

# CAPÍTULO 4

## SISTEMAS INTELIGENTES E O SISTEMA IMUNOLÓGICO

Este capítulo faz uma revisão de tópicos específicos da teoria de sistemas inteligentes, destacando as redes neurais artificiais e os algoritmos evolutivos. O objetivo é não apenas o de salientar similaridades e diferenças entre estas teorias, o sistema imunológico biológico (Capítulo 2) e os sistemas imunológicos artificiais (Capítulo 3), mas também o de apresentar alguns dos algoritmos que serão utilizados na comparação de desempenho com as ferramentas a serem propostas no Capítulo 5. Uma breve discussão sobre computação molecular e sistemas nebulosos também é feita. Serão apresentados pontos de vista cognitivos do sistema imunológico biológico, e a resposta imune adaptativa guiada pelo princípio da seleção clonal será caracterizada como uma microevolução segundo a teoria Darwiniana.

*“A inteligência é uma faculdade especial, própria de certas classes de seres orgânicos, aos quais dá, com o pensamento, a sua vontade de agir, a consciência de sua existência e de sua individualidade, assim como os meios de estabelecer relações com o mundo exterior e de prover as suas necessidades” – A. Kardec*

### 4.1. Introdução

Ao introduzir a proposta de engenharia imunológica na Seção 3.2, foram discutidos diversos componentes da teoria de sistemas inteligentes, como *inteligência artificial clássica*, *redes neurais artificiais*, *computação evolutiva*, *vida artificial*, *sistemas nebulosos* e *computação molecular* ou *computação de DNA*. Aprofundando-se no estudo de cada uma destas abordagens, seria possível perceber a existência de diversas características comuns entre elas, os sistemas imunológicos biológicos e os sistemas imunológicos artificiais. Entretanto, devido ao enfoque adotado nesta tese, daremos prioridade ao estudo das redes neurais artificiais (Seção 4.2) e da computação evolutiva (Seção 4.4). Alguns comentários serão feitos, na Seção 4.6, tentando relacionar o SI com os sistemas nebulosos e a computação molecular. Além disso, muitos dos algoritmos de treinamento de redes neurais e de computação evolutiva revisados neste capítulo serão empregados para comparações com as ferramentas a serem propostas no Capítulo 5.

Os leitores que detiverem conhecimentos sobre redes neurais artificiais e computação evolutiva podem ir direto às Seções 4.3, 4.5 e 4.6, sem perda da seqüência do trabalho.

## 4.2. Redes Neurais Artificiais

*Definição 4.1:* Uma *rede neural artificial* (RNA) é um sistema de processamento massivamente paralelo, composto por unidades simples com capacidade natural de armazenar conhecimento e disponibilizá-lo para uso futuro (Haykin, 1999).

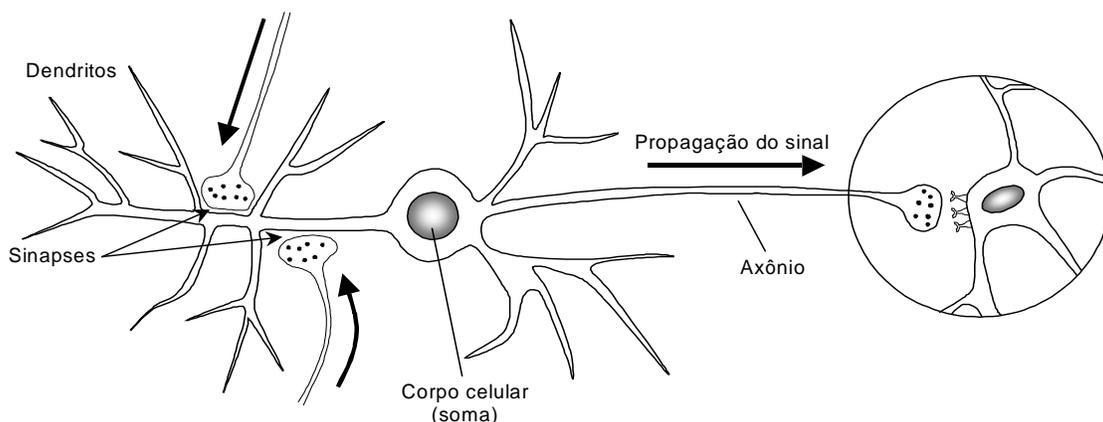
As RNAs assemelham-se ao cérebro em dois aspectos:

- Elas extraem conhecimento do ambiente através de um processo de *aprendizagem* ou *treinamento*; e
- Os pesos das conexões entre os neurônios, conhecidos como *pesos sinápticos*, são utilizados para armazenar o conhecimento adquirido.

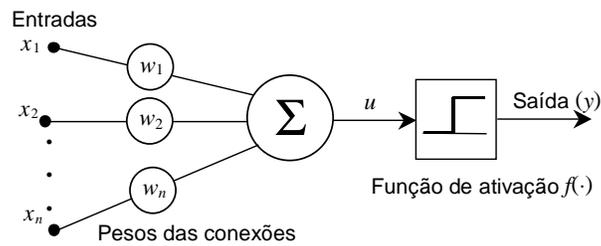
A Figura 4.1 apresenta um modelo de neurônio biológico com a seqüência de propagação dos sinais pela célula. Os neurônios artificiais também são chamados de *nós*, ou *unidades*. A natureza das RNAs faz com que seu estudo seja multidisciplinar, envolvendo pesquisadores de diversas áreas, como neurofisiologia, psicologia, física, computação e engenharia.

Neurofisiologistas e psicólogos estão particularmente interessados em compreender o funcionamento do sistema neural humano. As características de resposta a estímulos apresentada por neurônios individuais, bem como redes de neurônios, são alvo de estudo dos neurofisiologistas, enquanto os psicólogos estudam funções do cérebro tratando aspectos cognitivos e estão interessados na utilização de técnicas baseadas em redes neurais para criar modelos detalhados do comportamento humano.

Cientistas da área de computação têm em vista a construção de computadores dotados de processamento paralelo e distribuído, buscando superar as limitações impostas pelos computadores atuais, que realizam processamento serial simbólico.



**Figura 4.1.** Célula neural biológica (as setas largas indicam a seqüência de propagação de sinais pelos neurônios).



**Figura 4.2.** Representação funcional de um neurônio artificial.

Inspirados na habilidade apresentada pelos seres humanos e outros animais no desempenho de funções como o processamento de informação sensorial e a capacidade de interação com ambientes pouco definidos, os engenheiros estão preocupados em desenvolver sistemas artificiais capazes de desempenhar tarefas semelhantes. Habilidades como capacidade de processamento de informação incompleta ou imprecisa e generalização são propriedades desejadas em tais sistemas.

McCulloch & Pitts (1943) projetaram a estrutura que é conhecida como a unidade básica de uma rede neural. Estes pesquisadores propuseram um modelo de neurônio como uma unidade de processamento binária (Figura 4.2) e provaram que estas unidades são capazes de executar várias operações lógicas (OU, AND, etc.). Este modelo, apesar de muito simples, trouxe uma grande contribuição para as discussões sobre a construção dos primeiros computadores digitais, permitindo a criação dos primeiros modelos matemáticos de dispositivos artificiais que buscavam analogias biológicas. Matematicamente, o neurônio da Figura 4.2 pode ser expresso por:

$$y = f(u) = f(x_1w_1 + x_2w_2 + \dots + x_nw_n) = f(\mathbf{w}^T \mathbf{x}), \quad (4.1)$$

onde  $y$  é a saída do neurônio,  $u$  é a ativação do neurônio,  $f(\cdot)$  sua função de ativação,  $x_i$  ( $i = 1, \dots, n$ ) é o  $i$ -ésimo componente do vetor  $\mathbf{x}$  de entradas, e  $w_i$  ( $i = 1, \dots, n$ ) é o  $i$ -ésimo componente do vetor  $\mathbf{w}$  de pesos do neurônio.

#### 4.2.1. Características Principais

As redes neurais artificiais têm sido desenvolvidas como generalizações de modelos matemáticos de cognição humana ou neurobiologia, assumindo que:

- O processamento da informação ocorre com o auxílio de vários elementos chamados *neurônios*;
- Os sinais são propagados de um elemento a outro através de *conexões*;
- Cada conexão possui um *peso* associado, que, em uma rede neural típica, pondera o sinal transmitido; e
- Cada neurônio (ou unidade) possui uma *função de ativação* (geralmente não-linear), que tem como argumento a soma ponderada dos sinais de entrada, para determinar sua saída.

Uma rede neural pode ser caracterizada por três aspectos principais: (1) padrão de conexões entre as unidades (*arquitetura* ou *estrutura*), (2) método de determinação dos pesos das conexões (*algoritmo de treinamento* ou *aprendizagem*) e (3) *função de ativação*.

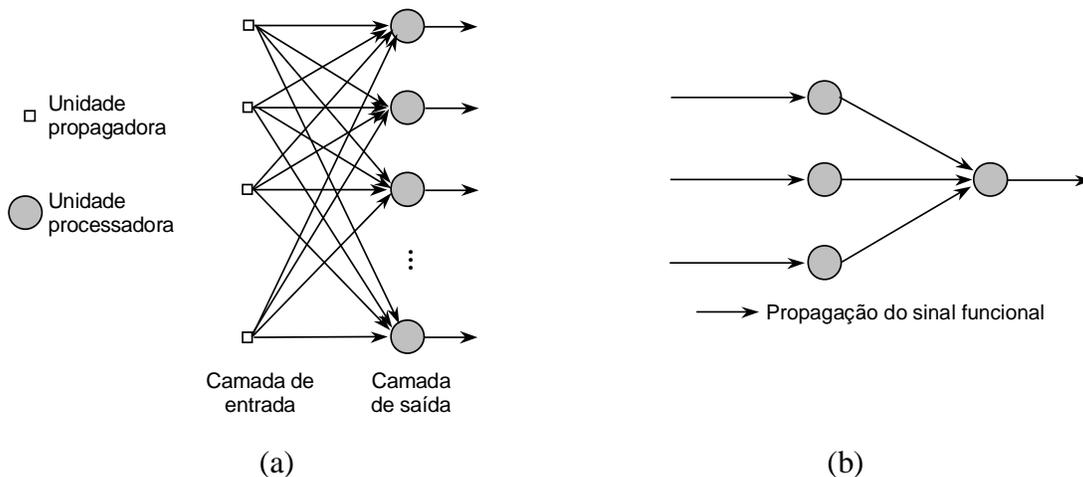
#### 4.2.1.1. Arquitetura

A forma pela qual os neurônios de uma RNA estão estruturados (interconectados) está intimamente relacionada ao algoritmo de aprendizagem a ser utilizado para treiná-la. A classificação dos algoritmos de aprendizagem será apresentada na Seção 4.2.1.2. Nesta seção, será feita uma breve descrição das principais arquiteturas de redes neurais artificiais.

Em geral é possível distinguir três classes fundamentais de arquiteturas: *redes feedforward de uma única camada*, *redes feedforward de múltiplas camadas* e *redes recorrentes*.

##### 4.2.1.1.1. Redes Feedforward de Uma Única Camada

No caso mais simples de redes em camadas (*layers*), tem-se uma camada de entrada com neurônios cujas saídas alimentam a última camada da rede. Geralmente, os neurônios de entrada são propagadores puros, ou seja, eles simplesmente repetem o sinal de entrada em sua saída distribuída. Por outro lado, as unidades de saída costumam ser unidades processadoras, como apresentado na Figura 4.2. A propagação de sinais nesta rede é puramente unidirecional (*feedforward*): os sinais são propagados apenas da entrada para a saída, e nunca vice-versa. Esta arquitetura está ilustrada na Figura 4.3(a) e a direção de propagação dos sinais na Figura 4.3(b).

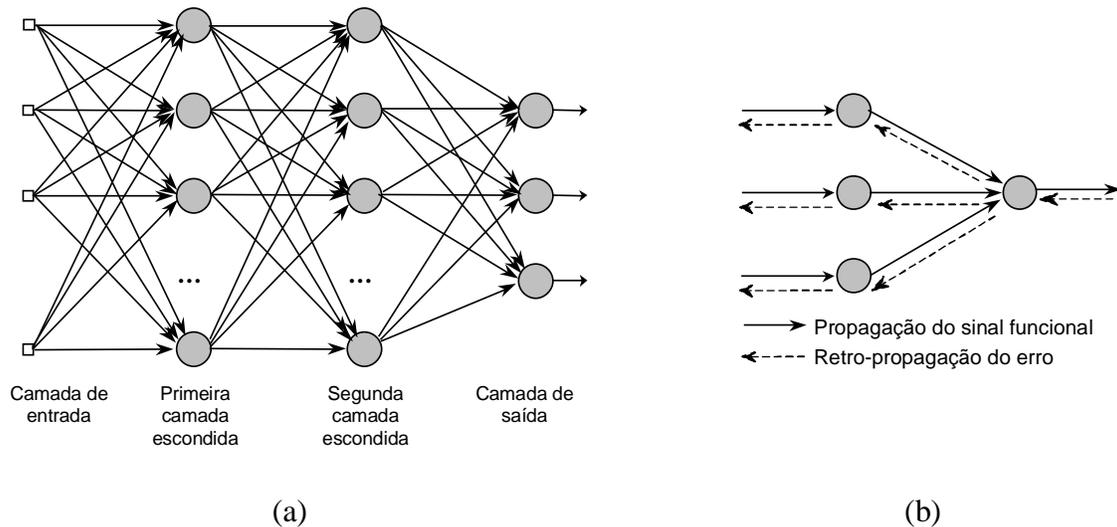


**Figura 4.3.** Redes neurais tipo *feedforward* com uma única camada de unidades processadoras. (a) Arquitetura. (b) Sentido de propagação do sinal funcional.

#### 4.2.1.1.2. Redes Feedforward de Múltiplas Camadas

A segunda classe de rede feedforward se distingue pela presença de uma ou mais camadas intermediárias ou escondidas (camadas em que os neurônios são efetivamente unidades processadoras, mas não correspondem à camada de saída). Adicionando-se uma ou mais camadas intermediárias, aumenta-se o poder computacional de processamento não-linear e armazenagem da rede. O conjunto de saídas dos neurônios de cada camada da rede é utilizada como entrada para a camada seguinte. A Figura 4.4(a) ilustra uma rede feedforward de múltiplas (duas) camadas intermediárias.

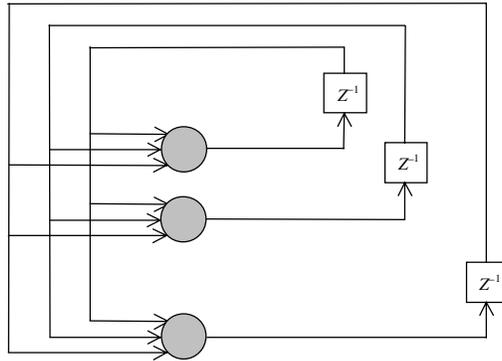
As redes feedforward de múltiplas camadas, são geralmente treinadas usando o algoritmo de retro-propagação do erro (*error backpropagation*), embora existam outros algoritmos de treinamento. Este algoritmo requer a propagação direta (*feedforward*) do sinal de entrada através da rede, e a retro-propagação (propagação reversa, ou *backpropagation*) do sinal de erro, como ilustrado na Figura 4.4(b).



**Figura 4.4.** Redes neurais tipo *feedforward* com múltiplas camadas. (a) Arquitetura. (b) Sentido de propagação do sinal funcional e do sinal de erro.

#### 4.2.1.1.3. Redes Recorrentes

As redes recorrentes distinguem-se das redes feedforward pela existência de pelo menos um laço (*loop*) de recorrência (*feedback*). Por exemplo, uma rede recorrente pode consistir de uma única camada de neurônios com cada neurônio alimentando seu sinal de saída de volta para a entrada de todos os outros neurônios, como ilustrado na Figura 4.5. O laço de recorrência possui um grande impacto na capacidade de aprendizagem e no desempenho da rede (dos Santos & Von Zuben, 2000). Além disso, este laço envolve a utilização de ramos particulares compostos de unidades de retardo ( $z^{-1}$ ), resultando em um comportamento dinâmico não-linear, assumindo que a rede possui componentes não-lineares.



**Figura 4.5.** Arquitetura recorrente de rede sem nenhuma camada intermediária.

#### 4.2.1.2. Métodos de Aprendizagem

A capacidade de *aprendizagem* é uma das características marcantes das RNAs. Uma rede neural aprende, basicamente, através de um processo iterativo de ajuste de pesos e limiares (bias). Atualmente, existem processos mais sofisticados de aprendizagem (ou *treinamento*), que são capazes de ajustar não apenas os pesos da rede, mas também sua arquitetura e as funções de ativação dos neurônios (Von Zuben, 1996, Kwok & Yeung, 1997, de Castro *et al.*, 1999a,b; de Castro & Von Zuben, 1999c).

*Definição 4.2:* *Aprendizagem* (ou *treinamento*) é o processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um mecanismo de apresentação de estímulos fornecidos pelo ambiente no qual a rede está inserida. O tipo de treinamento é definido pela forma na qual os parâmetros são modificados (Haykin, 1999).

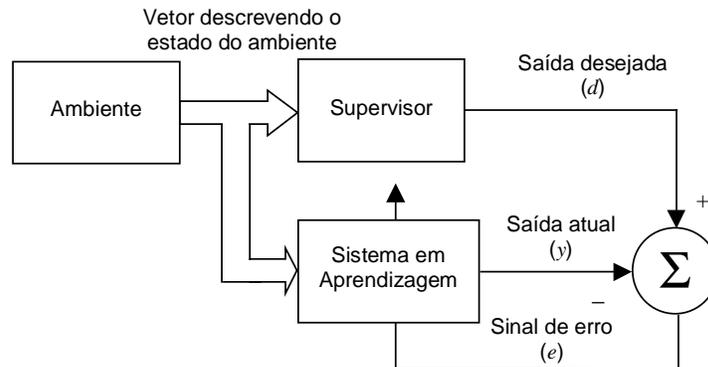
Esta definição de aprendizagem implica na seguinte seqüência de eventos:

- Apresentação de estímulos à rede neural;
- Alteração dos parâmetros livres da rede; e
- Novo padrão de resposta ao ambiente.

Os principais paradigmas de aprendizagem são: (1) supervisionada, (2) não-supervisionada, e (3) por reforço.

##### 4.2.1.2.1. Aprendizagem Supervisionada

Trata-se de um paradigma de aprendizagem, no qual um *supervisor* possui conhecimento sobre o ambiente em que a rede está inserida. Este conhecimento está representado sob a forma de um conjunto de amostras de *entrada-saída*. O ambiente, por sua vez, é desconhecido. A Figura 4.6 ilustra esta abordagem. Os parâmetros da rede são ajustados pela combinação do sinal de entrada com um sinal de erro, que é a diferença entre a saída desejada e a fornecida pela rede.



**Figura 4.6.** Diagrama de blocos do processo de aprendizagem supervisionada.

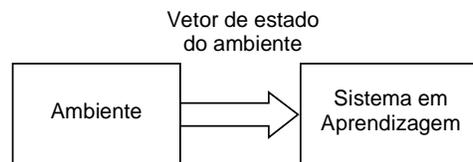
Seja  $t$  o índice que denota tempo discreto ou, mais precisamente, o intervalo de tempo do processo iterativo responsável pelo ajuste de pesos do neurônio  $k$ . O único sinal de saída  $y_k(t)$ , do neurônio  $k$ , é comparado com uma *saída desejada*, denominada  $d_k(t)$ . Conseqüentemente, um sinal de erro  $e_k(t)$  é produzido:

$$e_k(t) = d_k(t) - y_k(t). \quad (4.2)$$

#### 4.2.1.2.2. Aprendizagem Não-Supervisionada

No processo de *aprendizagem não-supervisionada* ou *auto-organizada*, não existe um supervisor para avaliar o desempenho da rede em relação ao conjunto de treinamento (Figura 4.7), ou seja, os dados são não-rotulados. A rede se adapta a regularidades estatísticas dos dados de entrada, desenvolvendo a habilidade de criar representações internas para codificar características da entrada e, assim, gerar novas classes automaticamente. Geralmente os algoritmos auto-organizados utilizam aprendizagem competitiva.

Na *aprendizagem competitiva*, os neurônios de saída da rede competem entre si para se tornarem ativos. Um único neurônio de saída é ativado a cada iteração. Esta característica torna o algoritmo apropriado para descobrir características estatísticas salientes, que podem ser utilizadas para classificar um conjunto de padrões de entrada.



**Figura 4.7.** Diagrama de blocos dos processos auto-organizados.

Existem três elementos básicos para uma regra de aprendizagem competitiva:

- Um conjunto de neurônios iguais, exceto pelos pesos das conexões;
- Um limite imposto ao peso de cada neurônio; e
- Um mecanismo de competição entre os neurônios. Aquele que vencer a competição é chamado de vencedor (*winner-takes-all*).

Neurônios individuais aprendem a se especializar em grupos (ou clusters) de padrões similares, tornando-se *detectores de características* para diferentes classes de padrões de entrada.

Em sua forma mais simples, uma rede competitiva possui uma única camada de neurônios de saída, totalmente interconectados. Também existem conexões laterais entre os neurônios, como indicado na Figura 4.8, capazes de efetuar uma inibição lateral entre os neurônios vizinhos.

Para um neurônio  $k$  ser o vencedor, seu campo induzido local  $v_k$  em relação a um determinado padrão  $x$  deve ser o maior de toda a rede. O sinal de saída  $y_k$  do neurônio vencedor  $k$  é setado em 1, e o sinal de saída de todos os outros neurônios que perderam a competição é setado em 0

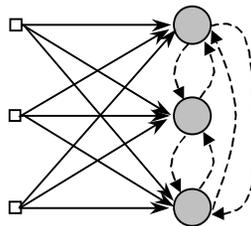
$$y_k = \begin{cases} 1 & \text{se } v_k > v_j \quad \forall j, j \neq k \\ 0 & \text{demais casos} \end{cases}, \quad (4.3)$$

onde o campo induzido local  $v_k$  representa a ação combinada das entradas positivas e laterais do neurônio.

Se um neurônio não responde a um determinado padrão de entrada, nenhuma aprendizagem ocorre. Por outro lado, se um neurônio ganha a competição, um ajuste  $\Delta w_{k,j}$  é aplicado ao vetor de pesos  $w_{k,j}$  deste neurônio vencedor

$$\Delta w_{k,j} = \begin{cases} \alpha(x_j - w_{k,j}) & \text{se } k \text{ vence a competição} \\ 0 & \text{se } k \text{ perde a competição} \end{cases}, \quad (4.4)$$

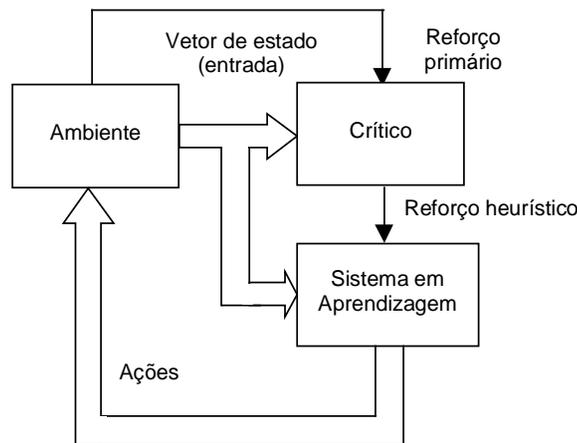
onde  $\alpha$  é a taxa de aprendizagem. Esta regra possui o efeito geral de mover o vetor de pesos  $w_{k,j}$  do neurônio vencedor  $k$  na direção do correspondente padrão de entrada  $x$ .



**Figura 4.8.** Arquitetura de uma rede competitiva simples com conexões diretas (feedforward) excitatórias da entrada para a saída e conexões laterais inibitórias (setas tracejadas).

#### 4.2.1.2.3. Aprendizagem por Reforço

A *aprendizagem por reforço* enfatiza a aprendizagem do indivíduo através da interação direta com o ambiente, sem se basear em uma supervisão ou um modelo completo deste ambiente, visando minimizar um índice escalar de desempenho. A Figura 4.9 ilustra um tipo de aprendizagem por reforço (Haykin, 1999) baseado em um crítico que converte um *signal primário de reforço* recebido do ambiente em um sinal de reforço de maior qualidade chamado *signal de reforço heurístico*. O objetivo da aprendizagem é *minimizar uma esperança do erro acumulado*, ao invés de simplesmente tratar o erro atual.



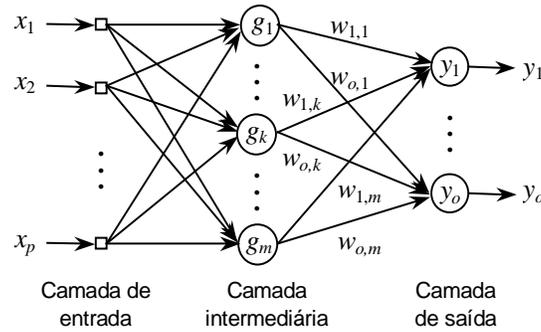
**Figura 4.9.** Diagrama de blocos do processo de aprendizagem por reforço.

## 4.2.2. Modelos Neurais e Algoritmos de Treinamento

Existem diversas arquiteturas e algoritmos de treinamento distintos para as redes neurais artificiais. Dentre elas destacam-se o perceptron de uma ou mais camadas, as redes neurais de função de base radial (RBF – *radial basis function*), as redes de Kohonen (1982) e as redes de Hopfield. Esta seção tem por objetivo descrever apenas as redes do tipo RBF, as de Kohonen (auto-organizadas) e as redes de Hopfield discretas, que foram utilizadas nesta tese principalmente para efeito de comparação de desempenho em aplicações em que foram utilizados os algoritmos de engenharia imunológica a serem propostos.

### 4.2.2.1. Redes Neurais de Base Radial (RBF)

Uma rede neural com função de ativação de base radial (RBF – *radial basis function*) pode ser vista como uma rede feedforward de múltiplas camadas (Seção 4.2.1.1.2), contendo três camadas com papéis completamente distintos (Broomhead & Lowe, 1988). A camada de entrada é composta por unidades sensoriais que conectam a rede a seu ambiente. A segunda camada (única camada intermediária) aplica uma transformação não-linear do espaço de entrada para o espaço intermediário. As unidades intermediárias não-lineares têm ativação local e suas respostas correspondem às ativações das funções de base radial.



**Figura 4.10.** Rede neural com funções de ativação de base radial (RBF).

A camada de saída é linear, fornecendo cada saída da rede como uma combinação linear das respostas das unidades intermediárias (Haykin, 1999; Karayiannis & Mi, 1997). A Figura 4.10 ilustra uma rede RBF com múltiplas saídas (note a presença de um único conjunto de pesos ajustáveis,  $\mathbf{w}$ , entre as unidades intermediárias e de saída).

Para um vetor de entrada  $p$ -dimensional  $\mathbf{x} = (x_1, \dots, x_p)$ , onde  $\mathbf{x} \in \mathbf{X} \subset \mathfrak{R}^p$ , a saída da rede RBF pode ser determinada da seguinte forma

$$y_i = \mathbf{w}_i^T \mathbf{g} = \sum_{j=1}^m w_{i,j} g_j, \quad i = 1, \dots, o \quad (4.5)$$

onde  $\mathbf{w}_i = [w_{i,1}, \dots, w_{i,m}]^T$ ,  $i = 1, \dots, o$ , são os vetores de peso para cada neurônio  $i$  de saída,  $\mathbf{g} = [g_1, g_2, \dots, g_m]^T$  é o vetor com os valores de ativação das funções base, e  $o$  é a quantidade de neurônios de saída da rede. Dado um conjunto de vetores protótipos  $\mathbf{c}_j \in \mathfrak{R}^p$ ,  $j = 1, \dots, m$ , a saída de cada RBF é

$$g_j = h(\|\mathbf{x} - \mathbf{c}_j\|, \rho_j), \quad j = 1, \dots, m \quad (4.6)$$

onde  $h(\cdot)$  é a função base e  $\|\cdot\|$  é uma norma, geralmente Euclidiana, definida no espaço de entrada. As escolhas de  $h$  consideradas na literatura incluem  $h(r) = r$  (linear),  $h(r) = r^3$  (cúbica),  $h(r) = r^2 \log r$  (thin-plate spline),  $h(r) = \exp(-r^2/2)$  (Gaussiana),  $h(r) = \text{sqrt}(r^2+1)$  (multiquádrica), and  $h(r) = 1/\text{sqrt}(r^2+1)$  (multiquádrica inversa), onde em todos os casos  $r$  é o raio escalonado  $\|\mathbf{x} - \mathbf{c}_j\|/\rho_j$  (Broomhead & Lowe, 1988; Poggio & Girosi, 1990; Chen *et al.*, 1991; Elanayar & Shin, 1994; Whitehead & Choate, 1996).

As funções de base radial foram originalmente propostas na implementação de métodos de interpolação, e suas propriedades como interpolantes têm sido extensivamente estudadas (Poggio & Girosi, 1990; Freeman & Saad, 1995). Neste contexto, uma função de mapeamento  $y: \mathfrak{R}^N \rightarrow \mathfrak{R}$  satisfazendo a Equação (4.5), para  $o = 1$ , deve ser determinada. Considere um conjunto de  $N$  dados amostrais  $\{\mathbf{x}_i \in \mathfrak{R}^p \mid i = 1, \dots, N\}$ . Se os valores desejados de  $y$  são conhecidos para todas as  $N$  amostras, ou seja  $\{d_i \in \mathfrak{R} \mid i = 1, \dots, N\}$ , então cada função base  $h(\cdot)$  pode ser centralizada em uma destas amostras. Assim, existem tantos centros (vetores protótipos)  $\mathbf{c}_j$  quanto dados de treinamento  $m = N$  (Powell, 1987). Em notação matricial

$$\mathbf{H} \mathbf{w} = \mathbf{d}, \quad (4.7)$$

onde os vetores  $\mathbf{d}$  e  $\mathbf{w}$  de dimensão  $N \times 1$  representam o vetor de saídas desejadas e o vetor de pesos de saída, respectivamente, e  $\mathbf{H}$  é uma matriz  $N \times N$ , denominada matriz de interpolação, que assume a forma:

$$\mathbf{H} = \begin{bmatrix} h(\|x_1 - c_1\|, \rho_1) & \cdots & h(\|x_1 - c_m\|, \rho_m) \\ \vdots & \ddots & \vdots \\ h(\|x_N - c_1\|, \rho_1) & \cdots & h(\|x_N - c_m\|, \rho_m) \end{bmatrix}. \quad (4.8)$$

A solução para o problema de interpolação da Equação (4.7) é dada por

$$\mathbf{w} = \mathbf{H}^{-1} \mathbf{d} \quad (4.9)$$

Resta-nos garantir que a matriz  $\mathbf{H}$  seja não-singular, tal que a inversa de  $\mathbf{H}$  exista.

O teorema de Michelli (1986) estabelece como único pré-requisito para a não-singularidade de  $\mathbf{H}$  (existência de  $\mathbf{H}^{-1}$ ) que as  $N$  amostras sejam diferentes, independentemente dos valores de  $N$  e  $p$ . Este teorema é válido para as funções multiquádrica e sua inversa e para as demais funções de base radial (RBF).

De acordo com Broomhead & Lowe (1988), o processo de interpolação descrito acima pode não ser uma boa estratégia para o treinamento de redes do tipo RBF para algumas classes de problemas, devido a uma má capacidade de generalização da rede resultante. Além disso, se  $N$  é muito grande, e/ou existem muitos dados redundantes, as chances de se obter uma matriz  $\mathbf{H}$  não-inversível também são grandes. A restrição de ter tantas funções de base radial quanto amostras de treinamento torna o problema sobre-determinado. Para superar estas dificuldades computacionais, a complexidade da rede deve ser reduzida, requerendo uma aproximação para uma solução regularizada (Poggio & Girosi, 1990). A abordagem empregada envolve a busca de uma solução sub-ótima em um espaço de dimensão reduzida. Um novo conjunto de funções base  $\{g_j, j = 1, \dots, m_1, m_1 < N\}$ , assumidas linearmente independentes, deve ser definido. O novo conjunto de centros  $\{c_j | j = 1, \dots, m_1\}$  deve ser determinado e o vetor  $\mathbf{w}^*$ , assumindo  $\sigma = 1$ , deve ser tal que resolva o seguinte problema de otimização  $\min_{\mathbf{w}} \|\mathbf{H}\mathbf{w} - \mathbf{d}\|^2$ . Desconsiderando o uso do parâmetro de regularização, a solução  $\mathbf{w}^*$  do problema para  $m_1 < N$ , é simplesmente dada por

$$\mathbf{w}^* = \mathbf{H}_1^+ \mathbf{d} = (\mathbf{H}_1^T \mathbf{H}_1)^{-1} \mathbf{H}_1^T \mathbf{d}, \quad (4.10)$$

onde  $\mathbf{H}_1$  é dada pela Equação (4.11), e  $\mathbf{H}_1^+$  é a pseudo-inversa da matriz  $\mathbf{H}_1$  (Broomhead & Lowe, 1988).

$$\mathbf{H}_1 = \begin{bmatrix} h(\|x_1 - c_1\|, \rho_1) & \cdots & h(\|x_1 - c_{m_1}\|, \rho_{m_1}) \\ \vdots & \ddots & \vdots \\ h(\|x_N - c_1\|, \rho_1) & \cdots & h(\|x_N - c_{m_1}\|, \rho_{m_1}) \end{bmatrix}. \quad (4.11)$$

Haykin (1999) sugere que este método é relativamente insensível à utilização do parâmetro de regularização, desde que haja uma escolha adequada dos centros das funções de base radial.

#### 4.2.2.1.1. Centros Fixos

A abordagem mais simples para o treinamento de uma rede RBF é assumir fixos os parâmetros  $\mathbf{c}_j$  e  $\rho_j$ ,  $j = 1, \dots, m_1$ , das funções de base radial. As localizações dos centros devem ser escolhidas de alguma forma, geralmente baseada nos dados de treinamento. Assumindo uma função Gaussiana como função de base radial e tomando  $\rho_j = \rho$ ,  $j = 1, \dots, m_1$ , onde  $\rho$  é uma medida de dispersão associada ao desvio padrão dos centros, resulta

$$h(\|\mathbf{x} - \mathbf{c}_j\|^2, \rho) = \exp\left(-\frac{1}{2} \frac{\|\mathbf{x} - \mathbf{c}_j\|^2}{\rho^2}\right) \quad j = 1, \dots, m_1, \quad (4.12)$$

onde  $\mathbf{x}$  é o vetor de entradas,  $\mathbf{c}_j$  a localização do  $j$ -ésimo centro e  $\rho$  é o desvio padrão das RBFs Gaussianas, dado pela seguinte equação

$$\rho = \frac{d_{\max}}{\sqrt{2m_1}} \quad (4.13)$$

onde  $m_1$  é a quantidade de funções de base radial e  $d_{\max}$  é a distância máxima entre os centros.

A Equação (4.13) garante que as RBFs individuais não serão muito achatadas ou alargadas (Haykin, 1999). Assim, os únicos parâmetros que ainda precisam ser definidos são os pesos da camada de saída da rede. Um procedimento direto para isso é empregar o método da pseudo-inversa apresentado na Equação (4.10).

#### 4.2.2.1.2. Seleção Auto-Organizada de Centros

Uma outra abordagem para estimar a localização apropriada dos centros das RBFs na camada intermediária é utilizando uma etapa de auto-organização antes de estimar os pesos da camada de saída da rede. O algoritmo de clusterização  $k$ -means tende a posicionar os centros das RBFs nas regiões do espaço de entrada onde dados significativos estão presentes. A determinação de um valor adequado para  $m_1$ , a quantidade de centros, pode requerer experimentação. Sejam  $\{\mathbf{c}_k(t)\}_{k=1}^{m_1}$  os centros das RBFs na iteração  $t$ . Assim, o algoritmo  $k$ -means opera como a seguir:

1. *Inicialização*: Escolha diferentes valores aleatórios para os centros  $\mathbf{c}_k(t)$ ,  $k = 1, \dots, m_1$ .
2. *Amostragem*: Escolha aleatoriamente uma amostra  $\mathbf{x}(t)$  a partir dos dados de entrada.
3. *Seleção*: Determine o índice  $k$  do centro mais próximo ao vetor  $\mathbf{x}(t)$  de acordo com

$$k(\mathbf{x}(t)) = \arg \min_k \|\mathbf{x}(t) - \mathbf{c}_k(t)\|, \quad k = 1, \dots, m_1, \quad (4.14)$$

onde  $\mathbf{c}_k(t)$  é o centro da  $k$ -ésima RBF na iteração  $t$ .

4. *Atualização*: Ajuste os centros das RBFs de acordo com a seguinte regra

$$\mathbf{c}_k(t+1) = \begin{cases} \mathbf{c}_k(t) + \alpha[\mathbf{x}(t) - \mathbf{c}_k(t)], & k = k(\mathbf{x}) \\ \mathbf{c}_k(t), & \text{outros casos} \end{cases} \quad (4.15)$$

onde  $\alpha \in (0,1)$  é a taxa de ajuste.

5. *Ciclo*: Repita os Passos de 2 a 5 até que nenhuma variação significativa em  $c_k$ ,  $k = 1, \dots, m_1$ , seja observada.

Uma limitação do algoritmo  $k$ -means é que ele só é capaz de atingir soluções ótimas locais, dependendo da escolha inicial dos centros. Assim, uma escolha inicial inapropriada dos centros pode resultar em centros posicionados em regiões do espaço com baixa densidade de dados.

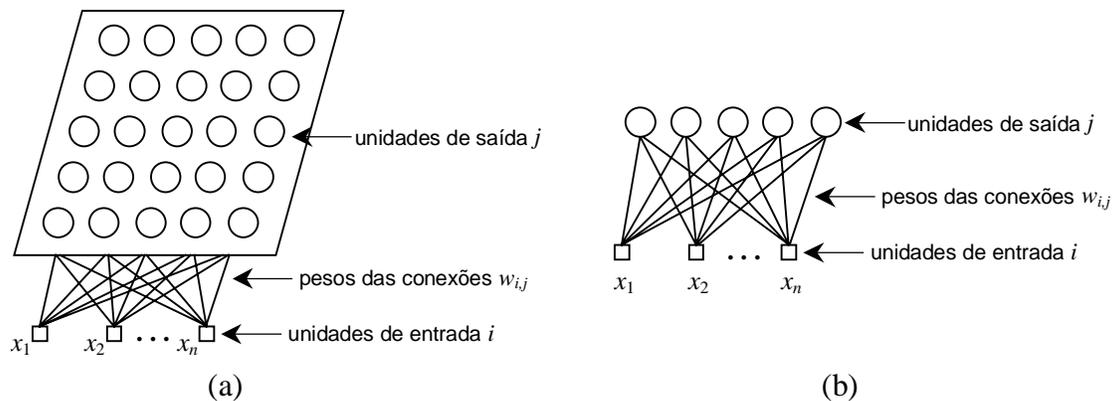
#### 4.2.2.2. Redes de Kohonen

As redes que possuem aprendizado competitivo ou auto-organizado (Seção 4.2.1.2.2) são chamadas *redes competitivas* ou *auto-organizadas*, onde os neurônios de saída competem entre si para estarem ativos ou não. Apenas um neurônio de saída, ou um por grupo, é ativado a cada iteração. As redes auto-organizadas de Kohonen (1982) são caracterizadas pela formação de um mapa topográfico dos padrões de entrada e, por isso, são denominadas de *mapas auto-organizáveis de Kohonen* (SOM – *self-organizing maps*).

No SOM, a localização espacial dos neurônios auto-organizados vai indicar características intrínsecas aos padrões de entrada. Seu principal objetivo é a transformação adaptativa e ordenada de um conjunto de dados de entrada em um mapa (*grid*) uni- ou bidimensional de saída. Os dados de entrada que apresentam semelhanças entre si são agrupados em regiões do mapa de saída, formando classes ou agrupamentos denominados *clusters*.

Durante o processo de auto-organização do mapa, a unidade do cluster cujo vetor de pesos mais se aproxima do vetor dos padrões de entrada é escolhida como sendo a *vencedora*. A *unidade vencedora e suas unidades vizinhas têm seus pesos atualizados*. Além disso, estas redes possuem parâmetros variáveis com o tempo, como a taxa de aprendizagem  $\alpha$ , e um raio de vizinhança  $N_R$  que indica quais unidades serão atualizadas simultaneamente.

A Figura 4.11 apresenta arquiteturas típicas de um SOM, considerando configurações de vizinhança unidimensional e bidimensional, embora dimensões mais elevadas possam ser consideradas (Costa, 1999). Além disso, dada a dimensão, a quantidade de unidades ou neurônios de saída pode ser arbitrada e mantida fixa, ou então definida automaticamente pelo algoritmo de treinamento (Fritzke, 1993; Cho, 1997; de Castro & Von Zuben, 1999b).



**Figura 4.11.** Arquiteturas típicas de um SOM. (a) Bidimensional. (b) Unidimensional.

O algoritmo de treinamento padrão do SOM é apresentado abaixo (Kaski, 1997):

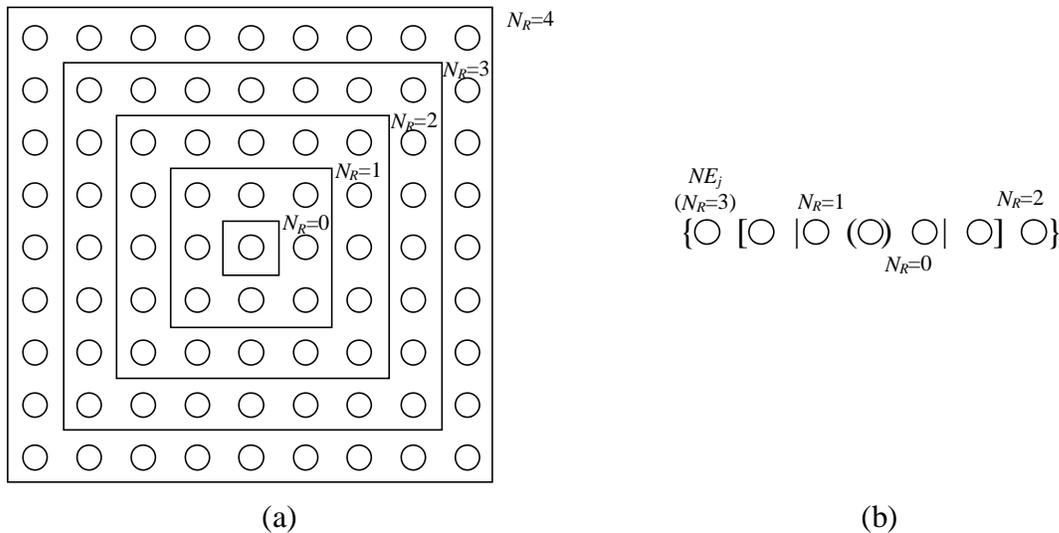
1. Inicialização e definições de parâmetros
  - Inicialize os pesos  $w_{i,j}$ ;
  - Defina  $N_R$  e  $\alpha$ .
2. Enquanto a condição de parada é falsa, faça:
  - 2.1. Para cada  $j$  determine:
    - 2.1.1.  $J = \arg \min_j \{ \|w_j - x_l\| \}$
    - 2.1.2.  $\forall j \in N_R$  de  $J$ , e  $\forall k$ :  

$$w_{j,k}(t+1) = w_{j,k}(t) + \alpha [x_{l,k} - w_{j,k}(t)]$$
  - 2.2. Atualize a taxa de aprendizagem
  - 2.3. Reduza o raio de vizinhança
3. Teste a condição de parada.

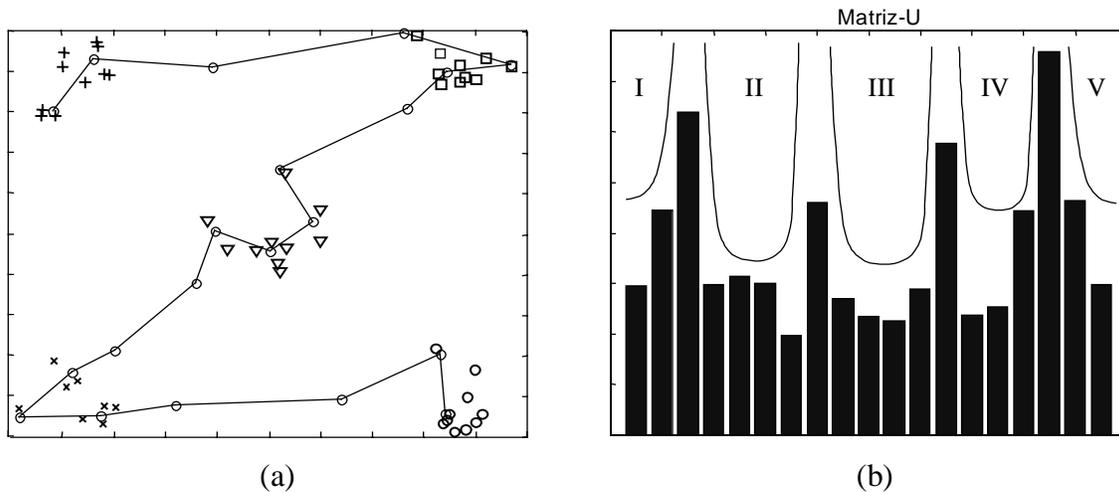
A taxa de aprendizagem decresce lentamente com o tempo. A formação de um mapa ocorre em duas fases: 1) formação inicial da ordem correta do mapa, e 2) convergência final.

No processo de treinamento,  $N_R$  é decrementado de 1 após um determinado número de iterações até que  $N_R$  seja igual a zero. Aqui, diz-se que ocorreu uma iteração quando o vetor de padrões  $x_1, x_2, \dots, x_N$  tiver sido apresentado uma vez. Os pesos das conexões para o nó  $J$  e todos os nós em sua vizinhança definidos por  $N_R$ , como mostrado na Figura 4.12, são atualizados. Outros critérios de vizinhança podem ser definidos.

Um método de visualização de um SOM treinado, denominado matriz de distâncias unificadas, ou matriz-U, foi desenvolvido por Ultsch (1993), com o objetivo de permitir a detecção visual das relações topológicas dos neurônios.



**Figura 4.12.** Vizinhos do nó  $j$ ,  $NE_j(N_R)$  para arquitetura bidimensional (a), e unidimensional (b).



**Figura 4.13.** Representação da matriz-U. (a) Dados de treinamento e topologia final de uma rede de Kohonen (SOM) para vizinhança unidimensional. (b) Matriz-U.

A idéia básica da matriz-U é usar a mesma métrica utilizada no treinamento da rede para calcular as distâncias entre os pesos dos neurônios adjacentes. Para o caso unidimensional, o resultado é um *histograma* no qual a altura de cada barra corresponde à distância do neurônio a seu vizinho mais próximo. Desta forma, é possível visualizar similaridades entre neurônios e portanto determinar a quantidade de clusters existente no conjunto amostral. A Figura 4.13 ilustra a matriz-U para um problema simples de classificação, onde o conjunto de treinamento possui cinco clusters linearmente separáveis. Cada vale da Figura 4.13(b) corresponde a um cluster distinto.

#### 4.2.2.3. Redes de Hopfield Discretas

As redes de Hopfield (1982, 1984) possuem suas raízes na Física Estatística, mais do que em observações biológicas. As redes de Hopfield são totalmente recorrentes e podem ser descritas como um *sistema dinâmico cujo espaço de estados contém um conjunto de pontos fixos (estáveis) que representam as memórias fundamentais do sistema*. Consequentemente, estas redes possuem uma propriedade emergente que ajuda a recuperar informações e lidar com ruídos. Devem ser vistas como uma *memória associativa* ou *memória endereçável por conteúdo*.

Memórias associativas fornecem mecanismos para armazenar e recuperar dados baseados em seu conteúdo, ao invés de baseado em um endereço de armazenamento. A informação está distribuída pelo sistema (rede neural).

A rede de Hopfield originalmente proposta utiliza uma matriz simétrica de pesos, de modo que as realimentações existentes geram um comportamento dinâmico não-linear caracterizado apenas pela presença de pontos fixos. Após um transitório inicial, qualquer estado inicial em que a rede se encontre será conduzido até um dos pontos fixos da dinâmica, geralmente associados a memórias fundamentais. A validade deste argumento

pode ser mostrada através de uma *função de energia* e da garantia de que a dinâmica é sempre dissipativa.

Nesta arquitetura não há neurônios intermediários (Figura 4.5). O estado completo da rede é visível em todos os momentos. A restauração de um padrão é feita colocando o padrão desconhecido na rede como um estado inicial e permitindo à dinâmica estabilizar. Iterações sucessivas do algoritmo mudam o estado da rede de forma a colocá-la próxima a uma das memórias fundamentais. O processo de restauração não é perfeito. Em alguns casos, a rede pode estabilizar em um padrão que não é uma das memórias fundamentais, conhecido como *estado espúrio*. Estas memórias são caracterizadas, por exemplo, por combinações lineares de um número ímpar de memórias fundamentais.

### Função de Energia

Considere uma rede de Hopfield com pesos simétricos  $w_{i,j} = w_{j,i}$  e  $w_{i,i} = 0$ , e seja  $s_i$  o estado do neurônio  $i$ , onde  $i = 1, \dots, N$ . A *função de energia* da versão discreta da rede de Hopfield considerada aqui é definida por

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{j,i} s_i s_j. \quad (4.16)$$

A variação da energia  $\Delta E$  devido a uma variação  $\Delta s_j$  do estado do neurônio  $j$  é dada por

$$\Delta E = -\Delta s_j \sum_{i=1}^N w_{ji} s_i. \quad (4.17)$$

Os mínimos locais da superfície de energia correspondem a *atratores* do espaço de estados, que são as memórias da rede. Para garantir a emergência da memória associativa, duas condições devem ser satisfeitas:

1. As memórias fundamentais armazenadas pela rede serem estáveis; e
2. Os padrões estáveis possuem uma base de atração (região de influência).

### Algoritmo de Treinamento

O algoritmo apresentado aqui é colocado para padrões bipolares  $\{-1, +1\}$ , mas a função de ativação pode ser modificada facilmente para aceitar padrões binários  $\{0, 1\}$ .

1. Utilize a *regra de Hebb* (Equação (4.18)) ou a regra da projeção (Equação (4.19)) para inicializar os pesos (Haykin, 1999):

$$\mathbf{W} = \frac{1}{N} \sum_{i=1}^p \mathbf{v}_i \mathbf{v}_i^T - \frac{p}{N} \mathbf{I}. \quad (4.18)$$

$$\mathbf{W} = \mathbf{P}(\mathbf{P} \cdot \mathbf{P}^+)^{-1} \mathbf{P}^T. \quad (4.19)$$

2. Para cada padrão de entrada  $\mathbf{x}$ , faça
  - 2.1. Coloque a ativação inicial igual ao vetor de entrada externo  $\mathbf{x}$ 

$$y = x$$
  - 2.2. Para cada saída  $y_i$ 
    - 2.2.1. Calcule a entrada de rede

$$y_{in_i} = \sum_j y_j w_{j,i}$$

2.2.2. Determine a nova ativação (sinal de saída)

$$y_i = \begin{cases} 1 & \text{se } y_{in_i} > 0 \\ y_i & \text{se } y_{in_i} = 0 \\ -1 & \text{se } y_{in_i} < 0 \end{cases}$$

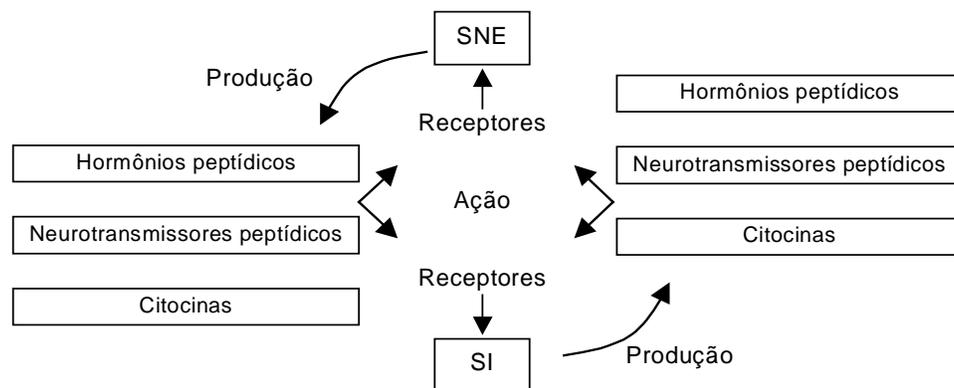
2.2.3. Propague o valor de  $y_i$  para todas as outras unidades.

2.3. Teste a convergência

### 4.3. O Sistema Imunológico e o Sistema Nervoso Central

A interação do sistema imunológico (SI), o sistema nervoso central (SNC) e o sistema endócrino (SE), os dois últimos algumas vezes denominados conjuntamente de sistema *neuroendócrino* (SNE), é importante para a garantia da integridade e homeostase do organismo. Nesta seção, pretendemos discutir brevemente sobre a relação entre estes sistemas, mais especificamente entre o SI e o SNC. Para isso, serão discutidos aspectos cognitivos do sistema imunológico e serão apresentadas similaridades e diferenças entre o SI e o SNC, e entre a engenharia imunológica e as RNAs apresentadas nas Seções 3.2 e 4.2, respectivamente.

A influência do SNC no SE pode ser claramente observada em situações como, por exemplo, a necessidade de se correr ao banheiro em um momento de estresse psicológico, como uma prova ou a defesa de uma tese. Sensações físicas de desconforto são comuns quando as pessoas estão submetidas a crises de ansiedade. Estudos tentando relacionar anomalias como câncer e outras doenças com o temperamento e condições psicológicas do indivíduo levaram ao surgimento da *psiconeuroimunologia* (Ader *et al.*, 1991), que aborda de forma integrada as diversas funções dos sistemas nervoso e imunológico, tentando esclarecer as doenças conhecidas como *psicossomáticas* (Bonamin, 1994).



**Figura 4.14.** Um mecanismo de comunicação entre o SI e o SNE através do compartilhamento de moléculas sinalizadoras e seus receptores (adaptado de Blalock, 1994).

Uma análise em sentido oposto, ou seja, de como o sistema imunológico influencia o sistema neuroendócrino, também tem sido de fundamental importância para a compreensão da relação entre eles. Várias observações, como a alteração de padrões de atividade neural e hormonais seguindo a inoculação antigênica em indivíduos saudáveis, permitiram a apresentação de um modelo molecular bidirecional de comunicação entre o SNE e o SI, como apresentado na Figura 4.14 (Blalock, 1994). Sem nos prendermos aos muitos detalhes desta interação, objetivamos salientar que os mesmos sinais moleculares e seus receptores podem ser utilizados tanto para a regulação do próprio sistema em questão quanto para a comunicação entre eles.

### 4.3.1. Aspectos Cognitivos

Existe uma grande dificuldade por parte da comunidade científica na obtenção de um consenso sobre o significado de palavras e expressões como *cognição*, *intencionalidade* e *mente*. Os conceitos variam para diferentes ramos de pesquisa, como psicologia, filosofia e neurociência. Esta seção tem por objetivo discutir sucintamente algumas abordagens sobre o que se entende por estes temas, enfocando o problema da cognição, e também apresentar argumentos da literatura sugerindo que o sistema imunológico possui propriedades suficientes para ser caracterizado como um sistema cognitivo. Além disso, o leitor será posicionado acerca de minha opinião pessoal quanto à visão cognitiva do sistema imunológico, a qual também é defendida por diversos pesquisadores, principalmente imunologistas. É indiscutível que os pontos de vista a serem apresentados, assim como a teoria cognitiva do sistema imunológico a ser defendida nesta seção, também não são de consenso geral e muito menos completas. Apesar disso elas visam fornecer subsídios para um estudo mais amplo da intitulada *cognição imunológica*.

O *cognitivismo* surgiu no ano de 1956 a partir de dois encontros realizados em Cambridge e Dartmouth, nos quais as principais idéias que forneceriam as bases da *ciência cognitiva* moderna foram propostas (Gardner, 1995). A ciência cognitiva é o resultado de um esforço multidisciplinar que traz para a psicologia, ciência da computação e inteligência artificial vários aspectos de neurobiologia, lingüística e filosofia (Edelman, 1992). Ela se baseia no conceito de *representações mentais* e em um conjunto de suposições coletivamente denominadas de *posição funcional*. Partindo deste ponto de vista, as pessoas se comportam de acordo com o conhecimento gerado por representações mentais simbólicas. Assim, a *cognição* consiste na manipulação destes símbolos. Os fenômenos psicológicos são descritos por processos funcionais. A eficácia de tais processos reside na possibilidade de interpretar itens como símbolos, de formas abstratas e bem definidas, de acordo com um conjunto de regras. Este conjunto de regras constitui o que é conhecido como *sintaxe*. O exercício das regras sintáticas é uma forma de *computação*, no sentido amplo de manipulação simbólica, ou operação com símbolos, de acordo com um procedimento definido. Tais processos bem definidos constituem representações *semânticas*, que indicam formas precisas de especificação do que os símbolos representam para o mundo.

Sob esta perspectiva computacional, a proposta central da cognição é de que a inteligência, incluindo a humana, pode realmente ser definida como uma forma de computação com representações simbólicas (Varela *et al.*, 1991). A noção central, neste caso, é a de

representação ou *intencionalidade*. *Sistemas intencionais* têm como função legitimar predicados mentais, assim como uma máquina de Turing deve estabelecer regras para a interpretação de objetos (Dennett, 1978). O argumento cognitivo defende que o comportamento inteligente pressupõe a habilidade de representar o mundo de alguma forma. Provavelmente, um dos aspectos mais controversos da cognição esteja na sua premissa de que a única forma de se obter um comportamento inteligente ou uma intencionalidade é através da suposição de que as ações ocorrem através da manipulação de representações que são fisicamente executáveis por um código simbólico no cérebro, máquina ou outro sistema. Assim, um dos problemas que precisam ser resolvidos é como relacionar a intencionalidade, ou os estados representativos, com as mudanças físicas ocorridas durante uma ação.

A expressão *cognitivo* chegou à imunologia vinda da psicologia, na qual ela se refere às funções superiores da mente, incluindo o reconhecimento de objetos, identificação do organismo e *intencionalidade* (Mitchison, 1994). O objetivo inicial era enfatizar que o sistema imunológico sabe o que ele está procurando quando encontra um antígeno, ou seja, sua *representação* (ou *organização*) *interna* lhe fornece uma certa intencionalidade. Alguns imunologistas como I. Cohen, F. Varela, A. Coutinho e N. Jerne tratam a *cognição imunológica* como sendo baseada principalmente no conceito de próprio/não-próprio. Outros autores mais conservadores preferem definir a imunologia como uma *ciência de reconhecimento* (Edelman, 1975, 1992 – Cap. 8), ou visualizar o sistema imunológico como um *sistema sensorial* (Blalock, 1994) e não um sistema cognitivo. Tada (1997) apresentou um paradigma diferente, propondo que sistemas como o imunológico e o nervoso central podem ser denominados de “supersistemas”.

A idéia de *reconhecimento* surgiu na imunologia desde seu início. A identificação dos elementos estranhos ao organismo implicitamente requer que algum componente esteja efetuando a identificação, ou reconhecimento, destes elementos. O reconhecimento é um evento *perceptivo* e, portanto, deve se apoiar em algum aparato cognitivo (Tauber, 1994, 1997). Esta visão reflete a riqueza escondida em termos como reconhecimento, aprendizagem e memória, características pertinentes ao sistema imunológico.

Cohen (1992a) define um sistema cognitivo como a seguir:

*Definição 4.3:* Um *sistema cognitivo* é capaz de extrair informações e experiências de dados de entrada através da manipulação de informações já contidas no próprio sistema, ou seja, ele age através de um senso de direção ou intencionalidade. Sendo assim, os sistemas cognitivos não são processadores passivos ou armazenadores de informação, e sim projetados para manipular informações particulares do domínio no qual eles operam.

Niels K. Jerne (1974a,b, 1984, 1985) é considerado o verdadeiro autor do modelo cognitivo do sistema imunológico (Tauber, 1997). A visão cognitiva da teoria da rede imunológica apresentada na Seção 2.10, é fundamentada em duas premissas:

- O sistema imunológico é constituído por um universo de imagens que só são reconhecidas por estarem expressas em uma linguagem conhecida pelo sistema;

- O sistema imunológico se auto define, ou seja, ele é projetado para conhecer a si próprio.

Dessa forma, os elementos próprios provocam um determinado tipo de resposta, enquanto os não-próprios induzem outra resposta, baseada não na natureza intrínseca do não-próprio, mas no fato de que o sistema imunológico enxerga o antígeno externo no *contexto* de invasão ou degeneração. O elemento chave é o anticorpo, que atua como um antígeno através de domínios idiotípicos. Existe portanto, uma *imagem interna* do universo antigênico. O reconhecimento mútuo entre os elementos do sistema imunológico (linfócitos B e anticorpos) forma uma vasta rede interconectada de elementos que se comunicam entre si, a chamada *rede idiotípica* ou *rede imunológica*.

Quando o sistema imunológico é visto como uma entidade cognitiva, ele representa uma complementação do sistema nervoso central (ou simplesmente sistema neural). Certamente, para que o SI reconheça um determinado elemento, ele, primeiramente, precisa percebê-lo e em seguida decidir que atitude tomar. Trata-se, portanto, de um tipo de modelo cognitivo, onde os sistemas imunológico e neural podem ser vistos de forma análoga. Cada um possui características perceptivas: a capacidade de discernir entre o universo interno e o externo. O processamento de informação é fundamental para a operação de ambos e as respectivas propriedades perceptivas estão ligadas a mecanismos efetores. Além das analogias funcionais, a crescente evidência da interdependência deles, através de moléculas mensageiras, neurotransmissores e hormônios, já foi brevemente discutida no início da Seção 4.3. Adotando-se o modelo em rede proposto por Jerne (1974a), as similaridades entre estes sistemas aumentam ainda mais.

Cohen (1992a,b) também apresentou uma *abordagem contextualista* para a cognição imunológica. Considerando os sistemas cognitivos como aqueles capazes de extrair informações e experiências de dados de entrada através da manipulação de informações já contidas no sistema, Cohen (1992b) sugeriu o chamado *homúnculo imunológico*:

*Definição 4.4:* O *homúnculo imunológico* é a imagem interna do próprio adquirida pelo reconhecimento primário dos antígenos próprios, tanto no timo, como na periferia (Cohen, 1992b).

Esta definição está fundamentada na idéia de que o sistema imunológico será capaz de realizar sua tarefa mais eficientemente através da coleta e processamento de informações caso ele seja dotado de uma representação interna de seu ambiente, definindo o foco de atenção junto ao qual ele deve operar. Os antígenos serão reconhecidos como não-próprios pois eles são apresentados em um *contexto* que denuncia sua patologia. A autoimunidade é tratada como uma característica normal do sistema imunológico, que constantemente tenta identificar e monitorar os elementos do hospedeiro. Se estes antígenos próprios forem alterados de forma contextual, seus significados mudam e uma resposta imunológica é iniciada. Assim, o próprio não é mais uma entidade, e sim emerge dinamicamente em um processo de auto-identificação, que muda continuamente ao longo da vida do indivíduo.

De forma similar, uma teoria baseada na definição do próprio e da rede imunológica foi desenvolvida por Coutinho e seus colaboradores (Coutinho *et al.*, 1984; Varela *et al.*, 1988; Coutinho, 1989; Varela & Coutinho, 1991). A postura adotada sugere que as propriedades

globais do sistema imunológico, como tolerância e distinção próprio/não-próprio, não podem ser compreendidas através da análise de componentes individuais (imunologia celular e molecular). Foi proposto que características essenciais das redes imunológicas, como estrutura (conectividade) e dinâmica, juntamente com a teoria da seleção clonal, fornecem uma abordagem poderosa para o estudo de alguns aspectos cognitivos do sistema imunológico, como a memória por exemplo. A memória imunológica é tida como uma característica clonal, pelo menos no contexto de respostas secundárias, e o reconhecimento antigênico (diretamente relacionado à memória) é provavelmente a propriedade imunológica de maior apelo cognitivo.

Baseado em um ponto de vista conceitualmente diferente, Blalock (1994) aborda o sistema imunológico como um mecanismo sensorial, assim como o sistema neural, porém ele assume que a cognição só pode ser atribuída a estímulos do tipo fisiológicos, emocionais, etc. Foi proposto que o sistema imunológico é capaz de reconhecer e responder a estímulos que não podem ser detectados pelo sistema neural. Estes estímulos foram denominados de não cognitivos, e incluem elementos como bactérias, vírus, tumores, antígenos, etc., que passariam despercebidos não fosse pelo sistema imunológico. Um vírus não pode ser visto a olho nu, nem cheirado, não possui gosto ou odor e não faz nenhum barulho, mas pode ser percebido pelos sintomas que ele causa. Isto ocorre através do reconhecimento deste estímulo por células do sistema imunológico, que o converte em informações químicas sob a forma de hormônios, neurotransmissores e citocinas (Figura 4.14). Estes sinais são recebidos pelo SNE resultando em mudanças fisiológicas e psicológicas. Aparentemente a função sensorial do SI imita o SNE no sentido de que um determinado estímulo promove uma resposta particular resultando em uma resposta fisiológica.

É interessante notar que o termo cognitivo nas abordagens imunológicas discutidas implica em consciência: a propriedade da intencionalidade aparece em conjunto com a da personalidade. Ambas são dependentes da história e experiência individuais de cada um e levam em consideração uma profunda noção de identidade (próprio). O objetivo é o de enfatizar que princípios cognitivos englobando a idéia de imagem interna podem ser aplicados de forma genérica ao sistema imunológico, que trata:

- *A busca por um contexto*: quando agir;
- *A extração de sinais a partir de ruídos*: como focalizar o reconhecimento; e
- *O problema da resposta*: que decisão tomar.

É possível afirmar que o SI não apenas pode ser visto como um sistema cognitivo, mas também é capaz de complementar e/ou regular as capacidades neurais de reconhecimento e tomada de decisão através da percepção de estímulos que não podem ser detectados fisiologicamente de forma direta. É evidente, como foi brevemente discutido nas seções anteriores, que existe uma reciprocidade (e comunicação direta) de ações entre estes e outros sistemas.

Finalmente, Tada (1997) cunhou o termo “supersistema” para designar sistemas vitais altamente integrados como o sistema imunológico, o sistema nervoso central e a embriogênese. Os diversos elementos de um supersistema se relacionam através de adaptação e co-adaptação mútuas entre seus componentes, criando um sistema dinâmico

auto-regulado através de auto-organização. Trata-se de um sistema auto-contido, porém aberto a estímulos ambientais que podem ser traduzidos em mensagens internas para os processos de auto-regulação e expansão. Um supersistema é caracterizado pela sua auto-regulação, pela geração de seus diversos tipos de componentes através de processos estocásticos seguidos de seleção e adaptação (conseqüências da auto-organização), individualidade e tomada de decisão em resposta a estímulos internos e externos.

### 4.3.2. Similaridades e Diferenças

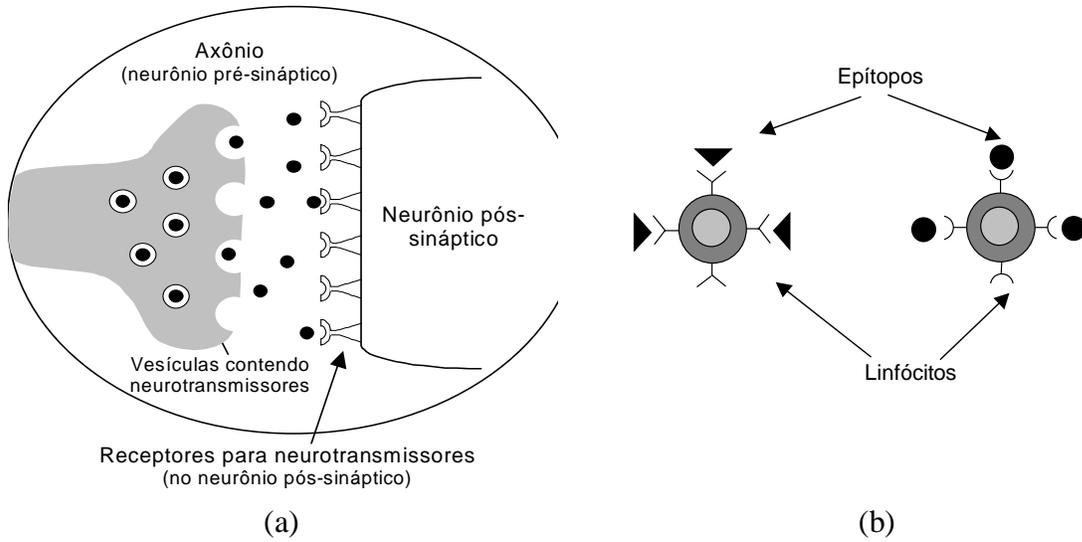
Durante a discussão sobre os aspectos cognitivos do sistema imunológico já foi possível perceber algumas similaridades e diferenças entre o SI e o SNC. Parte delas pode ser diretamente confrontada com os respectivos modelos artificiais, tanto na engenharia imunológica quanto nas redes neurais artificiais.

Dentre a vasta gama de similaridades e diferenças entre o SI e o SNC destacam-se (Jerne, 1974; Hoffmann, 1986; Vertosick & Kelly, 1989, 1991; Ishida, 1993, 1996; Dasgupta, 1997, 1999):

- Ambos os sistemas são constituídos por uma grande quantidade e diversidade de células. No homem o SI possui aproximadamente  $10^{12}$  linfócitos e o sistema neural consiste de aproximadamente  $10^{10}$  neurônios. Em ambos os casos células individuais são altamente específicas na execução de suas tarefas;
- A diversidade celular em cada sistema permite que respostas apropriadas sejam geradas para uma grande variedade de estímulos. O sistema imunológico reconhece e responde a formas diversas de moléculas. O SNC reconhece e responde a estímulos recebidos de nossos cinco sentidos: visão, audição, paladar, tato e olfato. Dessa forma, o SI pode ser visto como nosso “sexto sentido”, como proposto por Blalock (1994) e discutido na seção anterior;
- O SNC é geralmente decomposto em suas partes sensorial e motora. Uma divisão análoga em função de reconhecimento e efetora do sistema imunológico pode ser feita, onde os mecanismos efetores são responsáveis pela eliminação do antígeno;
- Nos sistemas neurais, os processos de aprendizagem e aquisição de memória são conseqüências da alteração dos pesos das conexões entre os neurônios, também chamados de sinapses (Figura 4.1), e não da alteração dos próprios neurônios. Além disso, o cérebro permite que as memórias sejam endereçadas por conteúdo, de forma que a morte freqüente de neurônios individuais não afeta drasticamente o desempenho do cérebro como um todo. No caso do sistema imunológico, o conhecimento está armazenado nas próprias células, ou em clones específicos de células e a aprendizagem ocorre através de variações nos padrões de receptores celulares;
- Sob o ponto de vista da teoria da rede imunológica proposta por Jerne (1974a), e apresentada na Seção 2.10, em ambos os sistemas as células estão funcionalmente conectadas entre si sob a forma de uma rede, apresentando funções estimulatórias ou supressivas (inibitórias) entre células vizinhas. No SNC as conexões entre as células são mediadas por axônios, sinapses e dendritos, podendo ser visualizadas em

um microscópio. No caso da teoria da rede imunológica, os componentes do sistema são capazes de reconhecer e serem reconhecidos, oferecendo uma rede de comunicação com características menos óbvias do que as redes neurais;

- As redes neurais utilizam sinais eletroquímicos de comunicação, enquanto o SI emprega ingredientes químicos;
- As redes neurais apresentam hierarquias, enquanto o sistema imunológico é descentralizado;
- Ambos utilizam processamento paralelo, onde os estímulos são processados simultaneamente em diferentes partes dos sistemas;
- A capacidade de apresentar memória distingue estes dois sistemas de todos os outros. A memória de um evento pode permanecer por muitos e muitos anos;
- Em ambos os casos, o conhecimento armazenado durante a vida não é automaticamente passado para as gerações subsequentes, mesmo que essa herança forneça uma aparente vantagem seletiva para o indivíduo. Este aspecto particular de como a evolução imunológica influencia na evolução da espécie será estudado na Seção 4.5;
- Ambos os sistemas possuem uma profunda noção de si mesmos, ou seja, do próprio. Uma das tarefas do sistema imunológico é distinguir entre o próprio e o não-próprio, e eliminar tudo aquilo que for categorizado como não-próprio (Seção 2.9). A distinção entre próprio e não-próprio é, de forma similar, uma das funções primitivas do sistema nervoso, e é fundamental para o enigma da consciência;
- As células dos dois sistemas possuem receptores que permitem a comunicação entre si e/ou com elementos externos (Figura 4.15);
- Algumas células do sistema imunológico aprendem a reconhecer e responder a formas particulares (antígenos) somente em contextos específicos. Este fenômeno é similar ao reconhecimento dependente do contexto dos sinais recebidos pelo SNC;
- O SI compartilha com o SNC a propriedade de que um reconhecimento definitivo pode ser obtido aplicando-se um estímulo “fuzzy”. Uma memória pode ser recuperada por um estímulo similar, mas não necessariamente idêntico, ao estímulo original. Em imunologia este fenômeno é chamado de memória reativa cruzada (Ada & Nossal, 1987; Smith *et al.*, 1997), como discutido na Seção 2.8.
- Ambos os sistemas penetram a maioria dos tecidos do corpo, mas não entram em contato direto devido à *barreira cérebro-sangue (blood-brain barrier)*;
- O funcionamento ideal de ambos os sistemas é aparentemente insensível a pequenos detalhes em seus elementos ou conexões. Os sistemas devem ser projetados de tal forma que, independente do padrão de conexões ou de interação celular, um repertório (população) de elementos deve existir e ser diverso o suficiente para descrever o universo a ser representado;
- O corolário do aspecto anterior indica que ambos os sistemas são altamente tolerantes a falhas, devido principalmente a redundância e diversidade de seus componentes.



**Figura 4.15.** Receptores celulares. (a) Os neurônios liberam neurotransmissores quando a atividade elétrica alcança outro neurônio. (b) Os linfócitos possuem receptores celulares capazes de reconhecer padrões de epítomos ou complexos MHC/peptídeo.

### 4.3.3. Engenharia Imunológica e Redes Neurais Artificiais

Baseados na discussão acima sobre as similaridades e diferenças entre o SI e o SNC, considerando a estrutura formal, apresentada na Seção 3.4, para a engenharia imunológica e os modelos de redes neurais artificiais discutidos no início deste capítulo, é possível fazer um paralelo entre a EI e as RNAs, como descrito na Tabela 4.1.

**Tabela 4.1.** EI × RNAs.

<b>Característica</b>	<b>EI</b>	<b>RNA</b>
Unidade básica	Célula composta por uma cadeia de atributos, conexões associadas e um limiar de afinidade	Neurônio composto por uma função de ativação, conexões associadas e um limiar de ativação
Interações com outras unidades	As células possuem conexões que a identificam (receptores) e permitem reconhecer outros elementos. Estas conexões podem ser ponderadas indicando o grau de interação com outros elementos	Os pesos sinápticos das conexões podem assumir valores positivos ou negativos indicando uma ação excitatória ou inibitória dos outros neurônios
Atividade	A célula possui uma imagem interna do ambiente que é comparada com a informação recebida	O neurônio processa a informação recebida
Conhecimento	Armazenado nos pesos das conexões e nas cadeias de atributos de cada célula	Armazenado nos pesos das conexões e na forma das funções de ativação
Aprendizagem	Ocorre através da modificação das cadeias de atributos das células e de seus pesos associados	Ocorre através da modificação dos valores e quantidade de pesos sinápticos, juntamente com a alteração das formas das respectivas funções de ativação
Limiar	Determina a ligação (reconhecimento) entre uma célula e o estímulo apresentado	Determina a ativação do neurônio
Robustez	Escalonável, auto-tolerante, flexível e tolerante a falhas	Flexível e tolerante a falhas
Localização	As células podem se deslocar	Os neurônios residem em posições fixas na rede
Comunicação	Ocorre através do contato celular, também representado por um conjunto de conexões	Ocorre através das ligações sinápticas
Estado	Concentração e/ou afinidade entre uma célula e seu estímulo	Nível de ativação dos neurônios de saída
Controle	Uma reação adaptativa determina o tipo de interação entre os componentes do sistema	Um algoritmo de aprendizagem determina o tipo de ajuste dos parâmetros livres da rede

#### 4.4. Computação Evolutiva

*Simular processos evolutivos naturais* objetivando resolver problemas de otimização tem demonstrado ser um método rápido e robusto. A otimização do comportamento de um sistema por processos evolutivos representa uma abordagem poderosa para a aprendizagem de máquina (*machine-learning*) e fenômenos auto-organizados. A implementação computacional de métodos de simulação da evolução, chamada *computação evolutiva*, possibilita a determinação de ótimos globais para várias classes de problemas cujas soluções eram, até então, desconhecidas ou insatisfatórias. A computação evolutiva pode ser dividida em três grandes grupos (Bäck & Schwefel, 1993; Bäck *et al.*, 2000a):

- Programação evolutiva (PE) introduzida por Fogel *et al.* (1966);
- Estratégias evolutivas (EE) inicialmente propostas por Ingo Rechenberg (1973); e
- Algoritmos genéticos (GAs – *genetic algorithms*), propostos por John Holland (1975).

Todos estes ramos da computação evolutiva seguem o mesmo *algoritmo evolutivo básico*, que será discutido na Seção 4.4.2. Porém, na forma como foram inicialmente propostos, eles possuem algumas diferenças entre si, como resumidas na Tabela 4.2.

Atualmente, o GA constitui o método evolutivo mais amplamente utilizado e, por isso, será discutido em maiores detalhes na Seção 4.4.3. Sua relevância nesta tese se dá sob a forma de aspectos comparativos com algumas das ferramentas propostas. Além disso, serão apresentados comentários gerais sobre a programação genética (PG), um outro ramo da computação evolutiva, devido a sua importância para uma das possíveis extensões desta tese. Por último, será apresentado um modelo de *espécies* em computação evolutiva, também para efeitos comparativos mais adiante com uma das ferramentas a serem propostas.

**Tabela 4.2.** Principais características dos algoritmos evolutivos (Bäck & Schwefel, 1993).

	EE	PE	GA
<b>Representação</b>	Valores reais	Valores reais	Cadeias binárias
<b>Auto-adaptação</b>	Desvio padrão e covariância	Variâncias (na meta-PE)	Nenhuma
<b>Fitness</b>	Valor da função objetivo	Valor escalonado da função objetivo	Valor escalonado da função objetivo
<b>Mutação</b>	Principal operador	Único operador	Operador secundário
<b>Recombinação</b>	Diversos tipos. Importante para a auto-adaptação	Nenhuma	Principal operador
<b>Seleção</b>	Determinística	Probabilística	Probabilística

#### 4.4.1. Evolução Natural

A *evolução* não é um processo imediato, pois ocorre através do acúmulo de comportamentos cada vez mais apropriados dentro de uma determinada espécie, ou de um determinado ambiente.

C. Darwin (1859) apresentou as seguintes hipóteses para explicar a origem das espécies:

1. Os filhos tendem a ser em maior número que os pais;
2. O número de indivíduos de uma espécie permanece aproximadamente constante;
3. De (1) e (2) conclui-se que haverá luta pela sobrevivência; e
4. Dentro de uma mesma espécie os indivíduos apresentam pequenas diferenças.

Este *princípio da seleção natural* indica que os indivíduos cujas variações se adaptam melhor ao ambiente terão maior probabilidade de sobreviver e reproduzir. Como complemento a estas hipóteses propostas por Darwin, três hipóteses adicionais foram incorporadas, conduzindo ao chamado *neodarwinismo*:

5. Algum processo de variação continuada deve ser responsável pela introdução de novas informações junto ao material genético dos organismos;
6. Não há limite para a sucessão de variações que podem ocorrer; e
7. A seleção natural é o mecanismo para a preservação das novas informações correspondentes a uma maior adaptação.

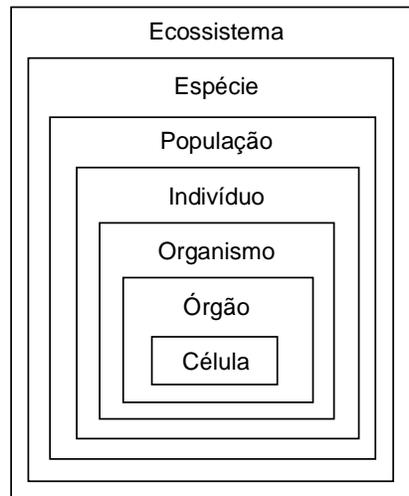
A *seleção* (competição e exclusão) existe para estatisticamente eliminar (ou reduzir o número de) indivíduos cujo comportamento é pouco apropriado, pois nem todos os indivíduos podem sobreviver na presença de recursos limitados. É importante notar que a evolução otimiza o comportamento, e não o material genético (Atmar, 1994). Sob o ponto de vista da computação, é possível definir evolução da seguinte forma:

*Definição 4.5:* *Evolução* é um método de busca (em paralelo) de indivíduos com um alto índice de adaptação a um determinado critério, recorrendo-se a um número finito de candidatos, sujeitos a condições de adaptação variáveis.

##### 4.4.1.1. Terminologia Biológica

A terminologia biológica a ser empregada nos algoritmos evolutivos representa uma analogia às entidades biológicas reais, sendo que as entidades computacionais corresponderão invariavelmente a estruturas bem mais simples que seus equivalentes biológicos.

A Figura 4.16 apresenta a hierarquia dos sistemas biológicos. Os algoritmos evolutivos geralmente operam com uma população de indivíduos, de uma única espécie, evoluindo dentro de um ecossistema (ou ambiente) pré-definido.



**Figura 4.16.** Hierarquia dentro dos sistemas biológicos.

Como apresentado na Figura 4.16, todos os organismos vivos são compostos por células, e cada célula contém o mesmo conjunto de um ou mais *cromossomos* – cadeias de DNA – que servem como identificadores do organismo. Um cromossomo pode ser conceitualmente dividido em *genes* (blocos funcionais de DNA), cada qual codificando uma proteína particular. Os tipos diferentes de características são chamados *alelos*. Cada gene está localizado em uma posição particular (*locus*) no cromossomo (Mitchel, 1998).

Muitos organismos possuem múltiplos cromossomos em cada célula. A coleção completa do material genético é chamada de *genoma* do organismo. O termo *genótipo* refere-se ao conjunto particular de genes contido em um genoma. Dois indivíduos que possuem o mesmo genoma têm o mesmo genótipo. O genótipo leva ao *fenótipo* do organismo, ou seja, suas características físicas e mentais, como a cor dos olhos, pele, etc.

Organismos cujos cromossomos aparecem em pares são chamados *diploides*, enquanto organismos com cromossomos que aparecem individualmente são ditos *haploides*. Geralmente, espécies com reprodução *sexuada* (entre dois indivíduos) são diploides, enquanto espécies com reprodução *assexuada* (clonagem de um único indivíduo, como por exemplo a reprodução dos linfócitos) são *haploides*. Durante a reprodução sexuada, ocorre a *recombinação genética* (ou *crossover*): em cada um dos pais, genes são trocados entre cada par de cromossomos para a formação de um *gameta* (cromossomo único), e os gametas dos dois pais formam pares constituindo um conjunto completo de cromossomos diploides. Na reprodução haploide, genes são trocados entre os cromossomos de cada um dos pais. Os *descendentes* (*offsprings*) estão sujeitos à *mutação*, na qual *nucleotídeos* (unidades elementares de DNA) são mudados de pai para filho. A *aptidão* (*adaptabilidade* ou *fitness*) de um organismo é tipicamente considerada equivalente à probabilidade que este indivíduo possui de se reproduzir (*viabilidade*), ou como uma função da quantidade de descendentes que ele pode ter (*fertilidade*).

**Tabela 4.3.** Terminologia biológica × terminologia computacional (assumindo representação binária).

<b>Biologia</b>	<b>Computação</b>
Cromossomo haplóide	Cadeia de bits que representa um candidato a solução
Gene	Um único bit, ou um bloco de bits
Alelo	Valores que podem ser assumidos para cada gene do cromossomo
Crossover	Troca de material genético entre cromossomos
Mutação	Troca aleatória de um determinado bit do cromossomo
Genótipo	Configuração de bits no cromossomo de um indivíduo
Fenótipo	Decodificação de um ou mais cromossomos

A Tabela 4.3 faz um paralelo entre a terminologia biológica apresentada e a terminologia computacional a ser utilizada nos algoritmos evolutivos. A grande maioria das aplicações de algoritmos evolutivos emprega codificação haplóide baseada em um único cromossomo.

#### 4.4.2. Algoritmo Evolutivo Básico

Os algoritmos evolutivos básicos são geralmente implementados da seguinte forma (Atmar, 1994):

1. Uma população inicial de indivíduos é escolhida aleatoriamente. O tamanho  $N$  da população é geralmente grande.
2. A população é reproduzida, com erro, de forma que cada pai gere  $n$  descendentes. O mecanismo de introdução de erro durante a reprodução deve ser estocástico e variável.
3. A qualidade do comportamento do sistema é avaliada para todos os indivíduos da população, pais e filhos, baseada em uma medida de adaptabilidade ou fitness de cada indivíduo. Uma de duas condições é geralmente implementada: 1) os  $N$  melhores indivíduos são selecionados para reproduzir e compor a próxima geração, 2)  $N$  dos melhores indivíduos são selecionados probabilisticamente. O tamanho da população é geralmente restrito.
4. O processo se repete, voltando ao passo 2. A convergência é assumida quando uma solução pré-definida é atingida, ou um número fixo de gerações foi executado.

*Definição 4.6:* O processo de busca de soluções dentro de um conjunto de candidatos a solução é chamado de *busca em um espaço de buscas*.

*Definição 4.7:* *Superfície de adaptação* é a representação de um mapeamento do espaço de todos os possíveis genótipos para as respectivas medidas de adaptabilidade (fitness) (Mitchel, 1995).

#### 4.4.2.1. Formalização do Processo

Dois espaços de estados são inerentes aos processos evolutivos: um espaço de estado genotípico,  $\mathbf{G}$ , (codificação), e um espaço fenotípico,  $\mathbf{P}$ , (comportamental ou de expressão do genótipo). Analogamente, dois alfabetos também são definidos; um alfabeto de entrada do ambiente,  $\mathbf{I}$ , e um alfabeto de saída,  $\mathbf{Z}$ . A evolução da população dentro de uma única geração está esquematizada na Figura 4.17.

Considere uma população de genótipos,  $g_i \in \mathbf{G}$ . Existe uma função de mapeamento  $f_1$  tal que cada  $g_i$  é transformado em um fenótipo  $p_i$ . O mapeamento é inerentemente de muitos para um, pois existem infinitos genótipos que podem resultar em um mesmo fenótipo.

*Definição 4.8:* *Epigênese* corresponde ao desenvolvimento celular sensível às condições locais. Dito de outra forma, epigênese é a teoria segunda a qual a constituição dos seres vivos se inicia a partir de uma célula sem estrutura e se faz mediante sucessiva formação e adição de novas partes que, previamente, não existiam no ovo fecundado.

O primeiro mapeamento da Figura 4.17, *epigênese*, incorpora regras de expressão genética, crescimento e desenvolvimento, em um contexto local:

$$f_1 : \mathbf{I} \times \mathbf{G} \rightarrow \mathbf{P} \quad (4.20)$$

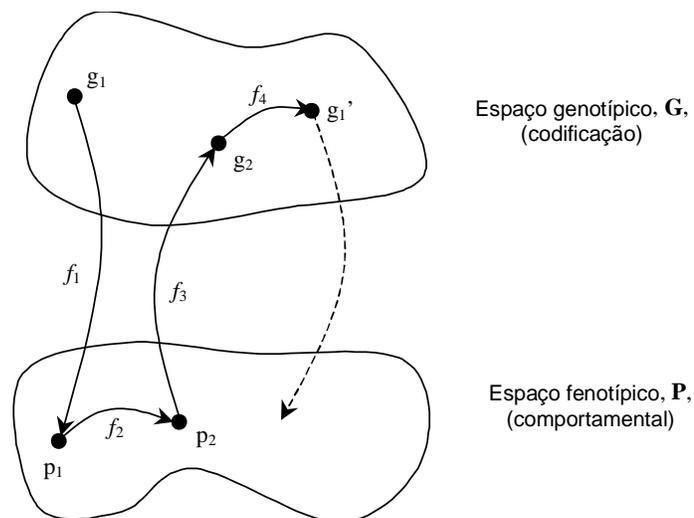
O segundo mapeamento, *seleção*, descreve os processos de seleção, imigração e emigração de indivíduos dentro de uma população local:

$$f_2 : \mathbf{P} \rightarrow \mathbf{P} \quad (4.21)$$

Os efeitos da seleção ocorrem somente em  $\mathbf{P}$  e não afetam diretamente as estruturas codificadas em  $\mathbf{G}$ .

O terceiro mapeamento, *representação*, descreve a representação fenotípica dentro de uma população antes da reprodução:

$$f_3 : \mathbf{P} \rightarrow \mathbf{G} \quad (4.22)$$



**Figura 4.17.** Evolução de uma população a cada geração.

O quarto mapeamento, *mutação*, incorpora variação aleatória e direcionada na codificação (incluindo reparo e recombinação):

$$f_4 : \mathbf{G} \rightarrow \mathbf{G} \quad (4.23)$$

Somente através da seleção repetida de comportamentos adequados em  $\mathbf{P}$ , geração após geração, uma aprendizagem apropriada pode ser acumulada em  $\mathbf{G}$ . A evolução ocorre através da repetição indefinida destes quatro mapeamentos.

### 4.4.3. Algoritmos Genéticos

*Definição 4.9:* Os *Algoritmos genéticos* (GAs) são procedimentos computacionais estocásticos cujos métodos de busca modelam os fenômenos biológicos de herança genética e seleção natural (Michalewicz, 1996).

Os algoritmos genéticos simples constituem modelos abstratos da evolução natural e operam com uma população de tamanho fixo e indivíduos representados por “cadeias genéticas” de comprimento fixo. Novas populações evoluem através da seleção probabilística proporcional ao fitness dos indivíduos, produzindo, via crossover e mutação, descendentes semelhantes aos pais.

#### 4.4.3.1. Características Principais e Escopo de Aplicações

Dentre as principais características dos algoritmos genéticos podemos citar (Mitchel, 1995):

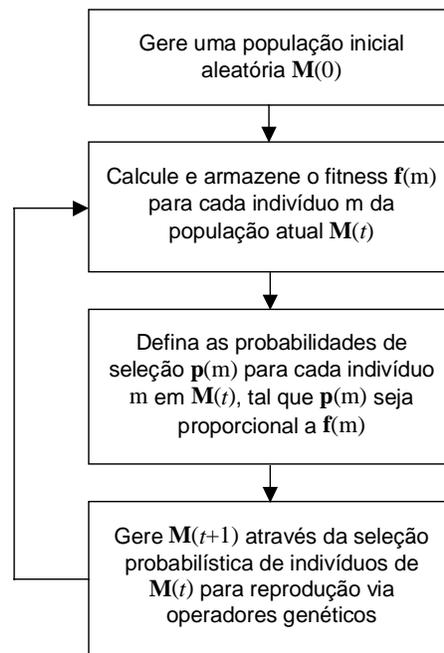
- Busca sobre uma população de pontos, e não sobre um único ponto;
- Utilização de funções de custo (funções objetivo, de fitness, ou de adaptabilidade), ao invés de derivadas ou outro tipo de conhecimento auxiliar; e
- Utilização de regras de transição probabilísticas, e não determinísticas.

Dado um problema, como saber se um algoritmo genético irá resolvê-lo de forma eficiente? Não existe nenhuma resposta rigorosa para esta pergunta, mas vários aspectos intuitivos podem ser levantados:

- Se o espaço de buscas é grande, não é perfeitamente suave e nem unimodal, não é conhecido, ou se a função de fitness é ruidosa, o GA se apresenta como uma ferramenta competitiva;
- Se o espaço é suave, ou unimodal, então métodos de *gradiente* ou de *hill climbing* serão muito superiores;
- Se o espaço de buscas é conhecido (como no caso do *caixeiro viajante*), heurísticas podem ser introduzidas em métodos específicos, de forma que eles apresentem bom desempenho.

#### 4.4.3.2. Algoritmo Genético Clássico e Modificado

Nesta seção, introduziremos o algoritmo genético clássico e suas modificações de forma a garantir um melhor desempenho em termos de qualidade da solução (busca pelo ótimo global) e velocidade de convergência.



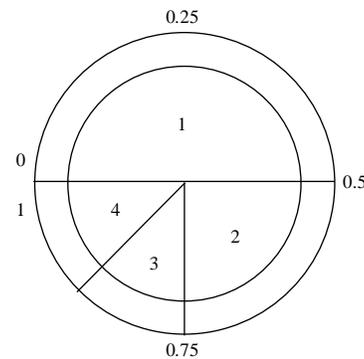
**Figura 4.18.** Algoritmo genético clássico.

Definida uma representação genética apropriada ao problema, o *algoritmo genético básico* pode ser descrito pelo fluxograma da Figura 4.18 (De Jong, 1994). Este algoritmo possui as seguintes características: 1) codificação binária, 2) reprodução e seleção natural via *Roulette Wheel*, 3) crossover simples, e 4) mutação.

Na seleção via *Roulette Wheel* (RW), a probabilidade de seleção de um cromossomo (indivíduo) é diretamente proporcional a seu valor da função de fitness. A Figura 4.19 ilustra o RW para uma população composta por quatro indivíduos. Rodar uma roleta significa obter um valor de um gerador de números aleatórios com distribuição uniforme no intervalo  $[0,1]$ . O valor obtido vai definir o cromossomo escolhido de acordo com as marcações na Figura 4.19(b).

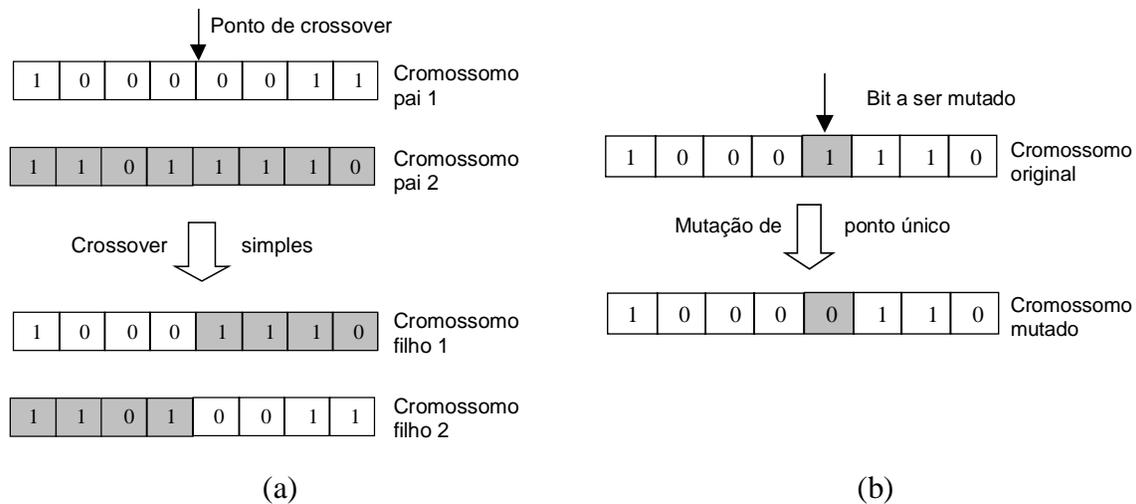
$N$	Cromossomo	Fitness	Graus
1	0001100101010	6.0	180
2	0101001010101	3.0	90
3	1011110100101	1.5	45
4	1010010101001	1.5	45

(a)



(b)

**Figura 4.19.** Exemplo de Roulette Wheel para uma população com quatro indivíduos.



**Figura 4.20.** Operadores b sicos de recombina o. (a) Crossover simples. (b) Muta o de ponto  nico.

O *crossover simples*   realizado da seguinte forma: dados dois cromossomos (pais), escolhe-se uma posi o (locus) aleat ria destas cadeias e efetua-se a troca de material gen tico, conforme indicado na Figura 4.20(a). Cada indiv duo tem uma probabilidade  $p_c$  de sofrer crossover. No processo de *muta o*, uma posi o (locus) do cromossomo   escolhida aleatoriamente e seu bit trocado de  $0 \rightarrow 1$ , ou de  $1 \rightarrow 0$ , conforme indicado na Figura 4.20(b). A probabilidade de muta o  $p_m$  define a taxa com que cada locus pode ser mutado.

Dentre os principais problemas com o algoritmo gen tico cl ssico, destacam-se:

- A pol tica de reprodu o e sele o permite a perda do melhor indiv duo;
- A posi o do gene no cromossomo influi na probabilidade de realizar crossover; e
- Dificuldade de codifica o quando os par metros s o n meros reais.

Para solucionar estes problemas, algumas estrat gias foram propostas:

- Utiliza o de mecanismos alternativos de sele o;
- Crossover uniforme; e
- Codifica o em strings (cadeias) de n meros reais.

V rias modifica es podem ser introduzidas no algoritmo gen tico cl ssico visando amenizar os problemas citados acima. Estas modifica es englobam basicamente outros mecanismos de (1) recombina o (crossover e muta o) e (2) sele o.

#### 4.4.3.2.1. Outros Operadores de Recombina o

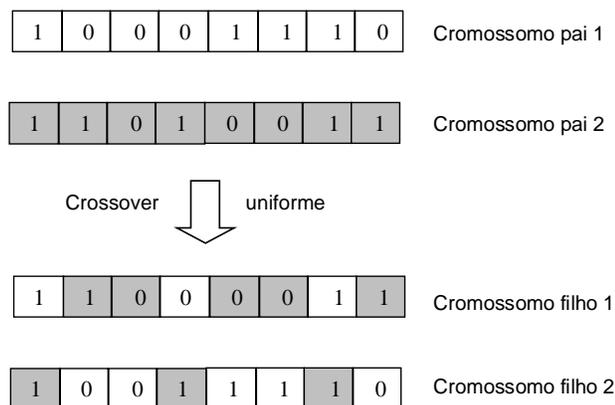
1. *Crossover* simples entre indiv duo aleat rio e o melhor indiv duo.  
O objetivo deste operador   preservar seq ncias gen ticas (partes da cadeia) do melhor indiv duo na popula o durante todo o processo evolutivo.
2. *Crossover* uniforme entre indiv duos aleat rios.

Este tipo de crossover tem a característica de não ser influenciado pela posição do gene na probabilidade de realizar crossover. São escolhidos dois indivíduos da população, determina-se a percentagem de genes que serão trocados, procede-se a um sorteio de alelos na quantidade determinada, efetuando o crossover somente dos alelos envolvidos, como ilustrado na Figura 4.21.

3. *Crossover* uniforme entre indivíduos aleatórios e o melhor indivíduo.
4. *Mutação* indutiva (somente para codificação real).  
Semelhante à mutação aleatória, só que ao invés de sortear um novo valor para o alelo, sorteia-se um valor a ser somado ao valor atual do alelo.
5. *Mutação* do melhor indivíduo.  
Pode ser aleatória ou indutiva.

#### 4.4.3.2.2. Outros Mecanismos de Seleção

1. *Elitista*: os  $n$  melhores indivíduos da população são selecionados.
2. *Aleatória*: são selecionados aleatoriamente  $n$  indivíduos da população intermediária.
  - 2.1. *Salvacionista*: seleciona-se o melhor indivíduo e os outros aleatoriamente.
  - 2.2. *Não-salvacionista*: todos os indivíduos são selecionados aleatoriamente.
3. *Por diversidade*: os indivíduos mais diversos são selecionados.
4. *Bi-classista*: são selecionados os  $p_b\%$  melhores e os  $p_w\%$  piores indivíduos.
5. *Por torneio binário*: dois indivíduos são escolhidos aleatoriamente. Um número aleatório  $r \in [0,1]$  é gerado. Caso  $r < k$  ( $k$  é um parâmetro, como por exemplo 0.75), o indivíduo com maior fitness é selecionado, senão o outro indivíduo é escolhido.



**Figura 4.21.** Crossover uniforme.

#### 4.4.4. Programação Genética

A principal questão que levou ao surgimento da programação genética foi: “Como os computadores podem aprender a resolver problemas sem serem explicitamente programados para tal?”.

A maior barreira para se atingir este objetivo está na característica determinística da grande maioria dos ramos de pesquisa em inteligência artificial. Nestes casos, as propriedades principais que um programa deve apresentar para ser um candidato à solução de problemas de interesse prático são: correção, consistência, motivação lógica, precisão, ordenação, parcimônia, e definibilidade.

A *programação genética* (PG) pode gerar programas que falham em todos estes quesitos. Ela manipula soluções corretas e incorretas, encoraja inconsistências e abordagens contraditórias, pode não apresentar um seqüenciamento lógico, é predominantemente probabilística, produz soluções não-parcimoniosas e não apresenta um critério de terminação claramente definido. Pode ser definida como a seguir:

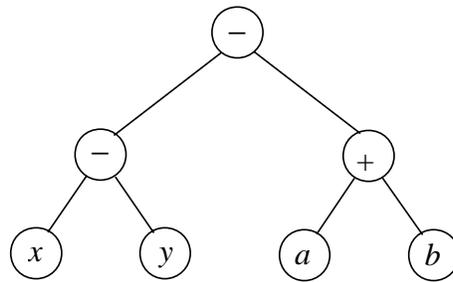
*Definição 4.10: Programação Genética* (PG) constitui um algoritmo evolutivo no qual as estruturas de dados que sofrem adaptação são programas executáveis de computador. A avaliação do fitness na programação genética envolve a execução dos programas evoluídos. Assim, a PG envolve uma busca baseada na evolução junto aos candidatos pertencentes ao espaço de possíveis programas computacionais (Bäck *et al.*, 2000a – Cap. 11).

A PG pode se constituir em uma opção especialmente interessante para o caso de programação de computadores com processamento paralelo. No contexto originalmente adotado para a PG, um programa computacional é basicamente uma seqüência de aplicações de funções a argumentos: *paradigma funcional*.

A implementação da PG é imediata quando associada a linguagens de programação que permitem a manipulação de um programa computacional na forma de estrutura de dados, inclusive por possibilitar que novos dados do mesmo tipo e recém-criados sejam imediatamente executados sob a forma de programas computacionais. Entretanto, qualquer linguagem computacional capaz de implementar (mesmo que indiretamente) a mesma estrutura de dados pode ser potencialmente empregada. Originalmente, sua linguagem é a LISP (Koza, 1992).

Para um sistema de programação genética, um problema é definido pela *representação* e pela função de *fitness*, tornando necessário um conhecimento profundo do problema em questão.

A *representação* de um problema consiste de *funções* e *terminais*, elementos básicos de um programa estruturado no paradigma funcional. Os programas computacionais na linguagem definida são os indivíduos, passíveis de representação em estruturas de dados do tipo árvore. Estes programas precisam ser executados para se obter o correspondente candidato à solução (Kinnear Jr., 1994). A Figura 4.22 ilustra uma estrutura em árvore representativa do programa computacional responsável pela avaliação da função  $f = (x - y) - (a + b)$ , sendo que  $x$ ,  $y$ ,  $a$  e  $b$  são terminais, e  $+$  e  $-$  são funções.



**Figura 4.22.** Árvore representativa de um programa em PG para avaliar a expressão  $f = (x - y) - (a + b)$ .

#### 4.4.5. Nichos e Espécies

Como discutido anteriormente (Definição 4.7), os algoritmos genéticos baseiam-se na teoria da seleção natural e operam de forma análoga à evolução natural. Entretanto, enquanto os processos evolutivos naturais mantêm uma variedade de *espécies*, cada uma ocupando um *nicho ecológico* diferente, os algoritmos genéticos tradicionais (ou clássicos) convergem rapidamente para um único ponto. Isso significa que todos os indivíduos da população tornam-se praticamente iguais. Mesmo quando existem múltiplas soluções para um determinado problema, o GA clássico localiza apenas uma delas (Mahfoud, 1995). No Capítulo 6, onde apresentaremos as aplicações das ferramentas propostas de engenharia imunológica, serão feitas comparações de desempenho com os algoritmos genéticos clássicos, um método de niching (mais especificamente o método de fitness sharing apresentado a seguir), e seus aspectos de convergência para um ou mais ótimos da função serão ilustrados e discutidos.

Intuitivamente, podemos visualizar uma espécie como sendo uma classe de organismos que apresentam características comuns, e um nicho como o local ocupado pelo elementos de uma mesma espécie.

*Definição 4.11:* Um método de niching é aquele que faz com que os algoritmos genéticos consigam manter uma população diversa de indivíduos, tornando-os capazes de localizar múltiplas soluções ótimas (subpopulações) dentro de uma única população (Goldberg, 1989).

O estudo de nichos e espécies em computação evolutiva, mais especificamente em algoritmos genéticos, está diretamente relacionado ao problema da *diversidade populacional*. Uma das principais motivações para o desenvolvimento dos métodos de niching foi a busca pela diversidade, que possui duas funções dentro dos GAs: retardar a convergência objetivando aumentar a exploração da superfície de adaptação, e permitir a determinação de múltiplas soluções.

Os mecanismos básicos de seleção dos algoritmos genéticos reproduzem os indivíduos de maior fitness e descartam aqueles com valores baixos de fitness, resultando em uma redução de diversidade da população. Três fatores fazem com que o GA clássico elimine indivíduos caracterizados por valores reduzidos de fitness e privilegie aqueles com valores

de fitness mais elevados: *pressão seletiva*, *ruído de seleção* e *quebra pelo operador*. A pressão seletiva é o resultado do valor esperado do processo de seleção: indivíduos com fitness baixos devem desaparecer de uma população com tamanho finito. O ruído de seleção é resultado da variância do processo de seleção: em uma população finita, escolhas aleatórias entre indivíduos com fitness idênticos adiciona ruído ao fitness esperado de cada indivíduo, forçando o surgimento de boas soluções. A quebra pelo operador é resultado da aplicação dos operadores de recombinação, como crossover e mutação, que são potencialmente capazes de eliminar bons candidatos à solução. As técnicas para diversificar populações geralmente reduzem um destes três fatores, ou uma combinação deles (Mahfoud, 1995).

Um método de niching deve ser capaz de formar e manter múltiplas e diversas soluções finais com valores iguais ou diferentes de fitness. Além disso, estas soluções devem ser mantidas por um longo período de tempo quando comparado ao tamanho da população. Mahfoud (1995) classificou os métodos de niching em duas categorias de comportamento: 1) espacial/temporal e de 2) único ambiente/múltiplos ambientes. Dentro da primeira categoria, um método espacial de niching forma e mantém subpopulações dentro de uma única população, enquanto um método temporal desenvolve nichos seqüencialmente ao longo do tempo. Na segunda categoria, um ambiente pode ser visto como uma função de fitness e, portanto, o algoritmo pode ser classificado como capaz de resolver problemas multi-objetivo no caso de múltiplos ambientes. A Tabela 4.4 ilustra estas categorias e como alguns algoritmos ou sistemas podem ser classificados dentro delas. Os *GAs ecológicos* induzem a formação de nichos através da utilização ou criação de múltiplos ambientes, onde cada ambiente pode ser entendido como sendo uma função de fitness (Mahfoud, 1995).

Por ser o método de niching mais empregado na literatura, descreveremos a seguir o algoritmo de *fitness sharing* para que possamos posteriormente discutir sua relação com alguns modelos imunológicos existentes, e compará-lo empiricamente com um dos algoritmos propostos nesta tese.

**Tabela 4.4.** Os métodos de niching podem ser classificados em duas categorias de comportamento: 1) espacial/temporal e de 2) único/múltiplos ambientes.

	Único ambiente	Múltiplos ambientes
Temporal	Localização seqüencial	Superespecificação GAs ecológicos
Espacial	Crowding Competição restrita Fitness sharing	GAs ecológicos Sistemas Imunológicos

#### 4.4.5.1. Fitness Sharing

Goldberg & Richardson (1987) propuseram um esquema prático que utiliza uma metáfora de *compartilhamento de fitness* (*fitness sharing*) para solucionar o problema de otimização de funções multimodais. Neste esquema, uma função de compartilhamento é definida para determinar a vizinhança e grau de compartilhamento para cada indivíduo da população. Para um dado indivíduo, o grau de compartilhamento é determinado somando-se o valor da função de compartilhamento de todos os outros indivíduos da população. Indivíduos próximos a outros requerem um alto grau de compartilhamento, enquanto indivíduos distantes entre si requerem valores pequenos de compartilhamento. O compartilhamento reduz o fitness de um indivíduo proporcionalmente à quantidade de indivíduos similares a ele dentro da população. Especificamente, o fitness compartilhado  $f_s(x_i)$  de um indivíduo é igual a seu fitness  $f(x_i)$  dividido pelo *contador de nichos*, que por sua vez é a soma dos valores das funções de compartilhamento entre este indivíduo e o restante da população:

$$f_s(x_i) = \frac{f(x_i)}{\sum_{j=1}^N sh(d(x_i, x_j))}, \quad (4.24)$$

onde  $N$  é a quantidade de indivíduos da população e  $sh(\cdot)$  é uma função de compartilhamento proporcional à distância  $d(\cdot, \cdot)$  entre dois elementos da população. A função de compartilhamento considera um *limiar de compartilhamento* (similaridade)  $\sigma_{share}$  tal que, se a distância entre dois indivíduos da população é maior ou igual  $\sigma_{share}$ , eles não afetam seus respectivos fitness. A função de compartilhamento mais empregada é (Bäck *et al.*, 2000b):

$$sh(d) = \begin{cases} 1 - \left( \frac{d}{\sigma_{share}} \right)^\gamma, & \text{se } d < \sigma_{share} \\ 0, & \text{outros casos} \end{cases}, \quad (4.25)$$

onde  $\gamma$  é uma constante (geralmente  $\gamma = 1$ ) capaz de regular a forma da função de compartilhamento. Medidas de distância genotípica ou fenotípica podem ser empregadas, de acordo com o problema a ser tratado.

## 4.5. Microevolução: Uma Evolução Dentro do Indivíduo

Na Seção 4.4, que descreve a computação evolutiva, discutimos como a evolução natural serviu de fonte inspiradora para o desenvolvimento de algoritmos computacionais capazes de simular processos evolutivos.

Os processos de *aprendizagem* ou *adaptação* podem se manifestar de três formas distintas na seleção natural:

- *Adaptação filogenética*: acúmulo de conhecimentos com reflexos nas células germinais de uma espécie;
- *Adaptação sociogenética*: comportamentos adaptativos são acumulados durante a vida dentro de um grupo (sociedade); e

- *Adaptação ontogenética*: comportamentos apropriados são aprendidos através de tentativa e erro durante a vida de um indivíduo.

Organismos individuais sofrem adaptação ontogenética de várias formas: os músculos ficam mais fortes quanto mais são utilizados, o comportamento varia ao longo do tempo, etc. Estas mudanças são rapidamente observáveis, porém um dos tipos de adaptação ontogenética mais estudado envolve o sistema imunológico dos mamíferos (Cziko, 1995).

Para que possamos compreender a evolução dentro do sistema imunológico, faremos um breve resumo da teoria da seleção clonal e maturação de afinidade dos linfócitos B (Seção 2.8). O princípio da seleção clonal pressupõe que uma grande quantidade de linfócitos B contendo receptores antigênicos, chamados de anticorpos, estão circulando constantemente pelo nosso organismo. A grande diversidade deste repertório é resultado da recombinação aleatória de fragmentos genéticos contidos em diversas bibliotecas, somada à inserção aleatória de sequências genéticas conforme a célula se desenvolve. Esta grande diversidade do repertório linfocitário virtualmente garante que pelo menos uma célula irá produzir um anticorpo capaz de reconhecer, e portanto se ligar a qualquer antígeno que invada o organismo (completude do repertório). A ligação Ag-Ab estimula a produção de clones desta célula (divisão celular), de modo que sucessivas gerações resultam no crescimento exponencial da quantidade de anticorpos circulantes do tipo selecionado. Alguns destes anticorpos permanecem circulando mesmo após o final da resposta, constituindo a memória imunológica. Outras células se diferenciam em plasmócitos produzindo anticorpos em grandes escalas. Finalmente, durante a reprodução, alguns clones sofrem o processo de maturação de afinidade, onde mutações somáticas são inseridas com altas taxas e, combinadas a uma forte pressão seletiva, melhoram a capacidade destes anticorpos em reconhecerem os respectivos antígenos (afinidade Ag-Ab).

O funcionamento da resposta imune adaptativa baseado no princípio da seleção clonal, brevemente discutido acima, revela que o sistema imunológico pode ser visto como um microcosmo da teoria da evolução proposta por Charles Darwin (Seção 4.4.1) com as três características fundamentais de *diversidade da população* (repertório), *variação genética* e *seleção natural* (Perelson *et al.*, 1978; Cziko, 1995; Adams, 1996). A diversidade do repertório é evidente, pois o sistema imunológico é capaz de produzir uma quantidade bem maior de anticorpos (repertório potencial) do que é utilizado instantaneamente (repertório ativo). Aparentemente, grande parte dos anticorpos produzidos não executam nenhuma tarefa durante a resposta imunológica. A variação genética ocorre durante a produção de novos anticorpos através da recombinação de fragmentos genéticos das diversas bibliotecas do nosso genoma. Por último, a seleção opera de forma que anticorpos capazes de reconhecerem e se ligarem aos antígenos terão maiores probabilidades de se reproduzir e serem mantidos como células de memória.

A similaridade entre a evolução biológica adaptativa e a produção de anticorpos ainda é mais forte quando consideramos que os dois processos centrais na produção de anticorpos, recombinação e mutação genética, são os mesmos responsáveis pela evolução biológica das espécies sexuadas. A recombinação dos genes que compõem as moléculas de imunoglobulina é responsável pela grande diversidade da população, enquanto a mutação serve como um mecanismo de ajuste da afinidade do anticorpo em relação ao antígeno. Nas

espécies com reprodução sexuada, estes mesmos processos são responsáveis por fornecer as variações sob as quais a seleção natural irá atuar para tornar o organismo melhor adaptado ao seu ambiente (Holland, 1975). Assim, uma variação cega cumulativa juntamente com a seleção natural, que durante milhões de anos resultaram na emergência dos mamíferos, permanece crucial na batalha diária pela sobrevivência desta espécie. Deve ser notado, entretanto, que a recombinação genética das moléculas de imunoglobulina envolvidas na produção de anticorpos difere consideravelmente da recombinação do material genético proveniente dos pais na reprodução sexuada. No primeiro caso, nucleotídeos podem freqüentemente ser inseridos e deletados aleatoriamente dos segmentos gênicos recombinados, enquanto a reprodução sexuada envolve quase sempre apenas a troca (crossover) de material genético dos pais, gerando filhos constituídos por uma mistura genética dos cromossomos dos pais.

Enquanto a evolução biológica adaptativa ocorreu através da seleção natural *entre* organismos (ou entre organismos e genes (Dawkins, 1989)), a pesquisa em imunologia revelou a primeira evidência clara de que uma mudança adaptativa ontogenética pode ser atingida através de variação e seleção cumulativas *dentro* dos organismos. A seleção natural pode ser vista como atuando no sistema imunológico em dois níveis. Primeiro, permitindo a multiplicação e variação das características dos linfócitos que apresentam maiores afinidades ao antígeno e, segundo permitindo a multiplicação dos indivíduos cujo material genético apresenta maior potencialidade de fornecer máxima defesa contra doenças infecciosas associadas ao menor risco de apresentação de doenças autoimunes.

Uma última similaridade entre o funcionamento do sistema imunológico e a evolução biológica pode ser vista como o nosso próprio conhecimento sobre como cada um destes mecanismos operam. A nossa compreensão da adaptabilidade de um organismo a seu ambiente evoluiu de uma explicação instrucionista (Seção 2.1), do tipo proposto por Lamarck, para um modelo puramente seletivo, do tipo neodarwinista discutido na Seção 4.4.1. Estágios semelhantes ocorreram no desenvolvimento da teoria da evolução atualmente aceita.

## **4.6. Outras Abordagens de Sistemas Inteligentes**

Esta seção visa discutir brevemente a relação entre outras abordagens de sistemas inteligentes, em particular a computação molecular e os sistemas nebulosos, o sistema imunológico biológico e a engenharia imunológica.

### **4.6.1. Computação Molecular**

O trabalho pioneiro apresentado por Adleman (1994) introduziu um experimento realizado com *moléculas de DNA* para resolver o problema do caminho Hamiltoniano (HPP – *Hamiltonian path problem*). Este problema pertence à classe NP-completo, é de natureza combinatorial e se assemelha bastante ao problema do caixeiro viajante (TSP – *travelling salesman problem*) a ser apresentado, discutido e resolvido por uma das ferramentas de engenharia imunológica que será proposta no Capítulo 5. O experimento de Adleman

(1994) levou às seguintes conclusões sobre a computação molecular: 1) é possível desenvolver *protocolos* capazes de coordenar moléculas individuais de forma que elas realizem tarefas específicas, e 2) existe uma grande potencialidade das moléculas de DNA na solução de problemas complexos de engenharia. De forma resumida, este experimento consistiu em arranjar cuidadosamente um conjunto de moléculas de DNA de forma que o comportamento químico por elas seguidas (*protocolo molecular*) naturalmente pudesse realizar o procedimento computacional desejado. As operações principais desta química são mecanismos de ligação que permitem que os nucleotídeos básicos dos ácidos nucleicos das moléculas de DNA formem estruturas maiores através de processos de *ligação* (*ligation* ou *matching*) e *hibridização* (*hybridization*). Estes processos de ligação se dão entre pares Watson-Crick complementares, A-T (Adenina-Timina) ou C-G (Citosina-Guanina), portanto, uma única cadeia irá se ligar a outra através da operação de complementaridade de Watson-Crick.

De forma genérica, os procedimentos mais importantes para a computação molecular podem ser caracterizados como a seguir (Garzon & Deaton, 1999):

1. *Eletroforese de gel*: funciona como um microscópio, permitindo a visualização das moléculas a olho nu;
2. *Enzimas de restrição*: funcionam como “tesouras” em operações de corte e ligação; e
3. *Reação em cadeia de polimerase* (PCR – *polymerase chain reaction*): opera como uma copiadora para a computação molecular, permitindo a duplicação de moléculas.

Estes procedimentos permitem executar alguns dos três passos principais da metodologia de computação molecular: a 1) *codificação* que mapeia o problema em cadeias de DNA, a 2) *hibridização/ligação* que realiza o processamento principal e a 3) *extração* que permite visualizar os resultados a olho nu.

Baseado na discussão acima sobre computação molecular e no Capítulo 3 que apresenta os conceitos de engenharia imunológica, é possível traçar um paralelo entre elas, de acordo com a Tabela 4.5.

O único trabalho conhecido na literatura até o presente momento, que desenvolve um sistema imunológico artificial baseado em moléculas de DNA foi apresentado por Deaton *et al.* (1997). Neste trabalho, os autores propuseram uma versão molecular do algoritmo de seleção negativa (ASN) introduzido por Forrest *et al.* (1994) e discutido na Seção 3.4.3.2. No artigo apresentado por Nagano & Yonezawa (1999), foram empregadas seqüências de DNA para representar os elementos próprios e não-próprios de um sistema de auto-reconhecimento, entretanto, a dinâmica deste sistema foi coordenada por um algoritmo genético e não por protocolos de computação molecular.

**Tabela 4.5.** Paralelo entre a engenharia imunológica e a computação molecular.

	<b>Engenharia Imunológica</b>	<b>Computação Molecular</b>
<b>Codificação</b>	Cadeias de atributos, em $S$ , que representam as células e moléculas do sistema imunológico	Cadeias de DNA
<b>Recombinação</b>	Mutação e edição de receptores	Enzimas de restrição
<b>Processamento</b>	Algoritmos (ex. ASN, CLONALG) resultantes da síntese de princípios imunológicos	Protocolos específicos de hibridização/ligação
<b>Ligação</b>	Complementaridade em $S$	Pares complementares Watson-Crick, A-T e C-G
<b>Reprodução</b>	Expansão clonal (mitose celular)	PCR
<b>Decodificação</b>	Interpretação das cadeias de atributos resultantes	Extração (via eletroforese de gel)

#### 4.6.2. Sistemas Nebulosos

O conceito de *conjunto nebuloso* foi introduzido, em 1965, por Lotfi A. Zadeh. A ele é atribuído o reconhecimento como grande colaborador da Teoria de Controle Moderno. Em meados dos anos 60, Zadeh observou que os recursos tecnológicos disponíveis eram incapazes de automatizar as atividades relacionadas a problemas de natureza industrial, biológica ou química, que compreendessem situações ambíguas, não passíveis de processamento através da lógica computacional fundamentada na álgebra (lógica) Booleana. Procurando solucionar estes problemas, Zadeh (1965) publicou um artigo resumindo os conceitos sobre *conjuntos nebulosos*, revolucionando o assunto com a criação dos *sistemas de inferência nebulosa*.

A inferência nebulosa envolve a representação e operação com *variáveis lingüísticas e conjuntos nebulosos*. Conjuntos nebulosos surgiram como uma nova forma de *representação de incertezas*, e têm sido bastante utilizados em problemas de mundo real e, principalmente, em conjunto com outras metodologias como redes neurais artificiais, computação evolutiva e, mais recentemente, sistemas imunológicos artificiais.

A *lógica* refere-se aos princípios formais que sustentam as leis do pensamento e espelham os fundamentos de várias áreas da ciência e engenharia. Sistemas com lógica de dois valores (lógica binária ou bi-valorada) tratam sentenças cujos valores são *verdadeiro* ou *falso*. Em lógica *multi-valorada*, uma sentença pode ser *verdadeira*, *falsa*, ou pode assumir *valores verdade intermediários*. A *lógica nebulosa* vai muito além da lógica multi-valorada, e admite valores verdade que são conjuntos nebulosos do intervalo unitário. Os valores verdade podem ser vistos como caracterizações lingüísticas de valores verdade

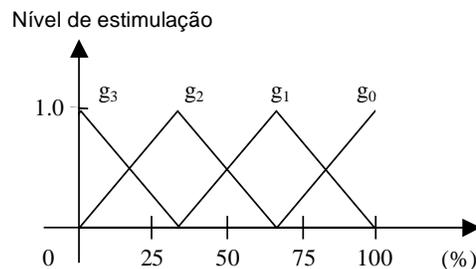
numéricos. Assim, a lógica nebulosa trata os princípios do raciocínio aproximado (Pedrycz & Gomide, 1998).

Krishnakumar *et al.* (1995) propuseram um *sistema computacional imunizado* empregando metáforas imunológicas e técnicas computacionais com o objetivo de reproduzir a robustez e adaptabilidade do sistema imunológico biológico. A estrutura do sistema híbrido proposto, simulando blocos de processamento do sistema imunológico, era composta por redes neurais artificiais, sistemas nebulosos e algoritmos evolutivos, de acordo com o problema a ser tratado. O desempenho desta abordagem foi avaliado em um problema de controle autônomo de aviões.

Nos trabalhos apresentados por Lee *et al.* (1999) e Jun *et al.* (1999), foram utilizados diferentes tipos de anticorpos, cada qual com uma tarefa específica. Neste caso, o nível de estimulação de um dado anticorpo era função do seu percentual de sucesso na execução de uma tarefa, de acordo com uma função semelhante à apresentada na Figura 4.23. Esta função é típica em sistemas nebulosos, onde o universo de discurso de uma variável (no caso o percentual de sucesso) é particionado em vários intervalos, permitindo uma representação aproximada do fenômeno tratado.

Além destes exemplos de aplicações, na Seção 3.4.2.1 foi definido um valor de ligação  $vl$  entre duas moléculas proporcionalmente à afinidade entre elas. Este  $vl$  é dependente de uma função de ativação capaz de permitir valores graduais de ligação. No sistema imunológico biológico, o reconhecimento antigênico acontece de forma aproximada, ou seja, uma resposta imune adaptativa pode ser promovida mesmo quando não ocorre uma ligação perfeita entre o antígeno e o anticorpo. Juntamente com a reatividade cruzada, estas características ressaltam a existência de processamentos nebulosos (aproximados) dentro do sistema imunológico, sugerindo que a lógica nebulosa pode ser bastante apropriada para modelar diversos mecanismos e aspectos do sistema imunológico biológico.

Finalmente, na Seção 5.5 apresentaremos um modelo de rede imunológica artificial para resolver problemas de clusterização e compressão de dados, e discutiremos uma forma de clusterização nebulosa (gradual) que pode ser desempenhada por este modelo.



**Figura 4.23.** Função de determinação do nível de estimulação de um dado anticorpo.

