 \\ P &= \left\{ \begin{array}{cccc} p * a \rightarrow P & ppb \rightarrow A & ppc \rightarrow B & ppd \rightarrow C \\ *pe \rightarrow D & npd \rightarrow E & npc \rightarrow F & npb \rightarrow G \\ n * a \rightarrow H & nnb \rightarrow I & nnc \rightarrow J & nnd \rightarrow K \\ *ne \rightarrow L & pnd \rightarrow M & pnc \rightarrow N & pnb \rightarrow O \end{array} \right\} \end{aligned} \quad (7.7)$$

onde, * é o símbolo de não-relevante.

A cadeia s_4 que descreve as direções compartilhadas pelos pontos consecutivos que compõem o contorno é do tipo:

$$s_4 = \{DBC.....JGFHHD\} \quad (7.8)$$

Este é o processamento, escrito de uma forma mais resumida, que o agente “Direção” realiza. Ele foi descrito novamente para mostrar o processamento dos nodos **Dir**₁ e **Dir**₂, que na verdade, utilizam este agente como processador interno.

O nodo **Sig** reescreve a cadeia s_4 na cadeia s_5 , que é composta pelos símbolos do seguinte dicionário:

$$D_7 = \{z = \text{sem mudança}, h = \text{mudança horária}, a = \text{mudança anti-horária}\} \quad (7.9)$$

Os símbolos descrevem as mudanças de direção entre as células do bitmap associadas a dois símbolos consecutivos de s_4 . Assim, s_4 descreve as mudanças de direção no contorno da figura. O nodo **Sig** procura pelas posições das seguintes sub-cadeias zhh , ahh , zaa ,

haa , aza , hzh , zzh , zza , zhz e zaz , em s_5 para gerar o vetor de saída v_6 , que contém as coordenadas (x, y) da célula bitmap onde as sub-cadeias acima indicam uma mudança de direção (pontos significativos).

O nodo **Sig** utiliza a gramática a seguir,

$$\begin{aligned}
 G_2 &= \{V_{s_2}, V_{n_2}, V_{t_2}, \mathcal{P}_2, \eta\} \\
 V_{s_2} &= \{s_4, v_4\} \\
 V_{n_1} &= \{z, a, h\} \\
 V_{t_2} &= \{S, N\} \\
 P &= \left\{ \begin{array}{cccc}
 zaa \rightarrow S & haa \rightarrow S & zhh \rightarrow S & ahh \rightarrow S \\
 aza \rightarrow S & hzh \rightarrow S & zza \rightarrow S & zzh \rightarrow S \\
 zaz \rightarrow S & zhz \rightarrow S & \#\#\# \rightarrow N &
 \end{array} \right\} \quad (7.10)
 \end{aligned}$$

onde, $\#\#\#$ significa que qualquer outra formação produz o símbolo N .

Esta gramática produz cadeias do tipo:

$$s_6 = \{SNN \dots S\} \quad (7.11)$$

que são usadas para construir o vetor v_6 , que indica as posições no espaço dos pontos significativos. O vetor seguinte foi gerado para o exemplo da fig. 7.1.c.

$$\begin{aligned}
 v_6 &= \{ \quad 300/380\&300/433\&313/513\&338/540\&\dots 398/327\&376/300\& \\
 &\quad 351/327\&313/380\&300/380\& \quad \} \quad (7.12)
 \end{aligned}$$

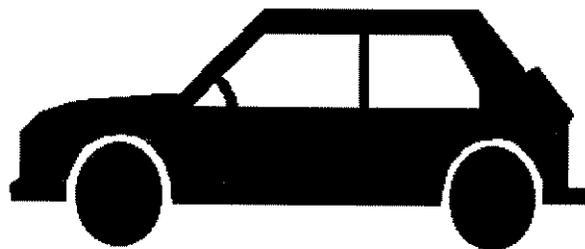
onde, / é o separador entre x e y , e & é o separador entre os pares (x, y) .

O nodo **Dir₂** recodificam v_6 na cadeia s_7 que descreve simbolicamente as direções entre os pontos significativos.

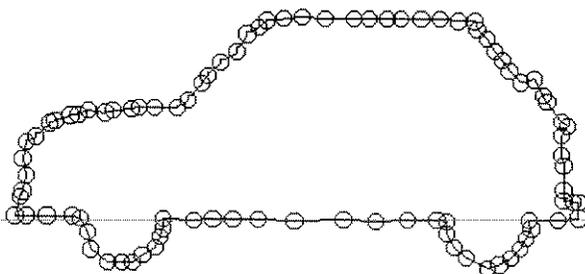
$$s_7 = \{DCBAPCPMONLNHKJGEHKJGEHGD\} \quad (7.13)$$

7.3 Aprendizagem

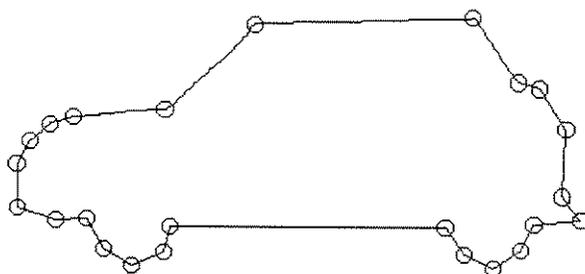
A aprendizagem de cadeias prototípicas de padrões é feita pela descoberta de símbolos comuns compartilhados por um conjunto de cadeias do tipo de s_7 , que descrevem um conjunto de exemplos de formas a serem reconhecidas.



a) Bitmap de entrada



b) Conjunto de pontos que definem o contorno.



c) Conjunto de pontos significativos

Figura 7.1: Um exemplo de extração de características básicas.

Para podermos aprender a forma do carro da fig. 7.1, utilizamos o conjunto de treinamento mostrado na figura 7.2, que contém figuras da mesma forma mas diferentes tamanhos e também formas distorcidas. Isto visa avaliar a sensibilidade do algoritmo de aprendizagem em encontrar variações na forma de um padrão.

O algoritmo usado para descobrir os símbolos comuns foi o *algoritmo de aprendizagem por molde*, descrito no capítulo anterior, e a cadeia aprendida (p_c) foi:

$$p_c = \{ DEC/5;0\&PAO/6;0\&BCA/4;1\&PAO/5;2\&MNL/7;0\&KJL/5;1\&HIG/5;1\&JKI/6;0\&HIG/3;0\&DEC/5;2\& \} \quad (7.14)$$

onde, ; separa o intervalo máximo do intervalo mínimo, no qual um caracter deve ser

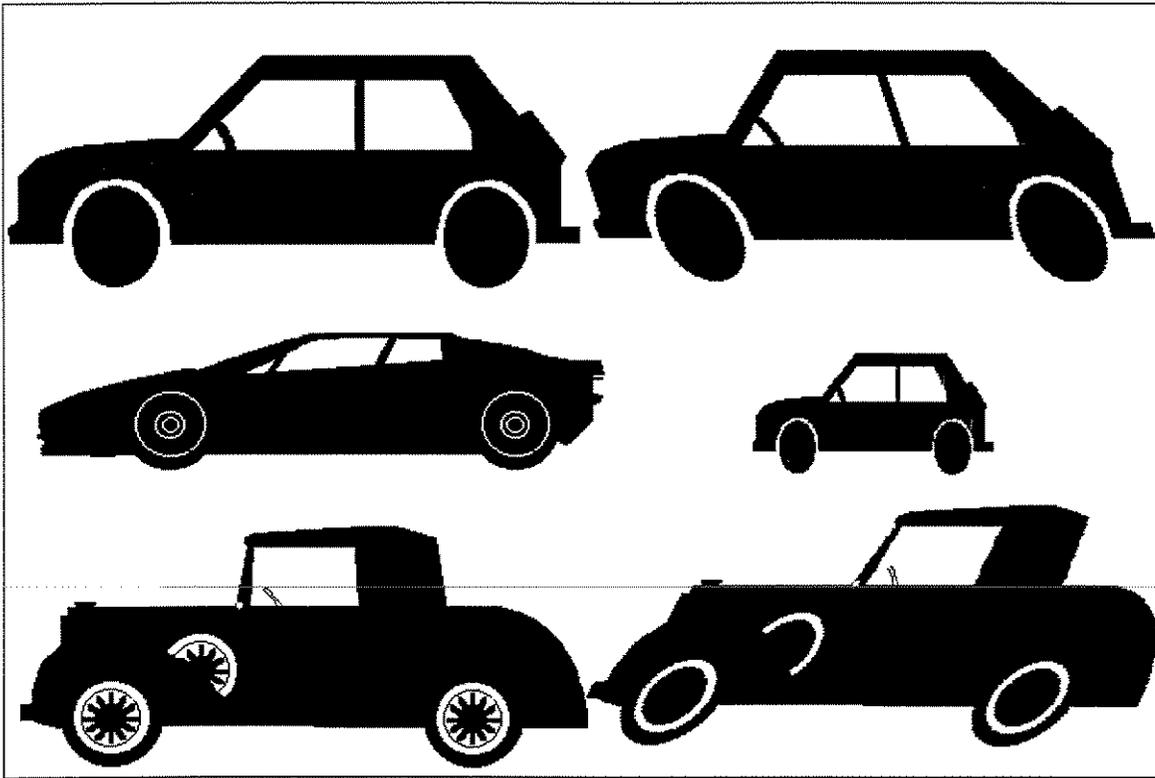


Figura 7.2: Alguns exemplos de padrões de treinamento

encontrado; / separa os caracteres de acoplamento (primeiro caracter) e de vizinhança (demais caracteres) do intervalo de procura destes caracteres; e & separa os conjuntos de caracteres e seus limites.

7.4 Reconhecimento do Padrão

Finalmente, a cadeia s_7 é comparada pelo nodo **Acoplamento** com as cadeias p_1 até p_n , aprendidas, respectivamente, pelos nodos P_1 à P_n , que representam formas prototípicas de figuras que compõem uma base de conhecimento. A comparação entre as cadeias é feita pelo *algoritmo de acoplamento nebuloso*. O conjunto calculado de graus de acoplamento $\mu(s_7, p_j)$ entre s_7 e p_j ($j = 1, n$), é a solução do processo de reconhecimento de padrões nebuloso suportado pelo sistema descrito pela rede da figura 6.1.

A seguir, mostramos uma tabela (tab. 7.1) com o resultado do treinamento e do acoplamento de um conjunto de exemplos.

CARROS			NAVIOS
Exemplo	Treinamento	Acoplamento	Acoplamento
1	1,00*	0,96	-0,30
2	-0,71	-0,80	-0,47
3	0,96*	1,00	-0,36
4	-0,73*	-0,76	-0,30
5	-0,33	-0,80	-0,50
6	-0,78*	-0,73	-0,57
7	-0,36	-0,83	-0,50
8	-0,53	-0,86	-0,43
9	-0,80*	-0,86	-0,57
11	-0,45	-0,80	
12	-0,64	1,00	
13	-0,34	-0,50	
14	-0,52	-0,80	
15	-0,74*	-0,66	
16	-0,56	-0,76	

Tabela 7.1: Reconhecimento do padrão carro.

O primeiro passo no procedimento de treinamento foi selecionar um carro e um navio como exemplos-moldes. Daí, todos os outros exemplos correspondentes a estas classes foram comparados a estes moldes, a fim de selecionar os sub-conjuntos das figuras mais similares como o conjunto de treinamento. A tabela 7.1 mostra o resultado no caso da seleção para o carro. A coluna rotulada *treinamento* mostra o grau de acoplamento entre a figura molde, que corresponde ao exemplo nº 1, e todos os outros carros. Os exemplos utilizados para o treinamento foram aqueles marcados com o carácter *. A cadeia prototípica aprendida é a mostrada em (7.14). As colunas chamadas *acoplamento* abaixo das seções *carros* e *navios* mostram o grau de acoplamento entre p_c e as cadeias s_7 que descrevem cada carro ou navio.

Lembramos que o (-) significa um acoplamento parcial, onde caracteres de acoplamento não foram encontrados na cadeia de comparação.

7.5 Análise de Cenas

Para a análise de uma cena, inicialmente é necessário extrair-se todos os componentes da mesma. O processo de segmentação de imagem sugerido anteriormente foi utilizado para extrair os padrões integrantes de uma cena e posteriormente, representá-la e reconhecê-la à partir de seus componentes isolados e previamente aprendidos.

Analisemos a figura 7.3. Ela é formada por uma casa, um carro, uma árvore e uma lata de lixo. Cada um dos componentes (P_i) da cena pode ser representado por descrições simbólicas (p) e vetoriais (v) que codificam suas características básicas:

$$D(P) = \{(p_1, v_1), (p_2, v_2), \dots, (p_n, v_n)\} \quad (7.15)$$

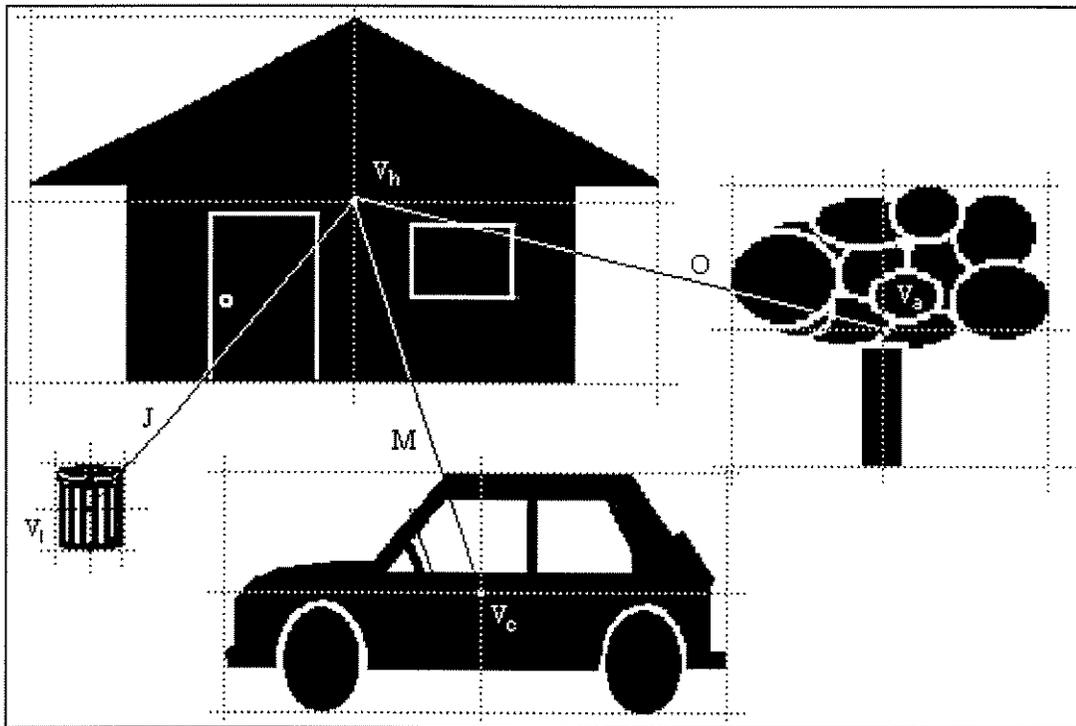


Figura 7.3: Exemplo de uma cena representativa.

Desta forma, uma cena C é composta de um conjunto de objetos distribuídos em um espaço n -dimensional, cuja descrição é:

$$C = \{D, V\} = \{\{D(P_i)\}, \{V_i\}\} \quad (7.16)$$

onde, $D(P_i)$ é a descrição do objeto i , e V_i é a sua posição no espaço. E a descrição da cena é suportada por uma gramática do tipo:

$$G_c = \{V_s, V_n, V_t, \mathcal{P}, \eta\} \quad (7.17)$$

onde:

V_s é o conjunto de símbolos iniciais;

V_n é o conjunto de símbolos não-terminais;

V_t é o conjunto de símbolos terminais;

η é o elemento nulo (vazio) e

\mathcal{P} é o conjunto de regras que reescrevem as relações permitidas e proibidas entre os elementos de \mathcal{C} .

O número de gramáticas passíveis de descrever uma cena \mathcal{C} é dependente da entropia do sistema observado.

Seja dado

$$\mathcal{C} = \{\{D(P_1), D(P_2), \dots, D(P_m)\}, \{V_1, V_2, \dots, V_m\}\}$$

O primeiro passo na análise de \mathcal{C} é classificar cada objeto P_i , de acordo com um conjunto de padrões $P = \{P_k\}$ previamente aprendidos, conforme discutido na seção 6.5. Esta fase é idêntica à etapa de análise de sintaxe do reconhecimento sintático de padrões. Assim, a cena da fig. 7.3 tem a seguinte descrição:

$$\begin{aligned} O_h &= D(P_h) && \text{(casa)} \\ O_c &= D(P_c) && \text{(carro)} \\ O_a &= D(P_a) && \text{(árvore)} \\ O_l &= D(P_l) && \text{(lata de lixo)} \end{aligned}$$

$$\mathcal{C} = \{D(P_i), V\} = \{O_i, V\} \quad (7.18)$$

Mas apenas o reconhecimento dos objetos da cena não é suficiente para garantir que uma cena possa ser identificada. É preciso avaliar também as relações existentes entre os objetos que compõem a cena.

O segundo passo no processamento desta análise da cena é, portanto, distribuir $\{O_i\}$ sobre V_s , V_n e V_t , segundo uma gramática, por exemplo, de perspectiva:

$$G_p = \{V_{s_p}, V_{n_p}, V_{t_p}, \mathcal{P}_p, \eta\} \quad (7.19)$$

produzindo uma outra descrição da cena, à partir das descrições das relações partilhadas por seus objetos:

$$C = \{\{D_j(O_i)\}, \{V_i\}\} \quad (7.20)$$

É necessária, portanto, a escolha um objeto como símbolo inicial, que servirá de referência para a análise das relações partilhadas pelos outros objetos da cena. Seja a casa o símbolo inicial s_0 de referência (foco de atenção primário). Os demais objetos da cena serão descritos em relação a ele, e uma gramática de perspectiva do tipo da fig. 7.4.a suportaria a descrição: *'existe uma casa com um carro parado à frente, e entre eles observa-se à direita uma árvore e à esquerda uma lata de lixo.'*

Quando o objeto casa (s_0) é reconhecido na cena, ativa a cadeia de derivação na fig. 7.4.b, que descreve as relações espaciais dos objetos da cena e a cadeia de derivação na fig. 7.4.a que descreve uma cena particular aprendida ou definida *ad hoc*. A ativação desta última cadeia de derivação seleciona carro (s_1) como foco de atenção secundário, o que por sua vez, ativa o cálculo de posicionamento dos outros objetos em relação a ele, conforme mostrado na fig. 7.4.b. Tais processamentos permitem agora, a ativação das regras de produção definidas nos ramos finais da árvore (a) da fig. 7.4, para tratamento de s_2 e s_3 .

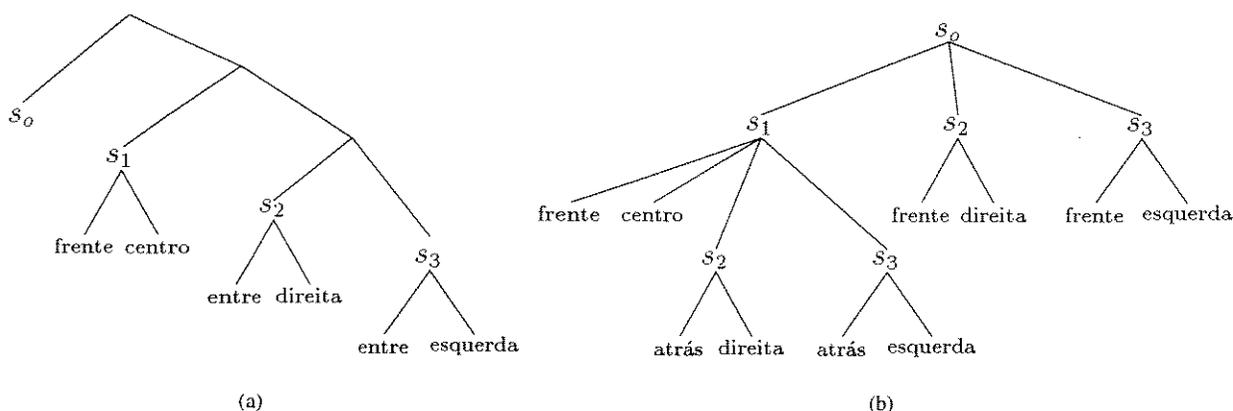


Figura 7.4: Exemplos de uma árvore de derivação.

As relações entre objetos na fig. 7.4.b são obtidas à partir do sistema de referência mostrado na fig. 7.5, tomando-se o objeto do foco de atenção (primário ou secundário) como origem, e localizando os demais com procedimento semelhante ao descrito para o agente "Direção", pag. 76. Estas relações são definidas como:

frente	=	{I, J, K, L, M, N, O}
atrás	=	{A, B, C, D, E, F, G}
direita	=	{A, B, C, M, N, O, P}
esquerda	=	{E, F, G, H, I, J, K}
centro	=	{C, D, E, K, L, M}
ao lado de	=	{A, O, P, G, H, I}
entre	=	frente de s_i e atrás de s_j

Para codificar a distância e o tamanho, as medidas são transformadas em variáveis lingüísticas que expressem estes conceitos, como, respectivamente, {*perto, médio, longe*} e {*pequeno, médio, grande*}, e assim por diante. Assim, teríamos um conjunto de símbolos que, arranjados por uma sintaxe, descreveriam a cena.

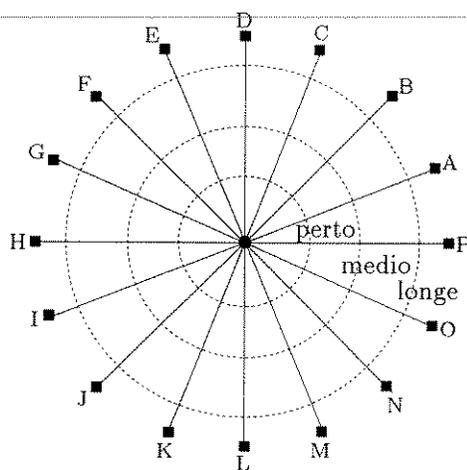


Figura 7.5: Possíveis localizações entre dois elementos de uma cena.

Em decorrência, a aprendizagem da cena poderia ser feita pelos mesmos mecanismos e algoritmos descritos anteriormente para a aprendizagem e reconhecimento de padrões. Outra possibilidade é que este conhecimento seja inserido *ad hoc*, por instrução de um supervisor.

A cena da fig. 7.3 é considerada uma *cena representativa* porque obedece as regras da gramática de perspectiva dada. A disposição dos objetos desta cena não é única, ou seja, alguns objetos podem variar de posição e continuamos a identificá-la, e uma gramática nebulosa tem que ser flexível a estas variações.

Observemos agora a fig. 7.6. Ela possui os mesmos objetos que compõem a cena anterior, entretanto, pelo modo como estão dispostos, ela é considerada uma *cena não-*

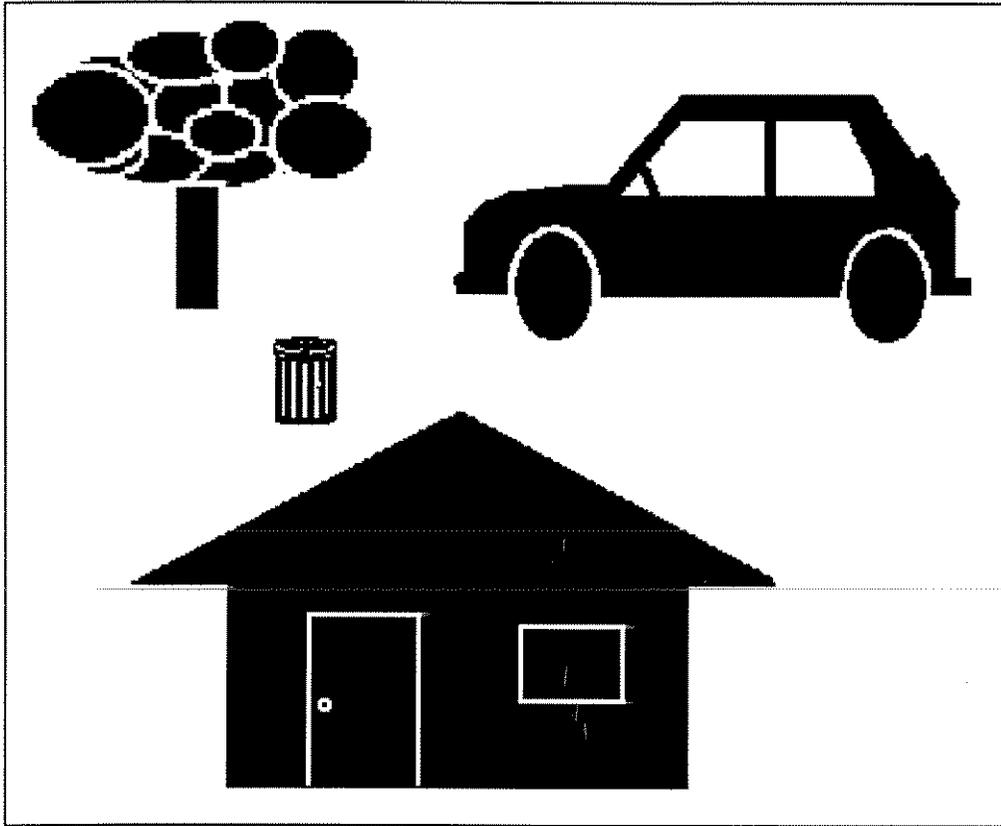


Figura 7.6: Exemplo de uma cena não-representativa.

representativa, pois a disposição espacial dos objetos não obedece a nenhuma gramática de perspectiva.

A figura 7.7 é uma cena que apresenta um conflito do tipo definido pela gramática mostrada na fig. 7.8, pois se considerarmos que:

- a) s_1 deve estar em uma rua, então s_2 e/ou s_3 não pode estar ao lado de s_1 ; mas
- b) s_1 pode estar em um estacionamento, então s_2 e/ou s_3 pode estar “ao lado de”.

Em outras palavras, uma restrição desta cena é ‘a árvore não pode estar ao lado do carro, a menos que o carro esteja em um estacionamento’.

Este conflito deve levar o sistema a buscar evidências na região “ao lado de” para definir uma rua ou estacionamento, isto é, modula o processo de atenção. Portanto, se houver uma marcação como a desenhada na fig. 7.9, que pode significar a delimitação

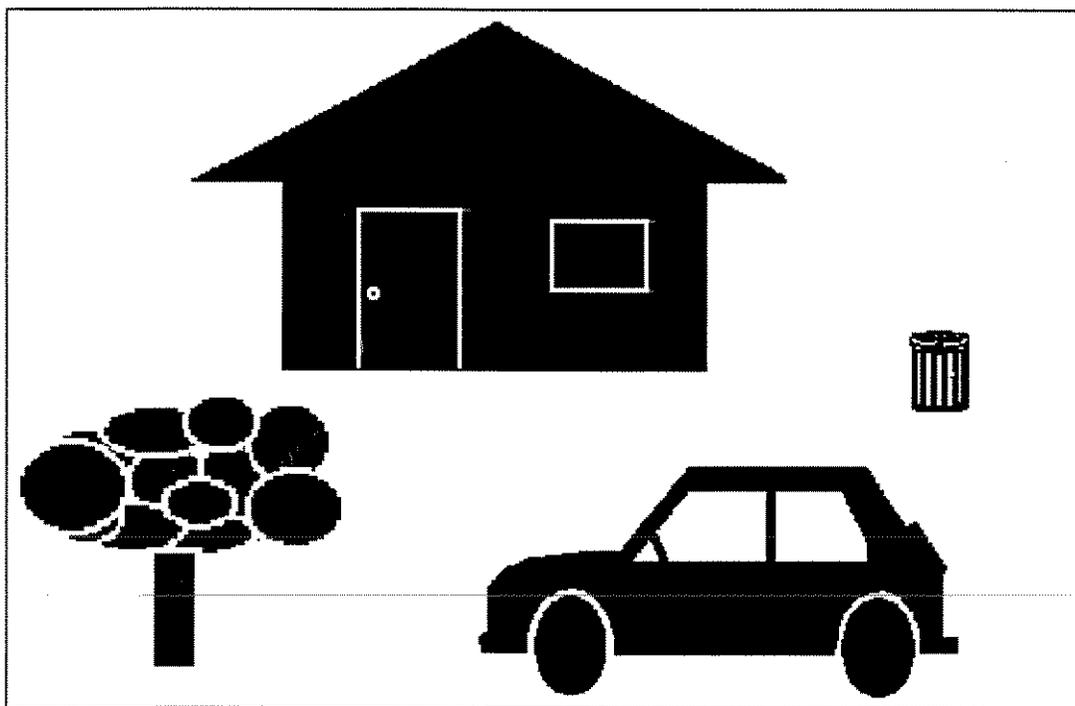


Figura 7.7: Exemplo de uma cena de conflito.

de uma rua ou um local para estacionamento, a cena passa a obedecer a gramática de perspectiva da fig. 7.8 e é chamada uma *cena permitida*.

Assim, dado um símbolo inicial e uma gramática, esta define as áreas onde jogar o foco de atenção para segmentação através da modificação do algoritmo de segmentação, procurando evidências para solucionar o conflito.

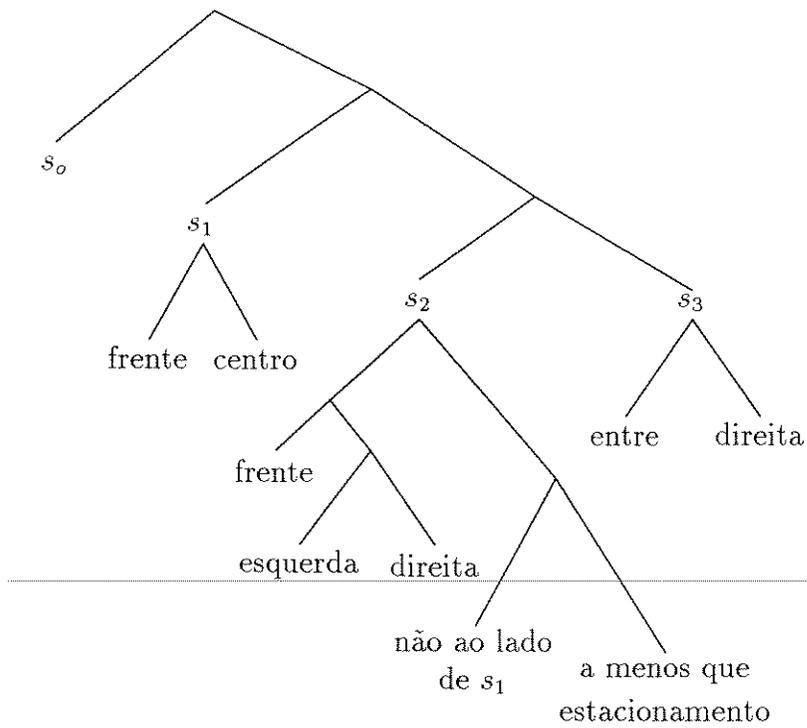


Figura 7.8: Exemplos de uma árvore de derivação mais complexa.

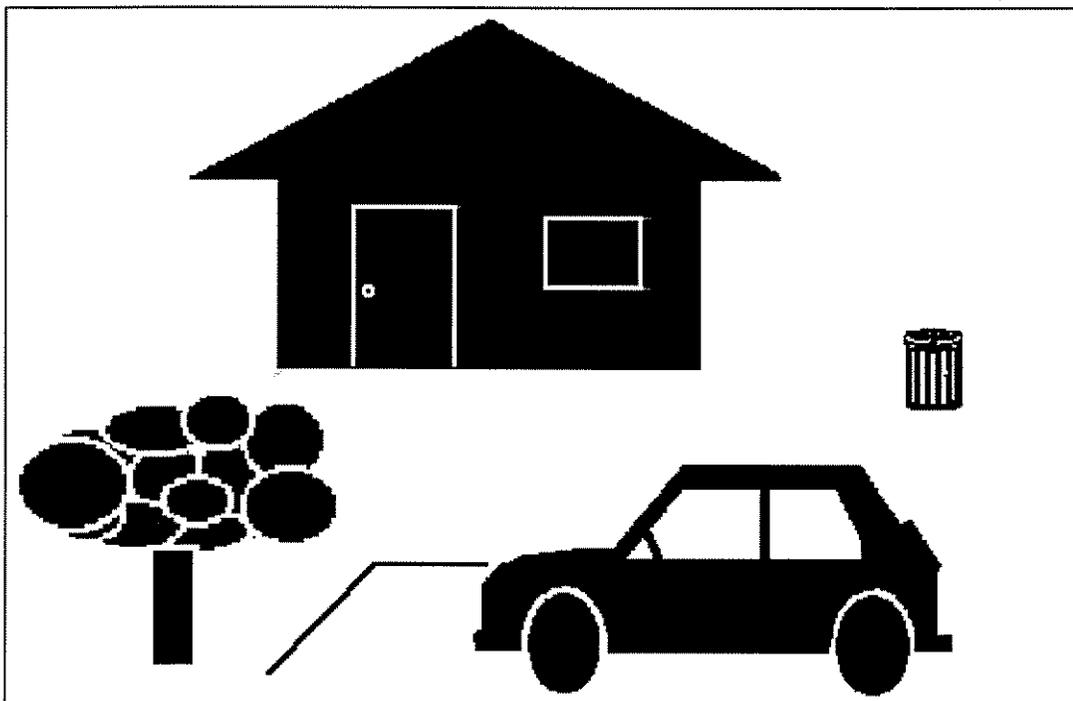


Figura 7.9: Exemplo de uma cena permitida.

Capítulo 8

Conclusão

O conhecimento sobre os genes reais foi usado para propor uma teoria formal para um sistema distribuído, como uma ferramenta inteligente capaz de resolver problemas através do uso de um raciocínio simbólico, suportado pela teoria de linguagens formais nebulosas. Mostramos que a linguagem formal nebulosa é um formalismo apropriado para modelar um sistema distribuído artificial “similar” aos organismos multicelulares, onde as células desempenham o papel central de agentes especializados no processamento de um sub-conjunto definido de uma dada linguagem formal.

Cada célula é equipada com uma membrana de interface cuja função é ativar os símbolos iniciais da gramática da célula, de acordo com as condições definidas no ambiente extra-celular. A ativação de símbolos iniciais inicializa um conjunto de regras de reescrita, cujos produtos finais são o resultado dos cálculos realizados pela célula. As relações que as células requerem para resolver um problema específico podem ser codificadas por meio de uma hierarquia adequada imposta por uma genética (gramática) G que define o sistema distribuído, assim como a hierarquia dos genes especializa células e organiza-as em organismos.

O neurônio, o tipo de célula modelada, mostrou ser um bom modelo para a implementação do sistema **SENSOR** pois fornece uma grande capacidade computacional tanto simbólica quanto numérica, suportando o formalismo das linguagens nebulosas, e proporcionando uma gama enorme de possibilidades de criação de agentes variados, variabilidade fundamentada no controle gênico celular. Estes são agentes especializados que, por meio de uma linguagem de comunicação comum, trocam mensagens entre si e são capazes de

solucionar, conjuntamente, tarefas específicas, baseados no comportamento dos sistemas neurais biológicos equivalentes.

Criado à partir das noções de especialização e integração de agentes neurais, o sistema visual artificial para análise de uma cena teve um desempenho condizente com o do sistema visual humano. Pois não só o sistema visual foi capaz de segmentar e identificar cenas representativas e não representativas, como também identificou cenas de conflito. O conflito, por sua vez, modula um processo de atenção que aprimora o sistema biológico de segmentação da cena.

Devemos lembrar que o objetivo deste trabalho não era criar um método de processamento de imagem concorrente com outros já existentes, e que tivesse um desempenho nitidamente superior a estes, e sim verificar se uma modelagem suportada pelas propriedades básicas dos neurônios visuais produzia resultados semelhantes aos dos mecanismos propostos para explicar a fisiologia do sistema visual. Este objetivo foi cumprido.

Os contornos extraídos de vários objetos visuais da mesma classe e de vários tamanhos foram usados para uma aplicação do **SENSOR** no reconhecimento de padrões. Tais contornos foram descritos por relações entre direções de seus pontos consecutivos montando uma cadeia de símbolos que as representassem.

Muitos tipos de algoritmos de aprendizagem podem ser usados para construir cadeias-padrões que possam ser utilizadas para tratar de um problema definido. O procedimento chamado *algoritmo de aprendizagem por moldes* foi usado para construir um sistema que utiliza agentes com funções específicas e distribuídas para reconhecer formas de padrões, e posteriormente, cenas. Uma característica marcante do sistema introduzido aqui é sua capacidade de aprender a identificar formas semelhantes de diferentes tamanhos e diferentes graus de distorção.

Para a análise de cenas, vimos que além da verificação de quais elementos compõem-na, é importante saber a orientação de cada um destes elementos em relação a um referencial, pois as suas posições podem ser ou não representativas para o reconhecimento da cena, e após o reconhecimento, classificadas como permitidas ou não. Isto significa que além da determinação do objetos que fazem parte de uma cena, outros aspectos devem ser considerados quando da análise, como relações possíveis e restrições impostas.

Seguindo esta aplicação e resultados do **SENSOR**, sugerimos como desenvolvimentos futuros, dentre duas perspectivas diferentes, as seguintes propostas:

- a) continuar a estudar o reconhecimento sintático de padrões para:
- explorar outras características básicas que possam ser extraídas dos padrões para serem usadas simultaneamente no reconhecimento, a fim de produzir um tratamento mais refinado e preciso dos mesmos, pois vários outros agentes foram desenvolvidos, apesar de não apresentados neste trabalho, tais como o agente “Periodicidade”, que verifica a ocorrência de seqüências de símbolos repetidas dentro de uma cadeia;
 - trabalhar padrões mais complexos, cujas primitivas possam ser segmentadas em termos das redes de raciocínio do **SENSOR**, a fim de gerar as árvores de derivação dos padrões; e
 - utilizar o reconhecimento de padrões em séries temporais.
- b) Utilizar os conceitos de linguagem formal nebulosa para simular os processos sinápticos que ligam a ativação do receptor pelo transmissor à modulação e especificação da leitura gênica, que resulta no controle da própria plasticidade neural, envolvida tanto no aprendizado qualitativo resultante da modificação da estrutura dos transmissores e receptores envolvidos, quanto na aprendizagem quantitativa, que envolve alterações das quantidades disponíveis destes mesmos transmissores e receptores.

“Em relação a todos os atos de iniciativa e de criação existe uma verdade elementar, ao mesmo tempo em que nos esforçamos, a providência também se põe em marcha. Toda uma gama de acontecimentos flui e surge a nosso favor. Como resultado da decisão, seguem todas as formas imprevistas de coincidência, encontros e ajuda que nenhum homem jamais pudera ter sonhado encontrar. Tudo o que se pode sonhar, pode-se realizar. A coragem contém em si mesma, o poder, a genialidade e a magia.”

Johann Wolfgang Goethe

Bibliografia

- [Adl94] L. M. Adleman. "Molecular computation of solutions to combinatorial problems". *Science*, 266, pag. 1021-1024, 1994.
- [Cho57] N. Chomsky. "Syntactic Structures". The Hague, Mouton, 1957.
- [DuPr80] D. Dubois e H. Prade. "Fuzzy Sets and Systems: Theory and Applications". Academic Press, 1980.
- [DuPr82] D. Dubois e H. Prade. "A class of fuzzy measures based on triangular norms. A general framework for the combination of uncertain information". *Int. J. General System*, 8/1, pag. 43-61, 1982.
- [DuHa73] R. O. Duda e P. E. Hart. "Pattern Classification and Scene Analysis". John Wiley & Sons, New York, 1973.
- [Fu74] K. S. Fu. "Syntactic Methods in Pattern Recognition". *Mathematics in Science and Engineering*, vol. 112, Academic Press, New York, 1974.
- [Guy92] A. C. Guyton. "Tratado de Fisiologia Médica". Ed. Guanabara-Koogan, Rio de Janeiro, 8ª edição, 1992.
- [HuWi62] D. H. Hubel e T. N. Wiesel. *J. Physiol.*, London, **160**, 106, 1962.
- [Kau75] A. Kaufmann. "Introduction to the Theory of Fuzzy Subsets". Academic Press, 1ª edição, 1975.
- [KSJ91] E. R. Kandel, J. H. Schwartz e T. M. Jessell. "Principles of Neural Science". Elsevier Science Publishing, 3ª edição, 1991.
- [Lee90] C. C. Lee. "Fuzzy Logic in control systems: Fuzzy Logic controller". *IEEE Transactions on systems, man and cybernetics*, 20(2): pag. 405-435, março/abril, 1990, partes I e II.

- [Leh93] A. L. Lehninger. "Princípios de Bioquímica". Sarvier Ed. de Livros Médicos, 1ª edição, São Paulo, 1993.
- [Lip95] R. J. Lipton. "DNA Solution of Hard Computational Problems". *Science*, vol. 268, abril, 1995.
- [Mar82] D. Marr "Vision". W. H. Freeman, San Francisco, 1ª edição, 1982.
- [Mau95] J. H. R. Maunsell. "The Brain's Visual World: Representation of Visual Targets in Cerebral Cortex". *Science*, vol. 270, pag. 764-769, novembro, 1995.
- [McSh95] H. H. McAdams e L. Shapiro. "Circuit Simulation of Genetic Networks". *Science*, vol. 269, pag. 650-656, agosto, 1995.
- [MTT73] M. Mizumoto, J. Toyoda e K. Tanaka. "Examples of Formal Languages with Weights". *Information Processing Letters*, 2, pag. 74-78, 1973.
- [NeRa75] C. V. Negoita e D. A. Ralescu. "Applications of fuzzy sets to systems analysis". John Wiley & Sons, New York, 1975.
- [ReRo95] M. P. F. Rebello e A. F. da Rocha. "KARDS: an Intelligent Neural Environment". *Proceedings of IFSA 95*, IFSA, pag. 697-699, São Paulo, Brasil, 1995.
- [RFB80] A. F. Rocha, E. Françoso e M. A. Balduino. "Neural Languages". *Fuzzy Sets and Systems*, 3/1, pag. 11-35, 1980.
- [Roc92] A. F. Rocha. "Neural Nets: A theory for brains and machines". *Lectures Notes in Artificial Intelligence*, Springer-Verlag, Berlin, 1992, vol. 638.
- [RoSe94] A. F. Rocha e A. B. S. Serapião. "Neurofuzzy Symbolic Systems and Pattern Recognition". *Proceedings of NAFIPS 94*, NAFIPS, San Antonio, Texas, 1994.
- [RSR96] A. F. Rocha, A. B. S. Serapião e M. P. F. Rebello. "Toward a Theory of Molecular Computing". *Submetido*.
- [RvZR93] M. Regattieri, F. J. von Zuben e A. F. Rocha. "Neurofuzzy interpolation II - reducing complexity of description". *Proceedings of the 2th International Conference on Neural Networks and Fuzzy Logic*, pag. 1835-1839, San Francisco, EUA, março, 1993.

- [SaJu85] D. Sagi e B. Julesz. "Where' and 'What' in Vision". *Science*, vol. 228, pag. 1217-1219, junho, 1985.
- [Sea93] D. B. Searls. "Formal Language Theory and Biological Macromolecules". *Submetido*.
- [Ser95] A. B. S. Serapião. "A Neural Distributed Sensory System". *Proceedings of IFSA 95*, IFSA, pag. 689-692, São Paulo, Brasil, 1995.
- [SeRo95] A. B. S. Serapião e A. F. Rocha. "Integrating Activities with Neurofuzzy Distributed Systems". *Proceedings of IJCAI 95*, IJCAI, Montreal, Canadá, 1995.
- [Sin95] W. Singer. "Development and Plasticity of Cortical Processing Architectures". *Science*, vol. 270, pag. 758-764, novembro, 1995.
- [SRRP96] A. B. S. Serapião, A. F. Rocha, M. P. F. Rebello e W. Pedrycz. "Toward a Theory of Genetic Systems". Em: *Genetics Algorithms and Soft Computing*. F. Herrera e J. L. Verdegay (Eds). Physica Verlag, Heidelberg. A ser publicado.
- [Ung95] L. G. Ungerleider. "Functional Brain Imaging Studies of Cortical Mechanisms for Memory". *Science*, vol. 270, pag. 769-775, novembro, 1995.
- [Zad73] L. A. Zadeh. "The concept of a linguistic variable and its application to approximate reasoning". *IEEE Transactions on systems, man and cybernetics*, SMC-3, pag. 28-44, 1993.